Modeling Content Accessibility Service Provision in Higher Education

Georgios Kouroupetroglou
National and Kapodistrian University of Athens, Greece,
Department of Informatics and Telecommunications,
Speech and Accessibility Laboratory
koupe@di.uoa.gr

Abstract

Despite the progress that has been made in Assistive Technologies and digital accessibility, not only in the technological field but also in the relevant legislation, today many print-disabled students in Higher Education Institutions still do not enjoy equal access to educational materials. We present the model we have developed for the efficient content accessibility service provision for students with disability. The workflow for the management and distribution of accessible textbooks is given first. Then, we briefly describe the general framework of the accessibility services provided, along with the following innovative methodologies and systems we have investigated to support the academic educational material accessibility service: meaningful high-quality alt-text image descriptions, inventories for free mobile Assistive Technology applications, Document-to-Audio advanced accessibility, data tables' accessibility, evaluation methodologies for Math-to-Speech, 8-dot braille code systems and braille embossed graphics with variable dot elevation. Finally, we present the actions we have taken in the context of three EU projects to disseminate the above know-how to 26 HEIs in eight countries at the regions of North Africa, West Balkans and Near East.

1 Introduction

The number of students with disabilities (SwD) in higher education institutions (HEI) has been steadily increasing over the past two decades [1-2]. The Convention on the Rights of Persons with Disabilities [3], which has been ratified by 183 countries, highlighted the obligation to ensure that persons with disabilities have access to tertiary education without discrimination and on the same terms as those without disability. A significant number of students with disabilities in HEIs are characterized as print-disabled as they cannot read effectively conventional print content of information because of a visual, physical, perceptual, developmental, cognitive, or learning disability. Print disabilities include students with low vision/partially sighted, color vision deficiency, blindness, motor impairment of the upper limbs (including manual dexterity), or dysanagnosia.

Printed or digital academic resources in HEIs include mainly textbooks, teaching notes, handouts, teaching presentations, workbooks, paper assessments, scientific journals or periodicals, webpages, reports, or articles. In some cases, they include complex mathematical expressions, symbols and formulas of physics and chemistry, or even music notations and partitures. During the past four decades, the evolution of computer science and information technology brought out new perspectives for educational resources [4]. Nowadays, the vast number of them is edited and stored in a digital or electronic format. Moreover, print educational resources of the past can be digitized using common scanners and optical character recognition (OCR) software.

Print-disabled students can use software or hardware Assistive Technology (AT) [5-7] to read printed or digital academic content, which can encompass a variety of different methods including braille, large print, audio formats, screen readers, text-to-speech, or a combination of multiple formats, e.g. audio-tactile.

Assistive Technology alone does not ensure content accessibility. Content accessibility refers to the design and delivery of content in a way that allows everyone to access and use it. Digital accessibility refers to the design and delivery of content in a way that allows everyone to access and use it. In practice, digital accessibility is a process which aims to ensure that content is presented in formats that can be easily accessed by persons with disability. Digital accessibility is processing the content to ensure that it has become compatible with the computer-based Assistive Technology of the user, under the condition that both of them follow the relative international standards, e.g. User Agent Accessibility Guidelines (UAAG) for AT and Web Content Accessibility Guidelines (WCAG) for content.

Despite the progress that has been made in Assistive Technologies and digital accessibility, not only in the technological field but also in the relevant legislation, today many print-disabled students still in Higher Education Institutions do not enjoy equal access to educational materials. This paper focuses on the model we have developed with the synergies of the Speech & Accessibility Research Laboratory and the Accessibility Unit of the National and Kapodistrian University of Athens (NKUA) for the efficient content accessibility service provision for SwD. We first present the workflow we follow for the management and distribution of accessible textbooks. Then, we briefly describe the general framework of the accessibility services provided in NKUA along with the following innovative methodologies and systems we have investigated to support the academic educational material accessibility service: meaningful high-quality alt-text image descriptions, inventories for free mobile Assistive Technology applications, Document-to-Audio advanced accessibility, data tables' accessibility, evaluation methodologies for Math-to-Speech, advanced 8-dot braille code systems and braille embossed graphics with variable dot elevation. Finally, we present the actions we have taken in the context of three EU projects to disseminate the above know-how to 26 HEIs in eight countries at the regions of North Africa, West Balkans and Near East.

2 Workflow Management of Accessible Textbooks

The provision of educational material in alternate formats for students with print-based disabilities is challenging and often time-consuming, expensive, and requires special knowledge and training of staff [10]. Legislation, universal guidelines [11] and universal standards [12] play an important role in providing accessible learning material provision (in either traditional or e-learning settings) within a HEI. Academic Accessibility Units or Disability Services' Offices undertake the role for the production and delivery of accessible academic educational content [13]. Standards of provision of accessible textbooks vary from place to place even in the same country and often require new workflows from inquire to access [14]. Moreover, copyright protection of textbooks in connection with the disability rights is another challenging issue [15] that varies across countries.

The HERMOPHILOS web-based Information Technology (IT) system has been designed and developed to automate and accelerate the accessible textbooks' production, workflow management, and delivery in NKUA [8]. It is based on a user-centric approach, i.e. it does not just highlight the individual students' needs with regards to skills and abilities, but also considers localization issues in terms of language and cultural differences [16].

The functional architecture of HERMOFILOS includes two main subsystems:

A. Users and Requests Management Subsystem

- *Users sign up service*: Three main user groups are using the web services: students, publishers, and the Accessibility Unit staff. Students can use the same credentials they use for all other online University services like MYSTUDIES (the secretariats' system where students declare the courses they will take) and EUDOXUS (where students register their preferred books for each course).
- *User authentication service*: Only the print-disabled students are entitled to the accessible digital textbooks and the web services, so this service ensures that only accredited students, personnel, and publishers will access the system.
- *User rights management service*: A general administrator, member of the Accessibility Unit staff, manages all user rights for all three user groups.
- *Students' accessible textbooks request service*: All new requests are made online, and no physical presence of the student at the Accessibility Unit's office is required.
- *Digital textbook requests to publishers' service*: A secure system for electronically sending new requests for textbooks to the publishers.
- *Requests' progress monitoring service*: Both students' requests to Accessibility Unit and the Unit's applications to the publishers are monitored and displayed online.
- *Requests' statistics service*: All stages of the procedure are logged, and executive reports are offered in real time.

B. Digital Content Management Subsystem

- *Scanning service*: A high-resolution scanner with a page feeder is connected to the IT system, and the scanning and storing process can be managed remotely.
- *Original digital textbook copy submission service*: This service is for the publishers that must send large files and they get an official receipt proving that they fulfilled a request.
- *Optical Character Recognition (OCR)*: runs on the high-speed server accelerating the process. We also use systems for math formulas recognition, as well as optical music recognition for partitures.
- *Version management service*: Many volunteers and the Accessibility Unit's staff contribute to OCR correction, document formatting, image description, math and science transcription, tactile graphics preparation, etc. All these produce a huge amount of data and numerous versions of intermediate documents managed online by administrators.
- *Accessible formats archive management service*: HERMOPHILOS supports the creation of multiple accessible formats: plain text (.txt), rich text (.rtf), accessible markup (.xml, .xhtml, and .html), large print (.doc), audio books (.mp3), DAISY 2&3 (text only or full text-full audio), Braille (.brf or .brl), MS-Word (.docx), portable document format (.pdf), ePUB 3, and LaTex (.tex). The required storing capacity is offered by the repository PERGAMOS provided by the central digital library system of the NKUA.
- *Copyright management service*: Digital IDs, electronic signatures, and watermarks are produced and managed centrally, offering a high degree of security and confidence to all stakeholders.
- *Accessible digital textbooks distribution service*: As soon as the preferred accessible format is ready, the student is informed that he or she can immediately download it.
- *Digital content usage statistics service*: File transfer reports and upload/download statistics are offered by HERMOPHILOS for all stages of preparation and distribution of the digital content.

3 Services of the Accessibility Unit

Accessible digital academic material production and distribution described above is one of the core services that the Accessibility Unit of the NKUA has undertaken to benefit students with print-disabilities. The accessibility services provision model of the NKUA [9] follows a student-oriented approach. It is based on the requirements' analysis of the students with disabilities during their studies. Moreover, this model influences their academic environment and the accessibility policy inside and outside the educational institution. The main pillar of this model is the Accessibility Unit which provides several supportive services, arranged in a three-tier architecture according to their "proximity" to the student (*we mark with italics the services that are related with content accessibility service provision*):

(i) Accessibility services addressed directly to the student:
- *Activity and participation restrictions' Registration*
- *Abilities Evaluation Service*
- *Personal Assistive Technologies Service*
- Transportation Service
- Accessible Academic Educational Material Service
- *Accommodations Service*
- Psychological Counseling Service
- Video Relay Service / Remote Sign Language Interpretation
- *Accessibility Support Voluntary Service*

(ii) Accessibility services applied to the student's environment:
- Buildings' Accessibility Service
- *Library Workstations for SwD*
- *Accessibility Guidelines & Tools*
- *Staff and Volunteers Training Service*

(iii) Accessibility promoting services:
- Web Accessibility Evaluation Service
- Events Service
- *Know-How Dissemination Service*
- *Research Service.*

4 Relative Innovative Actions

In the framework of the Research Service of the Accessibility Unit, numerus innovative methodologies and systems have been developed to support the Accessible Academic Educational Material Service, such as the following:

4.1 Meaningful high-quality alt-text image descriptions

Even though considerable efforts have been made to provide effective image descriptions for digital accessibility, a large portion of STEM images, especially complex STEM images, nowadays remains inaccessible to people with visual disability. We notice that It's important to note that having alt text, and having useful alt text are not the same thing. The quality of alt text is much more critical for university STEM textbooks as image descriptions must be accurate and detailed but not tire out the reader. We have created a large corpus of hierarchically classified STEM images from university textbooks [20] that are used in our internal guidelines for meaningful non-automatic high-quality alt-text image descriptions.

4.2 Inventories for free mobile Assistive Technology applications

The search process for mobile AT applications that fulfill specific user needs is not an easy task for the persons with disability, their facilitators, as well as the professionals in rehabilitation. Even, when they finally find what they are looking for, several questions are raised relative to the reliability, stability, compatibility, and functionality of the AT applications. These questions can be answered safely only by a team of AT experts. We have design and developed the ATHENA [17] and mATHENA [18] web-based inventories, for desktop/laptop computers and mobile devices (for smartphones and tablets) respectively, that aim to make the search and selection of freeware and Open-Source mobile AT applications simple and sound. The methodology we followed is based on the consistent and well-documented presentation of the information for each mobile AT application, after it is tested in an AT lab [19]. Both these inventories include a plethora of document accessibility AT applications that make easy for the user to read, listen, produce, and edit electronic texts of various kinds, such as plain text, rich text, text with mathematical and/or diagrams.

4.3 Document-to-Audio advanced Accessibility

We have developed the Document-to-Audio (DtA) advanced accessibility approach for supporting the accessibility of the text signals, i.e. the writing devices that emphasizes aspects of a text's content or structure, such as bold, italics, tables, or bullets [21-25]. DtA essentially constitutes the next generation of Text-to-Speech. DtA relies on two pillars: a) extraction methodologies we have developed for the formatting metadata on structured documents for enhancing their accessibility through prosody modeling and their augmented audio representation [26-35], and b) the emotional-based mapping methodology we have introduced for rendering typographic signals to the auditory modality [36-43].

4.4 Data Tables Accessibility

The semantics involved in simple and complex data tables result in poor and ambiguous text-to-speech synthesis. We first performed an in-depth analysis of the visual and semantic characteristics of data table structures to determine the types of attributes that should be taken into consideration for their aural rendering. After processing the acquired speech data from the most preferred human spoken renditions, we have introduced a corresponding prosody specification, including phrase accent and pause break information. The evaluation results showed that a degree of semantic structure essence is retained in the resulting speech synthesized tables, thus making the content easier for the listener to comprehend [44-47].

4.5 Evaluation methodologies for Math-to-Speech

First, we have introduced three approaches to calculate structure- and content-based performance metrics for user-based evaluation of math audio rendering systems: Syntax Tree alignment, Baseline Structure Tree alignment, and MathML Tree Edit Distance [48]. Then, we have proposed EAR-Math, a methodological approach for user-based evaluation of math rendering against a baseline. EAR-Math measures systems' performance using fine-grained error rates based on the structural elements, arithmetic operators, numbers, and identifiers in a formula [49]. Finally, we have developed a methodology to evaluate in a systematic way the correctness of the Math-to-Speech rendering by investigating specific steps during the production and rendering phases for non-English DAISY Digital Talking Math Books.

4.6 Advanced 8-dot Braille Code Systems

We have introduced a language-independent methodology for the systematic development of an 8-dot braille code. It includes a set of design principles that focuses on achieving an abbreviated representation of the supported symbols, retaining connectivity with the 6-dot representation, preserving similarity on the transition rules applied in other languages, removing ambiguities, and considering future extensions [51]. This methodology was successfully applied for: a) the development of an 8-dot literary Greek braille code that covers both the modern and the ancient Greek orthography, including diphthongs, digits, and punctuation marks [52], b) the rendering typographic signals [53-54], and c) the Nemeth code [55-56].

4.7 Braille Embossed Graphics with Variable Dot Elevation

Most of the braille embossers incorporate a mode for the creation of braille embossed graphics (BEG). Some of them can produce dots in different elevations, but also various dot densities. We have investigated the extent of what blind individuals can identify embossed lines and square areas in eight dot elevations H1–H8 and two dot densities [57-58]. The analysis of the results indicated that the participants classify with better accuracy the stimuli with the three lower dot elevations (between 0.03 mm and 0.15 mm) for both types of stimuli and in both dot densities of 10 and 20 dpi. Better classification accuracy is achieved with the combination of four only dot heights H1, H2, H4, and H8. These results constitute the first recommendations for the designers of braille embossed graphics in applying textures with variable dot height and dot density for educational purposes or for producing tactile maps to benefit users who are blind.

5 Dissemination actions

During the period 2019-2023 we have taken actions to disseminate the know-how described above to 26 HEIs in eight countries at the regions of North Africa, West Balkans and Near East in the context of the following three ERASMUS+ projects of the EU:

- InSIDE - Including Students with Impairments in Distance Education [59],
- Edu4ALL - Disability as Diversity: The Inclusion of Students with Disabilities in Higher Education [60],
- IDEA - Inclusive Tertiary Education in the Western Balkans [61].

The three projects included the following main activities: training (including workshops) of the academic and administrative personnel through study visits to NKUA and teaching visits of experts from the NKUA to the 26 partner universities, development and delivery of training material, purchase of Assistive Technologies and hardware/software for the production of accessible academic educational materials and establishment and pilot operation of an Accessibility Unit in each one of the 26 universities.

References

[1] Leake, D. W., & Stodden, R. A. (2014). Higher education and disability: Past and future of underrepresented populations. *Journal of Postsecondary Education and Disability*, *27*(4), 399-408.

[2] Newman, L. A., Madaus, J. W., Lalor, A. R., & Javitz, H. S. (2021). Effect of accessing supports on higher education persistence of students with disabilities. *Journal of Diversity in Higher Education*, *14*(3), 353. https://doi.org/10.1037/dhe0000170

[3] United Nations (2006). *Convention on the Rights of Persons with Disabilities and Optional Protocol*: Paris. https://www.un.org/disabilities/documents/convention/convoptprot-e.pdf

[4] Kouroupetroglou, G. (2015). Accessibility of Documents. In *Encyclopedia of Information Science and Technology*, 3d Edition, published by IGI Global, Hershey, Pennsylvania (USA), 563-571. https://doi.org/10.4018/978-1-4666-5888-2.ch437

[5] Kouroupetroglou G. (2013). *Assistive Technologies and Computer Access for Motor Disabilities*. IGI Global, Hershey, Pennsylvania (USA). ISBN: 9781466644380.

[6] Fellbaum, K., & Kouroupetroglou, G. (2008). Principles of Electronic Speech Processing with Applications for People with Disabilities", *Journal Technology and Disability*, 20(2), 55-85. https://doi.org/10.3233/TAD-2008-20203

[7] Freitas, D., & Kouroupetroglou, G. (2008). Speech Technologies for Blind and Low Vision Persons, *Journal Technology and Disability*, 20(2), 135-156. https://doi.org/10.3233/TAD-2008-20208

[8] Pino, A., Kouroupetroglou, G., & Riga, P. (2016). HERMOPHILOS: A Web-Based Information System for the Workflow Management and Delivery of Accessible eTextbooks. *Lecture Notes in Computer Science*, 9758, 409-416. https://doi.org/10.1007/978-3-319-41264-1_56

[9] Kouroupetroglou, G., Pino, A., & Kacorri, H. (2011). A model of accessibility services provision for students with disabilities in higher education. In *Proceedings of the International Conference Universal Learning Design*, (pp. 23-33). https://access.uoa.gr/Unit%20Publicity%20Files/Kouroupetroglou_Brno_2011.pdf

[10] Christensen, L. B., Keegan, S. J., & Stevns, T. (2012). SCRIBE: A model for implementing robobraille in a higher education institution. *Lecture Notes in Computer Science*, 7382, 77-83. https://doi.org/10.1007/978-3-642-31522-0_12

[11] Seale, J. (2013). *E-learning and disability in higher education: accessibility research and practice*. Routledge.

[12] Crombie, D., Lenoir, R., McKenzie, N., & Calvo, F. M. (2006). Identifying Trends in Accessible Content Processing. *In Proc. of ELPUB Conference on Electronic Publishing* (pp. 113-122).

[13] Eskay, M., & Chima, J. N. (2013). Library and information service delivery for the blind and physically challenged in University of Nigeria Nsukka Library. *European Academic Research*, 1(5), 625-635.

[14] Beisler, A., & Kurt, L. (2012). E-book workflow from inquiry to access: Facing the challenges to implementing e-book access at the University of Nevada, Reno. *Collaborative Librarianship*, 4(3), 3. https://doi.org/10.29087/2012.4.3.04

[15] Harpur, P., & Suzor, N. (2013). Copyright protections and disability rights: turning the page to a new international paradigm. *The University of New South Wales Law Journal*, 36(3), 745-778.

[16] Draffan, E. A., McNaught, A., & James, A. (2014). eBooks, Accessibility and the Catalysts for Culture Change. *Lecture Notes in Computer Science*, 8548, 543-550. https://doi.org/10.1007/978-3-319-08599-9_80

[17] Pino, A., Kouroupetroglou, G., Kacorri, H., Sarantidou, A., & Spiliotopoulos, D. (2010). An open source/freeware assistive technology software inventory. *Lecture Notes in Computer Science*, 6179, 178-185. https://doi.org/10.1007/978-3-642-14097-6_29

[18] Kouroupetroglou, G., Kousidis, S., Riga, P., & Pino, A. (2015). The mATHENA inventory for free mobile assistive technology applications. *Lecture Notes in Computer Science*, 9416, 519-527. https://doi.org/10.1007/978-3-319-26138-6_56

[19] Kouroupetroglou, G., Pino, A., & Riga, P. (2017). A methodological approach for designing and developing web-based inventories of mobile Assistive Technology applications. *Multimedia Tools and Applications*, 76, 5347-5366. https://doi.org/10.1007/s11042-016-3822-3

[20] Antonakopoulou, T., Riga, P., & Kouroupetroglou, G. (2022). Developing a Corpus of Hierarchically Classified STEM Images for Accessibility Purposes. *Lecture Notes in Computer Science*, 13341, 56-62. https://doi.org/10.1007/978-3-031-08648-9_8
[21] Kouroupetroglou, G. (2015). Text signals and accessibility of educational documents. In *Proceedings of the 11th Intern. Conf. on Educational Technologies: EDUTE* (pp. 45-51). http://www.wseas.us/e-library/conferences/2015/Dubai/EDUTE/EDUTE-05.pdf
[22] Kouroupetroglou, G., & Katsoulis, Ph. (2016). How Blind and Sighted Individuals Perceive the Typographic Text-Signals of a Document. *Lecture Notes in Computer Science*, 9737, 81-90. https://doi.org/10.1007/978-3-319-40250-5_8
[23] Stavropoulou, P., Spiliotopoulos, D., & Kouroupetroglou, G. (2011). Acoustic modeling of dialogue elements for document accessibility. *Lecture Notes in Computer Science*, 6768, 175–184. https://doi.org/10.1007/978-3-642-21657-2_19
[24] Kouroupetroglou, G. (2013). Incorporating Typographic, Logical and Layout Knowledge of Documents into Text-to-Speech. In *Proc. of the 12th European Conference on Assistive Technologies (AAATE)*, (pp. 708-713). https://doi.org/10.3233/978-1-61499-304-9-70
[25] Argyropoulos, V., Sideridis, G., Kouroupetroglou, G., & Xydas, G. (2009). Auditory Discriminations of Typographic Attributes of Documents by Students with Blidness. *British Journal of Visual Impairment*, 27(3), 183-203. https://doi.org/10.1177/0264619609106360
[26] Xydas, G., Spiliotopoulos, D., & Kouroupetroglou, G. (2003). Prosody prediction from linguistically enriched documents based on a machine learning approach. In *Proceedings of the 6th International Conference of Greek Linguistics* (pp. 135-138). Retrieved from https://speech.di.uoa.gr/sppages/spppdf/web-xydas_icgl2003.pdf
[27] Xydas, G., Spiliotopoulos, D., & Kouroupetroglou, G. (2003). Modelling emphatic events from non-speech aware documents in speech based user interfaces. Proc. Human-Computer Interaction-HCII 2003, Theory and Practice, 2, 806-810. Retrieved from https://shorturl.at/uvUX8
[28] Xydas, G. & ,Kouroupetroglou, G. (2001). Augmented Auditory Representation of e-Texts for Text-to-Speech Systems. *Lecture Notes in Artificial Intelligence*, 2166, 134-141. https://doi.org/10.1007/3-540-44805-5_17
[29] Xydas, G., Argyropoulos, V., Karakosta, T., & Kouroupetroglou, G. (2005). An experimental approach in recognizing synthesized auditory components in a non-visual interaction with documents. In *Proceedings of the 11th International Conference on Human-Computer Interaction*. Retrieved from https://shorturl.at/vxMS7
[30] Fourli-Kartsouni, F., Slavakis, K., Kouroupetroglou, G., & Theodoridis, S. (2007). A Bayesian network approach to semantic labelling of text formatting in XML corpora of documents. *Lecture Notes in Computer Science*, 4556, 299-308. https://doi.org/10.1007/978-3-540-73283-9_34
[31] Tsonos, D., Xydas, G., & Kouroupetroglou, G. (2007). Auditory accessibility of metadata in books: a design for all approach. *Lecture Notes in Computer Science*, 4556, 436-445. https://doi.org/10.1007/978-3-540-73283-9_49
[32] Spiliotopoulos, D., Petasis, G., & Kouroupetroglou, G. (2008). A framework for language-independent analysis and prosodic feature annotation of text corpora. *Lecture Notes in Artificial Intelligence*, 5246, 517-524. https://doi.org/10.1007/978-3-540-87391-4_66
[33] Kouroupetroglou, G. (2015). Acoustic Mapping of Visual Text Signals through Advanced Text-to-Speech: the Case of Font Size. *WSEAS Transactions on Computers*, 14, 559-569. Retrieved from http://www.wseas.us/journal/pdf/computers/2015/b105805-098.pdf
[34] Katsoulis, F., & Kouroupetroglou, G. (2015). How Sighted and Blind Students Perceive Relational Similarity between Font-Size and Loudness in Text-To-Speech. *The Turkish online Journal of Educational Technology*, 245-251. Retrieved from https://rb.gy/s9ngek
[35] Kouroupetroglou, G., & Tsonos, D. (2015). Rendering web-content text signals through advanced Text-to-Speech. In *Proc. of the International Conference on Internet Studies*, Tokyo, Japan, pp. 1-8. Retrieved from https://shorturl.at/ehrsW

[36] Tsonos, D., & Kouroupetroglou, G. (2011). Modeling reader's emotional state response on document's typographic elements. *Advances in Human-Computer Interaction*, Article ID 206983, 1-18. https://doi.org/10.1155/2011/206983
[37] Tsonos, D., & Kouroupetroglou, G. (2016). Prosodic Mapping of Text Font Based on the Dimensional Theory of Emotions: a Case Study on Style and Size. *EURASIP Journal on Audio, Speech, and Music Processin*8, 1-16. https://doi.org/10.1186/s13636-016-0087-8
[38] Tsonos, D., Kouroupetroglou, G., & Deligiorgi, D. (2013). Regression Modeling of Reader's Emotions Induced by Font Based Text Signals. *Lecture Notes in Computer Science*, 8010, 434-443. https://doi.org/10.1007/978-3-642-39191-0_48
[39] Kouroupetroglou, G. Tsonos, D., & Vlahos, E. (2009). DocEmoX: A System for the Typography-Derived Emotional Annotation of Documents. *Lecture Notes in Computer Science*, 5616, 550-558. https://doi.org/10.1007/978-3-642-02713-0_58
[40] Tsonos, D., Ikospentaki, K., & Kouroupetrolgou, G. (2008). Towards modeling of Readers' Emotional State response for the automated annotation of documents. In *IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)* (pp. 3253-3260). IEEE. https://doi.org/10.1109/IJCNN.2008.4634260
[41] Tsonos, D., & Kouroupetroglou, G. (2008). A Methodology for the Extraction of Reader's Emotional State Triggered from Text Typography. In *Tools in Artificial Intelligence*, In-Tech Publishing, (pp 439-454). https://doi.org/10.5772/6071
[42] Tsonos, D., Xydas, G., & Kouroupetroglou, G. (2007, October). A Methodology for Reader's Emotional State Extraction to Augment Expressions in Speech Synthesis. In 19th IEEE International conference on tools with artificial intelligence (ICTAI 2007) (Vol. 2, pp. 218-225). IEEE. https://doi.org/10.1109/ICTAI.2007.127
[43] Tsonos, D., Stavropoulou, P., Kouroupetroglou, G., Deligiorgi, D., & Papatheodorou, N. (2014). Emotional prosodic model evaluation for greek expressive text-to-speech synthesis. *Lecture Notes in Computer Science*, 8514, 166-174. https://doi.org/10.1007/978-3-319-07440-5_16
[44] Spiliotopoulos, D., Xydas, G., Kouroupetroglou, G., Argyropoulos, V., & Ikospentaki, K. (2010). Auditory universal accessibility of data tables using naturally derived prosody specification. *Universal Access in the Information Society*, 9, 169-183. https://doi.org/10.1007/s10209-009-0165-0
[45] Spiliotopoulos, D., Stavropoulou, P., & Kouroupetroglou, G. (2009). Acoustic rendering of data tables using earcons and prosody for document accessibility. *Lecture Notes in Computer Science*, 5616, 587-596. https://doi.org/10.1007/978-3-642-02713-0_62
[46] Spiliotopoulos, D., Xydas, G., Kouroupetroglou, G., & Argyropoulos, V. (2005, July). Experimentation on spoken format of tables in auditory user interfaces. In *Proceedings of the 11th International Conference on Human-Computer Interaction (HCII'05)* (pp. 361-370). Retrieved from https://speech.di.uoa.gr/sppages/spppdf/UA.pdf
[47] Spiliotopoulos, D., Xydas, G., & Kouroupetroglou, G. (2005). Diction Based Prosody Modeling in Table-to-Speech Synthesis. *Lecture Notes in Computer Science*, 3658, 294-301. https://doi.org/10.1007/11551874_38
[48] Kacorri, H., Riga, P., & and Kouroupetroglou, G. (2014). Performance Metrics and Their Extraction Methods for Audio Rendered Mathematics. *Lecture Notes in Computer Science*, 8547, 614-621, 2014. DOI 10.1007/978-3-319-08596-8_95
[49] Kacorri, H., Riga, P., & Kouroupetroglou, G. (2014). EAR-Math: Evaluation of Audio Rendered Mathematics. *Lecture Notes in Computer Science*, 8514, 111-120, 2. https://doi.org/10.1007/978-3-319-07440-5_11
[50] Riga, P., Kouroupetroglou, G., & Ioannidou, P. (2016). An Evaluation Methodology of Math-to-Speech in Non-English DAISY Digital Talking Books. *Lecture Notes in Computer Science*, 9758, 27-34. https://doi.org/10.1007/978-3-319-41264-1_4

[51] Kacorri, H., & Kouroupetroglou, G. (2013). Design and Developing Methodology for 8-dot Braille Code Systems. *Lecture Notes in Computer Science*, 8011, 331-340. https://doi.org/10.1007/978-3-642-39194-1_39

[52] Argyropoulos, V., Martos, A., Sideridis, G., Kouroupetroglou, G., Nikolaraizi, M., & Papazafiri, M. (2015). Reading comprehension issues and Individuals with visual impairments: the effects of using 8-dot and 6-dot braille code through a braille display. *Lecture Notes in Computer Science*, 9176, 71-81. https://doi.org/10.1007/978-3-319-20681-3_7

[53] Argyropoulos, V., Kouroupetroglou, G., Martos, A., Nikolaraizi, M., & Chamonikolaou, S. (2014). Patterns of Blind Users' Hand Movements: The Case of Typographic Signals of Documents Rendered by Eight-dot and Six-dot Braille Code. *Lecture Notes in Computer Science*, 8547, 77-84, https://doi.org/10.1007/978-3-319-08596-8_12

[54] Argyropoulos, V., Martos, A., Kouroupetroglou, G., Chamonikolaou, S., & Nikolaraizi, M. (2014). An experimental approach in conceptualizing typographic signals of documents by eight-dot and six-dot braille code. *Lecture Notes in Computer Science*, 8514, 83-92. https://doi.org/10.1007/978-3-319-07440-5_8

[55] Martos, A., Kouroupetroglou, G., Argyropoulos, V., & Deligiorgi, D. (2014). Towards the 8-dot Nemeth braille code. *Lecture Notes in Computer Science*, 8547, 533-536. https://doi.org/10.1007/978-3-319-08596-8_83

[56] Martos, A., Kouroupetroglou, G., & Argyropoulos, V. (2015). 8-dot Braille Code for Complex Nemeth Symbols. In *Proceedings of the International Conference on Enabling Access for Persons with Visual Impairment*, (pp. 115-119). Retrieved from https://access.uoa.gr/ICEAPVI-2015/proceedingsFiles/117-121.pdf

[57] Kouroupetroglou, G., Martos, A., Papandreou, N., Papadopoulos, K., Argyropoulous, V., & Sideridis, G.D. (2016). Tactile Identification of embossed raised lines and raised squares with variable dot elevation by persons who are blind. *Lecture Notes in Computer Science*, 9759, 77-84. https://doi.org/10.1007/978-3-319-41267-2_11

[58] Martos, A., Kouroupetroglou, G., Argyropoulos, V., & Papadopoulos, K. (2021). Tactile identification of embossed lines and square areas in diverse dot heights by blind individuals. *Universal Access in the Information Society*, *20*, 333-342 https://doi.org/10.1007/s10209-020-00729-4

[59] InSIDE - Including Students with Impairments in Distance Education. https://www.inside-project.org/

[60] Edu4ALL-Disability as Diversity: The Inclusion of Students with Disabilities in Higher Education. https://edu4all.psdpal.org/

[61] IDEA - Inclusive Tertiary Education in the Western Balkans. https://idea-cbhe.com/

New Accessibility Platform "Chatty Library"

Masakazu Suzuki[1,2*]
[1] Kyushu University, Fukuoka, Japan
[2] NPO: Science Accessibility Net, Fukuoka, Japan
msuzuki@sciaccess.net

Abstract

We are developing a new accessibility platform named "Chatty Library" that is a web site to provide dyslexic students with not only daisy books (we refer to Multimedia DAISY and Accessible EPUB3 all together as daisy books) themselves but also a variety of other innovative services related to them: playing them back with a popular browser, converting PDF into a daisy book automatically, etc. In Chatty Library Project, we have also organized a research group to develop a system to convert PDF into daisy books.

To realize automatic conversion, we have proposed a new type of daisy books, "Fixed-Layout DAISY." In it, the whole page is treated as a multi-layer picture, the front layer of which has the same form as the original PDF. A DAISY (EPUB3) player can read out any texts together with highlighting them. Fixed-Layout DAISY can be produced almost automatically from an original "e-born PDF" with the conversion service in Chatty Library, and it should be very helpful for the dyslexic students to get accessible version of their textbooks.

1 Introduction

As is well known, "DAISY (Digital Accessible Information System)" (or accessible EPUB3 that is essentially DAISY4) has already held the position of the international standard for accessible e-books [1]. In Japan, "The Japanese Society for Rehabilitation of Persons with Disabilities (JSRPD)" has been providing print-disabled students with e-textbooks in multimedia DAISY or accessible EPUB3 (audio-embedded DAISY/EPUB3 as media overlay; here, we refer to them all together as "daisy books") since 2008 [2]. It now organized 23 volunteer groups/organizations and produced around 300 titles of accessible textbooks for elementary and junior-high school (the ages of compulsory attendance at school in Japan). Now, around 20,000 print-disabled students (mostly ones with developmental reading disorder, namely, dyslexia) use those textbooks. Certainly, Braille and large-print versions are also provided to blind and low-vision students, respectively.

[*] Masterminded NPO: Science Accessibility Net and newly Chatty Library Project

Concerning textbooks in senior-high school and higher education, however, it is very hard for dyslexic students in Japan to obtain accessible textbooks. A number of textbook titles in elementary and junior-high school is limited, but extremely many titles (more than 1,000) are used even in senior-high school. As you know, recently, very complicated layout is usually used in a textbook, and conversion process into daisy books requires a lot of manual works to get a DAISY player to read out them properly. It is very difficult to convert all necessary titles into daisy books just for senior-high-school students. Furthermore, in higher education, the titles of textbooks reach an enormous number. Thus, no official organizations including JSRPD do not provide the dyslexic students with accessible textbooks. Several small volunteer groups help them individually with converting a limited number of textbooks into daisy books, but obviously, it is not good enough. Unfortunately, in Japan, many students who do need accessible textbooks due to the severe dyslexia decide to give up the idea of going to senior-high school or college.

To give a practical/effective solution to this problem, we are developing "Chatty Library" under the Nippon Foundation's subsidy [3] that is contributed to welfare-aimed innovative projects. This new accessibility platform is a web site that provides dyslexic people with not only daisy books themselves but also a variety of other innovative services related to them: playing them back with a popular browser, converting PDF/text files/contents on the Windows Control Panel into a daisy book automatically, etc. In addition, we have proposed a new type of daisy books, "Fixed-Layout DAISY," which can be produced almost automatically from an original "e-born PDF" by making use of our OCR (optical character recognition) system. To realize that, we have also organized a research group to develop a system to improve remarkably our OCR ability.

2 Outline of Chatty Library

We started Chatty Library Project since January 2023. The development of this project has been gone ahead as the following three parallel projects.

- Constructing a web site on which various online service described later are provided.
- Developing a system to improve our OCR technology.
- Producing daisy books initially stocked in the library.

We are now pre-opening it to examine its various operations and to fix bugs. Its official site will open in February 2024.

Chatty Library is an online e-library of daisy books for people with reading difficulties due to visual disabilities or developmental disorders. This library is a web site that consists of three sectors: Library, ChattyBox and Factory. This site provides them with not only DAISY books but also a variety of other services related to daisy books, which include playing back a daisy book, converting PDF/text files/contents on the Windows Clipboard into daisy contents, supporting production and publication of daisy books, etc.

- Library

Library provides print-disabled people with a collection of daisy books ranging from textbooks to comics. They can use several functions to find necessary books such as keyword search, classification by a genre, the top chart of well-read books, etc.

To use Library, a user should have a certificate to show their disability. In Japan, visually disabled people usually have official certificate, but dyslexic people do not. Concerning dyslexic people, they have to obtain alternative (private) certificate such as one that their teacher guarantees the disability.

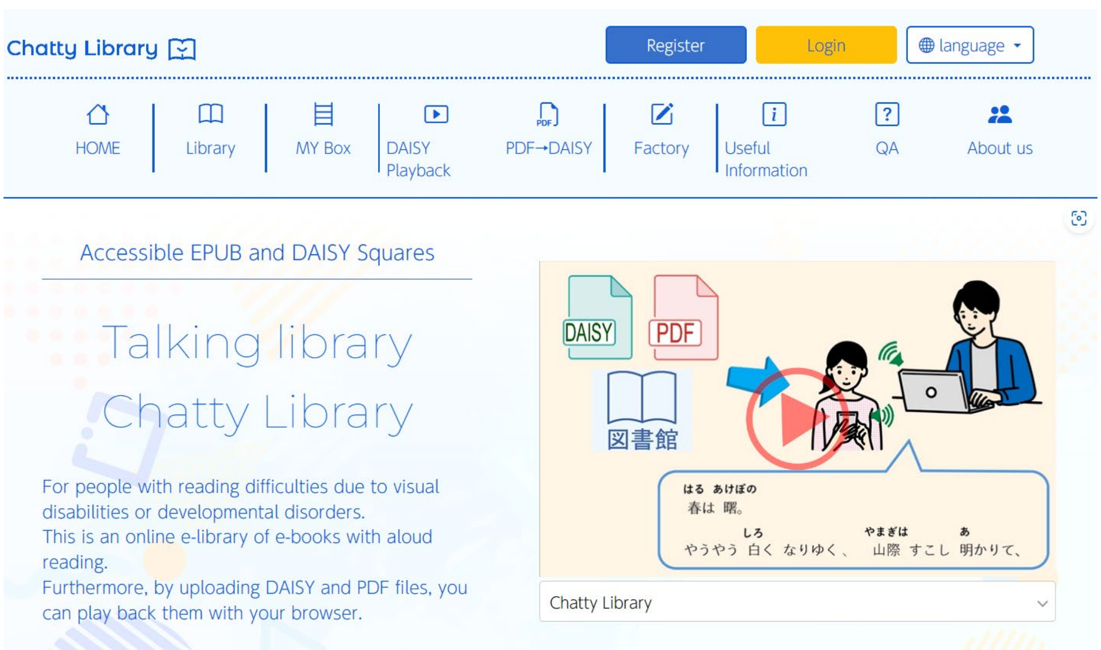
- ChattyBox

In ChattyBox, a user can play back a daisy book in Library with a popular browser. No special application is necessary to read it. Using a Progressive Web Application (PWA), a user can also store a daisy book in cache on their own computer/smart phone to play it back offline. You can upload your own daisy books onto ChattyBox to play them back directly on the web site. It can be used as an OS-independent DAISY player without installing any special (player) application.

By making use of our OCR technology, we provide you with an innovative service to convert your-uploaded PDF files, text files or contents on the Windows Clipboard into daisy books with aloud reading. The conversion result can be played back directly on ChattyBox or downloaded in Daisy-book format.

- Factory

Factory is still under construction. In near future, we intend to provide teachers/ volunteers with online services for supporting their production and publication of daisy books.



3 Problems in Automatic Conversion

3.1 PDF Accessibility

One of the most serious problems in digitized STEM (science, technology, engineering and mathematics) contents, which are provided in PDF in most cases, is their poor accessibility. So-called "PDF/UA" is supposed to be accessible in a certain level; however, making PDF/UA contents requires a lot of extra works [4], and for the present, print-disabled people hardly expect to obtain such accessible

PDF. Thus, they usually use OCR software to read PDF contents. In terms of STEM contents, however, the ordinary OCR software cannot recognize technical parts such as mathematical formulas properly, and it is hard for them to read PDF STEM contents. To solve this problem, we have been working on the development of OCR software for STEM contents, "InftyReader" [5].

There are essentially two different types in PDF. The first one, "e-born PDF" is PDF produced directly from a digital file such as a document in Microsoft Word, LaTeX, Adobe InDesign, etc. We refer to the other type as "image PDF," which is usually made by scanning or copying.

We have released InftyReader for almost twenty years. Through user supports for the software, we have realized that in recent years, most of the (individual) end users use InftyReader to read STEM contents in e-born PDF, and the importance of e-born PDF accessibility is definitely increasing.

3.2 Conversion of PDF into daisy books

From the viewpoint of computerized processing to convert e-born PDF into accessible form, its most significant advantage is that the accurate information on each character/symbol such as the character code, the font name, the coordinate on a page is embedded in it. Actually, in ICCHP2016, we reported the new method of mathematical OCR to improve recognition accuracy for e-born PDF, by combining analysis technologies in our mathematical OCR with character/symbol information extracted from the PDF by a PDF parser [6]. However, as far as a mathematical part is concerned, a font bounding box (a rectangular area being circumscribed on the font) extracted from e-born PDF by the PDF parser often differs significantly from the graphical (real) bounding box of the original character image. Consequently, in the previous works, it was impossible to realize mathematical recognition based only on character information extracted from PDF.

In the DEIMS2021 conference, Fujiyoshi, et al. reported a completely new approach to extract character information from PDF [7]. They developed an application named "PDFContentExtracter" that makes the vector information of drawing each character/symbol in scalable vector graphics (SVG) by trapping a function for printing PDF. This application allows us to get a correct graphical bounding box even in a mathematical part. By making use of their application, we have actually improved InftyReader so that its structure analysis of mathematical formulas is less dependent on the image-based OCR result for characters/symbols [8]. This approach has improved remarkably recognition accuracy for e-born PDF STEM contents represented in Latin characters. Furthermore, even if a Unicode-based local language is used in its text part, without a special OCR engine for that language, InftyReader can convert the PDF into various accessible e-formats: LaTeX, human-readable (HR) TeX, Microsoft Word, xHTML with MathML, daisy books, etc.

If page layout were not complicated, InftyReader could recognize and convert it into accessible formats automatically. As was pointed out, however, the textbook layout recently tends to be very complicated. Its page consists of not only Body texts but also many sub-texts such as foot notes or other marginal notes, balloons, figure captions, etc. OCR software probably can recognize those text blocks themselves, but unfortunately, it cannot judge their order in which they should be read out. Without reading-order annotation, a DAISY player would read out them from the left to the right, from the top to the bottom in disregard of word connection. To get the player to read out the content properly, you do need to consider the order and give appropriate annotation to the recognition result manually. In addition, DAISY contents should be displayed properly through so-called "reflow." An authoring process to optimize those things requires you a lot of extra time and efforts in addition to ordinary editing works such as error correction.

4 Some Extension in Daisy Books

A remarkable feature of recent textbooks in Japan is to use a lot of pictures to tell important points. They are often conveyed only by those pictures. To understand the content of the textbooks thoroughly, you do need to read texts included in the pictures. Certainly, many pictures are also used in textbooks especially ones for young students in other countries, but they are usually so-called illustrations to help a reader with understanding a story. In most cases, you should find word descriptions corresponding to them in the main article, and you could understand the content even if you could not read texts in the pictures. Thus, most popular DAISY/EPUB3 players in the world do not have a function to read out text in a picture. Instead, they would read out an alternative text for the picture if it were available. In Japan, however, many pictures in a textbook often play a role to tell a certain part of its main article, which is not explained sufficiently in words. Certainly, you could give them alternative text, but in many cases, a long passage should be necessary to tell. Thus, the alternative-text approach does not work enough in Japan. Furthermore, here, we would like to point out that the picture representation would rather be good for dyslexic students. They can see, and the picture representation should be easier for them to understand if texts in the pictures were read out. They should prefer that to the alternative text.

To mee this demand, we have introduced a layer structure in daisy books. A picture is represented in multiple layers, the front layer of which is an image that appears in the completely same layout as the original PDF. On the underlying layers, text-related information such as fonts, their coordinate on the page and bounding box, etc. (the recognition result) is stored, and by making use of it, a DAISY (EPUB3) player could read out texts in the picture with a TTS (text-to-speech) voice together with highlighting them. This multi-layer EPUB3 (DAISY) is authenticated by an EPUB3 validator [9], and it should be regarded as a kind of accessible EPUB3. Actually, some EPUB3 (DAISY) players in Japan can read out texts in a multi-layer picture. JSRPD has already adopted this- type daisy textbooks and provided print-disabled students.

To make DAISY/accessible EPUB3 contents more useful, we have also developed a Windows application named "ChattyBooks" that converts STEM contents in DAISY/accessible EPUB3 into audio-embedded HTML5 with JavaScript ("ChattyBook") [10]. It consists of two component modules: a converter and a file manager. If a DAISY/accessible EPUB3 content is dropped on the ChattyBooks icon or in its main window, the content is converted automatically into HTML5 with JavaScript (a ChattyBook content). A popular browser such as Microsoft Edge, Google Chrome, Safari can display the content which has the almost-same functionality and operability of high-quality as the original DAISY/accessible EPUB3. A special DAISY/EPUB3 player is no longer needed to read out the content. JSRPD has also adopted ChattyBooks as one of standard players for their accessible textbooks that include multi-layer pictures.

5 Fixed-Layout DAISY

In The layer structure allows us to develop a new-type accessible e-book named "Fixed-Layout DAISY." In it, the whole page is treated as a multi-layer picture, the front layer of which has the same form as the original (outline) PDF. A DAISY (EPUB3) player can read out any texts together with highlighting them as same as a multi-layer picture located locally on a page.

A main difference of Fixed-Layout DAISY from ordinary DAISY is that it does not have the reflow function. The page layout is always kept as same as the original. It does not have information either in which order texts on the page should be read out. These features look disadvantage for print-disabled students to read the content.

As is well known, DAISY was originally designed for visually disabled people, and subsequently, dyslexic people have become to use. Thus, DAISY is designed so that various-types of print-disabled people can read the same contents. However, as far as dyslexic people are concerned, some features of DAISY such as reflow, "automatic" aloud reading are not absolutely necessary requirements. In Fixed-Layout DAISY, its content cannot be read out automatically. You need to click a position on a page where you want to start reading; then, the text is read out up to the end of the text block. Certainly, this mechanism may not work for visually disabled people; however, dyslexic people can see and click a place where they want to read. Fixed-Layout DAISY should work for dyslexic people, we believe. In addition, Fixed-Layout DAISY would rather be appropriate to produce the accessible version of a book other than textbooks such as comics, picture books, (Pictorial) maps, etc.

Fixed-Layout DAISY, you do not need to author it either so that a DAISY/EPUB3 player can read out its texts in a proper order. Thus, it becomes possible to realize almost automatic conversion from e-born PDF into Fixed-Layout DAISY. Actually, we have implemented this function in InftyReader to realize automatic PDF conversion on ChattyBox. Using it, dyslexic people can convert PDF into Fixed-Layout DAISY for themselves. It should be very helpful for dyslexic students in senior-high school or higher education to get the accessible version of their textbooks.

6 Preliminary Evaluation

At some meetings in Japan, we have recently introduced Fixed-Layout DAISY and demonstrated how it worked. In the online lecture on 26 February 2023, more-than-200 people (teachers, volunteer-group members, parents and publishers who are associated with education for dyslexic students) participated, and we listened to their opinions. We realized that majority of them gave it strong support. In addition, some teachers pointed out that Fixed-Layout DAISY should be more appropriate for an exam paper. In it, keeping the original layout is often more important than a textbook. Someone said that Fixed-Layout DAISY should be very useful for students with slight dyslexia who were in a transition process to a reader of ordinary print books.

7 Conclusion and Future Tasks

Library and ChattyBox services on Chatty Library are nearly completed and expected to start since the mid-February 2024 while the start of Factory should delay slightly more. All services should be available by the mid-2024. Our remaining important task is to develop a method to infer correct reading order. We have been repeating one trial and error after another to find a good mechanism to give a correct order to text blocks in a page image; however, unfortunately, not yet. If we found it, automatic conversion of PDF into ordinary reflow-type daisy books would be available on Chatty Library.

(Acknowledgment)
We greatly appreciate support by the Nippon Foundation to achieve this project.

References

[1] DAISY Consortium (n.d.) http://www.daisy.org/

[2] Japanese Society for Rehabilitation of Persons with Disabilities (n.d.) (Japanese only) http://www.dinf.ne.jp/doc/daisy/book/daisytext.html

[3] Nippon Foundation (n.d.) https://www.nippon-foundation.or.jp/en

[4] sAccessNet (n.d.) http://www.sciaccess.net/en/

[5] Soiffer N. (2021) *Accessible PDF:* $2 \ngtr 1$, Proc. the 4th International Workshop on “Digitization and E-Inclusion in Mathematics and Science 2021” (DEIMS2021) online, Nihon University, pp.1-8.

[6] Suzuki M, Yamaguchi K. (2016) *Recognition of E-Born PDF Including Mathematical Formulas*, Proc. the 15th International Conference on Computers Helping People with Special Needs (ICCHP 2016), Linz, Springer, LNCS 9758, pp.35-42, DOI: https://doi.org/10.1007/978-3-319-41264-1_5.

[7] Nakamura S, Kohase K, Fujiyoshi A. (2021) *Extracting Precise Coordinate Information of Components from E-Born PDF Files,* Proc. the 4th International Workshop on “Digitization and E-Inclusion in Mathematics and Science 2021” (DEIMS2021), online, Nihon University, pp.15-18.

[8] Suzuki M, Yamaguchi, K. (2022) *Conversion of Multi-Lingual STEM Documents in E-Born PDF into Various Accessible E-Formats*, Proc. the 18th International Conference on Computers Helping People with Special Needs (ICCHP-AAATE 2022), Lecco, Springer, LNCS13341, pp.7-14, DOI: https://doi.org/10.1007/978-3-031-08648-9_2.

[9] pagina EPUB-Checker (n.d.) https://pagina.gmbh/startseite/leistungen/publishing-softwareloesungen/epub- checker/

[10] Yamaguchi K, Kanahori T, Suzuki M. (2021) March 09, *ChattyBook: Making Math Accessible for Online Education*, Presentation at the 36th CSUN Assistive Technology Conference, online.

Pre-Assessment Holistic Screening Tool (PRASHAST) for Disabilities

Dr.Bharti Kaushik , Associate Professor,CIET,NCERT
Ms.Latika Khullar,Junior Project Fellow, CIET,NCERT
Ms.Bhawna Malik, Academic Consultant,CIET,NCERT

Abstract

Disability is a complex and multifaceted concept. The understanding and significance of disability vary across different regions, influenced by various legal, political, and social constructs. It is generally viewed as a physical or mental condition(s), or both, that limit(s) an individual's movements, activities and sense perceptions. Persons with disabilities are forced to face a lot of discrimination due to prejudices and biases in society. The most vulnerable area that falls prey to this discrimination is "EDUCATION". However, it is worth noting that we have made significant progress, from the historical notion of exclusion to the various phases of integration, ultimately arriving at the concept of inclusive education.One of its primary indicators is the United Nations Convention on the Rights of Persons with Disabilities (UNCRPD), which advocates for promoting and safeguarding the human rights and fundamental freedoms of individuals with disabilities on an equal basis with others. This emphasises the integration of disability issues into sustainable development strategies. The guiding principles of the UNCRPD include respect for individual autonomy, non-discrimination, full societal inclusion, and equal opportunity. The process of creating an inclusive environment begins with the critical step of identifying the presence of disabilities. These disabilities should be appropriately addressed, and individuals should be supported to realise their full potential within society. This initial identification stage serves as the foundation upon which subsequent efforts are built. Without a clear understanding of the disabilities individuals may face, it becomes challenging to develop tailored and effective strategies to address their specific needs. (SSA,2019)In the context of India, the identification of disabilities is not only essential but also legally mandated. The 2016 Rights of Persons with Disabilities Act (RPWDA) in India recognises 21 different disabilities, highlighting the country's commitment to fostering inclusivity and support for individuals with diverse needs. Moreover, the Right to Education (RTE) Act of 2009 ensures free and compulsory education for all children, including Children With Special Needs (CWSN). However, India's vast

geographical expanse and population size present unique challenges when it comes to identifying and supporting individuals with disabilities, especially children in schools. One significant challenge is the shortage of adequately trained professionals, particularly special educators, who play a pivotal role in identifying and catering to the needs of disabled students. In the broader context of fostering an inclusive educational environment, the role of regular classroom teachers in the identification process is undeniably imperative. These educators serve as the frontline observers and caretakers of students, as they are in constant, daily contact with them. Their unique position grants them unparalleled insight into the behavioural patterns, learning styles, and needs of their pupils. Furthermore, regular classroom teachers serve as a bridge between students and specialised support services. Their early observations and insights can initiate the initial steps toward early intervention and support, which are often crucial in effectively addressing disabilities, thereby contributing to a more inclusive educational environment in India.Subsequently, The teachers also serve as credible points of contact with parents to elicit pertinent information. However, regular teachers, while crucial in the identification process, often lack the in-depth preservice and in-service training and knowledge base required to accurately recognise and effectively support disabled children. Recognising the significant gap in teacher preparation and the vital importance of early identification, the Central Institute of Educational Technology (CIET), which is a constituent unit of the National Council of Educational Research and Training (NCERT), has taken proactive measures to address this issue. CIET, as the national authority responsible for educational technological innovation and research in India, has introduced an innovative solution called "PRASHAST." This application is based on the screening checklist specifically designed to facilitate the school level screening of 21 disabilities recognized in the RPwD Act 2016. . Part 1 of the PRASHAST checklist is intended for completion by regular teachers, who may not possess in-depth knowledge of disabilities, as their prime responsibility is to deliver the curriculum. Part 2, on the other hand, requires vetting of observations of part 1 by the special educators who are expected to have better knowledge of disabilities. . The screening results of the PRASHAST provide requisite information for further assessment, identification and certification. . The development of PRASHAST began with the status survey to gather information and insights regarding the process adopted by all the States and Union Territories of India for screening, assessment, identification , and certification of disability. This survey revealed that 52.9% of states face difficulties in collecting required information from the schools due to lack of disability knowledge among the regular teachers and also due to the overlapping characteristics and symptoms of certain disabilities such as Mental Illness, Speech and Language Disability, Learning Disabilities, Haemophilia, and Thalassemia. Further, 76.4% of states actively engage special educators and medical experts in the identification process of Children with Special Needs (CWSN), hence, they didn’t face any challenge or confusion in the process of identification of disabilities**.** Lack of awareness and knowledge about the various disability conditions among regular teachers emerged as the major reason for not involving them in the identification process. Eight states—Sikkim, Tamil Nadu, Kerala, Assam, Chhattisgarh, Puducherry, Delhi, and Uttar Pradesh—were found to have either finalised or are in the process of finalizing their own checklists for identifying CWSN. Most of these states, however, were not considering all the 21 disabilities recognized in RPwD Act 2016. Nonetheless, the absence of adequate identification and screening tools emerged as a significant barrier to the prompt and accurate identification of CWSN. The pre-assessment holistic screening tool for schools was standardised with the help of 100 schools in Chandigarh, leading to screening of 3,482 pupils through PRASHAST Part-1. Subsequently, 637 students were referred to assessment and certification camps on the basis of PRASHAST Part-2. The PRASHAST part 1 checklist comprises 63 behavioural statements and part 2 contains 21 disability checklists, one for each disability condition recognized in RPwD Act 2016. The internal consistency of PRASHAST, measured using Cronbach Alpha ($\alpha = 0.801$), falls within an acceptable range, indicating good internal consistency. After establishing the statistical standards for the screening tool an Android App was developed, for the ease of end user and collating the real time data. This App was launched on 6 September 2022, by the Ministry of Education, Government of India. Since then the app has gained widespread usage, with 36 States and Union Territories employing PRASHAST. There are approximately 6,13,866+ registered users, 26,26,568+ students profile added on the PRASHAST App. In conclusion, PRASHAST, with its user-friendly approach as well as interface promotes the active participation of regular teachers, special educators, counsellors, school heads, and school management committees in the screening process. Importantly, PRASHAST shields children from "unscientific diagnosis and needless labelling," ensuring their well-being and educational progress.

Introduction & Rationale

In India, education goes beyond being a basic right, and focuses on including all learners, especially those with special needs. Achieving true inclusion in educational settings starts with identifying and diagnosing disabilities in the classroom. Without this, exists the risk of treating everyone the same, overlooking unique abilities and sidelining those who don't fit the predefined standards. Recognizing and supporting diverse capabilities is essential for a forward-looking and transformative education system.(Kaushik,2019).

Disability is generally viewed as a physical or mental condition(s), or both, that limit(s) an individual’s movements, activities and sense perceptions. However despite being a complex and multifaceted concept, understanding and significance of disability conditions varies across different regions, influenced by various legal, political, and social constructs and is still forced to face a lot of discrimination due to prevalent prejudices and biases in society. The discriminatory practices are seen not only in the larger society but they also manifests and translates itself to the classroom and hinders in getting quality education , as noted by the United Nations Convention on the Rights of Persons with Disabilities (UNCRPD), which advocates for promoting and safeguarding the human rights and fundamental freedoms of individuals with disabilities on an equal basis with others. The guiding principles of the UNCRPD include respect for individual autonomy, non-discrimination, full societal inclusion, and equal opportunity. One of the primary emphasis is on the integration of disability issues into sustainable development strategies.

The concerns associated with disabilities should be appropriately addressed, and individuals with disabilities should be supported to realise their full potential within society. This may not be possible without early identification which will serve as the foundation upon which subsequent efforts would be built. Without a clear understanding of the disabilities faced by individuals , it becomes challenging to develop tailored and effective strategies to address their specific needs.

In the context of India, the identification of disabilities is not only essential but also legally mandated. The enactment of the Rights of Persons with Disabilities (RPwD) Act in 2016 marked a significant shift from the earlier Persons with Disabilities (PwD) Act of 1995. The newer legislation, recognizing 21 disability conditions, reflects alignment with the United

Nations Conventions on the Rights of Persons with Disabilities (UNCRPD). This legal framework not only establishes the rights of individuals with disabilities but also serves to create awareness, promote equity-based inclusion, and provide a comprehensive basis for addressing their unique needs.

Moreover, the Right to Education (RTE) Act of 2009 ensures free and compulsory education for all children, including Children With Special Needs (CwSN). Furthermore National Education Policy, 2020 also recognizes the importance of creating enabling mechanisms for children With Special Needs (CwSN) or *Divyang*, that ensures the same opportunities of obtaining quality education as any other child. In this context ,quality education is seen as a catalyst for enhancing upward mobility in the society. Whereas the more recent National Curriculum Framework for School Education (NCF) in 2023 heralded as a transformative phase in educational reforms, are resolutely committed to fostering an educational system that is fair, accessible, and attuned to the diverse needs of all learners, irrespective of their backgrounds or abilities. One common thread amongst these legislative overlaps implies a dual responsibility: early identification of children with disabilities and the implementation of tailored educational interventions to facilitate their holistic development within the societal framework.

Navigating the landscape of inclusive education in India is a complex endeavour due to the country's expansive geography and vast population. This complexity is particularly pronounced in the identification and support of individuals with disabilities, particularly among school-going children. Within the educational framework, key stakeholders such as School Principals, Teachers, and Special Educators each play distinct roles, yet the challenge lies in establishing a cohesive network that seamlessly integrates their efforts, towards early identification, certification and offering the customised interventions for realising the optimum potential of Children with Special Needs (CwSN) In this backdrop, with the aim to promote coordination amongst various stakeholders of the school education towards early identification ,the Central Institute of Educational Technology (CIET), which is a constituent unit of the National Council of Educational Research and Training (NCERT), has taken a proactive measure. CIET, as the national authority responsible for educational technological innovation and research in India, has introduced an innovative solution facilitating screening of disability conditions at the school level, called "PRASHAST-Pre-Assessment Holistic Screening Tool”. The PRASHAST facilitates collaboration among stakeholders of school education in the process of disability screening at the level of school in accordance with the provisions of Samagra Shiksha and RPwD Act 2016.

PRASHAST recognizes that teachers play a pivotal role in the lives of students, PRASHAST empowers them to be in the frontline in identifying potential disabilities. As the tool is user-friendly and written in clear language, it enables regular teachers to observe and recognize behavioural manifestations associated with the 21 disabilities outlined in the RPwD Act 2016.

This application is based on the screening checklist specifically designed to facilitate the school level screening of 21 disabilities recognized in the RPwD Act 2016. Part 1 of the PRASHAST checklist is intended for completion by regular teachers, who may not possess in-depth knowledge of disabilities, as their prime responsibility is to deliver the curriculum. Part 2, on the other hand, requires vetting of observations of part 1 by the special educators who are expected to have better knowledge of disabilities. The screening results of the PRASHAST provide requisite information for further assessment, identification and certification by a panel of experts, in line with the recommendations of Samagra Shiksha.

Development of PRASHAST began with gathering information and insights regarding the process adopted by all the States and Union Territories (UTs) of India for screening, assessment, identification , and certification of disabilities. This revealed that 52.9% of states faced difficulties in collecting required information from the schools due to lack of disability knowledge among the regular teachers and also due to the overlapping characteristics and symptoms of certain disabilities such as Mental Illness, Speech and Language Disability, Learning Disabilities, Haemophilia, and Thalassemia. Further, 76.4% of states actively engage special educators and medical experts in the identification process of Children with Special Needs (CwSN). hence, didn’t face any challenge or confusion in the process of identification of disabilities. Lack of awareness and knowledge about the various disability conditions among regular teachers emerged as the major reason for not involving them in the identification process. Eight states—Sikkim, Tamil Nadu, Kerala, Assam, Chhattisgarh, Puducherry, Delhi, and Uttar Pradesh—were found to have either finalised or are in the process of finalising their own checklists for identifying CwSN. Most of these states, however, were not considering all the 21 disabilities recognized in RPwD Act 2016. Nonetheless, the absence of centralised identification and screening tools emerged as a significant barrier to the prompt identification of CWSN.

Objectives of the Study

- To develop a screening checklist for 21 disability conditions legally recognized in India.
- To standardise the screening checklist developed.
- To leverage the ICT solution (Android App) for dissemination of the screening checklist.

Key Terms

A. **SAMAGRA SHIKSHA(2019):** Samagra Shiksha is an all-encompassing program for the school education sector to improve school effectiveness as evaluated by equal opportunities for schooling and equitable learning outcomes that spans from preschool to class 12.

B. **Children with Special Needs (CwSN):**Children who are born with some kind of disability need considerable support and care in order to grow and thrive.

C. **Special Educators :** An individual who is trained in interacting with differently-abled children in a way that takes into consideration each child's unique needs and talents.

D. **Regular Teachers:** An individual who consistently performs their job, teaches regularly based on the number of hours they are assigned, and assists the students consistently.

E. The Disabilities mentioned in the RPWD Act, 2016 are as follows: Blindness, Low-vision, Leprosy Cured persons, Hearing Impairment (deaf and hard of hearing), Locomotor Disability, Dwarfism, Intellectual Disability, Mental Illness, Autism Spectrum Disorder, Cerebral Palsy, Muscular Dystrophy, Chronic Neurological conditions, Specific Learning Disabilities, Multiple Sclerosis, Speech and Language disability, Thalassemia, Hemophilia, Sickle Cell disease, Multiple Disabilities including deafblindness, Acid Attack victim, and Parkinson's disease.

Procedure and Development of PRASHAST

PRASHAST seeks to enhance the involvement of regular teachers irrespective of their background and expertise in disability studies, in systematically screening students for potential disability conditions within mainstream schools, considering the school as the primary unit for data collection.

The steps mentioned below were followed in the development of PRASHAST:

1. ***Status Study Survey*** was conducted in June 2019, receiving responses from 34 out of 36 states and union territories, revealing that only 8 states had initiated the development of checklists for identifying disabilities.

2. ***In-house drafting*** took place in July-August 2019.

3. Vetting and Finalization Workshop in August 2019 with input from various experts.

4. ***Field testing*** occurred in 29 schools across three states, demonstrating the draft's user-friendliness and usability across different cultural contexts.

5. ***Statistical Standardisation*** Establishing the psychometric properties, including validity and reliability, were established through administering the tool following the procedure finalised in the previous step, in the UT of Chandigarh, wherein the data from 3482 students, enrolled in 100 schools were collected. PRASHAST has good internal consistency as shown by the Cronbach Alpha is 0.801, which is within the acceptable range.

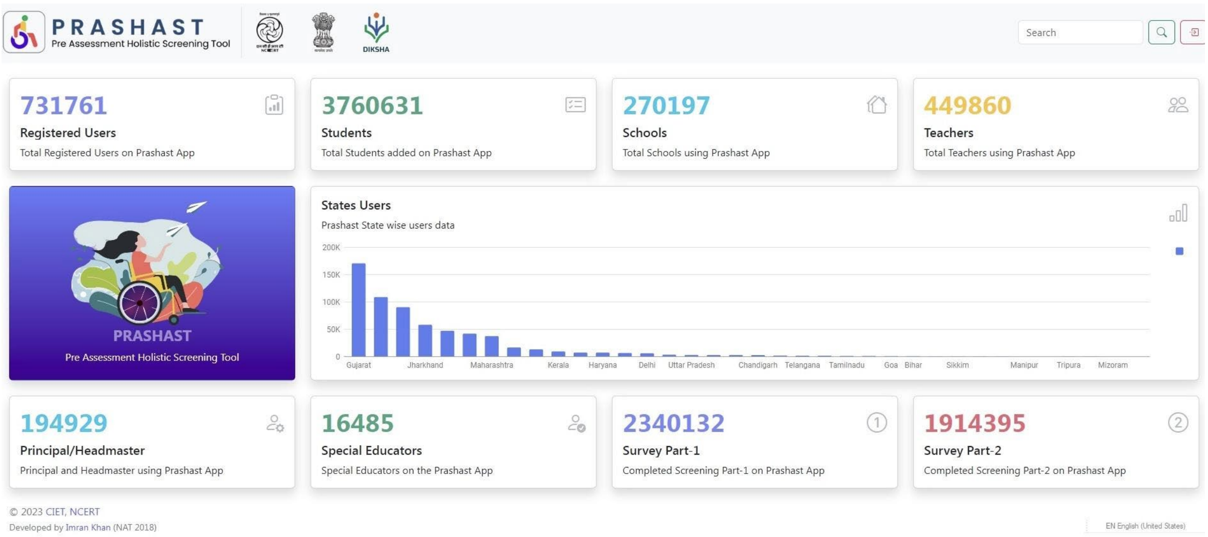
6. ***Language editing*** was done by professional language editors during its stages of development and finalisation. .

7. ***External vetting*** was done by national and international experts, in November 2021 which re-emphasised the importance of PRASHAST as a preliminary screening tool, discouraging its use for formal diagnosis. The division into sections for specialist and less-experienced teachers was commended, and the purpose of PRASHAST was recognized as a catalyst for professional intervention in identifying and supporting students with potential disabilities.

8. ***Android App –*** potential of ICT was leveraged and an android App was developed for speedy, paperless screening and quick data sharing.The App is available at google play store. The app is available at https://play.google.com/store/apps/details?id=com.dscs.app&hl=en&gl=US

Dissemination and Utilisation

The Ministry of Education (MoE), Government of India plays a pivotal role as the primary agency responsible for disseminating information on the usage of PRASHAST. The Android App was launched by the MoE on 6th September 2022, during Skhikshak Parv (Teacher Festival). Currently, the dissemination of PRASHAST is carried out through a multifaceted approach, including live interaction sessions with experts, online platforms, and specialised in-person demand based workshops for stakeholders at the Central Institute of Educational Technology (CIET),NCERT. The emphasis of PRASHAST dissemination focusses on the in-service teachers, educational administrators and special teachers. At the same time,information is also shared during pre-service teacher education. This aims to provide a comprehensive understanding of PRASHAST's benefits and challenges, ensuring that both preservice and in-service teachers are well-equipped in usage of screening checklists at the level of school. Furthermore, the dissemination extends to key stakeholders, such as principals, special educators, and other professionals who regularly utilise.



Key Features of PRASHAST

- Screening based on the behavioural **manifestation** of the disability conditions in the **classroom**
- Part 1 meant for the usage by regular teachers, comprises of 63 behavioural statements
- Name of disability conditions not shared in the part 1 with regular teachers to avoid unscientific labelling
- Each statement in the part 1 is coded, and linked with a disability condition
- The respondent had to select yes and no against the statement
- Participation of regular teachers irrespective of their training in disability studies
- The part 2 meant for the usage by special teachers comprises a separate checklist of dominant behaviour associated with each of the disability conditions recognized in the RPwD Act 2016. This part also serves as the ready reckoner for the special educators who may have had training in only one of the disability conditions but had to work in the multi-disability conditions.
- Special educator is expected to provide inputs on the part 1 filled by the regular teachers.
- Special educators may select the disability condition based on their detailed observation of the child plus the filled part 1.
- The disability condition screened after part 2 remains tentative, till its further formal assessment.
- Access to regular teachers and special educators only after verification by their respective school principles
- The school principles are auto-verified by the app based on the information provided during registration
- One of its kind standardised screening instrument catering to all 21 disability conditions recognised in the RPwD Act 2016
- The screening data of disabilities can be downloaded by the principle and shared further with the higher authorities engaged in the planning of assessment and certification camps for disabilities.

Benefits

The PRASHAST is a safeguard against *unscientific diagnosis and needless labelling of children.*With the help of PRASHAST, teachers have a safeguard to avoid making hasty and inaccurate conclusions that lack scientific basis. The use of PRASHAST is intended to promote clarity regarding the child's condition. Furthermore, it also helps the teacher, special educators in early recognition of disability conditions, and helps direct the attention of regular teachers towards children with special needs. Additionally, it expedites and ensures timely screening of children whose learning may be impeded by various disabilities, thereby improving the overall quality of school education.

Transparency, saving time, energy, and effort are another few of the overt benefits of screening for disability with the help of PRASHAST.

Challenges

The prime challenge lies in reaching the length and breadth of a vast country like India, which has a statistically huge number of schools (15 lakh), teachers (85 lakh), and students (26 crore). Next on the list is Internet connectivity in the remote areas for proper functioning of the App and training of stakeholders, that is teachers, special teachers and principles, who prefer to work with pen and paper. Besides , many stakeholders working in the school education system have a fear of technology.

Building synergies among the various government agencies working in the field with different purposes but gathering similar data from schools regarding disabilities is another major challenge. For example, the health department conducts health assessments of diseases and disability with the purpose of medication and other assistance. The department of disability affairs and social justice gathers data with the aim of providing need based assistance to different age groups for rehabilitation. Further, the lack of adequate number of human resources available at the school, adds to the list of challenges.

Nonetheless, this is a baby step towards adopting the scientific screening and thereafter following the process for issuing the disability certification.

Way Forward

Despite the legal provisions in place, the ground realities across India reveal instances where children with disabilities remain unidentified, even after gaining admission to schools. This oversight often stems from a lack of awareness among school functionaries and the general population. Notably, disabilities with less visible symptoms, such as mental illness, specific learning disabilities, hearing impairments, or autism, pose particular challenges in identification without specialised training.

At present the screening data is gathered at the school level which is further shared by the school principals with the higher authorities at cluster or block level for data-based planning. There is a need to develop an ecosystem around PRASHAST for easy access of data by the stakeholders at state and centre level. This will enhance the organisation of assessment and identification camps, distribution of aids and equipments', tracking of academic progress etc leading to more effective implementation of Inclusive Education provisions of Samagra Shiksha. The ecosystem needs to incorporate following–

- Role of educational administrators at the level above school
- Refined data extraction
- Tracking of the provisions in line with the Inclusive Education component of Samgara Shiksha
- Comparative performance assessment of states both at inter-state and intra-state levels

References

*Aysha(2021)*Does India have enough special teachers? Supreme Court asks,*India Today*

https://www.indiatoday.in/education-today/news/story/does-india-have-enough-special-teachers-supreme-court-asks-1870970-2021-10-29

Government of India. (2020). *National Education Policy 2020*. Ministry of Education. https://www.education.gov.in/sites/upload_files/mhrd/files/NEP_Final_English_0.pdf

Government of India. (2009). The Right of Children to Free and Compulsory Education (RTE) Act. New Delhi: Ministry of Human Resource Development. https://www.education.gov.in/sites/upload_files/mhrd/files/upload_document/rte.pdf

Government of India. (2016). Rights Of Persons With Disabilities Act. New Delhi: Ministry Of Law, Justice and Company Affairs, December 27, 2016. https://www.indiacode.nic.in/bitstream/123456789/15939/1/the_rights_of_persons_with_disabilities_act%2C_2016.pdf

United Nations. (2006). Convention on the Rights of Persons with Disabilities. United Nations General Assembly Session 61 Resolution 106. https://www.un.org/disabilities/documents/convention/convoptprot-e.pdf

Government of India.(2023)*National Curriculum Framework for School Education* .Ministry of Education.https://ncert.nic.in/pdf/NCFSE-2023-August_2023.pdf

Government of India(2019)*Samagra Shiksha Scheme* .Ministry of Education https://samagra.education.gov.in/

Kaushik, B. (Ed.). (2019). Creating Inclusive Schools: Theory, Process and Practice. SAGE Publications India Pvt Limited.

Website Links:

PRASHAST App Link:

https://play.google.com/store/apps/details?id=com.dscs.app&hl=en&gl=US

PRASHAST Booklet

https://ncert.nic.in/pdf/DSCS_booklet.pdf

Orientation to PRASHAST

https://www.youtube.com/live/HMknw1K6SnU?si=AJXnJ1gznmFXDPXH

Central Institute of Educational Technology

https://ciet.nic.in/

VisPhoto: Photography for People with Visual Impairments via Omnidirectional Imaging Post-Production*

Naoki Hirabayashi[1], Masakazu Iwamura[2], Zheng Cheng[1], Kazunori Minatani[3], and Koichi Kise[2]

[1] Graduate School of Engineering, Osaka Prefecture University
pepepor123@gmail.com, zheng386@hotmail.com
[2] Graduate School of Informatics, Osaka Metropolitan University
masa.i@omu.ac.jp, kise@omu.ac.jp
[3] National Center for University Entrance Examinations
minatani@rd.dnc.ac.jp

Abstract

Many people with visual impairments would like to take photographs. However, they often have difficulty pointing the camera at the target. In this paper, we address this problem by proposing a novel photo-taking system called VisPhoto. Unlike conventional methods, VisPhoto *generates a photograph* in post-production. When the shutter button is pressed, VisPhoto captures an omnidirectional camera image that contains the surrounding scene of the camera. In post-production, the system outputs a cropped region as a "photograph" that satisfies the user's preference.

1 Introduction

Many sighted people may be unaware that people with visual impairments (PVI) take photographs. Their purposes vary widely. With the widespread use of applications to assist PVI, such as Seeing AI and TapTapSee, they have more opportunities to use a camera at the present time. In addition to using apps, they want to take photographs to record their memories and share them with their sighted friends, in some cases through social networking services, in the same manner as sighted people [3–5, 8, 10–12]. According to an online survey of 118 PVI on why they use cameras, 61.9% of those who had recently used a camera said that it was for friends, family, trips, or fun [5].

PVI encounter great challenges when taking photographs. Harada et al. raised the issues of framing, lighting, and focus [4]. Balata et al. focused on composition [2]. However, PVI have an even more fundamental problem: how to include the target in the camera frame. To address this problem, much effort has been made to develop real-time applications. The common approach is to provide users with audio feedback, such as "left" and "right," and vibration feedback that helps them to point the camera in the direction of the target [2, 5–7, 9]. However, these applications require time and effort from users. Therefore, these applications can only be used when the user has sufficient time to aim at the target and the target is static or moving sufficiently slowly.

In this paper, we address the aforementioned problem by proposing a novel photography support system for PVI called VisPhoto (See Fig. 1 for more detail). The first key idea is to use an omnidirectional camera to take a photograph. An omnidirectional camera is equipped with two fish-eye lenses to capture a 360-degree view. It allows the user not to take time to

*This is a digest version of our paper presented in ASSETS2023. The full information is available at the project page at https://opu-imp.github.io/VisPhoto/.

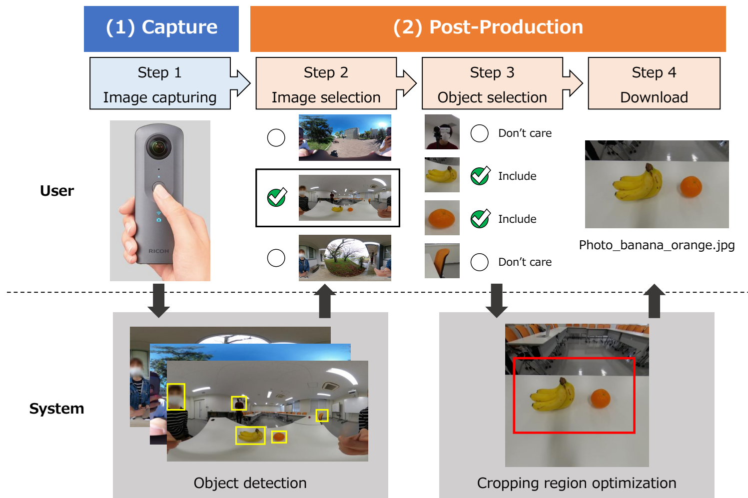
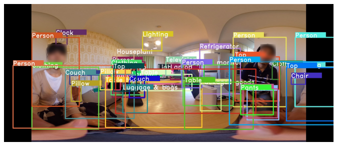


Figure 1: Overview of the proposed system. The procedure that generates a photograph is divided into two stages: (1) capture and (2) post-production. In (1), the user captures an omnidirectional camera image by pressing the shutter button. Unlike standard photography, the process is not complete until the photograph is cropped according to the user's preference in (2). In (2), we create two types of interfaces to enable users to generate a photograph: a manual method in which the user manually performs post-production on web pages and an automatic post-production method in which speech recognition allows the user to skip some manual operations.

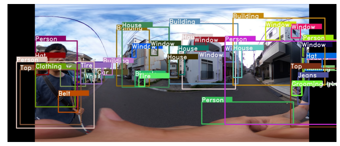
aim the camera at the target. However, the captured image (i.e., the omnidirectional camera image) includes many objects and a large amount of background that the user does not want to include in the photograph. Therefore, as the second key idea, in post-production, VisPhoto *generates a photograph* by outputting the cropped region that the user wants to include in the photograph. To help the user to determine which region to crop, our system applies an object detection technique to the image and provides the user with information on which objects exist in which direction. The advantages of the proposed system are as follows:

1. The user is free from the process of aiming at the target.
2. The skew of the camera can be automatically corrected. Hence, the user does not have to consider the camera angle when taking a photograph.
3. In principle, the user can take a photograph of a moving object, which is not easy using the current mainstream approach [2, 5–7, 9].
4. For accurate object detection, VisPhoto can run a computationally heavy process, such as a cloud service, even if it takes a long time. The mainstream approach needs to send feedback to the user in real time. Hence, accuracy is often sacrificed to achieve real-time processing. However, VisPhoto is free from such a constraint.

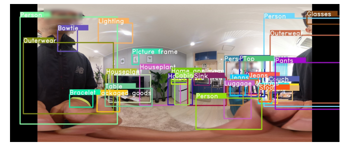
By contrast, generating a photograph is postponed until post-production, which forces the user to remember the target and scenario from the moment of taking the photograph until post-production. This can be regarded as a disadvantage because it increases the user's cognitive



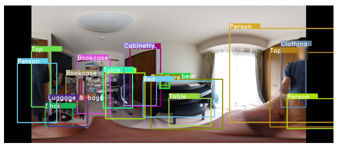
(a) Photographed by a 26-year-old female.



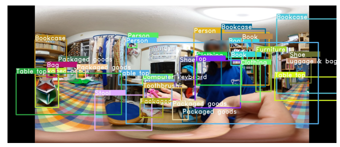
(b) Photographed by a 47-year-old male.



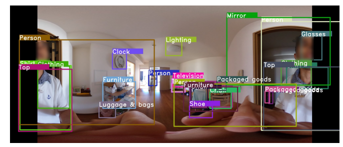
(c) Photographed by a 29-year-old male.



(d) Photographed by a 47-year-old male.



(e) Photographed by a 23-year-old female.



(f) Photographed by a 29-year-old male.

Figure 2: [Demonstration] Selected omnidirectional camera images photographed in the real environment by blind participants. The bounding boxes represent object detection results.

load. To compensate for this, we implement audio recording functionality to record a voice memo similar to that in [1, 4]. The recorded audio is provided in post-production to remind the user of the photographed scenario, along with the date and time the user photographed it. To further reduce the user's load, we automate the most laborious part of post-production.

We conducted a user study for 24 PVI to evaluate VisPhoto. We asked the participants to take photographs with a genuine iPhone camera app, a conventional method, and VisPhoto. After the study, 20 sighted people evaluated and compared the quality of the photographs.

2 Demonstration

We show omnidirectional camera images photographed in the real environment by eight blind participants. Fig. 2 shows six photographs selected from 30 photographs taken in total for demonstration purposes. The photographs show that blind people can use VisPhoto even in the real environment.

Acknowledgments

This work is financially supported by The Telecommunication Advancement Foundation and JST Kakenhi 20H04212.

References

[1] Dustin Adams, Sri Kurniawan, Cynthia Herrera, Veronica Kang, and Natalie Friedman. 2016. Blind Photographers and VizSnap. In *Proceedings of the 18th International ACM SIGACCESS Conference on Computers and Accessibility - ASSETS '16*. ACM Press. https://doi.org/10.1145/2982142.2982169

[2] Jan Balata, Zdenek Mikovec, and Lukas Neoproud. 2015. BlindCamera: Central and Golden-ratio Composition for Blind Photographers. In *Proceedings of the Mulitimedia,*

Interaction, Design and Innnovation on ZZZ - MIDI '15. ACM Press. https://doi.org/10.1145/2814464.2814472

[3] Cynthia L. Bennett, Jane E, Martez E. Mott, Edward Cutrell, and Meredith Ringel Morris. 2018. How Teens with Visual Impairments Take, Edit, and Share Photos on Social Media. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems - CHI '18* (2018-04). ACM Press. https://doi.org/10.1145/3173574.3173650

[4] Susumu Harada, Daisuke Sato, Dustin W. Adams, Sri Kurniawan, Hironobu Takagi, and Chieko Asakawa. 2013. Accessible Photo Album: Enhancing the Photo Sharing Experience for People with Visual Impairment. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems - CHI '13*. ACM Press. https://doi.org/10.1145/2470654.2481292

[5] Chandrika Jayant, Hanjie Ji, Samuel White, and Jeffrey P. Bigham. 2011. Supporting Blind Photography. In *The proceedings of the 13th international ACM SIGACCESS conference on Computers and accessibility - ASSETS '11*. ACM Press. https://doi.org/10.1145/2049536.2049573

[6] Marynel Vázquez and Aaron Steinfeld. 2012. Helping Visually Impaired Users Properly Aim A Camera. In *Proceedings of the 14th International ACM SIGACCESS Conference on Computers and Accessibility - ASSETS '12*. ACM Press. https://doi.org/10.1145/2384916.2384934

[7] Marynel Vázquez and Aaron Steinfeld. 2014. An Assisted Photography Framework to Help Visually Impaired Users Properly Aim a Camera. *ACM Transactions on Computer-Human Interaction* 21, 5 (Nov. 2014), 1–29. https://doi.org/10.1145/2651380

[8] Violeta Voykinska, Shiri Azenkot, Shaomei Wu, and Gilly Leshed. 2016. How Blind People Interact with Visual Content on Social Networking Services. In *Proceedings of the 19th ACM Conference on Computer-Supported Cooperative Work & Social Computing - CSCW '16*. ACM Press. https://doi.org/10.1145/2818048.2820013

[9] Samuel White, Hanjie Ji, and Jeffrey P. Bigham. 2010. EasySnap: Real-Time Audio Feedback for Blind Photography. In *Adjunct proceedings of the 23nd annual ACM symposium on User interface software and technology - UIST '10*. ACM Press. https://doi.org/10.1145/1866218.1866244

[10] Shaomei Wu, Jeffrey Wieland, Omid Farivar, and Julie Schiller. [n. d.]. Automatic Alt-Text: Computer-Generated Image Descriptions for Blind Users on a Social Network Service. In *Proceedings of the 2017 ACM Conference on Computer Supported Cooperative Work and Social Computing* (2017-02). ACM. https://doi.org/10.1145/2998181.2998364

[11] Yuhang Zhao, Shaomei Wu, Lindsay Reynolds, and Shiri Azenkot. 2017. The Effect of Computer-Generated Descriptions on Photo-Sharing Experiences of People with Visual Impairments. *Proceedings of the ACM on Human-Computer Interaction* 1, CSCW (dec 2017), 1–22. https://doi.org/10.1145/3134756

[12] Yuhang Zhao, Shaomei Wu, Lindsay Reynolds, and Shiri Azenkot. 2018. A Face Recognition Application for People with Visual Impairments. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems - CHI '18*. ACM Press. https://doi.org/10.1145/3173574.3173789

Employing Theorem Prover Technology to Develop an Accessible and Inclusive Mathematical Learning Environment

Bernhard Stöger[1], Klaus Miesenberger[1], and Walther Neuper[2]

[1] Johannes Kepler Universität Linz {bernhard.stoeger,klaus.miesenberger}@jku.at
[2] RISC Linz walther.neuper@jku.at

Abstract

A novel approach to design an inclusive and accessible mathematical learning environment is presented: The technology of theorem proving shall be employed to support a student in solving mathematical problems by giving hints to him/her based on formal proofs of each step in the process of solving a problem. The user will get access to a rich knowledge base of definitions, background and context information. As a customizable front-end, the system will allow educators to calibrate and personalise the level of support in the problem solving process. The system shall be made accessible by making use of the built-in accessibility coming with VSCode, a standard editor used as an experimental front-end for the theorem prover Isabelle.

1 Introduction

Although long-year research and development efforts have been carried out to enable blind and visually impaired people to learn and do mathematics effectively through information technology, until now no system exists which would be widely used for step-wise problem solving, actually one of the central activities in mathematics education at school and in academic engineering faculties. There do exist some partial solutions to the complex challenge of supporting blind individuals in mathematical problem solving:

- Standard screen readers such as *NVDA*[1] and *Jaws*[2] are able to read mathematical contents presented in MathML or written with MathJax. Readeing is done by synthetic speech or by Braille output in *Nemeth Code*[3]. They even support collapse/expand of a formula, but the underlying tree structure is based upon the presentation of a term, not upon its semantics.
- Math editors such as *MathType*[4] or *ChattyInfty*[5][18, 17] let a blind person edit mathematical contents by supporting LaTeX syntax and several Braille mathematics notations. However, they do not grant any help in organizing one's calculations, nor do they provide support in solving mathematical problems.
- One first step towards support in doing mathematics was carried out in a PHD Thesis by *Prajaks Jitngernmadan*[7, 3, 5, 6, 4]. The thesis investigated the question whether assistants or wizards might support a pupil in doing elementary tasks in arithmetic.

[1] https://www.nvaccess.org/
[2] https://www.freedomscientific.com/
[3] https://nfb.org/images/nfb/documents/pdf/nemeth_1972.pdf
[4] https://www.wiris.com/en/mathtype/
[5] https://www.sciaccess.net/en/ChattyInfty/

> By an assistant or wizard, a software program is meant which guides the user through the various steps of a calculation, giving him/her instructions and hints what to do in a particular step. The thesis showed that indeed a pupil could be supported by such a wizard, the problem being that the software granted too much support, in that the pupil's ability to work on his/her own was not sufficiently trained.

The present paper describes current efforts to bring new energy and inspiration into the process of developing a comprehensive solution to support blind individuals in mathematical problem solving by following a completely novel idea: A theorem prover shall be used to support a person learning and doing mathematics by assisting him/her through hints given by the prover's approach to a mathematical problem and shall give information about the formal correctness of the actions. By such an approach, the student will remain creative in solving a problem, but he or she will get confirmation and advice on his/her mathematical activity.

The accessibility of the environment to be designed will be decisively supported by the accessibility of VSCode, a front-of-the wave IDE with open source featured by Microsoft which is an experimental front-end for Isabelle [16], the theorem prover intended as the core of the learning environment. VSCode itself is well accessible because it rests upon the Chromium library, which is able to furnish enough accessibility to be the basis for Google Chrome, a web browser used by many blind people worldwide.

Already now, the Isabelle theorem prover furnishes valuable support to people working in mathematics, be they blind, visually impaired, or sighted. This support is due to the fact that, being a theorem proving system, Isabelle comes with a wealth of mathematical concepts with precise formal definitions, and theorems derived from them, with formally correct and well readable proofs. Therefore, with the Isabelle library we have a database of present mathematical knowledge which is already now accessible to everyone.

This paper outlines ideas, plans and preparatory work which emerge from joint efforts of three partners, where every two of them were collaborating pairwise for a long time, but never all three of them. One partner is from the community of the proof assistant Isabelle [16], which is promoted by several institutions around the world; another is IIS, the institute at Johannes Kepler University, and the third one is the Isac project.

The collaboration between Isabelle and Isac [9] results from the fact that the latter uses the former as a conceptual and technological base. For some time there are plans to intensify this collaboration [11] for technical reasons. Great advances of Isabelle's front-end suggest to stop developing a proprietary front-end for Isac and to advance Isac's integration into Isabelle such that it can re-use all of its front-ends (and eventually inherit accessibility). And the collaboration between IIS and ISAC concerns IIS's mission to support students with special needs and to research how to make software accessible for those students. In particular, the Isac prototype shall be improved such that it becomes usable for visually impaired and blind people. This goal fits well with Isac's original idea to narrow the gap between high-school mathematics and academia. For that purpose Isac reduces the complexity of proof to [1], that means, to calculations solving problems as taught in academic engineering courses as well as at high-school.

Recently an eureka moment triggered collaboration of all three partners: A member of IIS (and co-author of this paper) is a blind person, nevertheless he holds a degree in a formal science. At his eureka moment he installed Isabelle/VSCode without further ado and then for the first time he was able to interactively investigate a proof of a well-known theorem. This important experience showed us that, when we want to make Isabelle accessible, the right way is to use its VSCode front-end.

2 Accessible and Inclusive Software for Maths Education

The traditional way of using paper and pencil to write down mathematical contents and even more to do mathematical calculations is still the predominant method among all people who actively solve mathematical problems. This is especially remarkable if we consider the tremendous changes modern information technology brought into society over the past few decades. Although computer algebra systems such as Mathematica or Maple are widely used among mathematicians, these systems cannot help a person doing a calculation by hand, since they will do the calculation for him/her. A system where the computer with its ever increasing possibilities to support the human brain would help a person to actually solve a problem in reasonable steps is still missing, not only, as outlined here, for the group of blind or visually impaired people, but for the sighted mainstream as well - one will publish, or distribute, mathematical content in a digital format, but, as soon as one calculates by him/herself, it is generally paper and pencil which is used. Of course the situation is changing step by step and faster due to the growing number of touch/pen based systems including OCR for math, digital math-ink, which will allow developing a new digital culture of interactively doing math including and integrating the interaction with tools as computer algebra, theorem provers and tutoring systems. These new and innovative systems inviting to transfer from paper/pencil to digital doing math provide a much better and much richer source for supporting navigation and in particular doing math for blind and low vision people: What was implicit to the visual procedures and work flows of doing math with pencil and paper now has to be formally and therefore explicitly described in a machine readable and processible manner [5, 4]. This forms a rich digital base, to which accessibility can hopefully be efficiently added and which allows to implement assisting and supportive functionalities better in accordance with the needs of blind and low vision people. In spite of the undisputed advantages of the traditional paper-and-pencil method, it is our conviction that the computer has a strong potential to help in a calculation, be the user sighted, visually impaired, or blind. We thus envision a scenario where methods which apparently are needed by the relatively small group of blind people may as well be of great use and value for all people who are dealing with mathematics. We see this formal mark-up of the procedures of doing math as a first step towards a much more personalized approach to math problem solving which is able to much better respect the broad diversity of users in terms of physical, sensory, psycho-cognitive and social factors.

3 User Requirements

User requirements usually precede detailed design of a software system. However, this paper is backed by a concrete project proposal [10] and builds upon two decades of development in the Isac-project [13, 15, 14]. Thus the requirements do not serve design and development any more here, rather this paper uses them to clarify the novel features of the system.

General requirements about mathematical problem solving:

1. Separate problem solving into “what” to solve and “how” to solve, a long-serving principle in engineering education. Isac models this requirement in a **specify**-phase and a **solve**-phase respectively.
2. If a student gets stuck, the system can **propose a next step**, i.e. a missing element of the specification or the next formula towards a solution [12]). A suggested formula can be presented in several ways:

   (a) the system simply presents the next formula
   (b) the user requests a list of rules and selects an appropriate rule to be applied
   (c) the system suggests a rule requesting the user to provide the resulting formula
   (d) the system suggests the resulting teformularm requesting the user to provide a rule
   (e) the system suggests the resulting formula with gaps and a request to fill-in
   (f) the system suggests a rule with gaps requesting to fill-in
   (g) the system suggests the resulting formula and the rule with gaps requesting to fill-in
   (h) etc.

3. The **kind of problems** covered by Isac are restricted to "constructive proof", where some objects are given (restricted by a so-called "pre-condition") which are sufficient to construct a solution (characterised by a predicate called "post-condition", which relates the given objects and the solution). Solving such problems covers about 90% of mathematics education in academic engineering courses (the accompanying exercises respectively), as well as what is taught as high school mathematics.

4. The **formal specification** of a problem must be comprehensible for students in an academic course in engineering disciplines (and also at high school) without any instruction in formal logic (concerning $\exists$, $\forall$, $\land$, $\lor$ etc). So a *Specification* is given by *Given* (the given objects), *Where* (the pre-condition), *Find* (the solution to be constructed) and *Relate* (simple parts of the post-condition).

5. In general, mathematical problems can be solved in many ways. Thus a *Specification* must be **open for variants** which are handled within one and the same process of problem solving. The running example in §4 below illustrates two variants.

6. The **specify-phase can be skipped** on predefined settings. This requirement is more important than it seems: Isac may be not only used for solving new problems, but also for exercising *known* problems, for instance differentiation. In that case *Specification* will be folded in, such that problem solving immediately will be started by the solve-phase, i.e. by work within *Solution*.

7. A *Specification* as well as a *Solution* **can be completed** automatically (by pushing a button or the like), in case the course designer allows to do so by the predefined settings (see the remark an a dialog module below).

8. In the specify-phase there is the following **feedback on input of a formula**:
   (a) **Correct**: There is probably no specific indication.
   (b) **Syntax**: This feedback can be necessary, if the user switches the *Model* suddenly from *Problem_Ref* to *Method_Ref* or vice-versa.
   (c) **Incomplete**: Input is incomplete, in particular elements of type list.
   (d) **Superfluous**: An input can be syntactically correct, but have no evident relation to the problem at hand.

   In the solve-phase a formula input by the user is either correct, false or a syntax error; but there is promising research on "error patterns"[2]

These requirements address a system, where all of problem solving is covered by one application and the student needs not switching between worksheet, formulary, problem statement, textbook, etc – the main obstacle of blind students to catch up with sighted classmates in math classes. Several of the user requirements implicitly postulate the software requirement of a **dialog module**, which manages all the complex interactions with data from a user model according to many predefined settings.

Specific requirements for accessibility add to the above user requirements.

1. **Decomposition of terms into proper sub-terms** allows the user to break down complicated formulas along the term structure to sub-terms of reasonable size. [8] showed by experiments, that efficient navigation through sub-terms and super-terms *can* be realised via keyboard and Braille display — which supports understanding the structure of a term quite successfully.

2. **Representation of terms on Braille and on screen is related**. While the first requirement above is more or less accomplished by MathML already, inclusion of sighted(!) students in collaborative learning (and also support by a teacher!) requires, that a sub-term addressed by the blind classmate on Braille is also addressed on screen (by colour, by a box, etc).

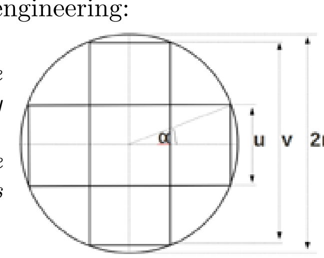
3. **Support for survey on structures**. For comprehension, arbitrary switching between detail and survey is crucial. Fortunately almost all mathematical structures, namely terms, specification, solutions, etc, are trees. So one single standard must be available for all these structures.

4. **Different Braille codes**, predefined by user settings, and other requirements are self-evident and not mentioned here.

4 Case Study: Solving a Complex Mathematical Problem

In the sequel an example is presented, which is already present in the Isac prototype and which will serve to illustrate the user requirements introduced in §3 above. The following text combined with a figure can be found in textbooks for electrical engineering:

The efficiency of an electrical coil depends on the mass of the kernel. Given is a kernel with cross-sectional area determined by two rectangles of same shape as shown in the figure.
Given a radius $r = 7$, determine the length u and width v of the rectangles such that the cross-sectional area A (and thus the mass of the kernel) is a maximum.



This problem will be used for demonstrating novel interactivity in step-wise construction of a solution. One point with this example is, that it can be solved in several variants, see the subsequent paragraph.

"Next-Step Guidance" from hidden knowledge: User requirement No.2 challenges the system to propose a next step (there is no decision procedure to solve all mathematical problems). So there must be some knowledge prepared by an author. For instance, the running example requires the following "formalisation" in order to provide "Next-Step Guidance":

$$F_I \equiv [\,[r=7],\ [A\ [u\ v]],\ [A = 2\cdot u\cdot v - u^2,\ (\tfrac{u}{2})^2 + (\tfrac{v}{2})^2 = r^2],\ \{0 < .. < r\}\,]$$
$$F_{II} \equiv [\,[r=7],\ [A\ \alpha],\ [A = 2\cdot u\cdot v - u^2,\ \tfrac{u}{2} = r\sin\alpha,\ \tfrac{v}{2} = r\cos\alpha],\ \{0 < .. < \tfrac{\pi}{2}\}\,]$$

Freedom of input and variants in problem solving: Input to a *Specification* is not only supported by hints like for *SideConditions* below, all algebraically equivalent formulas are accepted as input, for instance *Extremum* $-u^2+2\cdot u\cdot v = A$ in the field *Relate* in Fig.1 below, too. As already mentioned, the running example can not only be solved by the law of Pythagoras

```
Example "Diff_App-No.123a"
  Specification:
    Model:
      Given: ‹Constants [r = 7]›
      Where: ‹0 < r›
      Find:  ‹Maximum A› ‹AdditionalValues [u, v]›
      Relate: ‹Extremum (A = 2 * u * v - u ↑ 2)›
        ‹SideConditions [__ = __, __ = __]›
    References:
      Theory_Ref: "Diff_App"
   (*⊗*) Problem_Ref: "univariate_calculus/Optimisation"
   (*⊙*) Method_Ref: "Optimisation/by_univariate_calculus"
(*Solution:*)
```

Figure 1: *Specification* almost finished. "[__ = __, __ = __]" indicates a list of equations.

as *SideConditions* $[(\frac{u}{2})^2 + (\frac{v}{2})^2 = r^2]$. Looking at the angle α in the figure of the problem statement on p.5, one also finds two other appropriate *SideConditions* $[\frac{v}{2} = \sin\alpha,\ \frac{u}{2} = \cos\alpha]$.

Specify knowledge for solving a problem: "Next-Step Guidance" requires not only a "formalisation", but also a specific type of problem (referenced by *Problem_Ref: "univariate_calculus/Optimisation"*), a method (referenced by *Method_Ref:*) for solving the problem at hand and also a theory, which defines all the notions involved in a calculation. All this knowledge needs authoring – but then one gets "Next-Step Guidance" with little efforts, see the respective paragraph above.

At least for advanced students the *References* into the knowledge should be transparent. For beginners the *user model* can be preset to collapse the *References* or even the *Specification* as a whole (and the specify-phase is accomplished automatically behind the scenes).

Interactivity and feedback: Fig.2 shows a sub-problem solving an intermediate step in the calculation of *Maximum A*, the step to make one of the two variables $u,\ v$ explicit [6]. Given the equation as $\frac{u^2}{4} + \frac{r^2}{-1} + \frac{1}{4}\cdot u^2 = 0$ as shown in Fig.2, let us assume, that the student gets stuck and asks for help. Then the system (compare requirement Nr.2) can answer with

- presenting a next formula $\frac{1}{4}\cdot u^2 = 0 - (\frac{r^2}{-1} + \frac{u^2}{4})$ frankly
- suggesting the next formula partially and request the student to complete $\frac{1}{4}\cdot u^2 = 0 - (\frac{r^2}{-1}\dots)$
- suggesting a rule to apply, e.g. *rule* $(a+b=c) \;=\; (b=c-a)$
- suggesting a rule together with a partial formula, e.g. *rule* $(a+b=c) \;=\; (b=c-a)$ and $\frac{1}{4}\cdot u^2 = 0 - \dots$
- presenting the formula $\frac{1}{4}\cdot u^2 = 0 - (\frac{r^2}{-1} + \frac{u^2}{4})$ and asking for the rule applied
- etc

[6] Standard equation solving first simplifies to the pattern "__ = 0" in order to determine the degree of an equation.

The abundance of possibilities for interactions particularly calls for a strong dialog module and a detailed user model – and for possibilities of a course designer to prepare setting appropriate for the level of the students.

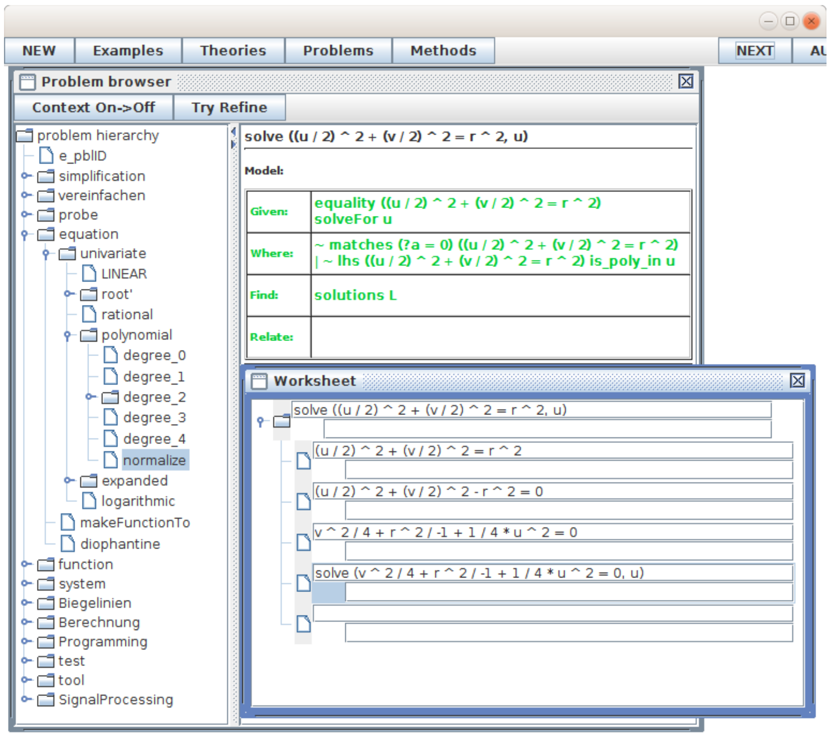


Figure 2: Subproblem of the *Solution*: equation solving.

Feedback in the solve-phase is restricted to correct / incorrent, whereas in the specify-phase it is more detailed (see requirement No.8).

Iterate between specifying and solving The above Fig.2 shows a sub-problem, which occurs while constructing a solution for *Maximum A* – solving the overall problem is broken down into simpler sub-problems. So solving more complex problems, like the one shown here, iterates between specifying (sub-)problem and constructing solutions for them [7].

5 Summary and Conclusions

This paper presents a derivative of a project proposal A-I-MAWEN for an "**A**ccessible and **I**nclusive **M**athematical **W**orking **En**vironment", where the presented system shall act as a demonstrator for a new kind of formula editor. The system's features have almost fully been realised in the Isac prototype, but the respective front-end, implemented in Java Swing, did not meet accessibility requirements. Meeting such requirements (and developing several novel additional features) are expected by adaptation of Isabelle/VSCode. While the rich language

[7]emphSpecification of an equation mainly involves determining the type of equation, its degree etc.

for proofs in Isabelle is just downgraded to structured derivations by Isac (a format close to what is written by hand in a notebook during problem solving) — a major challenge with the envisaged A-I-MAWEN system is, however, to locate a dialog module somewhere in the architecture of Isabelle/VSCode – which is document-oriented rather than dialog oriented.

The paper nourishes the longstanding hope for a system that supports stepwise problem solving in an electronic notebook with reliable feedback. This includes the expectation, that part of learning mathematics can be done similar to learning chess games, going back a few moves and try other variants.

Since mathematics is the science of reasoning, the paper argues, that the basis for respective computer support must be reasoning technology as given by proof assistants. These are highly elaborated software products, which postulate a completely different interaction design in comparison to users originally targeted by proof assistants, experts in computer mathematics. The paper gives an extensive case study for such interaction in mathematics education.

Thus the plan to provide proof assistants with a novel interaction design comes to meet recent developments in browser technology. Browsers were the first software devices made optimally accessible, and by now some browsers have become generic software components such, that several proof assistants started to adopt VSCode as their standard front-end. This technology seems to promise enabling to integrate much of the ideas and tools developed in the accessibility community since decades step by step towards inclusive environments, where sighted and blind students receive lessons together without need for further social segregation.

References

[1] Back, R.J.: Structured derivations as a unified proof style for teaching mathematics. TUCS Technical Report 949, TUCS - Turku Centre for Computer Science, Turku, Finland (July 2009)

[2] Daróczy, G.: Cognitive Aspects of Designing Dialogues in Theorem-Prover Based Mathematics Assistants; Implementation of Error-Patterns Guiding Dialogues in Isac. Master's thesis, ME:CogSci, Vienna, Austria (2013), http://www.ist.tugraz.at/projects/isac/publ/gdaroczy-ma.pdf

[3] Jitngernmadan, P., Miesenberger, K.: A comparative study on java technologies for focus and cursor handling in accessible dynamic interactions. In: Sikne Lanyi, C. (ed.) Proceedings of AAATE. IOS Press, Amsterdam, Budapest (2015)

[4] Jitngernmadan, P., Miesenberger, K.: Extraction methodology of implicit didactics in math schoolbooks for the blind. In: Cudd, P., De Witte, L. (eds.) Proceedings AAATE Conference 2017. Harnessing the Power of Technology to Improve Lives. IOS Press, Amsterdam (2017)

[5] Jitngernmadan, P., Petz, A., Stöger, B., Miesenberger, K.: Analysis of implicit didactics in math schoolbooks for interactive non-visual user interface development. In: Miesenberger, K., Bühler, C., Penaz, P. (eds.) Computers Helping People with Special Needs, 15th International Conference, ICCHP 2016. Springer Heidelberg, Linz Austria (July 13-15 2016)

[6] Jitngernmadan, P., Stöger, B., Petz, A., Miesenberger, K.: IDMILE: An interactive didactic math inclusion learning environment for blind students. Technology and Disability **29**(1-2), 47–61 (2017)

[7] Jitngernmadan, P.: A Framework of Support Functionalities for Multimodal Interface Design for Mathematic Working Environment. Ph.D. thesis, Johannes Kepler University Linz, Austria (2017)

[8] Karl, N.: Developing an Inclusive Approach for Representing Mathematical Formulas. Master's thesis, Hagenberg University of Applied Sciences, Linz, Austria (2016), https://static.miraheze.org/isacwiki/0/02/Masterthesis_NatalieKarl.pdf

[9] Krempler, A., Neuper, W.: Prototyping "systems that explain themselves" for education. In: Quaresma, P., Neuper, W. (eds.) Proceedings 6th International Workshop on Theorem proving components for Educational software, Gothenburg, Sweden, 6 Aug 2017. Electronic Proceedings in Theoretical Computer Science, vol. 267, pp. 89–107. Open Publishing Association (2018). https://doi.org/10.4204/EPTCS.267.6, https://arxiv.org/abs/1803.01470v1

[10] Miesenberger, K., Neuper, W., Stöger, B., Wenzel, M.: Accessible and inclusive mathematics working environment (A-I-MAWEN) (2023), https://static.miraheze.org/isacwiki/9/90/A-I-MAWEN.pdf

[11] Miesenberger, K., Neuper, W., Stöger, B., Wenzel, M.: Towards an accessible mathematics working environment based on Isabelle/VSCode. In: Quaresma, P., Marcos, J., Neuper, W. (eds.) Proceedings 11th International Workshop on Theorem Proving Components for Educational Software, Haifa, Israel, 11 August 2022. Electronic Proceedings in Theoretical Computer Science, vol. 375, pp. 92–111. Open Publishing Association (2023). https://doi.org/10.4204/EPTCS.375.8

[12] Neuper, W.: Lucas-interpretation on Isabelle's functions. In: Quaresma, P., Neuper, W., Marcos, J. (eds.) Proceedings 9th International Workshop on Theorem Proving Components for Educational Software (ThEdu'20). p. 79–95. No. 328, Electronic Proceedings in Theoretical Computer Science, Paris, France (30th October 2020). https://doi.org/2010.16016v1, https://arxiv.org/abs/2010.16016v1

[13] Neuper, W., Dürnsteiner, C.: Angewandte Mathematik und Fachtheorie mithilfe adaptierter Basis-Software. Tech. Rep. 683, IMST – Innovationen Machen Schulen Top!, University of Klagenfurt, Institute of Instructional and School Development (IUS), 9010 Klagenfurt, Sterneckstrasse 15 (2007), https://www.imst.ac.at/imst-wiki/images/f/f9/683_Kurzfassung_Neuper.pdf, https://www.imst.ac.at/imst-wiki/images/f/f9/683_Kurzfassung_Neuper.pdf

[14] Neuper, W., Reitinger, J., Gründlinger, A.: Begreifen und Mechanisieren beim Algebra Einstieg. Tech. Rep. 1063, IMST – Innovationen Machen Schulen Top!, University of Klagenfurt, Institute of Instructional and School Development (IUS), 9010 Klagenfurt, Sterneckstrasse 15 (2008), https://www.imst.ac.at/imst-wiki/images/9/9d/1063_Langfassung_Reitinger.pdf, https://www.imst.ac.at/imst-wiki/images/9/9d/1063_Langfassung_Reitinger.pdf

[15] Neuper, W., et al.: Angewandte Mathematik und Fachtheorie. Tech. Rep. 357, IMST – Innovationen Machen Schulen Top!, University of Klagenfurt, Institute of Instructional and School Development (IUS), 9010 Klagenfurt, Sterneckstrasse 15 (2006), http://imst.uni-klu.ac.at/imst-wiki/index.php/Angewandte_Mathematik_und_Fachtheorie, http://imst.uni-klu.ac.at/imst-wiki/index.php/Angewandte_Mathematik_und_Fachtheorie

[16] Paulson, L.C., Nipkow, T., Wenzel, M.: From LCF to Isabelle/HOL. Formal Aspects of Computing **31**, 675–698 (September 2019), https://doi.org/10.1007/s00165-019-00492-1, springer, London

[17] Suzuki, M., Yamaguchi, K.: On automatic conversion from e-born pdf into accessible epub3 and audio-embedded html5. In: Proc. the 17th International Conference on Computers Helping People with Special Needs (ICCHP 2020). pp. 410–416. No. 12376 in LNCS, Springer (2020). https://doi.org/10.1007/978-3-030-58796-3_48

[18] Yamaguchi, K., Suzuki, M.: An accessible STEM editor customizable for various local languages. Journal of Enabling Technologies **13**(4), 240–250 (2019). https://doi.org/10.1108/JET-12-2018-0064

Using LaTeX in Schools: Simplifying Inclusive STEM Education

Barbara Henn[1], Michael Schäffler[1], Davide Cervone[2], Volker Sorge[2], and Dorine in't Veld[3]

[1] SBBZ Ilvesheim, Germany
[2] MathJax Consortium
[3] Dedicon, The Netherlands

1 Introduction

We report on experience of the use of LaTeX in Schools to teach mathematics to blind and visually impaired children. For the last 20 years the German education has phased out the use of sophisticated mathematical Braille notations and replace it by teaching children from year one LaTeX (or LaTeX-like) notation in mathematics. This notation made tactile by translating it into 8-dot Euro Braille thus removing the use of indicators and the need to express single characters with multiple Braille cells. The aim was to reduce the learning curve for students as well as to improve their abilities to author and communicate mathematics with peers and teachers that are not trained in Braille, which is paramount for inclusive education.

There has been previous work on using LaTeX as the basis for assistive technologies, such as text-to-speech (TTS) translation [11] or for transcriptions of Braille textbooks [10]. These works would transform LaTeX into dedicated math speech systems or specialist math Braille dialects like Marburg or Nemeth, rather then using the power of LaTeX directly and making it available as a communication medium for blind and visually impaired (BVI) learners. We present how our approach can technically be supported via web technology that can not only render LaTeX, but also provide 8-dot Braille output for expressions as well as compute meaningful LaTeX commands for sub-expression and support for direct copying from web sites.

Finally, we will discuss our initial ideas to transfer this model to other education systems, namely the Dutch system, where the current approach is ambivalent, in that there exists an AsciiMath-like linearized math notation for teaching to BVI students, while tactile math textbooks use an outdated Braille notation, that is effectively no longer used in teaching. As a result, students are actively discouraged from studying mathematics.

2 LaTeX in Education for the Blind

The choice of mathematical notation in education for blind and visually impaired (BVI) students vary widely throughout the world. While there exist many national and regional variations in regular mathematical notation, these differences are particularly strong for primary and secondary education and become less pronounced in advanced mathematics and higher education. For BVI learners however this is generally not the case.

Traditionally, mathematics is presented in tactile format on the basis of highly specialized Braille notations, which increases in complexity for more advanced mathematical notation and more unconventional symbols. In particular, in traditional 6-dot Braille systems symbols have to be embellished with a plethora of indicators (e.g., for numbers, capital letter, fonts) and modifiers (for accents, positioning etc.). For a comparison of notational systems see [13].

This approach has a number of drawbacks: (1) The learning curve for mathematics, already steep for sighted students, becomes even steeper for BVI students. (2) Students can not easily communicate their Braille math with their sighted peers or teachers not trained in the formalism, which is an obstacle for inclusive education. (3) Math Braille notation is usually easier to read but less suitable for writing mathematics. This is particularly compounded by the use of computers even with Braille input devices. (4) Many traditional Math Braille notations have a 2D variant for complex elements like nested fractions or matrices, which are difficult to display on a computer Braille display. (5) Different Math Braille notations are generally not compatible and often even mutually intelligible. Not only have different countries different notations, but even the same language has different Math Braille dialects, e.g., in English there exist Nemath and UEB as well as formerly British Math Braille etc.

As a consequence more than 20 years ago Germany has chosen a different path for specialist education for BVI students [6, 7, 9], replacing the teaching of the local mathematical Braille notation (Marburg System) based on 6-dot Braille by adopting a uniform LaTeX notation for writing and transcribing that notation into 8-dot Euro Braille.

Since the early 1990s, the number of blind students in mainstream schools in Germany rose significantly. This increase was mainly due to the technical developments, with students working consistently with a PC and a connected Braille display, using 8-dot Braille, as opposed to the traditional six dots. Employing 8-dot font did not only prove less difficult for use with a computer, but also advantageous for inclusive education, as it turned out that 8-dot Braille was more compatible with the world of the sighted classmates.

For every character, letter, and number that had to be learned by the sighted student, there was exactly one equivalent in 8-dot Braille. There was no need for duplicate meanings of characters or indicators and modifiers as every single ASCII character can be translated into its equivalent Euro Braille cell.

Consequently, in the 2000s, a debate broke out in Germany between the competing writing systems of 6-dot and 8-dot Braille. While 6-dot script with and without contraction were mainly used in schools for the blind, the 8-dot script was the Braille system for integrated education and inclusion [4]. This long-standing practice was scientifically underpinned by the Zubra study conducted by Lang et al [8, 14], which demonstrated that sufficient reading speed is also possible with 8-dot Braille. Moreover, in combination with screen reader output, reading speed is much higher than reading contracted 6-dot Braille on paper.

In mathematical writing the script that goes hand in hand with 8-dot Braille is the LaTeX typesetting system as every single character of the LaTeX code corresponds to exactly one Braille cell. In elementary school mathematics, LaTeX code is only necessary in a few places, as the digits and arithmetic symbols are already available in the 8-dot system. However, some characters, such as the element symbol, are introduced in LaTeX notation. With the transition to secondary school, the LaTeX commands for fractions, root signs and exponent notation are added. Thus, on the way to the secondary school examination (Abitur), the students are gradually taught all the necessary LaTeX.

For teachers in mainstream schools, LaTeX is not a major hurdle, as many mathematics teachers are already familiar with this system from University. Those who are new to it can quickly learn it due to its intuitive and semantic notation. In contrast, both 6-dot contracted Braille and the 6-dot mathematical notation in Germany — the Marburg system — are much more complex to learn. As a result, learning these notations is hardly feasible for teachers in mainstream schools. The 8-dot Braille system, together with the linear notation of LaTeX on the one hand and the PC in conjunction with a screen reader and a Braille display on the other, represent an ideal bridging technology between the sighted and blind worlds. LaTeX not only helps students to communicate mathematics with their teachers but also with their sighted peers. Simple expressions are easy to understand even for those not yet exposed to LaTeX syntax. And if that fails it can be visually rendered for easier reading.

With the increasing digitalization in schools for blind students and the introduction of computers in every classroom, the 8-dot system and LaTeX have become the widely accepted standard. Many students thus achieve the University entrance qualification and can continue to work seamlessly at Universities with LaTeX [1]. As LaTeX is the lingua franca among mathematicians and as such internationally understood notation, students going on to higher education are already equipped with the most important tool to communicate mathematics with their professors.

3 LaTeX to Braille

As already mentioned, in early primary education very little LaTeX is actually needed. And even then the main goal is to use as much as possible symbols that are available on the keyboard. For example, asterisk `*` is used instead of `\cdot` for multiplication and simple fractions are written in beveled notation e.g., $\frac{1}{2}$ would be written as `1/2` instead of `\frac{1}{2}`. Nevertheless, some symbols like $\in$ are already introduced in their LaTeX notation `\in`.

Other conventions are adopted for easier reading, such as avoiding curly braces as much as possible, and inserting spaces before operators and relation symbols. Consider the example $n! = n * (n - 1)!$ that is written by this convention as `n! =n *(n -1)!` In addition, certain command abbreviations are introduced, primarily with the goal of reducing spatial requirements on the Braille display. E.g., writing `\f` instead of `\frac`, or `\ol` instead of `\overline` . Also commands can be replaced by meaningful symbol combinations, such as `\le` being replaced by `<=` (see also `https://augenbit.de/`).

Initial online support for working with LaTeX and Euro Braille was developed at the SBBZ Ilvesheim as an extension of the MathJax v2.7 [2] library. The extension allowed to expose the LaTeX sources in the web page from which the MathJax would compute the visual rendering as textual underlay for the expression. This allowed easy selection and copying and pasting into a text editor or word processor from which the expression could be automatically translated into 8-dot Braille by a correctly configured screen reader. While this technique was sufficient for students to work with, in particular during the Covid pandemic, it had the drawback that, while some cleanup on the LaTeX source could be done, the expression would usually not follow all the rules described above.

In MathJax version 4 [3] we will support these features natively, in particular, copying and Euro Braille translation and rewriting LaTeX sources into the required format. Figure 1 shows the Braille translation for the commands describing the quadratic equation:

```
x = \frac{-b \pm \sqrt{b^2-4ac}}{2a}
```

As an additional feature MathJax v4 exposes correct LaTeX sub expressions for parts of formulas that can be interactively explored. While this sounds straightforward, these sub expressions are non-trivial to compute. Firstly, LaTeX is a touring complete language, which requires a recursive stack automaton for parsing and thus partial expressions are not as readily available as in a simple LR-parser. Secondly, the exploration is based on a semantic model that is computed using Speech Rule Engine [12], which produces a canonical purely semantic representation of the math expression regardless of the incoming syntax, i.e., LaTeX, MathML, or AsciiMath..

$$x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

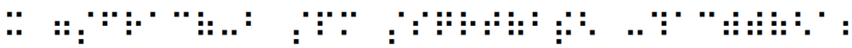


Figure 1: Translation of the quadratic equation into 8-dot Euro Braille

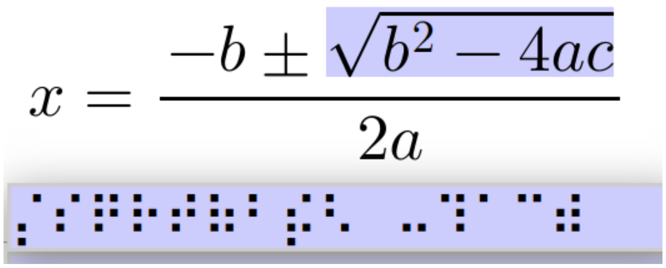


Figure 2: Translation of the quadratic equation into 8-dot Euro Braille

Figure 2 shows the Braille for the square root sub expression in the quadratic formula.[1] Note that there are nevertheless limits to the LaTeX sub expression MathJax can produce. For example, AMS matrix environments like `pmatrix`, `bmatrix`, or `vmatrix` implicitly generate fences, that are rendered. However, at the moment the parsing algorithm cannot produce corresponding LaTeX output.

4 Math Notation in the Netherlands

While similar approaches to Math Braille notation have been taken in other European countries using established mathematical syntax notations (e.g., LaTeX in Slovenia and AsciiMath in Sweden), other countries have pursued a different approach.

We will examine the situation in The Netherlands. Over twenty years ago in The Netherlands math had become practically inaccessible, the primary problem being the lack of a Math Braille code. Consequently, a linear notation was developed and introduced by Dedicon[2], which eases math communication in particular in inclusive education, but did not follow any pre-existing standards. In fact, the Dedicon notation is a mix of several linear notations, borrowed from symbolic calculation languages, like Maple, Mathematica, Excel. Pragmatic choices were made to make the notation as intuitive as possible; e.g. using SQR, SQRT or the dutch word WORTEL for abbreviations of square root.

In [5], Dorine in't Veld and Davy Kager presented the status of this notation, which in practice works quite well for students, since it meets all the needs summed up above for LaTeX:

1. It allows direct communication with sighted peers and teachers.

[1] Additional examples of the technique can be found here: https://mathjax.github.io/MathJax-demos-web/euro-Braille/

[2] http://Braille.dedicon.nl/wiskunde

2. It remains consistent notwithstanding in what application it is used. It can be converted from Microsoft Word or HTML to plain text or Markdown without any problems, because, like LaTeX, the notation only uses ASCII characters that can be typed with a qwerty keyboard.

3. It requires no additional assistive or conversion software. The screenreader translates the notation to 8-dot Braille and TTS.

The weaknesses are that the notation initially only covered primary and secondary education, not higher education. And while it was good for writing, it was too ambiguous for more complex expressions, mainly due to its use of spaces for delineation. For example, there is a rule that spaces terminate superscript and subscript text, which is usually very convenient but can go wrong when combining the two: Consider `x_2 ^2` versus `x_2^2`, where the former is x_2^2, while the latter represents x_{2^2}. Note that the space between the 2 and the caret symbol makes the difference, which can lead to easy mistakes. The main issue is the lack of clear notation that defines argument boundaries, which is particularly concerning with more complex expressions containing fractions, roots, vectors etc. As a consequence, automatic conversion is difficult and there currently exists no implementation of a renderer that can visualize expressions.

As already mentioned, the Dedicon notation is quite popular with students and teachers in The Netherlands and it works sufficiently well in practice. And since Dedicon has the responsibility to make textbooks accessible, they must use formats that are actually taught to students, which currently means that books are transcribed into Word documents with math in Dedicon notation.

As a consequence the necessary improvements indicated in 2016 have not been made yet. Likewise there is no pressure to change to AsciiMath or LaTeX for schools or to extend the current Dedicon notation to become suitable for more advanced mathematics. However, this leads to a disconnect on the other stages of the education ladder:

1. Students in higher education have no support of specialized itinerant teachers who are experts in a discipline and in the use of screenreaders. This means for example that those students who need to learn LaTeX at their University, have to find a practical accessible tutorial on their own. Additionally, if they do not have the luck to find someone in their IT-department who can help out, they have to find out what software to use and how it can be installed in the network of their University, etc.

2. Students in secondary pre academic education already have no support of specialized itinerant teachers. In fact they too are pretty much on their own with their laptop and screenreader. The last couple of years there is much more attention for science education; there is a working group that develops (mainly tactile) materials and there is a helpdesk. But there was no attention for standard mathematics notation like LaTeX so far; there were many other priorities and – again – the Dedicon notation works well enough.

This matter is compounded by another peculiar problem in primary schools: children often want printed Braille, also for math. In these tactile books an old 6-dot Braille math code is used, that is not taught to the students. When they start from scratch they will easily learn and accept that spacing rules are different in Braille and if there might be a notation they do not understand, they can ask their teacher or peers what the original book says or they can look up things in a symbols list. For secondary and higher education, we no longer print Braille. The educational institutes for the BVI students do not want it since the code for more advanced math is too complex and nobody learns it. For tactile images this means, that it is often not possible to transcribe formulas into Braille, if they cannot be expressed in 6-dot Braille.

Moreover math is mostly done electronically on the laptop in secondary school. Until fairly recently students would choose their own preferred 8-dot Braille table on their Braille display, mostly the 'German' or 'American' table. While students are now encouraged to mostly use Euro Braille, those who are used to other Braille tables, are often reluctant to change to Euro Braille.

Due to the fact that students are very much on their own finding their way and there is no uniform notation or code, still only very few blind students choose science studies where they need more advanced mathematical notation and specialized software applications. Because of the same reasons, many students experience big obstacles in disciplines that require statistics. Here R becomes more prevallent with strong community support and improving accessibility support based on LaTeX. This, together with the advantages to be learned from the German system, in combination with the latest MathJax developments, should help to improve the situation in The Netherlands and bring about a feeling of urgency to have books from which you can simply copy (human readable) LaTeX that one can communicate, render or paste into accessible software thereby increasing the number of BVI students taking up mathematical and scientific subjects in secondary and higher education. Hopefully this article will raise awareness and involve experts from the above mentioned working group to push this agenda forward.

References

[1] Erdmuthe Meyer zu Bexten and Martin Jung. Latex at the university of applied sciences giessen-friedberg—experiences at the institute for visually impaired students. In *Computers Helping People with Special Needs: 8th International Conference, ICCHP 2002 Linz, Austria, July 15–20, 2002 Proceedings 8*, pages 508–509. Springer, 2002.

[2] Davide Cervone, Peter Krautzberger, and Volker Sorge. Towards universal rendering in mathjax. In *Proceedings of the 13th Web for All Conference*, page 4. ACM, 2016.

[3] Davide Cervone and Volker Sorge. Mathjax: The present and the future. *SIAM News*, November 02 2020. https://sinews.siam.org/Details-Page/mathjax-the-present-and-the-future.

[4] Richard Heuer gen. Hallmann. Beliebt, unbeliebt, beliebig – 6- oder 8- Punkt-Braille. *horus*, 6:211–214, 2001.

[5] Dorine in 't Veld and Davy Kager. Math notation used in the netherlands: quick fix or lasting solution? In Katsuhito Yamaguchi and Masakazu Suzuki, editors, *3rd International Workshop on Digitization and E-Inclusion in Mathematics and Science (DEIMS2016)*, 2016.

[6] Ulrich Kalina. LaTeX (nicht nur) eine Lösung für das Problem sehgeschädigter Computerbenutzer, Mathematik schriftlich darzustellen. *Blind sehbehindert*, 113:86–89, 1993.

[7] Ulrich Kalina. Welche Mathematikschrift für Blinde soll in der Schule benutzt werden? *Blind sehbehindert*, 118(3):78–94, 1998.

[8] Markus Lang, Ursula Hofer, and Fabian Winter. Lese- und Schreibkompetenzen von Braillenutzerinnen und -nutzern in Allgemeinen Schulen und Schulen mit dem Förderschwerpunkt Sehen. Ergebnisse aus dem Forschungsprojekt "ZuBra" (Zukunft der Brailleschrift). *Zeitschrift für Heilpädagogik*, 71:280–292, 2020.

[9] Ernst-Dietrich Lorenz. 6-punkt-mathematikschrift und/oder latex. stellungnahme der fachgruppe zur frage der für blinde menschen zweckmäßigsten notation mathematischer und naturwissenschaftlicher ausdrücke. *Blind sehbehindert*, 122(4):265–267, 2002.

[10] Tomás Murillo-Morales, Klaus Miesenberger, and Reinhard Ruemer. A latex to braille conversion tool for creating accessible schoolbooks in austria. In *Computers Helping People with Special Needs: 15th International Conference, ICCHP 2016, Linz, Austria, July 13-15, 2016, Proceedings, Part I 15*, pages 397–400. Springer, 2016.

[11] TV Raman. Aster: Audio system for technical readings. *Information Technology and Disabilities*, 1(4), 1994.

[12] Volker Sorge. Speech rule engine, v4. http://speechruleengine.org, 2022.

[13] Annemiek van Leendert, Michiel Doorman, Paul Drijvers, Johan Pel, and Johannes van der Steen. Towards a universal mathematical braille notation. *Journal of Visual Impairment & Blindness*, 116(2):141–153, 2022.

[14] ZuBra — Zukunft der Brailleschrift: Schriftsprachkompetenzen von Brailleleserinnen und Brailleleser. https://www.ph-heidelberg.de/blinden-und-sehbehindertenpaedagogik/forschung/zubra-zukunft-der-brailleschrift/.

Enhancing LaTeX to Automatically Produce Tagged and Accessible PDF*

Frank Mittelbach[1] and Ulrike Fischer[2]

[1] LaTeX Project, Mainz, Germany
frank.mittelbach@latex-project.org
[2] LaTeX Project, Bonn, Germany
fischer@troubleshooting-tex.de

Abstract

At the TUG 2020 online conference the LaTeX Project Team announced the start of a multi-year project to enhance LaTeX so that it will fully and naturally support the creation of structured document formats, in particular the "tagged PDF" format as required by accessibility standards such as PDF/UA.

In this talk we present the current achievements of this project[1] and the issues we encountered along the way; we also outline open areas of research and the future steps that we shall take to automatically produce well-tagged PDF that supports accessible standards (in particular, the recently finalized PDF/UA-2) as well as general reuse and further conversions This will be achieved by embedding in the PDF a comprehensive description of the document structure.

Contents

*This conference submission is a fully tagged and accessible PDF produced by the project software it describes.

[1]This project is carried out by a small number of developers. Besides the authors, the following individuals from the LaTeX Project Team are actively involved: Chris Rowley, David Carlisle, Joseph Wright, Marcel Krüger, and Phelype Oleinik.

We also wish to acknowledge the contributions from the various members of the TeX community and beyond; these have been made through comments and suggestions, and more recently through the testing of the new functionality made available in the form of prototype implementations. Without such feedback it would be difficult to finish this project with satisfactory results.

1 General Overview

For over 30 years now, the LaTeX system has been used, widely and successfully, for document production in the STEM world and also in other places where high-quality output is required; but until recently its focus was solely on page-oriented output for print (on paper) or as paged output using the PDF format. Therefore, the structural information about the document that was present in the LaTeX source did not get incorporated into the PDF output. Rather, this information was discarded as soon as possible during the processing; this was necessary so as to conserve the limited computer resources (memory and storage) that were typically available at that time (when the core of the LaTeX processing model was first designed).

As long as the intention is only to print a document on a physical medium, then this is all that is required. However, for quite a while now other uses of documents have been increasing in importance so that nowadays many documents are never printed, or printed only as a secondary consideration.

Coming into the 21st century, for many reasons great interest has arisen in the production of PDF documents that are "accessible", in the sense that they contain information to assist screen reading software, etc., and, more formally, that they adhere to the PDF/UA (Universal Accessibility) standard [2, 5], which is explained further in [1].

At present, all methods for producing such "accessible PDFs", including the use of LaTeX, require extensive manual labor[2] during either the preparation of the source or the post-processing of the PDF (maybe even at both stages); and these labors often have to be repeated after making even minimal changes to the (LaTeX or other) source.

1.1 The Goals of the Multi-Year "LaTeX Tagged PDF" Project

The main goal of the project is to enhance LaTeX so that it can *automatically* produce tagged PDF without the need to add additional data or commands to the LaTeX source, or to do any of the post-processing work necessary in other workflows.

If it remains necessary to alter substantially, or to extend, each individual document in order to provide tagged PDF that conforms to some accessibility standard, then we shall see very few document authors willing to go through the pain of making such additions (unless they are forced to). It is therefore of utmost importance that the generation of tagged PDF be done essentially behind the scenes, with the only cost to the authors being a somewhat longer compilation time.

Another important aim is to make already existing documents accessible by simply recompiling them without the need to alter the source in any substantial way.[3]

The project will support the PDF 2.0 standard [3] (with the very widely supported PDF 1.7 as a fallback solution), because the PDF 2.0 standard offers a more comprehensive tag set, and it supports associated files and many other important features; its use is also a requirement of the new PDF/UA-2 standard [5].

Unfortunately, even though PDF 2.0 has existed for six years already it has yet to be adopted for industry solutions: e.g., most viewers and other applications are still incapable of making correct use of the new PDF 2.0 features. This is largely a chicken-and-egg problem: because nobody produced 2.0 files, no application was specifically extended to enable processing such

[2] If not using the already existing code extensions to LaTeX provided by the project.

[3] Of course, required data that is not part of the document source (such as alternative text for figures or additional metadata) will need to be manually added, so as to ensure that the document is compliant with PDF/UA. But even if this work is not undertaken, the fact that the document gets automatically tagged will mean that it can be easily navigated and consumed in ways that were impossible before.

files; and due to the fact that no viewer could handle such files, the developers of PDF writers saw no need to invest in the technology to produce PDF 2.0 files.

As a result, LaTeX is one of the first authoring applications that can produce PDF 2.0 files automatically and in large quantities. In particular, LaTeX is capable of producing documents compliant with PDF/UA-2, the new standard for Universal Accessibility [5] that was finalized in 2023 and will be officially released in early 2024.[4] No doubt other suppliers will follow our lead when there is sufficient demand for the production and processing of PDF 2.0 and PDF/UA-2 conformant files.

The document entitled "LaTeX Tagged PDF Feasibility Evaluation" [9], available from the LaTeX Project website [7], explains in detail both the project goals and the tasks that need to be undertaken, concluding with the project plan that is currently being executed.

For the time being the project will focus primarily on PDF output (generated either directly by the TeX engine or through a DVI-based workflow). However, as a bonus outcome of the design approach, the implemented solution will make it easy to add other such output formats to the workflow by simply replacing the output (backend) module. Instead of PDF output, HTML5 or some other format can thus be written. As of now, such alternative backends are not part of the project coverage, but once LaTeX is able, using well-defined interfaces, to pass structure information to a backend, we expect that support for other structured output formats will follow. Such work may be undertaken by us or by other teams, possibly in parallel to later phases of the project.

1.2 Current Status and Achievements

As mentioned earlier, LaTeX was originally designed (as was essential 40 years ago) to be very economical with computer resources; the implementation therefore worked very hard to discard information as soon as it was no longer needed for the compilation of a document. For print output, which was all that was produced back then, these discards included most of the structural information since this was no longer useful once the visual representation had been determined. An important part of the early work on this project was therefore to alter LaTeX's inner workings by adding code that preserves this structural information from the source and adds it to the PDF.

Another part of this early "background" work was to standardize (and often to provide, for the first time) code interfaces into which extension packages can safely hook. The use of these interfaces, rather than directly overwriting internal LaTeX functions (as was commonly done in the past), avoids the problem that such packages would often break when used in certain combinations, or break when LaTeX internals changed. Moreover, it means that these packages can automatically benefit from the existence of extended workflows (such as those which produce tagged PDF).

Most of these interfaces are now in place in the LaTeX kernel. What remains (as a huge task) is to upgrade many of the core extension packages so that they make use of the new functionalities; this will enable the retirement of some of the existing code that directly overwrites LaTeX internals, or that makes assumptions (about those internals) that will become invalid in the future.

The next large phase of the project was to provide automatic tagging for a subset of LaTeX documents. This task is largely finished and therefore most documents that are restricted to using only the commands and environments described in Leslie Lamport's "LaTeX Manual" [6] can be automatically tagged by adding a single configuration line at the top of the document.

[4] At the moment we can only claim that the project software is capable of producing documents that comply with the latest draft of the immanent PDF/UA-2 standard.

We say "largely finished" because a few such elements, or element combinations, are not yet covered at the time of writing.

On the other hand, a number of extension packages that go beyond Lamport are already supported, most importantly much of `amsmath` (providing extended math capabilities) and `hyperref` (enhancing LaTeX with interactive hyperlinking features). Also already supported are some of the major bibliography packages, such as `natbib` and `biblatex`.

The project is thus by now capable of producing PDF 2.0 documents that conform to the new PDF/UA-2 standard. In fact, after correcting a small number of issues (not directly related to tagging) in the class file for this conference we have been able to deliver this article as a fully tagged PDF 2.0 document.

1.3 Ongoing and Future Project Tasks

At present, tagging support for the core document elements in a LaTeX document is still at the prototype level, which means that it works for the standard LaTeX classes and for the document elements provided by the LaTeX kernel, but it may or may not work with extension packages or classes that alter the implementation of these document elements, or that provide completely new elements.

To make further progress, some of the interfaces for tagging will first need to be finalized. Then all major extension packages, as well as all important third-party document classes, will need analyzing and possibly updating.[5] The tasks here are to identify all of the legacy low-level code for which the kernel now provides tagging-aware replacements, and then, in cooperation with their maintainers, to make the necessary updates to all these packages and classes.

In addition, any packages and classes that provide new document elements will need to specify how these elements are supposed to be tagged. Some interfaces already exist to help with this process, but it is likely that most of these will require further refinement when tested in the field.

The remaining phases of the project, as outlined in the "Feasibility Evaluation Study" [9], cover further support for other PDF standards, and an improved interface to comprehensive metadata. There are also a number of research problems that need to be solved in order for authors to easily generate high-quality tagged PDF documents from their LaTeX sources. These are outlined below.

2 Specific Aspects of the Project Work

We now take a look at a few specific aspects of the project work that are related to challenging problems and pose interesting research questions. These topics are: the development of a more granular tag set; the handling of formulas; the need for an extended table specification syntax; and the handling of language and script related requirements.

2.1 The Existing Tag Set Support in PDF

When PDF (already in version 1.3) first introduced a structure tree into the format, to support the inclusion of the document's logical structure, it used only a fairly minimal set of structure tags that were largely modeled after the basic HTML tag set.

[5] The number of widely used packages, e.g., those described in "The LaTeX Companion, third edition" [8], amounts to roughly 500, so this evaluation and code adjustment forms a substantial part of the remaining project work, and most likely requires additional volunteer support.

For example, for mathematical formulas there was only the one `<Formula>` tag, with no possibility to add further structure within the formula. For accessibility, all that was available was an “alt” attribute in which one could add a textual description of the formula’s content. In a similar manner, other areas of document structures were (over)simplified in the tag set: e.g., for all types of floating elements there is only the tag `<Aside>` that they must share with margin notes (and even that tag is available only in PDF 2.0). For code elements, whether they are small snippets or long, commented listings, there is only a single `<Code>` tag, and there is no option to accurately describe the handling of spaces and new lines within code listings. There is a tag (again only in PDF 2.0) to denote footnotes; but if the document contains several types of structured (and possibly nested) notes, then there is no way to adequately describe this without losing possibly crucial information.

As is also the case with HTML, the relationships between these tags define a fairly simple document model that is not sufficiently rich, so that it cannot express (or not correctly express) many real-life documents; this is often due to the fact that certain elements appear in such documents with nesting relationships that are not permitted by the inclusion rules defined in ISO 32005 [4].

All this means that, when preparing a PDF to be PDF/UA-2 or PDF/UA-1 compliant, compromises have to be made and some of the structural information may thus get lost.

As part of the project we are therefore developing an extended tag set (currently called the “LaTeX namespace”) that describes the logical structure of (complex) documents in more granular detail; this will help PDF processors (such as viewers) that understand this namespace to make better use of a document’s structure. Ideas from this development may also prove useful in conjunction with future HTML5 developments.

2.2 The LaTeX Namespace

LaTeX is an open system that allows for structural extensions (and even changes to structures) in every direction. It is therefore not possible to define a fixed (definitive) document model that is both valid and comprehensive for each and every conceivable LaTeX document.

However, it is possible to define a document model which captures the majority of LaTeX documents that are out there in the real world. If this is combined with methods to extend (and possibly alter) the document model whenever necessary for special structural extensions or changes, we are confident that a comprehensive solution can eventually be provided.

As part of the project we are therefore developing a “standard namespace” that fully describes the LaTeX document model (in the sense outlined above). This tag set will thus be noticeably more detailed and comprehensive than those offered by PDF 2.0 and HTML5. We are working with the PDF Association [10] and various application producers to ensure that this namespace will, when complete, become a recognized resource (preferably acknowledged in future revisions of the PDF standard); it may also be more generally useful as an XML schema. This will, for example, allow PDF and other applications to directly use the extended tag set it provides; and this will enable such applications to make better use of the information contained in the document, whether for accessibility support or for other purposes.

For applications that do not (yet) understand this new namespace, we provide role-mapping back into PDF 2.0 (or PDF 1.7) as necessary; but of course, in that case the more granular information provided by the tags in the new namespace will get at least partially lost.

Additionally, we will be providing interfaces that allow package or class developers who extend the standard LaTeX structures to specify how their new commands or environments map into the LaTeX namespace (and from there, if necessary, are role-mapped back to the PDF tag set).

2.3 Formulas in STEM Documents

LaTeX is well-known and appreciated for its ability to describe and format mathematical or other formulas with a high degree of flexibility and unsurpassed quality. This is one of the reasons why we see a huge proportion of the documents in STEM disciplines such as mathematics, physics, and computer science being produced using LaTeX.

As described by Neil Soiffer in his keynote for the DEIMS 2021 [11], there are basically three methods for making such formulas accessible in a tagged PDF. One option is to use static text that can be attached as "alternative" natural language text to the formula structure, which is then read by an AT application. While rather easy to implement, this method has various drawbacks: it does not allow for braille generation or for exploration of the equation; and the text often must be hand-crafted to avoid problems with reading software whose heuristic usually ignores certain symbols such as punctuation or braces.

A second option is to add marked-content operators to the PDF stream and then build a MathML structure tree that references this marked content. This method leads to a large, since extremely fine grained, structure tree with many objects.

There are a number of problems with this second approach. It is difficult for LaTeX (without the help of a real programming language) to generate correct and useful MathML while building the TeX math list. Furthermore, when the still widely used pdfTeX engine is producing the PDF, there is a high chance that the combined processes (of simultaneously adding the necessary tagging-related material to the content stream whilst formatting the formula) will alter the spacing of the formula and thus render the visual representation invalid. There may be technical solutions to circumvent these issues and this is an area of active research. However, it is likely that this option can only be implemented successfully if the LuaTeX engine is used, because then a suitable programming language (i.e., Lua) is available and, furthermore (because of extended functionality in LuaTeX), it becomes possible to delay adding the necessary extra material to the content stream until after LaTeX has completed the formatting of the formula with the correct spacing.

The last option is to make use of so called "associated files" (AF) that were introduced in PDF 2.0: these are files embedded into the PDF that can be attached to a structure element. Such files can contain, for example, a MathML representation for a formula, or its LaTeX source or some additional commentary text—more than one such file can be attached to each structure. The AF approach is simpler and easier to implement, and it also allows the use of MathML representations that do not closely follow the visual output; but it has the drawback that the MathML in the AF file is associated only to the formula as a whole and it is therefore not possible to synchronize parts of the MathML representation with the corresponding parts of the formula in the typeset document, as is necessary to support navigation of formulas and highlighting them. This method may require that the AT software overlays the printed output with its own rendering of the MathML (which may differ substantially from the original rendering).

The two last options, MathML in the structure tree or or in associated files, both suffer from a lack of support in current PDF viewers and AT software: Neil Soiffer's optimistic statement in 2021 that "*Adobe's API will likely incorporate this ability in the future*" has not come true yet.

Because of this, LaTeX currently follows a three-fold strategy in the prototype for math tagging: it incorporates the LaTeX source as alternate text for the formula, under the assumption that the LaTeX syntax is understandable to most readers; and it also embeds the LaTeX source as an associated file. Additionally, for all (or a selection of) the formulas, MathML representations can be provided in an external file that are then also embedded as associated files. Such an external file can be created, for example, with the help of `tex4ht` or with the LuaTeX engine. At this point in time there is no fully automatic workflow implemented for this, but with only a few adjustments it was already possible to add MathML associated files to all the formulas in the `amsmath` user documentation.

The form of the final solution for formulas, and whether or not it is necessary to offer customizable alternatives—to cater for different reader deficiencies or different user preferences—are questions that need active research to understand how to best serve consumers given the currently limited functionality of AT tools with respect to AF files, etc.

There is another aspect of common LaTeX usage that affects all three of these method, and is also found in many other areas beyond formulas and STEM: the inclination of authors to invent new symbols, notation systems, and command names. This is nowadays exacerbated somewhat by inevitable failures to take accessibility into account. Such ad hoc extensions make it difficult to fully automate any tagging process. Overcoming this will need both technical support for such extensions and also, perhaps more importantly, encouragement of authors to keep accessibility in mind when writing documents.

The approach that will most likely be adopted to deal with this issue is as follows: by default, assume that new commands are simply abbreviations and that, by recursively replacing them each with its definition, we eventually get to something that can be automatically tagged by using standard methods. For cases where this does not work, there will be interfaces with which the author of the document (or the package developer, if the command is defined there) can specify how the command should be interpreted when providing tagged output (e.g., MathML).

2.4 Extensions to LaTeX's Table Handling

In most cases, the LaTeX source will contain all the necessary information about the logical structure of a document, so that it is possible to automatically transform the source into richly tagged PDF output. There is one noticeable exception, LaTeX's handling of tabular data: standard LaTeX, and most extension packages, do not describe table data through structural information; rather, they do this in a purely visual fashion, describing only the content that should go into each cell. Thus no information is supplied concerning important relationships between cells, such as which are the header or sub-header cells, or to which cells some header cell applies.

Thus, while it is fairly trivial to tag tables as simply consisting of table rows and table data cells, determining the header cells can be done only by the use of heuristics (e.g., cell formatting changes done through `\multicolumn` are likely to represent header cells, or certain rules in a table may indicate header rows). However, any such heuristic will have a noticeable number of counterexamples.

It is therefore an important task to develop good heuristics that correctly cover a large proportion of the tables in legacy documents; and in addition to develop a syntax extension for LaTeX that allows authors to specify such logical structure explicitly in case the heuristics fail or they wish to specify explicitly the logical structure of the table. This syntax extension has to be done in a light-weight way, i.e., without putting an unnecessary burden onto the authors. Furthermore, it should be upwardly compatible with the existing syntax so that it is easily possible to enhance documents with only small alterations to the original source.

Also important are methods that enable authors to easily check LaTeX's interpretation of the logical structure of a table without the need to examine the final PDF, so that they can overwrite the heuristics when necessary. This is an area of active research.

2.5 Support for Different Scripts and Languages

Historically, the TeX engine and LaTeX were developed for ASCII-based, English documents and then (with TeX 3.0) extended to support other languages and scripts—at first, because of the restrictions to 8-bit codepages, mostly for languages using Latin scripts, but later also to non-Latin scripts, such as Greek or Cyrillic, as well as more diverse scripts. Initially, the solutions for all such scripts required complex font setups (as done, e.g., by the `CJK` package), special processors to handle transliterations, and engine extensions to handle, for example, right-to-left scripts or special input encodings.

The advent of Unicode and the Unicode-aware engines (XeTeX, LuaTeX and upTeX) led to the existence of simpler, and much more powerful, setups; therefore most scripts are now well supported in LaTeX—perhaps with the exception of scripts that change the writing direction, since this isn't part of the original LaTeX design and thus often requires overwriting many standard commands.

The project currently concentrates on documents that use Latin scripts or scripts with similar characteristics. The correct tagging conventions to use with other types of script are not yet known by us: e.g., how to deal with direction changes or ruby characters. When using scripts (such as Latin) that typically use "whitespace" to delimit "words", tagged PDF has a requirement that even within the typeset content stream these words must remain delimited by an explicit "whitespace character" [3, §14.8.2.6.2]. This conflicts with the normal practice of TeX typesetting engines since they do not naturally add such delimiter characters; however, both pdfTeX and LuaTeX have been modified to provide workarounds for this.[6] We also do not yet know to what extent the many external packages supporting diverse scripts and languages will need to be adapted for the support of tagging. To research these topics, help from users and developers with in-depth knowledge of such scripts will be needed.

3 Some TeXnical Details

In this final section we take a brief look at two technical aspects of the project work. Both will be covered in more depth during the demonstration session at the conference.

The first subsection explains how to set up a document (such as this one) so that it will automatically produce tagged PDF. This should, we hope, enable you to immediately experiment with the addition of such tags to your own works. If you want to provide feedback on any issues that you encounter, or to provide suggestions for improvements, we suggest they are added to the project repository `https://github.com/latex3/tagging-project`, using either the `issues` or the `discussions` page, as appropriate.

This is followed by some background information on the experiments we are currently conducting to automatically include in the PDF MathML representations of all the formulas in a document. We expect that this work to become available for public testing during 2024/Q2.

3.1 How to Enable Tagging

Until recently there was no dedicated location in LaTeX documents to declare settings that affect the document as a whole. Settings had to be placed somewhere in the preamble or as class options, or sometimes even as package options. For some such settings this was problematical, e.g., setting the PDF version is only possible while the PDF output file has not yet been

[6]Engines, such as XeTeX, that do not offer this workaround can therefore not be used to produce PDF/UA documents involving scripts that separate words with spaces.

opened, which can be caused by loading one or the other package. For the "LaTeX Tagged PDF project" [9, p. 17] further metadata about the whole document (and its processing) needs to be specified, and again all this data should be placed in a single well-defined place.

For this reason we introduced (in June 2022) the command `\DocumentMetadata` so as to unify all such settings in one place. This command takes one argument that should contain a key/value list specificying all the document metadata for the current document.[7] This should be placed at the very beginning of the document, i.e., *before* `\documentclass`; it will produce an error if found later. The `\DocumentMetadata` command also loads the LaTeX PDF management bundle, which provides various PDF-related commands that are needed to create a tagged PDF.

The `\DocumentMetadata` command also accepts the `testphase` key, which is of a temporary nature since it is needed only while new functionality is being introduced for testing. This key is used to load specific tagging support: this article, for example, uses the following:

```
\DocumentMetadata{
   testphase={phase-III,table},
   pdfversion=2.0,
   pdfstandard=a-4,
}
```

which loads the tagging support from `phase-III` (basic document elements) and `table` (newly developed prototype code for tagging of `tabular`-like environments not yet integrated in any test phase). In addition, the PDF version was set to 2.0 and it specifies that the PDF should be compliant with the PDF/A-4 standard. So that is all that was necessary to produce the tagged version of this article.[8]

Eventually, the testphase code will move (once all components are considered stable) into the LaTeX kernel itself and the `testphase` key will vanish. Tagging will continue to require a `\DocumentMetadata` declaration, but then uses a simple `tagged=true` key (name to be decided).

3.2 Inclusion of External MathML

As outlined in section 2.3 on page 6 we are currently experimenting with a scheme in which externally provided MathML is embedded as AF files. The MathML for the formulas is provided in an external file containing one or more `\mml` commands with the following format:

```
\mml
 {65}
 {\begin{math}\sqrt [\beta ]{k}\end{math}}
 {656E4D3BB4F29D20A1B2CBCB35C35E7E}
 {%
  <math xmlns="http://www.w3.org/1998/Math/MathML" display="inline">
   <mroot>
    <mi>k</mi>
    <mi>&#x03b2;</mi>
   </mroot>
  </math>}
```

[7] At this point in time only a few keys are accepted, e.g., to set the PDF version, the language, a PDF standard and to load a color profile.

[8] The paper does not contain any tabular material, thus `table` is actually unnecessary for tagging this article. It was added to show how the interface can be used when new functionality is made available.

The first argument to `\mml` is a label (e.g., a number) to that uniquely identifies this MathML snippet; the second argument contains the LaTeX source for the MathML. The third argument is the MD5 hash of the LaTeX source. Its use ensures that the PDF file will contain only one AF file for any formula, even if a formula is repeated several times: for example, if the document repeatedly uses `$\beta$`, then each repetition will reference the same AF file, which means that the PDF file does not become unnecessarily large.

The final argument contains the corresponding MathML. In our current experiments the MathML is generated from the LaTeX source by processing it with `tex4ht`, with some further processing to add the md5 hash values and with some manual corrections to improve the resulting MathML. One advantage of using an external file at this stage is to allow the MathML to be validated before being included as an AF in the PDF. The MathML could potentially be generated by other TeX to MathML conversion programs such as `latexml` or `luamml`, which would allow experimentation with different pipelines to construct MathML associated files.

This file is then input at the beginning of the document and each MathML (with a unique hash value) is embedded in the PDF as the content stream of an AF file. The LaTeX code to produce tagged PDF then checks, for each math formula in the document, whether a MathML file with the same MD5 hash already exists; and if one does then its AF is simply referenced in the structure.

Currently the generation of the file of MathML fragments requires some manual editing and explicit execution of conversion programs. The next step will be to create scripts that will: run directly in a LuaTeX compilation; fully automate the generation of the MathML fragments from the LaTeX source; and validate this output.

References

[1] Olaf Drümmer and Bettina Chang. *PDF/UA in a Nutshell — Accessible documents with PDF*. PDF Association, August 2013. https://pdfa.org/resource/pdfua-in-a-nutshell/.

[2] *ISO 14289-1:2014; Document management applications — Electronic document file format enhancement for accessibility — 1: Use of ISO 32000-1 (PDF/UA-1)*, 2nd edition, 2014. https://www.iso.org/standard/64599.html.

[3] ISO. *ISO 32000-2:2020(en); Document management — Portable document format — Part 2: PDF 2.0*, 2nd edition, 2020. https://iso.org/en/contents/data/standard/07/58/75839.html.

[4] *ISO/TS 32005:2023; Document management — Portable Document Format — PDF 1.7 and 2.0 structure namespace inclusion in ISO 32000-2*, 1st edition, 2023. https://iso.org/en/contents/data/standard/04/58/45878.html.

[5] *ISO/FDIS 14289-2; Document management applications — Electronic document file format enhancement for accessibility — Part 2: Use of ISO 32000-2 (PDF/UA-2)*, 1st edition, 2024. https://www.iso.org/standard/82278.html.

[6] Leslie Lamport. *LaTeX: A Document Preparation System: User's Guide and Reference Manual*. Addison Wesley, 2nd edition, 1994.

[7] LaTeX Project Team. Website of the LaTeX Project. https://latex-project.org/.

[8] Frank Mittelbach and Ulrike Fischer. *The LaTeX Companion*. Addison-Wesley, Boston, MA, USA, third edition, 2023.

[9] Frank Mittelbach, Ulrike Fischer, and Chris Rowley. *LaTeX Tagged PDF Feasibility Evaluation*. LaTeX Project, September 2020. https://latex-project.org/publications/indexbyyear/2020/.

[10] PDF Association (PDFA). Website of the PDF association. https://pdfa.org/.

[11] Neil Soiffer. Accessible PDF: $2 \not> 1$. In *The 4th International Workshop on Digitization and E-Inclusion in Mathematics and Science 2021*. The DEIMS2021 Organizing Committee, 2021. https://workshop.sciaccess.net/deims2021/DEIMS2021_Proceedings.zip.

PDF Document Object Model Support for Math

ROMAN TODA[1],JING MU[1] ,YOUQIANG WU[1], NEIL SOIFFER[2]

[1] Foxit Software Incorporated
[2] Talking Cat Software Abstract

Accessible math in PDF documents is a long-standing issue. PDF 2.0 and PDF/UA-2 (in development) allows a mathematical expression can be tagged as a formula and the MathML representation of that expression to be stored as an Associated File (AF). Most Assistive Technology (AT) is capable of speaking and/or brailling MathML. To date, PDF readers do not expose AFs (and hence any MathML) to AT. This paper describes a method implemented in Foxit PDF Reader and NVDA that makes math in PDF as accessible as math in HTML documents. We also describe a method using Foxit PDF Editor to add MathML to any PDF document.

1 Math in PDF

ISO 32000:2008 (PDF 1.7) [1] only specifies alternate text as the method to make math expressions accessible. However alternative text does not allow for navigation of the expression, nor does it allow for braille generation because braille codes are based on the mathematical expression, not the words used to speak the expression. ISO 32000:2020 (PDF 2.0) [2] allows two methods of describing math:

- either use MathML elements directly as children of Formula structure element
- the use of MathML in an AF with Formula structure element

Although PDF 1.7 doesn't specifically mention it, the first method is legal in PDF 1.7. However, practice has shown that it is very difficult to produce. Despite being three years old, there are currently no PDF readers that expose AFs to AT and the standard PDF accessibility API [3] has not been updated to PDF 2.0 with a method to access an AF. Because of this, currently, there is no way to access math in PDF files other than as alternative text with current software.

2 DOM support for MathML

PDF documents consist of drawing commands that render text and graphics on a page. There is no requirement about the location of where the drawing occurs. For AT to access the document and know the reading order of the document, a separate structure (tag) tree is accessed whose traversal describes the reading order and meaning of those drawing commands [4]. For example, nodes in the tree might represent headings, lists, paragraphs, and for math "Formula"s as described by PDF 2.0.

The standard accessible PDF API provides a way for programs to access the PDF contents and hence is the DOM that AT can access. The API presents two ways to access the DOM on Windows: through the generic Microsoft (Windows) Active Accessibility interface (MSAA) or through a PDF-specific interface to the PDF DOM. AT can use either or both interfaces.

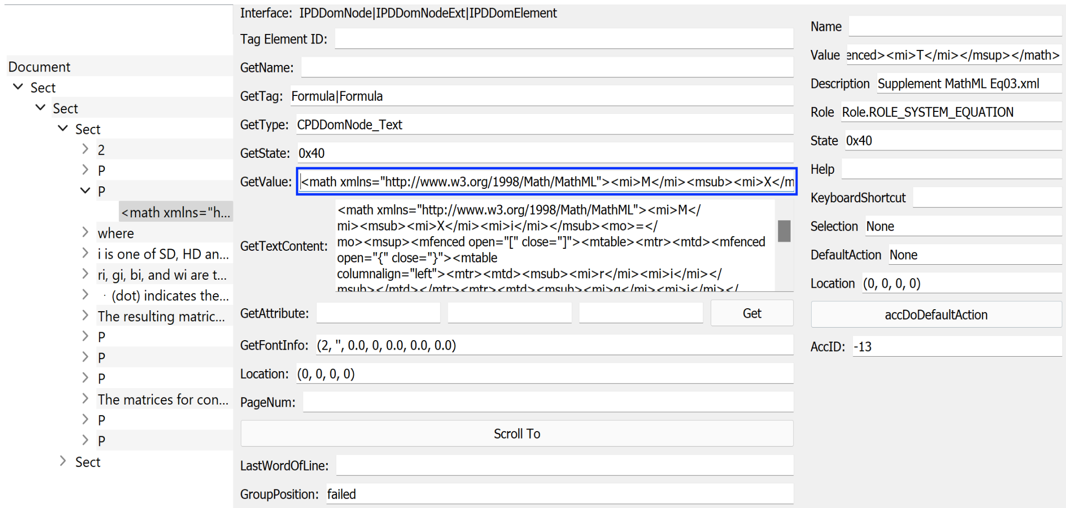


Figure 1: Visual representation of Dom for PDF containing Formula with associated MathML

Every DOM element has an MSAA role. For example, a PDF list is mapped to ROLE_SYSTEM_LIST and a paragraph is mapped to ROLE_SYSTEM_GROUPING. MSAA has a role for math: ROLE_SYSTEM_EQUATION. However, because in PDF 1.7 there was no way to expose math, the API does not return that value when a program asks for the role (get_accRole) of a Formula DOM element.

In PDF 2, a Formula element can have an AF containing data in MathML. When MathML is present in the AF, our implementation in Foxit PDF Reader[*] extends the functionality of the API to:

- Return ROLE_SYSTEM_EQUATION for the role of the DOM node
- Return the MathML for the textual content of the DOM node

This simple modification avoids modifying the accessibility API (i.e., it avoids adding a query to access an AF). No less important, it provides access to the MathML that is both easy to detect and to use. It is easy to detect because ROLE_SYSTEM_EQUATION was not used before. It is easy to access because the MathML is accessed just like text.

Figure 1 shows a dialog from the Foxit PDF Reader implementation of this idea. The left hand pane shows a portion of the DOM focused on an inline math expression. The middle pane shows various values that PDF-specific API queries would return, and the right hand pane shows the MSAA values. Notice that MathML content is shown in GetValue in the middle and the role is ROLE_SYSTEM_EQUATION on the right.

To test out this idea, we modified the open source NVDA screen reader to access this information. This was particularly easy because for over a decade, NVDA has had the ability to speak, navigate, and braille math in PDF using the first technique mentioned in section 2 (MathML tags as children of the formula tag). When marked up like this, ROLE_SYSTEM_EQUATION was used. Only a handful of documents were ever marked up in this way. Because the dispatch code already existed, only a few lines of code were added to the existing function that retrieved the MathML to detect this new usage.[†] In NVDA, regardless of the source, all MathML is handled by a MathML class that knows how to speak, navigate, and braille MathML, so no change was needed in that code. In other AT that reads PDF files,

[*] Custom build of Foxit Reader with this addition is at https://github.com/foxitsoftware/PDF_UA-2/tree/main/installer

[†] The NVDA modification can be found at https://github.com/NSoiffer/nvda.

it is likely there is code that examines the role. ROLE_SYSTEM_EQUATION isn't currently used, but it is likely easy to add a case for this role. After accessing the MathML, whatever path the existing code already uses for handling MathML in HTML documents. This relative ease of implementation was confirmed to be true for JAWS [personal communication, Oct 18, 2023].

3 Adding MathML to a PDF Document

Most authoring tools do not make use of AFs when generating PDF. As a result, most documents contain math in the form of an image or as a combination of vector graphics and textual symbols making the document inaccessible. Remediators have no other choice than to tag such page objects as a graphics element and at best add alternative text. In our implementation of Foxit PDF Editor‡, we have added the ability to attach MathML representation during manual tagging, allowing the generation of accessible PDF files as detailed in the previous section.

Manual tagging and manual conversion of images of math is time-consuming and error-prone. To improve upon this situation, we have a process that makes use of AI to generate MathML from the image semi-automatically. To convert an image to MathML and automatically modify the PDF to insert it in an AF, the user selects some page content representing math. The AI determines the appropriate MathML, renders it for user verification, and then modifies the PDF appropriately. The proposed user interaction and tagging are independent of the math recognition methods we used.

Software already exists that can do this. Examples that make use of AI include the commercial software MathPix and open source software [5, 6, 7]. An algorithmic approach with good results was taken by InftyReader [8]. Our goal is not to create a new solution but to take advantage of existing solutions to ease the burden on remediators. In the future, we expect that the work of the LaTeX project [9] will vastly reduce the need for remediation.

We tested the tools listed above with Foxit PDF Editor to remediate math in various types of PDF files with unsatisfactory results. Some of the tools generate LaTeX and some AsciiMath. The resulting experience is therefore dependent also on the conversion to MathML. It is important to note that the quality of the source file, especially with old scanned documents played a significant role. To provide a comprehensive comparison of OCR tools we decided to show results on a set of images of math provided by Foxit PDF Editor to each tool and compare results in MathML if the tool directly outputs it or in a native format. Table 1 shows a simplified version of the table with our evaluation of the result in percentage. See the github repository for original test files and detailed results.

4 Future work

Full compliance with PDF/UA-2 requires supporting more PDF 2.0-specific features that are mandatory. Many of these features like labeling, a combination of various namespaces, and navigational elements can all help with math consumption. We are planning on adding support for all PDF 2.0 features in our consumption and creation tools.

While UA-2 is a file format specification, we are lacking processor requirements and proper guidance. It is not clear what should a processor do if, for example, the authors provide only source representation of math in LaTeX with missing MathML representation. We are expecting further discussion in PDF community with the hope of reaching a consensus in providing unified experience with various ATs. Our work and participation information is available at https://github.com/foxitsoftware/PDF_UA-2.

‡ Custom build of Foxit Editor with PDF UA-2 features is at: https://github.com/foxitsoftware/PDF_UA-2/tree/main/installer.

	MathPix	InftyReader [8]	Pix2Text [5]	Nougat [6]	LaTeX-OCR [7]
1*	100%	100%	70%	100%	70%
2	100%	50%	100%	70%	80%
3	100%	0	100%	90%	10%
4	80%	0	60%	0	80%
5	100%	50%	0	0	100%
6	100%	100%	100%	100%	100%
7	100%	100%	100%	100%	100%
8	100%	0	90%	0	70%
9	100%	0	90%	80%	100%
10	100%	0	100%	0	100%
11	100%	0	100%	100%	100%

Table 1: Simplified table with results of math recognition

References

1. ISO 32000-1:2008 (PDF 1.7), https://www.iso.org/standard/51502.html.
2. ISO 32000-2:2020 (PDF 2.0), https://www.iso.org/standard/75839.html.
3. PDF Accessibility API Reference, retrieved from opensource.adobe.com/dc-acrobat-sdk-docs/acrobatsdk/PDFs/acrobatsdk_access.PDF on Oct 17, 2023.
4. Soiffer, N. (2021). Accessible PDF: 2≯1. In *The 4th International Workshop on "Digitization and E-Inclusion in Mathematics and Science 2021.* Retrieved October 17, 2023, from https://workshop.sciaccess.net/deims2021/.
5. Off-Line Math Formula Recognition Using Deep Neural Networks, retrieved October 17, 2023, from https://github.com/jungomi/math-formula-recognition.
6. Nougat: Neural Optical Understanding for Academic Documents, Retrieved October 17, 2023, from https://github.com/facebookresearch/nougat.
7. Latex-ocr, retrieved October 17, 2023, from https://github.com/UW-COSMOS/latex-ocr.
8. Suzuki, M., Yamaguchi, K. (2022). Conversion of Multi-lingual STEM Documents in E-Born PDF into Various Accessible E-Formats. In: Miesenberger, K., Kouroupetroglou, G., Mavrou, K., Manduchi, R., Covarrubias Rodriguez, M., Penáz, P. (eds) Computers Helping People with Special Needs. ICCHP-AAATE 2022. Lecture Notes in Computer Science, vol 13341. Springer, Cham. https://doi.org/10.1007/978-3-031-08648-9_2.
9. Fischer, U. Automated tagging of LaTeX documents—what is possible today? In TUG 2023. Retrieved October 18, 2023, from tug.org/tug2023/program.html.

HTML papers on arXiv: why it's important, and how we made it happen

Charles Frankston[1], A. Jonathan R. Godfrey[2], Shamsi Brinn[1], Alison Hofer[1], and Mark Nazzaro[1]

[1] arXiv
cbf@arxiv.org, shamsi@arxiv.org, ahofer@arxiv.org, mark@arxiv.org
[2] Massey University, New Zealand
a.j.godfrey@massey.ac.nz

1 Introduction

arXiv is the world's largest and oldest scientific preprint server, and a champion of open science. Started in 1991, arXiv presently holds more than 2.4 million articles and is growing at an ever-increasing rate.

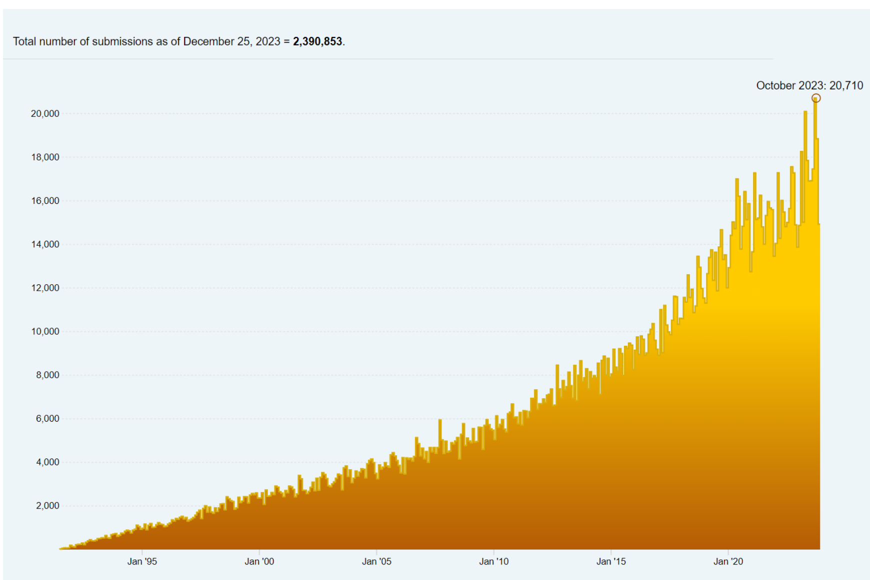


Figure 1: arXiv Monthly submissions from 1991 to present

In many fields of Physics, Math, and Computer Science, cutting edge research is first made available on arXiv. Examples:

- LLM research (OpenAI, Deepmind, etc.)
- LIGO (Gravity wave research; 2017 Nobel Prize in physics)
- Proofs of famous theorems (Grisha Perelman)

arXiv has a mandate to continuously improve access to scientific research, and our long-term mission is to simply serve the needs of the scientific community through openness, collaboration and scholarship. Everyone has the right to participate in the wealth of scientific knowledge contributed to arXiv by researchers from all over the world. Accessibility is inherent to our mission of championing open science, and when we asked scientists with disabilities how arXiv could help make research more accessible they told us in no uncertain terms: add HTML as a format for papers.

Over the past few years, arXiv has made good progress in making our website more accessible according to W3C WAI guidelines. While this allows people with disabilities to more easily find and access papers, they often cannot read them because they are available almost exclusively in PDF format which has low native accessibility.

What we heard from scientists with disabilities, standards experts, and accessibility researchers is that, when it comes to accessibility, PDF will always be playing catch up with HTML. Though we will retain PDFs on arXiv as always, adding HTML as a format will bring us closer to fulfilling the promise of truly open science.

2 PDF limitations

2.1 Layout

PDF was originally designed as a digital representation of a printed page and there is frequently a mismatch between the paper's geometry and the screen being used to view it. Zooming in to enlarge the text will often require horizontal scrolling. Responsive Design and content re-flow are not natural characteristics of PDF. Adobe is attempting to address these issues with features like "Liquid Mode", but Liquid Mode is presently mobile-only and Adobe proprietary. It is not clear if this retroactive reinterpretation of PDF document layout will work well in all cases.

2.2 Structure

Screen readers rely on structural elements with semantic meaning to efficiently map and navigate content. In a PDF, the original structure of the paper that screen readers depend on is typically lost. Section headings, captions, links, and more are reduced to typographic elements like font changes and positioning instructions. Mathematical notation in PDF is reduced to symbols and positioning directives, or simply turned into an image.

Adobe introduced tagged PDF to improve this situation, but despite its introduction in the early 2000s the vast majority of PDF documents in the world are not fully accessible. Properly tagging a PDF to make it accessible takes specialized knowledge and proprietary software. Presently, arXiv's pipeline from TeX → PDF has poor support for tagged PDF (though the core LaTeX team is working on a solution). Tagged PDF also has a sub-optimal solution for math: displaying it as an image with an alt-text description for screen readers.

3 HTML

3.1 A better solution

A lot of work has been done to make HTML accessible. When formatted correctly, HTML preserves the structure and intent of the document and renders fluidly on mobile devices. There is a rich ecosystem of assistive technologies and tools that build on this foundation.

For most people, HTML is the gateway to all the content the internet has to offer. But PDF still dominates scientific publishing and there is a substantial body of scientific work online that is available only in PDF format.

Converting that PDF content to HTML poses challenges, even using modern AI-assisted techniques. The Allen Institute for AI created such a converter, SciA11y. While it produces some nice looking HTML documents, the job it is trying to do—reconstruct the paper's structure that was lost in the process of producing the PDF—is fundamentally difficult, and the SciA11y convert has serious limitations. To quote from Wang et al. [9]:

> In the current iteration of the HTML render, we do not display author affiliations, footnotes, or mathematical equations due to the difficulty of extracting these pieces of information from the PDF.

In conversations with the SciA11y team, they agreed that, in the case where source TeX files are available for a paper, it is better to produce HTML starting from that source than use their tool to convert from the PDF.

3.2 A Strong HTML Ecosystem

Because use of the World Wide Web has become fundamental to human existence, a great deal of effort has been expended to make the web accessible. There's a vibrant ecosystem of screen readers for HTML web pages and the open-source structure of HTML enables the creation of add-ins that can change page appearance to aid visually impaired and people with dyslexia. Dark mode is now widely supported for those with light sensitivities, while adjusting font size is trivial on a well formatted HTML page. Modern web browsers also build in useful features like language translation, which expands access in a different way (something that is at-present not smoothly supported in any PDF viewer we know of). An well-formatted article will re-flow to present nicely on small devices like mobile phones, which sometimes might be the only internet device available to scientists in poorer countries.

In the early days of the web, text in PDF documents was not indexed by search engines. This isn't the case anymore, but PDF documents make search engines work harder – particularly PDFs where much of the text is actually represented in image form. By comparison, text harvesting from HTML documents is relatively easy. arXiv's corpus is a substantial part of nearly all the Large Language Models today, and we've been told by researchers attempting to use our corpus that having HTML versions of documents available would support their work.

4 TEX and LATEX

TeX is a computer program that Donald Knuth wrote in the late 1970s to automate the typesetting of his books on the Art of Computer Programming [1, 2]. It was born out of his frustration with the typesetting systems of the times, particularly as it related to mathematics and tables. TEX is similar to other "text formatters" that existed at the time, such as runoff and Scribe, but is uniquely capable in that TEX allows programming constructs that are powerful enough to make it Turing complete. This is both a blessing and a curse. The blessing is that users can create very sophisticated TEX documents that layout their papers exactly as they wish. The curse is that it makes it difficult to re-interpret those papers for a target other than the printed page they were intended for.

4.1 LATEXand reinterpretation

TEX's powerful programming features enabled Leslie Lamport to create a package of extensions called LATEX [7]. LATEX defines tags/macros to create structured documents. Tags look like: `\section` or `\paragraph`. Since almost all scientific documents built with TeX use LaTeX now, in theory it should be easy to convert LaTeX tags to HTML, for example: `<h1>` or `<p>`. In practice most LaTeX papers have additional complexity. But the semantics of tags like `\section` only partially match up to `<h1>`. (`\section` has more semantics than `<h1>` — mostly related to print formatting.)

Very few papers submitted to arXiv use 'base' LaTeX. Most commonly, the paper will use a package of settings and extensions intended to produce papers that are formatted in a particular manner. Many of these packages are document templates provided by journals or societies so authors can deliver papers formatted according to that publisher's standards.

For example, the American Physical Society offers the Physical Review Journal's REVTeX template, built on LaTeX. A document formatted using REVTeX will be "publication ready" for inclusion in an APS journal. These templates ease the burden for authors, but at the same time complicate the job of converting from LaTeX. There are hundreds, if not thousands, of such templates; some are provided by large societies, some by small; some are just extensions shared amongst individual groups of scientists; and some eager scientists create extensions just for their own use, sometimes because they want to fine-tune the layout of their papers on a Letter size or A4 page. They might do this fine-tuning in ways that look quite wrong in any other medium or format.

5 So what's the problem?

90% of arXiv's papers are submitted as TeX (lately almost all LaTeX) format. So why can't we just run the TeX compiler to output to HTML instead of PDF?

Unfortunately that is not supported and is not easy to support. Donald Knuth was concerned with the precise typesetting of his technical content. The TeX compiler he released in 1978 is exclusively intended for that purpose. Even though LaTeX adds structure to the TeX language, that structure is almost completely lost at compile time when the LaTeX constructs are converted back to TeX primitives.

The core LaTeX team (see latex-project.org), headed by Frank Mittelbach, is working on revising TeX/LaTeX processing to preserve the structured information. They are also adding new constructs to LaTeX for things like alt-text. Currently, their work is exclusively focused on producing better tagged PDF and is several years from completion. But we are hopeful that as this project nears completion it will be easy to produce structured HTML with a small amount of additional work.

5.1 But my journal offers HTML now

Many journals offer HTML for the online versions of their papers, including those submitted as LaTeX. Why can't arXiv use the same techniques that journals use? The simple answer is that the journals accomplish this through a combination of tooling (that might even be specific to that journal's LaTeX template), and manual labor. There are a specialized providers of formatting and accessibility remediation for publishers that can perform this work as a "reasonable" cost. But while it may be reasonable for a journal to spend $30, $50, or even $100 to prepare an article for publication, it's not feasible for arXiv. We now process around 20,000 papers per month, with most papers getting announced within 24 hours of submission. There is simply no budget for manual or custom processing, and we have not found any service that can keep up with our announcement pace.

It must continue to be the case that the labor for producing an article on arXiv is provided entirely by the author. At the same time, we cannot burden our authors with additional work to generate an HTML version of their papers, or to become accessibility experts. (Note: we are not averse to encouraging authors to make small adjustments to their submissions to help us produce better HTML, but it should be reasonably easy and because the authors wish to do so.) So we began our quest for an entirely automated solution.

5.2 arXiv's pragmatic approach

We know that researchers with disabilities need solutions now and did not want to wait for the LaTeX team to finish their tagging work, so we investigated existing tools. There have been at least ten attempts to create TeX/LaTeX to HTML tools (see texfaq.org for a partial list).

We evaluated all the tools we could find and the conclusion was that the best tools were:

- LaTeXML maintained by Bruce Miller and Deyan Ginev at NIST (National Institute of Standards and Technology)
- Tex4ht, created by Eitan M. Gurari, now maintained by Michal Hoftich

These two tools were roughly tied in the quality of the HTML produced, but LaTeXML has a larger library of supported packages, and the predecessor ar5iv Labs project at arXiv used LaTeXML, which made it a logical choice.

5.3 The ar5iv Project

ar5iv was a project started by Dr. Michael Kohlhase from KWARC, and Ph.D. student Deyan Ginev (see [8]). The intent was to offer HTML versions for arXiv's entire LaTeX corpus, using LaTeXML for the conversions.

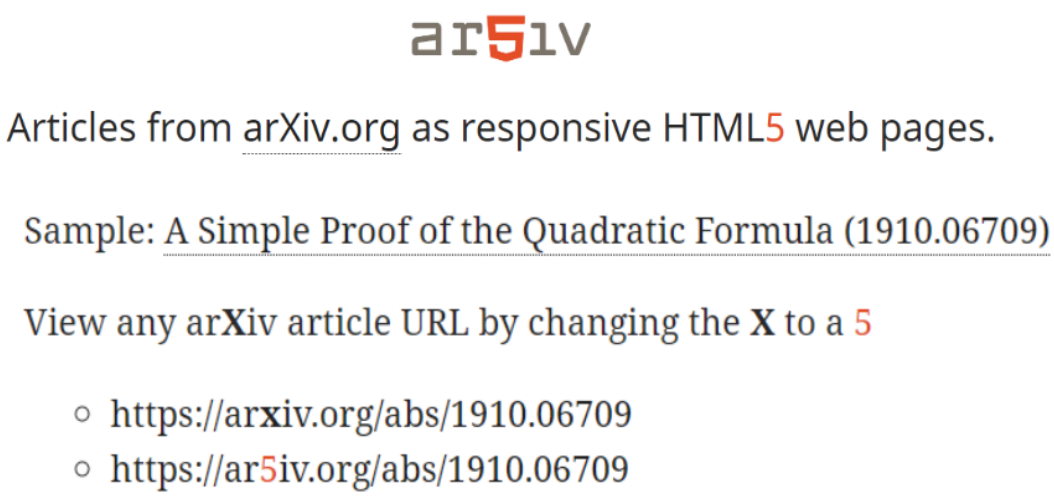


Figure 2: ar5iv.org home page

It mostly succeeded at this, but some 25% of papers had conversion errors. (Note that papers with conversion errors may still be readable. Only 3% totally fail conversion). As LaTeXML coverage has improved success rates have increased and errors decreased, but old articles must be reconverted to remove glitches. This is computationally expensive. At an average of $0.015 per article on Google Cloud, it would cost $30,000 every time we wanted to reconvert the entire arXiv corpus. Fortunately, the KWARC team in Germany has generously provided the ar5iv project with access to compute resources to run the conversions, but it takes several weeks to re-convert the entire corpus.

In many ways, arXiv's current effort to offer HTML for all papers, in parallel to the PDF, can be considered bringing the HTML effort in-house.

5.4 The difficulty of converting LaTeXto HTML

LaTeXML, like nearly all the TeX to HTML tools, does not use the actual TeX compiler. As explained earlier, that compiler in its present state loses structural information. LaTeXML therefore does its own processing of the TeX/LaTeX documents and packages, emulating just enough of TeX to preserve the information provided by the author. Since the LaTeXML compiler is does not (yet) implement a complete TeX engine many macros and extensions have to be handled as special cases. That is, the LaTeXML converter has built in code to handle the use of some packages, rather than trying to interpret the actual package definition as the core TeX engine would. The LaTeXML team has done such special case implementations for more than 400 of the most commonly used LaTeX packages in arXiv, but that still leaves a considerable number of packages it does not recognize.

5.5 Unfinished work

In addition to "one-offs"—papers where clever scientists defined their own macros and extensions—there is still a long tail of less common packages, and some popular packages that are hard to implement are not yet supported as well. The most substantial example is tikz, a popular diagramming package that produces vector graphics from a geometric description.

The LaTeXML team is currently working on improving their raw interpretation of TeX to automatically cover the long-tail of missing packages, reducing the number that need to be handled as special cases. And there are still some LaTeX constructs that in edge cases don't render quite right. The present conversion results are mostly successful, but many papers still have rendering glitches, although most are minor. Offering HTML papers now, even with rendering glitches, is far better than nothing and brings immediate benefits to scientists with disabilities and improves legibility on mobile devices.

Even papers that might use an unsupported package can still be useful for readers, as the paper might only use the construct defined in that package in one or two places. If the LaTeXML converter doesn't recognize a LaTeX command (because it didn't know about the package that defined it), it will simply output that command to the text of the paper. arXiv readers who are working members of the scientific community will often be able to deduce the meaning of the construct, and thus still be able to fully understand the content of the paper. This would also be true for those using screen-readers who have some familiarly with LaTeX.

5.5.1 Sample HTML Paper

Figure 3 below shows a sample HTML paper. Note:

- the "Experimental" label
- the prominent "Report Issue" button

We are describing the whole HTML effort as an experiment. This is to make the arXiv community aware that the present iteration is not perfect, and to expect improvements in the future as the "experiment" continues.

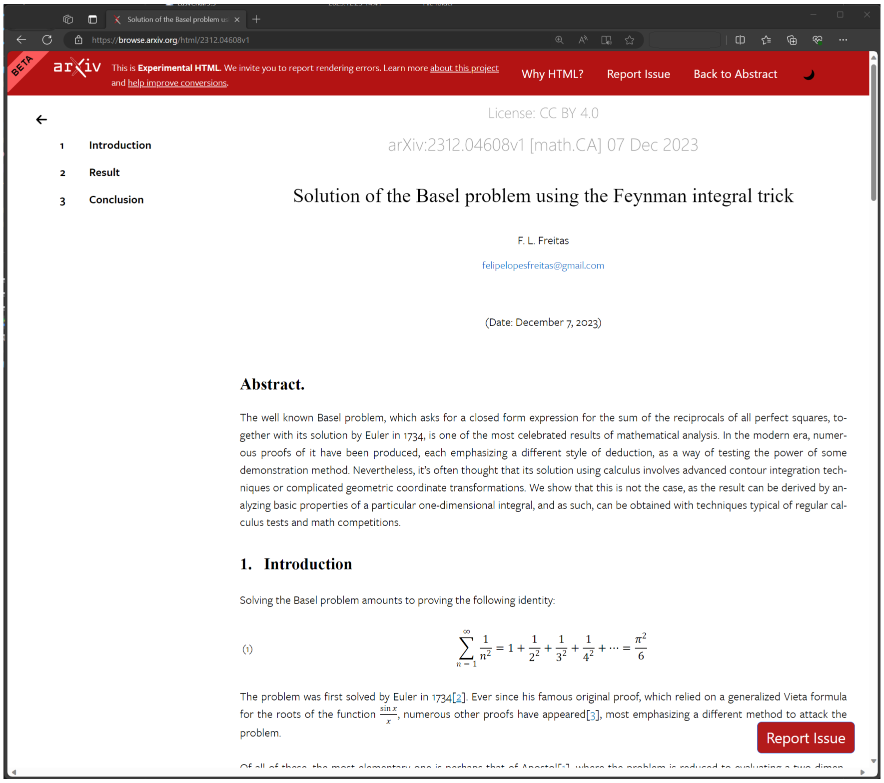


Figure 3: Sample HTML paper

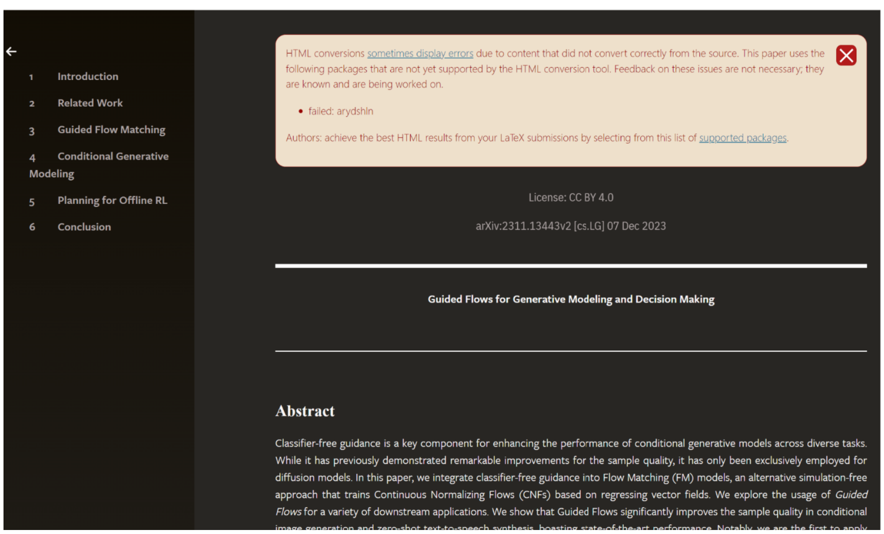


Figure 4: HTML paper sample, dark mode

The “Report Issue” button is present on all the HTML documents so the arXiv community can help us identify issues in LaTeXML. The developers of LaTeXML indicated to us that they would welcome bug reports on LaTeX constructs that did not render or read correctly. The button automatically records which paper the issue is on and captures a snippet of the relevant content when there is an active text selection. This has quickly produced a number of bug reports, largely of good quality. The main issue now is duplicate issue reports. We plan to improve the bug reporting feature and help reduce duplicates by showing users the reports already received on the same article.

Figure 4 above shows a paper rendered in dark mode, and a message shown at the top of any paper that called for package that LaTeXML didn't know about – an indication to readers that most likely something in this paper isn't going to render correctly. Dark mode is automatically chosen if that is the current setting for the user's browser. Dark mode has become a popular preference in recent years and is a common accessibility issue for those with light sensitivities or viewing their screen under suboptimal conditions like glare or low light.

6 Early Impact

We rolled out HTML formatted papers on arXiv on December 1, 2023. These are early days, but the reaction from the scientific community has been very positive. Most importantly, initial testing with screen reader and Braille users confirm that it is a much better experience than PDF. One early tester told us “pleasingly, in the most part the Braille translation is very good here.” We heard from another that “the reading of the HTML file by VoiceOver is impressive even for mathematical formulae.” We have also heard from many people who can now read papers on their mobile devices, and appreciation for being able to easily adjust font size and use dark mode.

This early feedback is heartening, but there is a long way to go. As noted earlier, we are receiving feedback from authors and readers about errors and issues they are coming across in some papers. However, we are keeping in mind that scientists with disabilities asked us to not let the perfect be the enemy of the good. Our ongoing challenge is to balance author's desires to present the highest quality version of their work with the tremendous accessibility gains from even imperfect HTML. We look forward to building on this important foundation in the future weeks, months, and years.

7 Future Work

- Continue to work with the LaTeXML team to improve the conversion process.
- Figure out a cost-effective way to periodically re-compile the whole corpus to pick up these improvements.
- Revisit tooling in a few years when the LaTeX team is further along. We are hopeful that the work to produce tagged PDFs will also enable the generation of HTML output using the core TeX engine.
- Make charts and graphs more accessible, possibly by providing a way for users to access the data behind the graph.
- Explore pathways to better image captions, including auto-caption or crowd-sourcing missing alt-text. It is a pressing accessibility issue and challenging AI problem. An early user tester shared a long-standing wish that “it would be great if there was a system that incentivised authors to provide their own alt text.”

- Continue to listen to feedback from authors, and share best practices as we learn more, working together to improve HTML papers on arXiv.

8 Acknowledgements

Kudos to Paul Ginsparg & co for insisting from the start in 1991 that scientists submit the source code for their papers, instead of just the PDF, which means we can now at least try to convert the 90% of our corpus that is in TEX to HTML.

We thank the many scientists with disabilities who so generously shared their expertise, insights, and feedback, and guided arXiv's efforts in the most useful direction.

This material is based upon work supported by the National Science Foundation under Award No. OAC-2311521 and by NASA under award No. 20-OSTFL20-0053. Any opinions, findings and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation or of NASA.

References

[1] B. Beeton, K. Berry, and D. Walden. "TeX: A Branch of Desktop Publishing Evolution, Part 1". In: *IEEE Annals of the History of Computing* 40.03 (July 2018), pp. 78–93. ISSN: 1934-1547. DOI: 10.1109/MAHC.2018.033841114.

[2] B. Beeton, K. Berry, and D. Walden. "TeX: A Branch of Desktop Publishing Evolution, Part 2". In: *IEEE Annals of the History of Computing* 41.02 (Apr. 2019), pp. 29–41. ISSN: 1934-1547. DOI: 10.1109/MAHC.2019.2893731.

[3] Jeffrey P. Bigham et al. "An Uninteresting Tour Through Why Our Research Papers Aren't Accessible". In: *Proceedings of the 2016 CHI Conference Extended Abstracts on Human Factors in Computing Systems.* Available at https://doi.org/10.1145/2851581.2892588. 2016.

[4] Shamsi Brinn et al. *A framework for improving the accessibility of research papers on arXiv.org*. 2022. arXiv: 2212.07286 [cs.DL].

[5] Donald E. Knuth. *Computers & Typesetting*. Reading, Mass.: Addison-Wesley Publishing Company, 1984-1986.

[6] Leslie Lamport. *LaTeX: A Document Praparation System*. Reading, Mass.: Addison-Wesley Publishing Company, 1986.

[7] Frank Mittelbach. "LATEX anniversaries – A look into two direction". In: *TUGboat* 44.1 (2023), pp. 77–86. DOI: 10.47397/tb/44-1/tb136mitt-history.

[8] Heinrich Stamerjohanns et al. "Transforming Large Collections of Scientific Publications to XML". In: *Mathematics in Computer Science* (2010).

[9] Lucy Lu Wang et al. *Improving the Accessibility of Scientific Documents: Current State, User Needs, and a System Solution to Enhance Scientific PDF Accessibility for Blind and Low Vision Users*. 2021. arXiv: 2105.00076 [cs.DL].

Accessible Papers from Source: The Quest for the Holy Grail

Zenaib Ali[1] and David Austin[2] Volker Sorge[3,4]

[1] Kebab CA, Ltd., London, UK
[2] Grand Valley State University, Allendale, MI, USA
[3] University of Birmingham, UK
[4] Indian Institute of Technology, Delhi, India

Abstract

We present an approach to authoring scientific content that is fully web accessible out of the box, despite including math formulas, diagrams and code snippets. Our work is based on PreTeXt, a format that facilitates the creation of accessible scholarly documents. A PreTeXt author describes the structure of their document in XML while keeping the mathematical content in standard LaTeX. As generating accessible math from LaTeX sources is effectively a solved problem, we concentrate on the hard problems of providing access to diagrams as well as listings and code snippets. For the former we provide a PreTeXt extension with a relatively small, but intuitive, vocabulary that allows all authors to generate accessible mathematical diagrams easily by creating a high-level description of its mathematical components. For the latter we employ techniques from code indentation and highlighting, to compute aria annotations that allow for meaningful grouping, voicing and navigation of code snippets in structured programming languages.

1 Background

Although mathematical literature forms the basis of many scientific and technical (STEM) subjects, it still poses a considerable hurdle for Visually Impaired (VI) learners. STEM content in particular in the mathematical and computational sciences is generally comprised of a mixture of text, mathematical formulas, diagrammatic images and listings of pseudo code or code snippets.

A plethora of techniques have been researched to turn text and formulas into accessible content, from sources like LaTeX[3] or by adapting documents retrospectively [11]. However, relatively little work has been done to enable the automatic generation of accessible content directly and automatically for the whole gamut of STEM artifacts, including diagrams and code snippets, from one unique source.

While specialist authoring tools like Equatio [10] or Desmos [6] can generate accessible content they are not only restricted to certain graphics only (e.g., xy-graphs) but need to be used outside the usual authoring workflow and often included explicitly into published material on the web. Similarly well structured and authored source code is often mishandled in the publishing pipeline, leading to badly structured content or even turned into images in online publications, thus making it inaccessible.

Our approach aims to provide an automatic means to make all these components accessible from a single source while not disturbing the creative process of the authors. We use the PreTeXt format as a basis, a language that facilitates the creation of accessible scholarly documents. A PreTeXt author describes the structure of their document in XML while keeping the mathematical content in standard LaTeX. Likewise, graphics or code can be input in the formats authors are familiar with, such as Tikz [9] and various listing environments for code.

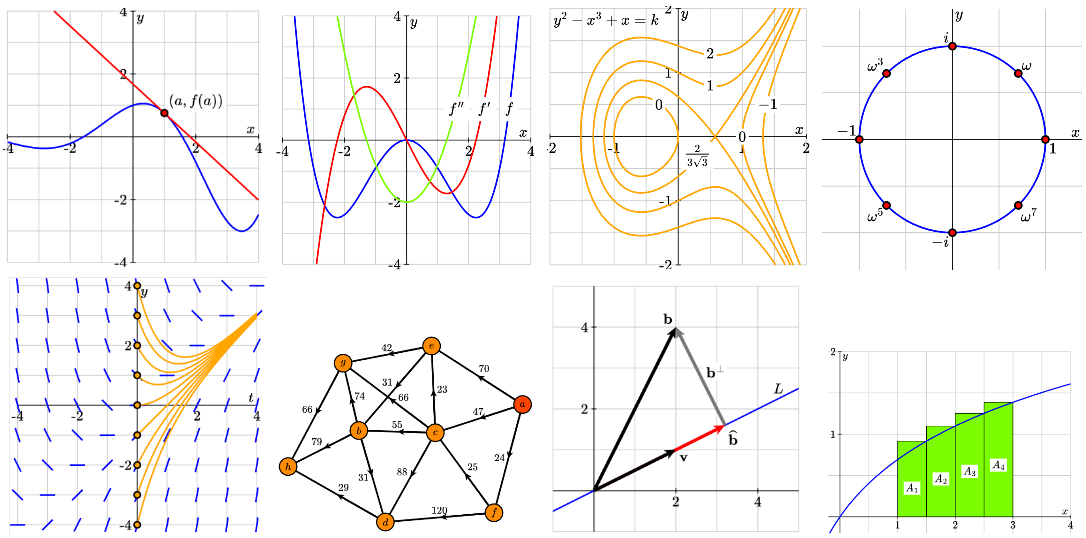


Figure 1: Examples of targeted mathematical diagrams.

Thus, similar to LaTeX, documents are written without concern for their visual appearance while authoring, and PreTeXt will create an accessible version of the document in a variety of formats, such as HTML, PDF, ePub, or Braille.

Within a document the mathematical content is straightforwardly rendered and made accessible using MathJax [5] putting no onus on the author. Similarly code snippets need no special attention and are made web accessible using the Shellac library [1] as describe in Section 3. The only constraint imposed on the author is the need to replace the Tikz formalism with a simple XML based language decribed in the next section.

2 Authoring of Accessible Diagrams

The graphical authoring langugage is designed in a similar spirit to PreTeXt and allows authors the ability to create accessible graphics directly in their documents. It thereby is not restricted to a particular set of diagrams but aims to deal with as broad a selection as possible. Examples of the type of diagrams we currently handle are given in Figure 1.

Accessible mathematical diagrams are authored by creating an XML description of their mathematical components. Annotations are likewise included in a simple format that specifies how they are to be traversed using a screen reader. A Python script converts XML input directly into SVG output that be viewed in a browser or sent to an embosser. The basic diagram vocabulary, designed to mirror mathematical vocabulary, drives the SVG generation and can roughly be divided into five functional groups: (1) definitions, control, and grouping, (2) basic grids and axes, (3) one-dimensional objects, like graphs and lines, (4) two-dimensional objects, such as areas and polygons, and (5) labels.

As an example, Table 1 gives the XML description used to create the top-left diagram in Figure 1. The `graphics` element specifies the basic dimension of the output graphics. Note that since we are constructing a scalalable vector graphics this dimension only serves to give the aspect ratio of the resulting diagram and to assist PreTeXt when compiling the diagram

```
<graphics id="figure" width="300" height="300" margins="5">
  <definition> a = 1 </definition>
  <definition> f(x) = exp(x/3)*cos(x) </definition>
  <boundingbox mbox="[-4,-4,4,4]"/>
  <grid-axes xlabel="x" ylabel="y" />
  <group id="graph-tangent">
    <graph id="graph" function="f" stroke="blue" />
    <tangent-line id="tangent" function="f" point="a" stroke="red" />
    <point id="point" p="(a, f(a))" fill="red" label="(a, f(a))" />
  </group>
</graphics>
```

Table 1: XML specification for the top-left image in Fig 1.

```
<annotations>
  <annotation id="figure" text="The graph of a function and its tangent line
                              at the point a equals 1">
    <annotation id="graph-tangent" text="The graph and its tangent line">
      <annotation id="graph" text="The graph of the function f" sonify="yes"/>
      <annotation id="point" text="The point a comma f of a"/>
      <annotation id="tangent" text="The tangent line to the graph of f at the point"/>
    </annotation>
  </annotation>
</annotations>
```

Table 2: The annotations that accompany the example diagram.

for inclusion in PDF output. Notice also that several graphical components, such as the graph, tangent line, and point, may be grouped together to facilitate the diagram's exploration with a screen reader through the set of author-supplied annotations.

The annotations that accompany our example diagram are given in Table 2. Annotations are included separately from the graphical descriptions since the order in which elements are drawn may differ from how they are traversed in a screen reader. The set of annotations forms a hierarchical structure that begins with a global description of the diagram and proceeds inward by nesting annotations to examine graphical elements in greater detail. This provides the basic interaction model that allows readers to explore the diagram in detail. Some common elements, such as the grid and axes, are annotated automatically, leading to a standardised user experience for VI learners when screen reading and for VI authors when inspecting the diagram during its creation. In addition, authors can provide bespoke manual annotations directly.

The diagram's XML description can then be translated into a fully accessible SVG diagram that can be inspected in a browser either visually or interactively via aural rendering of the annotations using the diagcess library [8, 7]. This library also supports custom color adaptation, highlighting and magnification as well as sonification of curves.

A second option allows the author to translate the diagram's XML description into an SVG that may be sent to an embosser resulting in a tactile realization of the diagram. Labels are converted into braille using MathJax [5] and are positioned carefully to align with the embosser's capabilities. The sizes of some components, such as points, are scaled to be easily detected by touch, and foreground features, such as lines and points, are outlined to create an empty "moat" so that these features clearly stand out from the background.

Using these tools, a full set of tactile diagrams was created for the open-access precalculus textbook *Active Prelude to Calculus* [4]. Two VI readers of the book provided guidance on certain decisions, such as an appropriate size for points and the width of moats, and reported that the tactile diagrams were a helpful companion to the textbook, which they read online using a screen reader.

3 Accessible Listings

Listings are an integral part of many computer science literature but also in many applied mathematical subjects, where algorithms and procedures are presented via pseudocode or concrete code snippets or exercises are given for use in computational systems. Although code generally is well-structured ASCII syntax, in online publishing listings are often poorly formatted or even transformed into graphical elements with poor alternative text leaving them for all practical purposes inaccessible for contemporary assistive technology.

We use a the Shellac library [1] as a plugin in PreTeXt, to make code snippets accessible in a structured fashion. Shellac itself uses tree-sitter [2], an incremental parsing library for highlighting and indentation. This allows Shellac to work with arbitary structured languages and exploit their abstract syntaxt trees to add aria notation for accessibility and to impose reading order.

In detail, Shellac computes a concrete syntax tree for a give piece of code, then transforms that tree into an html tree structure. The html tree is formatted using CSS grid and made accessible using the aria tree view. That is, elements are annotated with `tree`, `group`, and `treeitem` roles and `aria-labels` are added for summarizing code blocks.

Note that Shellac is not limited to brace-based languages but can run on any language with a tree-sitter grammar including markup languages like LaTeX and XML. For example the listing in Table 1 is made accessible by providing groups to for each of the tags, that either include the tag itself if it self-closing or the XML subtree that is rooted in the tag. The aria tree view then allows the reader to dive into the content of a group either by exploring the tags in the subtree or the content of the tag itself. For example, while initially the entire code block would be focused, pressing the down arrow focuses on the initial tag line as shown below

```
<graphics id="figure" width="300" height="300" margins="5">
  <definition> a = 1 </definition>
  ...
```

The screenreader would pick up the associated aria label and read "tag graphics with four attributes". The right arrow key allows us to switch focus and read the next code block or tag:

```
<graphics id="figure" width="300" height="300" margins="5">
  <definition> a = 1 </definition>
  ...
```

Alternatively, we could dive deeper into the graphics tag with the down arrow, to explore tag name and attributes step by step. For instance, stepping through the first tag line we get `graphics` followed by `id="figure"`, `width="300"`, etc. Attributes are announced and spoken as a key-value pair, for example "Attribute id with value figure". Each attribute can again be further explored, which can be helpful for comprehending more complex attribute values.

References

[1] Shellac: An accessible code syntax highlighter. https://git.sr.ht/~zainab.

[2] Tree-sitter. https://tree-sitter.github.io/.

[3] Dragan Ahmetovic, Tiziana Armano, Cristian Bernareggi, Anna Capietto, Sandro Coriasco, Doubrov Boris, Kozlovskiy Alexandr, Nadir Murru, et al. Automatic tagging of formulae in pdf documents and assistive technologies for visually impaired people: the latex package axessibility 3.0. In *ICCHP open access compendium" Future Perspectives of AT, eAccessibility and eInclusion"*, volume 1, pages 69–73. ICCHP, 2020.

[4] Matthew Boelkins. *Active Prelude to Calculus*. 2020. http://activecalculus.org.

[5] Davide Cervone and Volker Sorge. Mathjax v3.2. http://mathjax.org, April 2022.

[6] Desmos. Graphing calculator. www.desmos.com/calculator.

[7] A. Jonathan R. Godfrey, Paul Murray, and Volker Sorge. An accessible interaction model for data visualisation in statistics. In *International Conference on Computers Helping People with Special Needs*. Springer, 2018.

[8] Volker Sorge, Mark Lee, and Sandy Wilkinson. End-to-end solution for accessible chemical diagrams. In *Proceedings of the 12th Web for All Conference*. ACM, 2015.

[9] Till Tantau. *The TikZ and PGF Packages*, 2023. https://tikz.dev/.

[10] Texthelp. Equatio. www.texthelp.com/en-gb/products/equatio.

[11] Katsuhito Yamaguchi, Toshihiko Komada, Fukashi Kawane, and Masakazu Suzuki. New features in math accessibility with infty software. In *Computers Helping People with Special Needs*, pages 892–899. Springer, 2008.

Fully Accessible PDF/UA documents. Case study: NOAA fish stock reports.

Ross Moore

Macquarie University, Sydney, Australia
ross.moore@mq.edu.au

Abstract

It has frequently been said that PDF files are not accessible, yet much work has been done to develop standards such as PDF/UA to add 'accessibility features' into PDF documents. Using an 80+ page real-world PDF/UA document, built using LaTeX methods, its accessibility is tested using the PAC 2021 validation software, passing all tests with neither violations nor warnings. Yet there is no testing of a significant number of the WCAG recommendations.

To better test we "derive" PDF/UA documents to HTML-5 format and check Accessibility using the AInspector for Firefox browser plug-in; again with neither violations nor warnings, but with numerous tests marked for 'Manual Checking'. In particular, top-level structuring via 'Landmarks' and 'Regions' is confirmed, as well as all hyperlinks having a meaningful and unique 'Accessible Name'. Links, images and other structures have 'Accessible Descriptions', which allow a better understanding of the purpose of the content by a non-visual reader.

With some manual checking of these, helped by an understanding of how the 'Accessible Name' and 'Accessible Description' are constructed, one concludes that the appropriate guidelines are indeed satisfied, thus resulting in a document that is very Accessible. Since the HTML page was derived from the PDF/UA, one concludes that the PDF itself contains all the information to support it being very Accessible also. The only problem is that there may not yet be readily available software that can extract the information in a way that is sufficiently convenient for users with disabilities.

Conclusion: one should not complain that the PDF format is inaccessible; rather, ask for better software that can take advantage of the features that PDF/UA can make available. And similarly demand that publishers prepare PDF/UA documents that are enriched with these features.

Non-readability of a PDF document.

Most PDF documents are built using compression, not just of images, but encryption of entire document segments including textual portions, as in Figure 1. There are many advantages to using compressed PDFs[1]. While such a file may be able to be opened with simple text editor software, and some parts of it may well be readable, most of the information content cannot even be located, let alone extracted. Any attempt to edit it will almost certainly compromise the integrity of the file. This is true for a fully-sighted user, as much as for a non-visual one.

When uncompressed, a PDF file can be seen to be essentially a container for numerous resources known as 'objects', together with an '`xref`' table listing the byte offset from the start of the file to where each object commences. The filesize of the PDFs used for the figures here is roughly 6.8 Mbytes compressed, but ≈34.8 MB uncompressed, containing ≈9600 objects. For each visual page, there is a 'content stream' containing the graphics instructions to place each

[1] https://www.quora.com/What-are-the-advantages-of-using-compressed-Portable-Document-Format- PDF-files-over-normal-not-compressed-PDFs

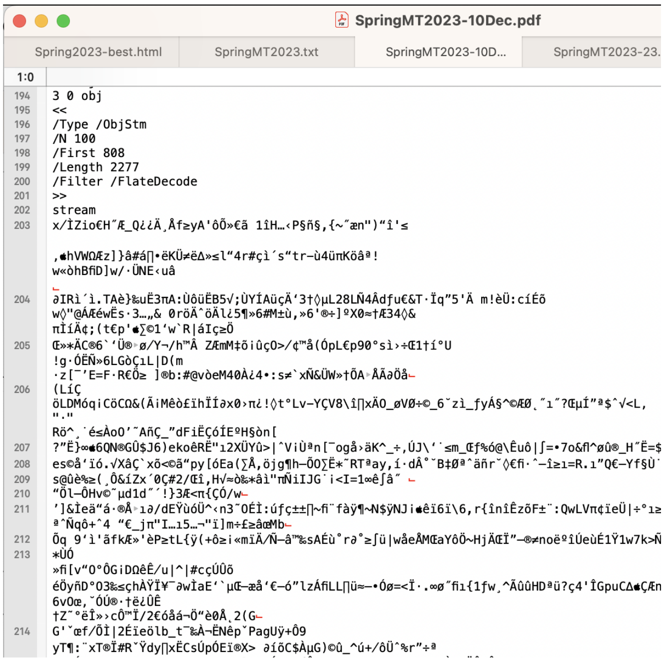


Figure 1: Encryption of portions of a PDF document — intentionally unreadable.

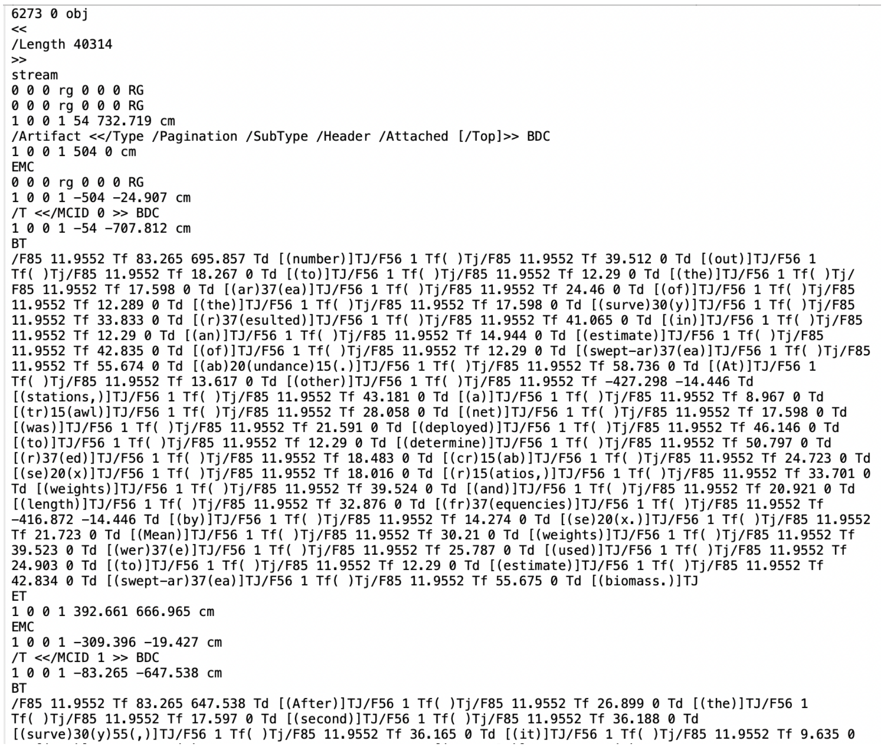
word onto that page, often broken into pieces. This is mixed-in with commands for selecting the font, and exact positioning on the page. Other non-visible page elements can be present, especially when tagging is included. This is still quite intractable, as in Figure 2, though some (parts of) words can be identified. The length (in bytes) is also included with such stream objects. If a stream object were to be edited, resulting in a change of length, then not only does the new length need to be recorded, but the byte offsets to every subsequent object would need to be adjusted in the '`xref`' table. This can amount to thousands of edits for just a simple adjustment; clearly this is not a viable way to do it.

Put simply, text-editing software is just not appropriate for handling a PDF file.

Proper text-extraction from a PDF document.

Clearly special software is needed to sift through the complexity of a PDF file's contents in order to build the picture for on-screen display, or to just locate all the textual pieces. One such well-known piece of software is Adobe's free Acrobat Reader[2]. As well as displaying a high-quality visual view, this allows Copy/Paste of the contents; see Figure 3(a). It puts all the word pieces back together, modulo hyphenation, but gives no real idea of the semantics and includes 'Artifact' text such as the running headers and footers including page-numbering.

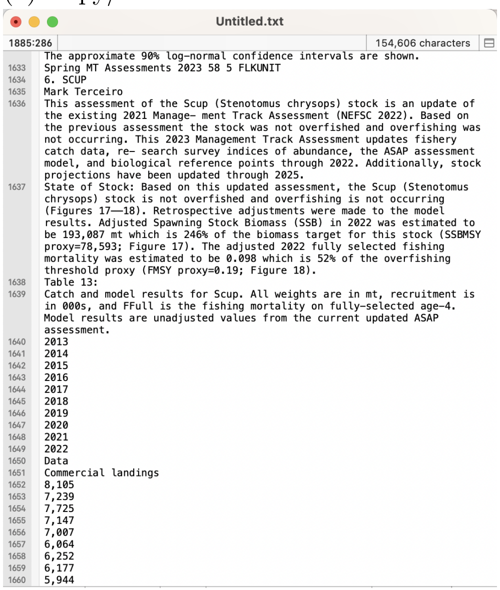
Acrobat Reader[2] also allows the contents of a PDF to be 'Read Out Loud', as well as saving as Text(Accessible); see Figure 3(b). This latter method is saving the words that would be spoken; which can include extra 'spoken tags' to convey semantics, should the PDF have been constructed to include such a feature. With punctuation not being explicitly spoken, the ';' and ':' characters introduce slight pauses; just a little longer than that of a comma. Notice that 'Artifacts' are excluded, and acronyms have been setup to be spoken letter by letter. With



```
6273 0 obj
<<
/Length 40314
>>
stream
0 0 0 rg 0 0 0 RG
0 0 0 rg 0 0 0 RG
1 0 0 1 54 732.719 cm
/Artifact <</Type /Pagination /SubType /Header /Attached [/Top]>> BDC
1 0 0 1 504 0 cm
EMC
0 0 0 rg 0 0 0 RG
1 0 0 1 -504 -24.907 cm
/T <</MCID 0 >> BDC
1 0 0 1 -54 -707.812 cm
BT
/F85 11.9552 Tf 83.265 695.857 Td [(number)]TJ/F56 1 Tf( )Tj/F85 11.9552 Tf 39.512 0 Td [(out)]TJ/F56 1
Tf( )Tj/F85 11.9552 Tf 18.267 0 Td [(to)]TJ/F56 1 Tf( )Tj/F85 11.9552 Tf 12.29 0 Td [(the)]TJ/F56 1 Tf( )Tj/
F85 11.9552 Tf 17.598 0 Td [(ar)37(ea)]TJ/F56 1 Tf( )Tj/F85 11.9552 Tf 24.46 0 Td [(of)]TJ/F56 1 Tf( )Tj/F85
11.9552 Tf 12.289 0 Td [(the)]TJ/F56 1 Tf( )Tj/F85 11.9552 Tf 17.598 0 Td [(surve)30(y)]TJ/F56 1 Tf( )Tj/F85
11.9552 Tf 33.833 0 Td [(r)37(esulted)]TJ/F56 1 Tf( )Tj/F85 11.9552 Tf 41.065 0 Td [(in)]TJ/F56 1 Tf( )Tj/F85
11.9552 Tf 12.29 0 Td [(an)]TJ/F56 1 Tf( )Tj/F85 11.9552 Tf 14.944 0 Td [(estimate)]TJ/F56 1 Tf( )Tj/F85
11.9552 Tf 42.835 0 Td [(of)]TJ/F56 1 Tf( )Tj/F85 11.9552 Tf 12.29 0 Td [(swept-ar)37(ea)]TJ/F56 1 Tf( )Tj/F85
11.9552 Tf 55.674 0 Td [(ab)20(undance)15(.)]TJ/F56 1 Tf( )Tj/F85 11.9552 Tf 58.736 0 Td [(At)]TJ/F56 1
Tf( )Tj/F85 11.9552 Tf 13.617 0 Td [(other)]TJ/F56 1 Tf( )Tj/F85 11.9552 Tf -427.298 -14.446 Td
[(stations,)]TJ/F56 1 Tf( )Tj/F85 11.9552 Tf 43.181 0 Td [(a)]TJ/F56 1 Tf( )Tj/F85 11.9552 Tf 8.967 0 Td
[(tr)15(awl)]TJ/F56 1 Tf( )Tj/F85 11.9552 Tf 28.058 0 Td [(net)]TJ/F56 1 Tf( )Tj/F85 11.9552 Tf 17.598 0 Td
[(was)]TJ/F56 1 Tf( )Tj/F85 11.9552 Tf 21.591 0 Td [(deployed)]TJ/F56 1 Tf( )Tj/F85 11.9552 Tf 46.146 0 Td
[(to)]TJ/F56 1 Tf( )Tj/F85 11.9552 Tf 12.29 0 Td [(determine)]TJ/F56 1 Tf( )Tj/F85 11.9552 Tf 50.797 0 Td
[(r)37(ed)]TJ/F56 1 Tf( )Tj/F85 11.9552 Tf 18.483 0 Td [(cr)15(ab)]TJ/F56 1 Tf( )Tj/F85 11.9552 Tf 24.723 0 Td
[(se)20(x)]TJ/F56 1 Tf( )Tj/F85 11.9552 Tf 18.016 0 Td [(r)15(atios,)]TJ/F56 1 Tf( )Tj/F85 11.9552 Tf 33.701 0
Td [(weights)]TJ/F56 1 Tf( )Tj/F85 11.9552 Tf 39.524 0 Td [(and)]TJ/F56 1 Tf( )Tj/F85 11.9552 Tf 20.921 0 Td
[(length)]TJ/F56 1 Tf( )Tj/F85 11.9552 Tf 32.876 0 Td [(fr)37(equencies)]TJ/F56 1 Tf( )Tj/F85 11.9552 Tf
-416.872 -14.446 Td [(by)]TJ/F56 1 Tf( )Tj/F85 11.9552 Tf 14.274 0 Td [(se)20(x.)]TJ/F56 1 Tf( )Tj/F85 11.9552
Tf 21.723 0 Td [(Mean)]TJ/F56 1 Tf( )Tj/F85 11.9552 Tf 30.21 0 Td [(weights)]TJ/F56 1 Tf( )Tj/F85 11.9552 Tf
39.523 0 Td [(wer)37(e)]TJ/F56 1 Tf( )Tj/F85 11.9552 Tf 25.787 0 Td [(used)]TJ/F56 1 Tf( )Tj/F85 11.9552 Tf
24.903 0 Td [(to)]TJ/F56 1 Tf( )Tj/F85 11.9552 Tf 12.29 0 Td [(estimate)]TJ/F56 1 Tf( )Tj/F85 11.9552 Tf
42.834 0 Td [(swept-ar)37(ea)]TJ/F56 1 Tf( )Tj/F85 11.9552 Tf 55.675 0 Td [(biomass.)]TJ
ET
1 0 0 1 392.661 666.965 cm
EMC
1 0 0 1 -309.396 -19.427 cm
/T <</MCID 1 >> BDC
1 0 0 1 -83.265 -647.538 cm
BT
/F85 11.9552 Tf 83.265 647.538 Td [(After)]TJ/F56 1 Tf( )Tj/F85 11.9552 Tf 26.899 0 Td [(the)]TJ/F56 1
Tf( )Tj/F85 11.9552 Tf 17.597 0 Td [(second)]TJ/F56 1 Tf( )Tj/F85 11.9552 Tf 36.188 0 Td
[(surve)30(y)55(,)]TJ/F56 1 Tf( )Tj/F85 11.9552 Tf 36.165 0 Td [(it)]TJ/F56 1 Tf( )Tj/F85 11.9552 Tf 9.635 0
```

Figure 2: Uncompressed portion of a PDF content stream — still intractable.

(a) Copy/Paste



(b) Text(Accessible)

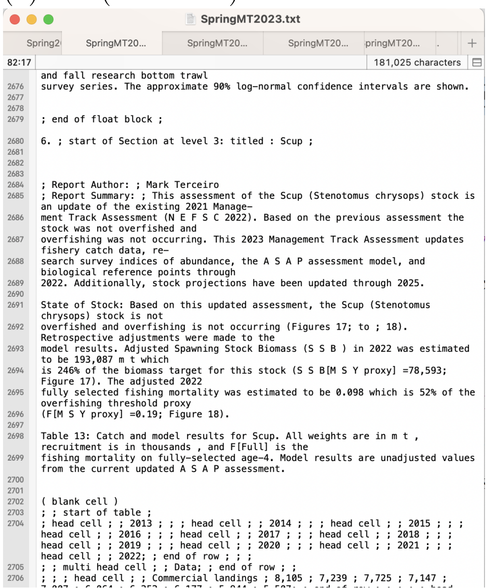


Figure 3: Extraction of text, using Adobe's Acrobat Reader[2] or Acrobat Pro[1]; (a) result from Copy/Paste; (b) result using Save As Text(Accessible).

3

tabular material, row and cell boundaries are clearly identifiable, as well as which cells are meant to be a header for their column or row.

Adobe's Acrobat Pro[1] is a significant step up from Acrobat Reader[2]. It allows alternative ways to view and access the document contents, and Export into other document formats. Although non-free, Acrobat Pro[1] is used by many academic institutions and can be obtained through site-licensing, perhaps bundled with other useful PDF software. There is a slight difference in its reading and Export to Text(Accessible) as follows. With hyperlinks, Acrobat Reader[2] speaks extra words 'jump to destination . . . ', with the dots being the internal identifier of the destination within the document; the anchor text word(s) are lost. On the other hand, it is precisely these anchor text words that are read (and saved) by Acrobat Pro[1], giving a more natural reading; Figure 3(b) actually shows this result. The Reader words could be useful giving an audible indication of a hyperlink to follow — a precursor of the 'Accessible Name' concept (see below) — but there seems to be no way to choose between these behaviours.

This is a vast improvement but we can do even better, using 'Tagged PDF'.

PDF/UA and Accessibility

As well as capturing large scale structural constructs, such as Parts, Sections with Headings, Lists, Tables, Figures, etc. and active elements such as hyperlinks, buttons and other controls, the 'Tagged PDF' format also supports having attributes for structural items. Being similar to formats such as HTML, SGML and XML, this allows great flexibility in associating layout specifications to content. It also permits conveying semantics by short 'hints' or with extra descriptive text, especially for content where there is implicit meaning in layout choices such as margin-widths, line-spacings or use of italiced or bold-faced text. The WCAG[33] and WAI-ARIA[32] recommendations describe attributes such as alt, role, aria-label, aria-roledescription used to provide such hints to Assistive Technology, as well as having aria-details, aria-labelledby and aria-describedby to reference parts of the document where descriptions may be found.

The published PDF/UA-1 standard has explicit rules about which structural elements may be parents or siblings of other elements, thereby covering the main requirements of Accessibility. However it is rather weak when it comes to specifying when aria-* attributes should be used to better convey meaning, when it is not so easily deduced from context. PDF/UA-2, to be released early in 2024 [24] by the PDF/UA Technical Working Group[26], has some reference to ARIA[32], but not to the extent exhibited in the documents studied in a later section.

To date there is no validation software[2] for PDF/UA documents that checks whether such attributes are present, thus whether the document can truly be regarded as being Accessible for WCAG 2.0, WCAG 2.1, or WCAG 2.2 [34], as is required to satisfy various Government document guidelines [4, 6, 9, 29].

On the other hand, the requirements of PDF/UA-1 are such that a document can be 'derived to HTML'[7] preserving the main document structures and tagging all of the visible content. One means of doing this is by using the online ngPDF[15] by Dual Lab[8] powered by the iText PDF library[11]. With ngPDF any applicable attributes specified for structure items are mapped to attributes on the resulting HTML tags. Where the resulting file is not strictly valid for HTML-5, incorrect patterns can be fixed with multi-line sed (Stream Editor) commands which reorganise the tagging without creating or destroying any content. HTML documents can be tested for WCAG 2.1 [34] accessibility and conformance to ARIA recommendations [32] using the AInspector for Firefox[5] plugin based upon the OpenA11y library[18].

[2]Although PAC 2021[20] and PAC 2024[21] are the best available for PDF/UA-1 and make claims for WCAG compliance, there are many WCAG/ARIA recommendations that are not checked, as we shall see.

Links to two real-world PDF/UA-1 documents, and the HTML derived and corrected to be fully HTML-5 conformant, can be found in the bibliography, for items FallMT2022[10] and SpringMT2023[30].

We discuss these 'fish stock review' documents further in a later section.

'Accessible Name' and 'Accessible Description'

These concepts, described in detail in [35], are like street signs and brief explanations in a city guide or theme park map. The 'Accessible Name' gives an indication of what is found down the street, while the 'Accessible Description' provides more detail. Their purpose in an electronic document is to make it easier to decide whether to go that way (by following a hyperlink, say) or continue reading at the current location. Assistive Technology (AT) is expected to be able to present the Name and/or Description, so that a choice can be made without losing 'focus'.

Most simply, for the 'Accessible Name' of a structural element (or HTML tag) one uses either the value of an aria-label attribute, or the document text indicated by an aria-labelledby reference, with the latter preferred. For the 'Accessible Description' it is aria-details and aria-describedby references that are followed, if available. Availability of references is highly dependent upon the use of 'Structure Destinations' within the PDF, whether for PDF 1.7 or PDF 2.0.

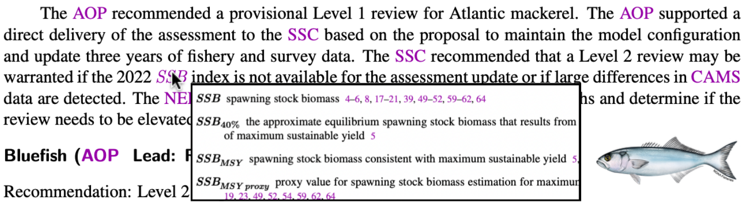


Figure 4: A view of a small portion of the target location "pops up" when the mouse hovers over a link anchor. Move the mouse slightly to dismiss the popup.

In a fully visual context this idea is well displayed as in Figure 4, where in PDF browser software under MacOS, a small popup window shows the contents found at the destination of a hyperlink. There the expansion and meaning of an Acronym is shown without jumping to the Glossary entry itself. Using AT, the 'Accessible Description' would include the expansion 'spawning stock biomass' and there could be an audible cue, perhaps speaking the 'Accessible Name', here the string `Glossary:SSB`. If desired, a button press would deliver the Description; but perhaps not used if the Name is meaningful or familiar as to not require a repeat. Figure 5 shows a representation of the internal PDF objects that encode the link information.

This would be especially relevant when navigating via `taborder` (i.e., using just the `tab` key) in either a PDF or HTML document; especially with a Braille display, where the surrounding context is extremely limited. The text used could be located anywhere within the document, not necessarily at the destination of a hyperlink; though usually there would be some relevant structure involved, such as a caption on an image, chart or photograph. There are WCAG recommendations concerning use of 'Accessible Name' and/or 'Accessible Description', under categories 1.1.1, 1.3.1, 2.4.4, 2.4.6, 3.2.4, 3.3.2 and 4.1.2; related also to headings and landmarks, as well as links, list items, and captions for images, figures and tables.

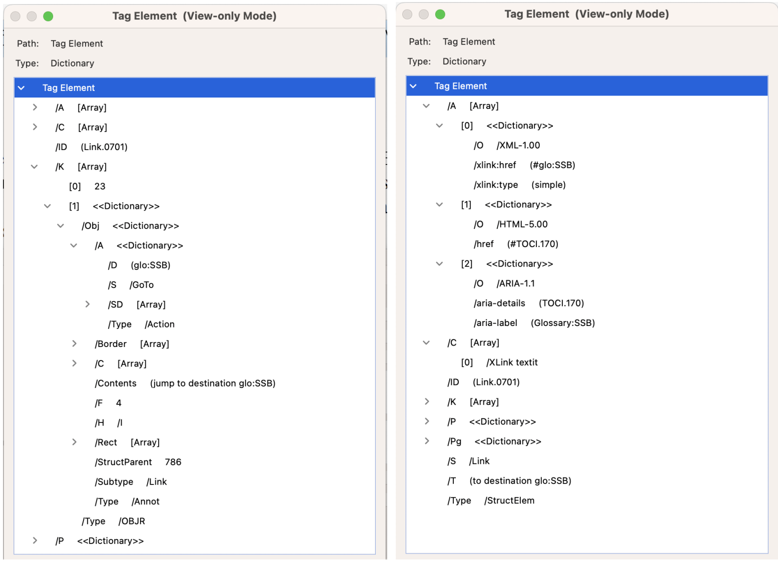
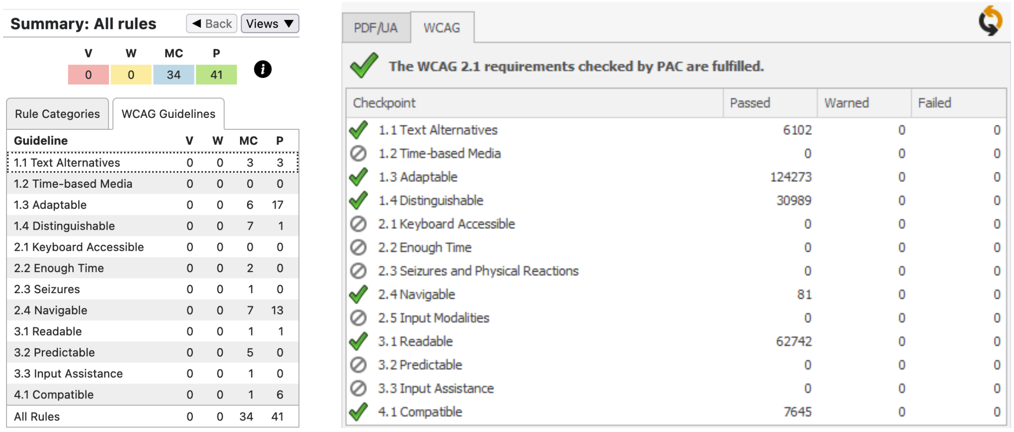


Figure 5: Link structure showing (on left) the `GoTo` action and target destination `glo:SSB`; and (on right) the attributes for Export to XML and HTML, with extra ARIA attributes used to create the 'Accessible Name' and 'Accessible Description' for the Acronym link of Figure 4.

Case study: Fish stock reviews

As an aside from Accessibility, the FallMT2022[10] and SpringMT2023[30] example documents are Technical Memoranda prepared for the NEFSC[14], containing Reports and Peer- and Oversight-reviews of the state of fish stocks in the Atlantic Ocean, off the northeast coast of the U.S.A. Data is collected from fishing vessels at sea or when unloading at a port. This is saved in database files and used with statistical modelling to confirm previously-set parameters, and make predictions for expected catches in future years. Research analysts upload statistics and textual descriptions. At least twice yearly Daniel Hennen (NEFSC[14]) extracts reports, using R[28] programs to create LaTeX source files for tables and figures and to present the textual materials for each fish stock under review. The LaTeX-based structure for these documents was developed by Daniel and Prof. Thomas Price (emeritus, University of Akron). In all there are up to a dozen input files for each fish stock, as well as many other information files that are read during the LaTeX preamble, defining macros for later use. NOAA[16], a U.S. government agency, must meet Accessibility requirements [4, 29]. Due to previous work that Ross has done on 'Tagged PDF' using LaTeX[13], he was invited to join the team to work at augmenting the high-quality visual view with Accessibility enhancements to satisfy all WCAG requirements for this kind of subject matter.

That LaTeX was used to create the PDF/UA documents is not the main issue here. Rather it is the enhancements for Accessibility that can be included through aria-* attributes that allow many WCAG/ARIA [32] recommendations to be met. These can be examined in the HTML version using the AInspector[5] when using the Firefox browser, and with other plug-ins for other commonly-used browsers. The latest, AInspector v3.0 tests 120 separate rules of which 51 have no applicable tag to test within the HTML version of the SpringMT2023[30] example. This leaves significantly more applicable Accessibility tests than are listed in the Matterhorn Protocol[12], or are tested for within PDF/UA documents by veraPDF[31] or PAC 2021[20] or



Guideline	V	W	MC	P
1.1 Text Alternatives	0	0	3	3
1.2 Time-based Media	0	0	0	0
1.3 Adaptable	0	0	6	17
1.4 Distinguishable	0	0	7	1
2.1 Keyboard Accessible	0	0	0	0
2.2 Enough Time	0	0	2	0
2.3 Seizures	0	0	1	0
2.4 Navigable	0	0	7	13
3.1 Readable	0	0	1	1
3.2 Predictable	0	0	5	0
3.3 Input Assistance	0	0	1	0
4.1 Compatible	0	0	1	6
All Rules	0	0	34	41

Checkpoint	Passed	Warned	Failed
1.1 Text Alternatives	6102	0	0
1.2 Time-based Media	0	0	0
1.3 Adaptable	124273	0	0
1.4 Distinguishable	30989	0	0
2.1 Keyboard Accessible	0	0	0
2.2 Enough Time	0	0	0
2.3 Seizures and Physical Reactions	0	0	0
2.4 Navigable	81	0	0
2.5 Input Modalities	0	0	0
3.1 Readable	62742	0	0
3.2 Predictable	0	0	0
3.3 Input Assistance	0	0	0
4.1 Compatible	7645	0	0

(a) AInspector summary (b) PAC 2021, WCAG panel summary

Figure 6: Validation summaries of conformance with applicable WCAG recommendations.

the recently released PAC 2024[21]. With 0 violations and 0 warnings, 41 of the remaining tests are algorithmically checked to be Passing, leaving 34 where a manual check (MC) is advised to determine whether a recommendation is actually satisfied; see Figure 6(a) for a summary.

Many of these remaining rules are indeed automatically satisfied, due to the nature of the PDF/UA format and the HTML document derived from it. The website for SpringMT2023 (see [30] for the link address) gives more details on the specific tests, explaining what needs to be checked or a justification for why a Pass can be presumed. E.g., for rule NAVIGATION 3: 'Consistent ordering of H1 and H2 labels' (ARIA Success Condition 3.2.3), there is an equivalent test in PDF/UA. With a valid PDF/UA there is no manual checking needed. Similarly for rule LINK 3: 'Target focus should be in content window', which is part of ARIA Success Condition 3.2.1. As there is just a single window containing the complete HTML site, this is satisfied for all internal hyperlinks. External hyperlinks would have been confirmed when the link target was harvested for use in the PDF document. Hence all 1500+ instances are effectively confirmed. And with rule TABLE 1: 'Data cells must have row/column headers', part of ARIA Success Condition 1.3.1, we have Pass for 484 table cells. The remaining 28 cells can be easily confirmed to be blank, for formatting purposes only, which is what the test requires.

On the other hand, rule IMAGE 1: 'Alt text must summarize purpose', part of ARIA Success Condition 1.1.1, requires checking that the Alt text for each Figure structure is applicable to the actual image. This should be part of the normal proof-reading and editorial checking. Other tests have confirmations that similarly should be part of normal checking; nevertheless it could be useful to have the reminder.

With essentially all the rules checked using the considerations detailed in the 'SpringMT2023' website (see the link for [30]), one concludes a very high degree of WCAG conformance for the HTML document. Now since that HTML was 'derived' from a PDF, one realises that the same information that enhances the HTML file's conformance is present also within the PDF; so it too should be considered to have a high degree of WCAG conformance. There is a difficulty with the PDF however, in that there may not be adequate software, easily available, to take advantage of that information. In summary we have the following:

It is not the PDF, or PDF/UA, format that should be considered as not being Accessible; rather, documents can be created supporting a high degree of Accessibility. The issue is that *software able to take full advantage of this Accessibility is not yet readily available.*

Direct validation in the PDF

Figure 6(b) shows a summary of the validation results from PAC 2021[20] working directly on the PDF/UA document, as organised by WCAG 2.1[34] success criteria. Showing no violations, nor even any warnings, one could easily presume that the document is fully compliant with all WCAG recommendations. However that is over-simplifying, somewhat.

In Figure 6(b) we see that 6 out of 13 categories are indicated as Passing, having run thousands of individual checks on structure elements and associated content. However, there are 7 categories with 0 tests recorded. Possible explanations for having 0 tests can be one of the below, or a combination thereof.

(a) There are no appropriate tests implemented; or

(b) no structure or content in the PDF is appropriate for what the tests need to check.

The WCAG/ARIA[32] recommendations are about ways to make certain kinds of information content and structures more accessible to users/readers having various kinds of disability. Figure 6(a) shows that AInspector[5] has run many tests in those categories, which include some for which the 'Accessible Name' and/or 'Accessible Description' are relevant. As the HTML was derived from the PDF, it is inconcievable that (b) alone could be the correct explanation.

Also it is remarkable that only 81 tests were run in Category '2.4 Navigable'. This includes sub-categories 2.4.4 and 2.4.6 which require looking at the 'Accessible Name' for the 1500+ internal hyperlinks. As well as requesting manual checks that the algorithmically constructed 'Accessible Name' is meaningful, and describes the purpose of the link, there is the requirement that links having the same 'Accessible Name' jump to the same target. The total number of tests run should certainly be in the thousands, well above 81 as reported. So one can only conclude that the 'Accessible Name' concept has not been addressed by PAC software.

Furthermore, within the 6 categories tested by PAC[20], AInspector[5] finds 25 applicable tests that require manual checking. Not all of these can be due to the change of format from PDF into HTML. As well as 2.4.4 and 2.4.6 covered in the previous paragraph, there are tests in subcategories 1.1.1, 1.3.1 and 4.1.2 which are about having extra hints via aria-* attributes.

Recommendations

The 'Tagged PDF' document format already has the capability for production of PDF/UA documents that support WCAG Accessibility criteria, to a very high level of Accessibility. However, there is currently a lack of software that can take full advantage of this, to give users with disabilities a satisfying reading experience.

Thus for better Accessibility (i.e., satisfying more WCAG/ARIA Success Criteria), we make recommendations as follows.

1. Software producing PDF/UA documents should include 'hints' by populating the aria-* attributes used in creating a meaningful, descriptive 'Accessible Name' and 'Accessible Description', with all structure elements where this can reasonably be helpful.

2. PDF consuming Assistive Technology (AT) applications should be written to look for and act upon the presence of aria-* attributes, to provide a better reading experience.

3. Future updates of the PDF/UA standard should describe the purpose and usage of aria-* attributes, along with recommendations for their support.

Acknowledgements

We thank others who have been either directly involved in doing this work, or indirectly via discussions concerning tagging and/or Accessibility. Firstly Daniel Hennen, an Operations Research analyst at NEFSC[14], and Thomas Price, emeritus at University of Akron for inviting me to work on the layout and accessibility aspects of these Technical Memoranda. Next there is Boris Doubrov from Dual Lab[8] and head of the LaTeX Project LWG[27], and Roman Toda from the 'Deriving HTML from PDF' TWG[7]. We wish to acknowledge many discussions (email and online) with other members of the LaTeX Project LWG[27], of which Ross is a member; namely Chris Rowley, Frank Mittelbach, David Carlisle, Ulrike Fischer and others. Some thanks also go to Jon Gunderson of University of Illinois Accessible IT group, a developer of the OpenA11y Evaluation Library[18] and AInspector[5], for sharing his latest versions for beta-testing using the HTML derivations, prior to the release of version 3.0, now renamed as AInspector for Firefox. More recently, we thank Thomas Schempp from axes4 GmbH Zürich, for help with interpreting results from PAC 2021[20] resulting in minor fixes and allowing pre-release testing of PAC 2024[21]. Chris, Dan and Tom also for comments on details in this paper.

Software tools and techniques, organisations, websites

[1] Acrobat Pro, Adobe Inc.; non-free software for viewing, reading, printing, editing and manipulation of PDF files. https://www.adobe.com/acrobat.html

[2] Acrobat Reader, Adobe Inc.; free software for viewing, reading, printing PDF files. https://get.adobe.com/reader/

[3] U.S. Department of Justice, Civil Rights Division: Americans with Disabilities Act of 1990, as amended. https://www.ada.gov/law-and-regs/ada/

[4] U.S. Department of Justice, Civil Rights Division: Guidance on Web Accessibility and the ADA[3], March 2022. https://www.ada.gov/resources/web-guidance/

[5] AInspector for Firefox. University of Illinois Accessible IT group. Uses the OpenA11y Evaluation Library[18]. https://ainspector.disability.illinois.edu.
GitHub: https://github.com/opena11y/ainspector-for-firefox

[6] BS 8878:2010 "Web accessibility. Code of practice." British Standards Institute (BSI), December 2010. https://knowledge.bsigroup.com/products/web-accessibility-code-of-practice?version=standard. Wikipedia: https://en.wikipedia.org/wiki/BS_8878

[7] Deriving HTML from PDF, 2019. A usage specification for tagged ISO 32000-2 files. Publ. by PDF Association[23, 25]. https://pdfa.org/resource/deriving-html-from-pdf/

[8] Dual Lab, software developers providing product development services in multiple domains. https://duallab.com/

[9] European accessibility act: Accessibility standardisation. https://ec.europa.eu/social/main.jsp?catId=1485&langId=en

[10] FallMT-2022: Management Track Assessments Fall 2022. Available in PDF/UA and WCAG 2.1 accessible HTML-5 versions. http://science.mq.edu.au/~ross/TaggedPDF/FallMT2022/

[11] Apryse: iText PDF library: toolkit offering well-documented and versatile PDF engines, written in Java and .NET). https://itextpdf.com

[12] Matterhorn Protocol 1.1. PDF/UA Conformance Testing Model. PDF/UA Technical Working Group[26]. https://pdfa.org/resource/the-matterhorn-protocol/

[13] Moore, Ross: Examples of Tagged PDF documents built using LaTeX. http://science.mq.edu.au/~ross/TaggedPDF/index.html. last updated December 2023.

[14] Northeast Fisheries Science Center. Office of NOAA Fisheries[17]. https://www.fisheries.noaa.gov/contact/northeast-fisheries-science-center-0

[15] ngPDF, by Dual Lab[8], using the iText PDF library[11];
online service at https://ngpdf.duallab.com. Requires validation with veraPDF[31].

[16] NOAA. National Oceanic and Atmospheric Administration; U.S. Department of Commerce. https://www.noaa.gov

[17] NOAA Fisheries. National Oceanic and Atmospheric Administration. https://www.fisheries.noaa.gov. Science & Data: https://www.fisheries.noaa.gov/science-and-data.
Research: https://www.fisheries.noaa.gov/resources/peer-reviewed-research.

[18] OpenA11y Evaluation Library, University of Illinois Accessible IT group. https://accessibleit.disability.illinois.edu/tools/ainspector-wcag/evaluation-library/.
GitHub: https://github.com/opena11y

[19] Open Preservation Foundation: global not-for-profit membership organisation working to advance shared standards and solutions for the long-term preservation of digital content.
https://openpreservation.org

[20] PAC 2021: PDF Accessibility Checker; PDF/UA Foundation. PDF/UA and WCAG[33] validation software, for Windows only. https://pdfua.foundation/en/pdf-accessibility-checker-pac/

[21] PAC 2024: PDF Accessibility Checker; PDF/UA Foundation. PDF/UA and WCAG[33] validation software, for Windows only; developed by axes4 GmbH, Zürich (CH); released December 2023. https://pac.pdf-accessibility.org

[22] PDF 2.0; no-cost access to the latest PDF standard: ISO 32000-2 (PDF 2.0). PDF Association[23]. https://pdfa.org/sponsored-standards/

[23] PDF Association: home of the worldwide PDF technical community. https://pdfa.org

[24] Usage of Tagged PDF in PDF 2.0. PDF Association[23], PDF Reuse & PDF/UA TWG[26]. To appear in 2024.

[25] 'Deriving HTML from PDF' TWG of the PDF Association[23].
https://pdfa.org/community/deriving-html-from-pdf-twg/

[26] PDF/UA Technical Working Group, PDF Association[23]. Drives development of ISO 14289 (PDF/UA) through its association with ISO TC 171 SC 2 WG 9 and publishes resources for developers who need to understand the International Standard for universally accessible PDF. https://pdfa.org/community/pdf-ua-technical-working-group/

[27] LaTeX Project Liaison Working Group (LWG), PDF Association[23]. Working on a project to enhance LaTeX to fully and naturally support creation of structured document formats, in particular the "tagged PDF" format as required by accessibility standards such as PDF/UA. https://pdfa.org/community/latex-project-lwg/

[28] The R Project for Statistical Computing. A programming language and environment for statistical computing and graphics. https://www.r-project.org

[29] U.S. Access Board, Section 508 of the Rehabilitation Act: Information and Communication Technology, Revised 508 Standards and 255 Guidelines, 2018: https://www.access-board.gov/ict/, WCAG 2.0 & PDF/UA: https://www.access-board.gov/files/ict/ict-final-rule.pdf.

[30] SpringMT-2023: Management Track Assessments Spring 2023. Available in PDF/UA and WCAG 2.1 accessible HTML-5 versions. http://science.mq.edu.au/~ross/TaggedPDF/SpringMT2023/

[31] veraPDF, Open source industry supported PDF/A validation maintained by the Open Preservation Foundation[19]. Java application; https://github.com/veraPDF.

[32] Web Accessibility Initiative (WAI). Making the Web Accessible: strategies, standards, and supporting resources to help you make the Web more accessible to people with disabilities. https://www.w3.org/WAI/

[33] Web Content Accessibility Guidelines (WCAG) 2.1. W3C Recommendation, September 2023. Accessibility Guidelines Working Group[32]. https://www.w3.org/TR/WCAG/

[34] Web Content Accessibility Guidelines 2, Overview. Web Accessibility Initiative[32].
https://www.w3.org/WAI/standards-guidelines/wcag/

[35] Accessible Name and Description Computation 1.1; W3C Recommendation, 18 December 2018. Web Accessibility Initiative[32]. https://www.w3.org/TR/accname-1.1/

A method to overcome graphical perception difficulties for the visually impaired in higher education and advanced profession

Tsuneyoshi Kamae[12*] and Tomio Koide[3†]

[1] Department of Physics, University of Tokyo, Tokyo, Japan
tune.kamae@gmail.com
[2] SLAC/KIPAC, Stanford University, Miami, Florida, U.S.A.
kamae@slac.stanford.edu
[3] Createsystem Inc., Tachikawa, Tokyo, Japan.
koide@createsystem.co.jp

Abstract

We present an effective solution to overcome graphical perception difficulties the visually impaired suffer in higher education and advanced profession. The solution combines existing ready-to-use technologies include multimodal ebooks, tactile graphical sheets, touch-activated audio-graphical web applications, linear Braille displays, and dot-matrix displays. Although multimodal combination of tactile and auditory senses has been applied in picture books for visually impaired children, any solution has not been developed well for visually impaired university students and professionals. Since graphical perception difficulties they face differ individually, we need volunteer groups versed in graphical communication tools to provide the needed assistance locally. Hence our goal is to provide an easy-to-use and fine-tunable set of technologies to volunteer groups formed around students and professionals with visual impairments.

1 Introduction

Communication relies heavily on graphics in recent years. Graphics are used in a variety of forms in textbooks, manuals, public documents, and business transactions. Social and professional lives would be badly hindered without effective graphic communications. People with visual impairments and learning difficulties may be unprepared for emergencies if communication goes only through standard tools for the sighted. Fortunately, new technologies have become available to overcome or mitigate such graphical perception difficulties.

Many technologies have been applied historically to assist graphical perception for the visually impaired. Traditionally, parents and guardians gave audio description of what blind infants and children touched. Teaching graphics in combination of tactile and auditory senses has been the most common and effective method throughout the history of mankind. Very little have been documented, however, in early history except for a few episodes [1] [2] [3] [4]. Invention of Braille in 1824 [5] has made a huge impact in developing the tactile sense among the visually impaired comminity. Using the tactile sense is the most common method for teaching graphics to blind infants and children [6] and in secondary education [7] [4].

Louis Braille's invention of Braille embossing on papers revolutionized the text reading for the blind [8]. By the first half of the 20th century, it has become the standard method to emboss Braille dots on papers with stencils and to read them by fingers [5].

It remained difficult, however, to translate advanced textbooks to Braille or to write new advanced textbooks in Braille. American Printing House for the Blind (APH) produced the

*Designed the project and built the hardware and tactile sheets
†Built the browser-based graphics, the output system for Braille and Dotpad

Braille version of the 1959 World Book Encyclopedia. It consisted of 38,400 Braille pages [9]. Small-size dictionaries have been translated to Braille in United Kingdom [10], Japan [11] and other countries. They all ended up very voluminous and not portable. Advanced textbooks are as voluminous as these small-size dictionaries. Students and professionals with vision impairments must find other methods to acquire and convey knowledge.

As computer use has become popular, graphics are presented more in digital forms. In 1975, the Kurzweil Machine [12] combined the optical character reader and the synthetic speech engine to open a new communication path for the blind. Those with visual impairments began to read printed books and journals without reading assistance in libraries. The machine was, however, too expensive for small libraries and individual users. As the computational power of personal computers (PCs) increased, screen-readers became available around 1986 [13]. The technology has brought another revolution to the visually impaired. They have gained access to books and dictionaries through the auditory function in PCs.

Introduction of OCR and screen-reader, however, has not solved the graphical perception difficulties. Books and journals in science, technology and engineering, medical and life sciences are filled with graphics. User manuals and public notices also rely heavily on graphics. Luckily, graphics have become available also in digital forms. Digital technologies have been taking the challenge to improve their usefulness for the handicapped ever since. We have been selecting easy-to-use technologies and combining them to solve or ease a wide spectrum of difficulties visually-impaired students and professionals face.

The vacuum thermoformer technology has become a major technology in tactile graphics between 1960s and 1970s. The technology became available to mass-produce plastic lids for bottles in 1960 [14]. Princeton Braillists [15] applied this technology to produce thin tactile sheets by thermovacuum pressing onto hand-drawn masters embossed on aluminum foils. They published a series of tactile graphic books, many pages of maps and anatomy drawings. Then, silk-screen printing with UV-hardening ink [16] has began to print books with Braille and raised lines around 1970. This technology has developed rapidly to mass-produce calendars, maps, and manuals to the visually-impaired community. Paper embossing machines were developed to print Braille dots and used almost all schools for the visually impaired. They are also used widely to print graphics by delineating it approximately as a series of dots. Later another kind of embossing machine has been developed to produce continuous tactile lines as well as Braille dots on thick papers (e.g. [17]) The paper can be color-printed before embossing.

Another technology appeared about 20 years ago that produces tactile graphics using special papers coated with microcapsuls. This method does not require special training. It has some weakness: It cannot produce complicated graphics and the tactile lines are not as stable as those made by the thermoforming. Because the heating machine to inflate microcapsuls requires less effort, the technology is widely used now.

Some combination of currently available technologies should enable to overcome graphical perception difficulties for the visually impaired in higher education and advanced profession.

2 Graphical presentation tools currently used among visual impaired university students and professionals

Traditionally blind children have been educated through auditory explanation of the objects they touch. The explanation have been given by human voices. Modern studies on the human sensory system have found that the sensory system begins to merge in the children's brains through the synchronization of auditory explanations, touch-and-feel experiences, and other sensory inputs [18]. This integration process slows down by aging but continues through our lives.

In fact, teachers had been creating various tactile materials for blind students long before the invention of Braille [8]. Saunderson, for example, learned alphabets by tracing engravings late 17th century [19] [2]. Letters and shapes had been taught to the blind by embossing papers with sticks and verbal explamation in early 19th century. This led to the invention of Braille in 1829 [2]. Braille was not adopted however, as the universal tool in the English-speaking world until around 1932. We note that Braille anone does no communicate graphical contents and hence tactile graphics are combined with audito presentation now.

Embossing papers and engraving hard materials to create tactile graphics had been a popular method to use graphical material in the education [20]. Descriptions are found about the method used in the early lives of Nicholas Saunderson(1682-1739). In Japan, Hanawa Hokiichi (1746-1821) [21] learned Chinese characters (Kanji) by touching stone engravings [22].

The technologies currently used in schools for special needs include tactile graphics transcribed onto special composite papers. Hundreds of books and teaching materials are provided in this form. The distributors include National Braille Press [23], Braille Authority of North America [24], Perkins School for the Blind [25], Beyond Braille [26], ClearVision [27], and Raised Lines Foundation [28].

The swell paper technology [29] are used in small numbers in classrooms and tutoring. Schools and organizations for visually impaired are likely to use paper embossing machines to print graphics as a series of dots. Embossing machines to produce continuous tactile lines on thick papers are used mostly to meet individual demands.

3 Cost of preparing graphical presentation tools we have considered

Transferring tactile graphics from masters to plastic films is a multi-step process and requires special care. The method used by Princeton Braillists [15] requires long technical training to produce the masters and cost long hours of labor. Preparing simple masters requires detailed guidance [30]. A machine dedicated for tactile graphics and Braille is soled by American Thermoformer [31] as EZ-Form Brailon Duplicator. It costs aeound 5 thousand US dollars. A4-size composit papers (called Brailons) cost about 10 US cents per sheet [32]. Large non-profit organizations have acquired thermal vacuum formers like EZ-Form Brailon Duplicator and have been publishing technical graphics books and teaching materails as mentioned inSection 2.

The swell paper technology [29] is popular for its handiness. Graphics are copied using a carbon-toner based photocopier to swell papers first and then heated by infrared light. Dedicated infrared heaters and swell papers sold by Zychem [33] and PIAF [34]. The heater costs around 1500 US dollars and the swell papers a few US dollars per page.

The Braille embossing printers used in schools and non-profit organizations cost more than ten thousand US diollars and too large for offices of small voluteer groups. One exception is a personal embosser called EasyTatix [17]. It is small and costs about 5 thousand US dollars.

Because of the costs described above, we choose to design an affordable thermoformer dedicated for Brailons. We also developed affordable methods to prepare the master used to transcribe to Brailons: one is to rent a lasaer cutter by minutes and cut 1mm thick Medium Density Fiberboard (MDF) sheets. A lasar cutter can be rented out at about 50 US cents a minute and a MDF board costs about 1 US dollar. The drawing to feed to the cutter can be prepared with computer graphic tools such as Illustrator and Inkscape.

Lack of graphical teaching material has been one important factor in narrowing curriculum and narrowing career option for the visually impaired in many countries including Japan. Acquiring adequate dgital skills and constantly updating them are also essential because digital

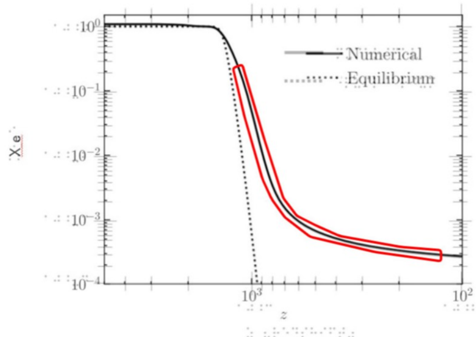


Figure 1: An enlarged HTML image on the browser and on the tactile sheet for a figure linked from the ePUB or DAISY textbook. If the region surrounded by the red line is touched, the MP3 audio file corresponding to the line will be activated and the black line will be explained. Taken from Modern Cosmology by Dodelson and Schmidt Chapter 4 [49].

technologies evolve constantly. For those who have only limited access to technical assistances, well-arranged teaching materials need to be provided. The method and materials we provide are to serve such needs.

Methods similar to what we propose here have been already tried and used. The first such method uses the QR code to activate auditory explanations on tactile sheets [35]. The second relies on a dedicated tablet which is activated by touch on tactile sheets [36]. This is very similar to ours except that the hardware and software are proprietary. Users alone cannot produce their own sheets nor explanatory audio files. The third such method uses the touch-sensing function of the iPAD and tactile sheets set on the screen [37]. Audio files are activated by finger touch on the sheet. The important difference between the last two methods and ours is that ours are applicable to almost all digital devices, including PCs and Android tablets [38] equipped with A4-size touch screens.

We re-emphasize that affordable and easy-to-use technologies have been selected to allow a volunteer group can provide high level assistance locally. Unfortunately, we have received feedbacks only from limited number of users in Japan. It is our goal to widen the user community to universities and workplaces world-wide. For that goal we are developing an inexpensive easy-to-use thermo-vacuum former to produce tactile sheets as described later.

4 Proposed multimodal teaching methods: ePUB/DAISY

Marrakesh Treaty [39] gives us the legal basis to use the contents published in advanced textbooks. The treaty allows us to produce multimedia ePUB [40] and DAISY [41] as well as the graphical tactile sheets and HTML images [42] based on the published textbooks.

The proposed method consists of the following steps and components:

1. Obtain the pdf version of an advanced textbook.
2. Copy the text part to a multimedia ePUB/DAISY writing/editing package, ChattyInfty3 [43]. On Windows OS several SAPI TTS voices [44] are available to read the texts. We have to check the pronunciation of terminologies and proper nouns in the chosen voice. If incorrected, add the pronunciation for the word in the designated entry.

3. Convert mathematical expressions and equations to corresponding LaTeX scripts using a PyTorch OCR program [45]. The LaTeX scripts are read aloud, verbatim or luculent.
4. Prepare enlarged HTML images [42] as shown in Fig.4 and post them on a cloud platform like Heroku [46].
5. Add links to corresponding HTML images posted at the cloud platform.
6. To assist comprehension, assign different TTS voices for plain texts, math expressions and figure captions.

ChattyInfty3 saves locally or export the edited file to multimedia ePUB or multimedia DAISY to its portal site. ChattyBooks [47]. They display files in a structured format, read aloud with multiple TTS voices, and highlight the lines being read. Thorium Reader [48] works similarly.

The enlarged graphics are prepared in tactile sheets with HTML images for all figures and equations in textbooks. On the tactile sheets for graphs, the axes and tick marks, the function curves, and the partitioned regions are converted to raised lines and patterns. For equations, mathematical symbols are shown by raised images and prints. We used a package conveniently assembled as HTML Imagemap Generator [50] to define polygon regions around graphical entities in HTML images [42].

We wish to add a comment that knowing the shapes of mathematical symbols is very important for the visually impaired university students because they have to show the shapes to sighted colleagues in schools and workplaces when they got jobs.

5 Proposed multimodal teaching methods: Preparing tactile sheets

As described in section 2, several technologies have been applied to produce tactile sheets. A concise review of technologies to produce raised tactile lines and Braille is in YouTube [51].

We choose two methods among these technologies. One embosses the graphics directly to paper. The other transcribe the masters to composite sheets. The transcription method consists of two steps: preparation of masters and transcription of from masters to tactile sheets.

Silkscreen printing with UV-hardening ink provided, for example, by Obun Printing in Japan [52] allows us superimposes tactile lines and Braille on color printing as shown in Fig.4, Such silkscreen printing become a best solution when hundreds of copies are needed. We note that a live display on dot-matrix display [53] is coming to be available in a small format (24x24 dot-matrix). The display costs multi-thousand dollars now. If the matrix size becomes larger and its cost is reduced, the graphical display technology will change dramatically.

1. Embossing directly to papers: We find one company selling an embossing printer worldwide at a price around 5 thousand US dollars [17]. The special papers for embossing cost around 1-2 US dollar a page. The machine embosses tactile lines and Braille onto the sheets color-printed beforehand. Fig.2 shows the embossed tactile lines and the embossed Braille. Note that the paper in Fig.2 was color-printed in to show embossing clearly.
2. Thermovacuum forming of thin sheets: There are several tabletop thermo-vacuumformers on market. They are primarily for forming plastic sheets around 3D masters and may not duplicate well shallow tactile lines nor Braille. The machine costs typically several thousand US dollars. For example, American Thermoformer [31] sells a thermo-vacuum former dedicated to duplicate the tactile lines and Braille dots on masters to composite papers (EZ-Form Brailon Duplicator). A4-size composit papers (called Brailons) cost about 10 US cents per sheet [32]. It is possible to superimpose color printing with pigment

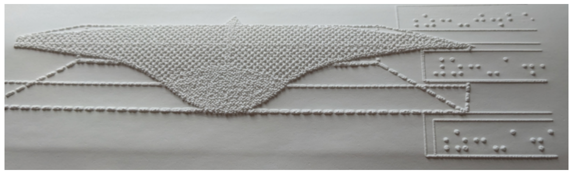


Figure 2: Easy Tactix can add tactile lines and Braille on color-printed papers. This shows embossed lines and Braille without color printing.

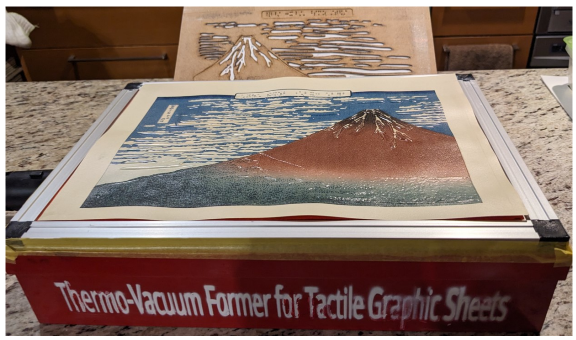


Figure 3: DIY thermal vacuum former design and assempled in this project. The box provides vacuum and the MDF sheet in the back is the masster. The color-printed tactile Brailon is shown on the vacuum box.

ink to Brailon sheets as shown in Fig.3. In our setup, we prepare masters by cutting or engraving MDF sheets either with laser cutters or engraving chisels. Laser cutters accept properly formatted graphic files. There are DIY shops where users can operate a cutter at about 5 US cents a minute.

6 Will our combined method open a new career path?

One goal of ours has been to provide the multimedia ePUB/DAISY versions of advanced text-books, the tactile sheets for the graphics, and the associated web-based applications bundled together. We hope to see visually impaired students and professionals follow the footsteps

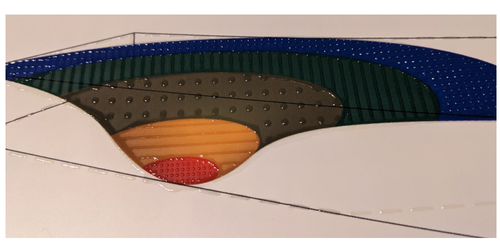


Figure 4: Tactile graphics added with UV ink on a color print. Taken from our teaching materials on vector analyses.

of those who have built-up very successful professional careers. They include mathematician Bernard Morin [54], biologist Geerat Vermeij [55]. and computer scientist T.V. Raman [56]. They are all talented and highly motivated, and received best education. Such education can be provided to all needed students and professionals by local volunteer groups equipped with adequate tools.

Serious problems remain world-wide in the quality of education given to average visually impaired students [57]. In universities, students with impairment learn in integrated classes as an absolute minority. Some universities offer assistance to the impaired students, give guidance to the class instructors, and provide teaching materials such as tactile graphics and multimedia ePUB/DAISY books. Most other universities, however, provide only note-taking assistants. Students with difficulties have to look for high level assistants to access teaching materials.

The problems are more sever for middle-aged professionals who lost vision in their mid-career [58] [59]. Supports they need are more diverse. We note that the CareerConnect page maintained by American Printing House is under construction. It may be difficult to prepare a generic recipe to assit wide variety of needs. Hence our method is designed to be adaptable to individual needs at a cost affordable for the volunteer groups formed around handicapped professionals.

University of Washington has been running a program to advance the success experience targetted to people with disabilities [60]. The program demonstrates importance of wide-range involvement by the students with various impairments. At the same time, their activities tell us that the technologies alone would not work effectively. Community around individuals with impairment must work together to raise their motivations. The method we propose will be useful to enhance such collaborative efforts.

7 Future prospects and concluding remarks

Within the limitation described in Sections 5 and 6, we are planning to prepare several sets of multimedia teaching materials for advanced textbooks. They include LaTeX cheat sheet [61], NVDA manual [62], Computational Mathematics with Sagemath [63] and Modern Cosmology [49]. We use A4 size Android Tablet [38] as the default interactive multimodal device. The tablet costs less than 400 US dollars and operates stand-alone on the internal battery for hours.

The most expensive hardware is the thermovacuum former. The best choice available now is Thermovacuum Former [31]. It cost nearly 4 thousand US dollars, a price beyond reaches of small volunteer groups assisting visually impaired students and/or professionals. We have assembled a kit combining materials available at Do-It-Yourself stores as shown in Fig3. Because of the international product liability law [64] we donot intend to distribute commercially.

We are also trying to form a community to spread our method to non-technical volunteers world-wide. The sample preparing for non-technical people are:

1. Woodblock prints by Hokusai [65] and Hiroshige. Engraved lines will be explained when touched the tactile sheet placed on an A4-size touch screen.

2. Music scores in tactile form [66]. The note will be played by touching musical symbols.

To increase future users of our technology, we are considering to take up the series of books titled “A Very Short Introduction” by Oxford University Press [67]. These books are less voluminous than the textbook in cosmology and probably attract wider readers.

Acknowledgements

This work has been supported by the CTC Next Generation Foundation, grant number 2022-013. We thank ITOCHU Techno-Solutions Future Foundation for the support. Assistive Technology Developing Organization (ATDO) is providing support for administrative works. We wish to express appreciation to Hiroshi Kawamura, Shigeru Yamauchi, and Tsuyoshi Maruichi. Makoto Amakawa and Ryusuke Kawamata have been feeding back their opinion. Akiyoshi Kurita has given detailed comments about availability of user friendly thermoformers. Takuya Nishimmoto and Hirofumi Fujii participated in testing our setups. Thanks are due to Masakazu Suzuki and Chifusa Ueno for answering questions about ChatteyInfty3 and ChattyBooks.

References

[1] ThinkerBellLabs, Before Braille: A Brief History of Visual Impairment and Education PT 1, retrieved form the following blog post, https://thinkerbelllabs.com/blog/before-braille-brief-history-visual-impairment-education/

[2] Denis Diderot, Lettre sur les aveugles a l'usage de ceux qui voyant, (known in English as Letter on the Blind for the Use of Those Who Can See), 1749. Photocopy of the book at, https://archive.org/details/lettresurlesave00didegoog/page/n17/mode/2up

[3] Valentin Hauey (1745-1822) is probably the first person to emboss paper as a means of reading for the blind. See, https://www.britannica.com/biography/Valentin-Hauy

[4] Charlotte Cushmann, Blog retrieved on April 09, 2023 Tactile Graphics: Overview of tactile graphics, which are the means by which non-textual informations are conveyed to people who are blind, visually impaired, deafblind. Retrieved from: https://www.pathstoliteracy.org/tactile-graphics/

[5] BrailleWork, History of Braille from, https://brailleworks.com/braille-resources/history-of-braille

[6] American Printing House, Creating Collage Tactile Graphics, retrieved from: https://www.youtube.com/watch?v=xsEjdUHkwy8; https://imagelibrary.aph.org/portals/aphb/#login

[7] Liz Eagan, Tactile Books for Students with Visual Impairments at https://www.pathstoliteracy.org/tactile-books-students-visual-impairments/

[8] Louis Braille, Method of Writing Words, Music, and Plain Songs by Means of Dots, for Use by the Blind and Arranged for Them (English translation), 1829 . The original book is not reprinted now. The photo is at, https://www.flickr.com/photos/perkinsarchive/sets/72157647987143143/
and https://www.perkins.org/brailles-most-famous-book/

[9] Robert Jaquiss and American Action Fund for Blind Children and Adults, The Encyclopedia and Me (2019). Retrieved from
https://nfb.org/images/nfb/publications/fr/fr38/3/fr380309.htm

[10] Royal National Institute for Blind, Little Oxford Dictionary.

[11] National Association of Institutions of Information Service for Visually Impaired Persons (Japan) translated dictionaries to Japanese Braille. They are a few 100 volumes each and about 100 pages each volume.

[12] Aaron Kleiner and Raymond C. Kurzweil, Bulletin of Prosthetics Research, 1977. A pdf book titled as "A description of the Kurrzweil Machie and a status report on its testing and dissemination" is available from https://www.rehab.research.va.gov/jour/77/14/1/kleiner.pdf; See also https://nfb.org/sites/default/files/images/nfb/publications/bm/bm06/bm0607/bm060703.htm

[13] Jim Thatcher, IBM Screen Reader, 1986. Description at, https://blog.tenon.io/the-legacy-of-dr-jim-thatcher-accessibility-pioneer/

[14] The first vacuum thermoformer machine was invented by ILLIG Maschinenbau GmbH & Co. in 1960. A Brief History of Vacuum Forming, posted at https://www.ansini.co.uk/news/a-brief-history-of-vacuum-forming/ See also https://www.illig.de/en-de/illig/history

[15] Princeton Braillists was established by Nancy Scott Amick. Raised lines of various heights and areas with various textures have been embossed to alminum and other metal sheets to be the masters. The group began operation in 1965. See https://princetonbrail.wordpress.com

[16] History of UV ink at https://imprintnext.com/blog/using-uv-ink-in-screen-printing-a-beginners-guide/

[17] Sinka KK; Easy Tactix. See https://www.easytactix.com/
The machine is sold as TactPlus outside Japan. See https://www.tactplusprinter.com/about

[18] Barry E. Stein and M. Alex Meredith, The Merging of the Senses, MIT Press 1993

[19] J. J. Tattersall; Historia Mathematica 19, 356370, 1992. A section is dedicated for Nicholas Saunderson, The Blind Lucasian Professor.

[20] Valentin Haüy had helped develop embossed teaching materials by the end of 1784. https://www.encyclopedia.com/science/encyclopedias-almanacs-transcripts-and-maps/valentin-hauy

[21] Descriptions about Hanawa Hokiichi in Wikipedia: https://en.wikipedia.org/wiki/Hanawa_Hokiichi

[22] Onko-Gakkai; http://onkogakkai.com/hokiichi-hanawa/

[23] National Braill Press; https://www.nbp.org/

[24] Braille Authority of North America: "Guidelines and Standards for Tactile Graphics, 2010" (Web Version - February 2012) https://www.brailleauthority.org/tg/web-manual/index.html

[25] Perkins School for the Blind; "Accessible Science", "Paths to Technology" https://www.perkins.org/accessible-science/

[26] Beyond Braille; Tactile picture series and books published by the visually impaired community in India and supported by Pocket Pixels. https://beyondbraille.com/

[27] Clear Vision Project; Making Tactile Books for the ClearVision Library, at http://www.clearvisionproject.org/resources/Tactile-book-makers-guide.pdf

[28] Books based on raised lines and Braille are produced mostly by thermoforming and distributed in India. https://raisedlines.org/

[29] The first swell paper is described in: T. P. Way and K. E. Barner, IEEE Trans. on Rehabilitation Engineering, Vol.5, No.1, pp.81-94 (1997). A historical review below, describes how the capsuling technology patented for pharmaceutical applications in 1957 applied to make the swell paper around 1997. R. Urbas, R. Milocevic, N. Kasikovic, Z. Pavlovic, U. Stankovic Elesini, Iran Polym Journal, vol.26, p.541?561 (2017)

[30] Ron Hinton; Tactile Graphics: https://www.ssc.education.ed.ac.uk/resources/vi&multi/Hinton/index.html

[31] American Thermoform sells tabletop Braille ducplicators. The one fits our need is EZ-Form Brailon Duplicator. See, https://americanthermoform.com/

[32] American Thermoform, A4 size Brailon paper. The composite paper costs about 10 US cents a paper. https://americanthermoform.com/product/brailon-21x29-7cm-a4-paper/

[33] Zychem, https://zychemltd.com/

[34] Harpo, https://piaf-tactile.com/

[35] Catherine M. Baker, Lauren R. Milne, Jeffrey Scofield, Cynthia L. Bennett, Richard E. Ladner Proc. the 16th Int ACM SIGACCESS Conf. on Computers and Accessibility, p.75, 2014.

[36] D. Parkes, Audio-tactile systems for designing and learning complex environments as a vision impaired person: static and dynamic spatial information access, in Cognitive Mapping: Past, Present and Future, eds. J. Steele and J.G. Hedberg, London: Routledge, 1994. See also, Steven Landau and Karen Gourgey, Information Technology and Disabilities Journal Vol.7(2), 2001. and Talking Tactile Tablet at http://itd.athenpro.org/volume7/number2/tablet.html and Touch Graphics, Inc. at https://www.touchgraphics.com/

[37] Giuseppe Melfi, Karin Mueller, Thorsten Schwarz, Gerhard Study C Jaworek, Rainer Stiefelhagen Proc. the 2020 CHI Conference on Human Factors in Computing Systems, p.1-12, 2020. See, https://dl.acm.org/doi/10.1145/3313831.3376508

[38] LUCA 15-inch Android 12 tablet by Iris Oyama sold at Amazon Japan: https://www.amazon.co.jp/-/en/TM152M4N1-Android-Viewing-Japanese-Support/dp/B0BD3M287G?th=1

[39] Wikipedia: Marrakesh Treaty facilitates access to published works to visually impaired persons and persons with print disabilities. It is known also as Marrakesh VIP Treaty or Marrakesh Treaty. This treaty entered into force on September 30, 2016. The types of works had been expanded to all literary works, plus musical works fixed in the form of text or notation.

[40] The distribution and interchange format for digital publications and documents by World Wide Web Consortium (W3C). The latest version (V3.3) is defined in:. https://www.w3.org/TR/epub-33/

[41] The DAISY Consortium, posted at https://daisy.org/. The specifications for the DAISY format are given below for easy reference. https://daisy.org/guidance/info-help/guidance-training/standards/daisy-format-specifications/

[42] W3C Consortium HTML Tutorials, HTML Images. See https://www.w3schools.com/html/html_images.asp; https://www.w3schools.com/html/html_images_imagemap.asp

[43] ChattyInfty Ver.3 series is an authoring tool of Multimedia Daisy and accessible ePUB developed by Nonprofit Organization, Science Accessibility Net. The manual is available at the following site: ChattyInfty, the Version 3 Series Manual (March, 2014)

https://sciaccess.net/en/doc/ChattyInfty3_Eng_Manual.pdf. The program runs on Windows 11 and Mac, and available for download at: https://sciaccess.net/en/ChattyInfty/index.html

[44] Wikipedia: SAPI is an API developed by Microsoft to allow the use of speech recognition and speech synthesis within Windows applications. TTS voices are a part of SAPI. Users can add their own TTS voices in Windows OS. https://learn.microsoft.com/en-us/previous-versions/windows/desktop/ms720151(v=vs.85)

[45] PyTorch-based program extracts the LaTeX code from equations in pdf. We have added options to improve either audio-recognition or explicit matching of opening and closing parentheses, braces, and brackets. The full LaTeX script is copied to the clipboard. https://github.com/lukas-blecher/LaTeX-OCR

[46] Heroku cloud platform described at, https://www.heroku.com/

[47] ChattyBooks for Windows PC: https://appsonwindows.com/apk/7246361/
for Android: https://play.google.com/store/apps/ details?id=net.sciaccess.chattybooks.billing&hl=en
and for iPhone: https://apps.apple.com/jp/app/chattybooks/id1151023858?l=en

[48] Thorium Reader is the EPUB reader for Windows, MacOS and Linux developed by EDRLab.
https://www.edrlab.org/software/thorium-reader/

[49] Scott Dodelson and Fabian Schmidt, Modern Cosmology (Second edition), Academic Press, 2020. The ePUB and DAISY version will be provided upon request.

[50] We use a packaged application to extract coordinated of polynomial regions called HTML Imagemap Generator. Downloaded from, https://labs.d-s-b.jp/ImagemapGenerator/

[51] YouTube by EMRC MKU: Technology and Education of Visual Impaired, gives a concise overview of methods to make tactile sheets. https://www.youtube.com/watch?v=G3rxW9wY_vo

[52] Obun Printing Co. offers UV-hardened varnish deposition processing with full-color or monochrome offset printing. https://en.obun.jp/

[53] DotPad see: https://pad.dotincorp.com/

[54] Bernard Morin received his Ph.D. in mathematics (topology) in 1972 from the Centre National de la Recherche Scientifique. He spent most of time at the Univ. of Strasbourg, He is known for his work to exhibit an eversion of the sphere and parameterization of Boy's surface.

[55] Geerat Vermeij graduated from Princeton Univ. and received his Ph.D. in biology and geology from Yale Univ. in 1971. He is a professor at Univ. of California, Davis and studies marine molluscs both as fossils and as living creatures. His life and career is written in: Geerat Vermeij, Privileged Hands - A Remarkable Scientific Life. Freeman and Co. 1997

[56] T. V. Raman is a computer scientist who specializes in accessibility research. Raman earned BS in mathematics from Univ. of Pune, MS from Indian Institute of Technology (IIT), and PhD in applied mathematics from Cornell Univ. in 1994. Dr. Raman also have developed an audio-based version of Emacs called Emacspeak. See, https://github.com/tvraman/emacspeak; https://emacspeak.sourceforge.net/

[57] Kim T. Zebehazy and Adam P. Wilton. Jour. of Visual Impairment and Blindness 108(1):5, 2014; 108(4),263, 2014; 108(4), 275, 2014.

[58] Those who lost vision partially or totally midway in their career need training to continue their career. The difficulties they face are differents. Learning to read Braille is not easy. See, https://lighthouse-sf.org/programs/skills/

[59] American Publishing House, CareerConnect: https://aphconnectcenter.org/ Also see the page titled 'Maintaining Your Job and Succeeding at Work': https://aphcareerconnect.org/succeed-at-work/maintaining-your-job/

[60] Univ. of Washington offers a wide range of assistances to students and faculty members under a code name, DO-IT. See https://www.washington.edu/doit/; https://www.washington.edu/doit/programs/advance

[61] Github, LaTeX cheat sheet, https://wch.github.io/latexsheet/

[62] NVDA Users Manual, posted at https://www.nvaccess.org/files/nvda/documentation/userGuide.html

[63] Paul Zimmermann et al., Computational Mathematics with Sagemath, Society for Industrial and Applied Mathematics, 2018. See https://epubs.siam.org/doi/book/10.1137/1.9781611975468

[64] See Wikipedia: https://en.wikipedia.org/wiki/Product_liability

[65] The HTML images for for Hokusai's Mt. Fuji can be seen at, https://tactilestem2.herokuapp.com
One can hear audio explanations in English by touching the lines and regions on the print image.

[66] Music scores in tactile form were in the Enlightenment Era. See https://www.britannica.com/topic/history-of-the-blind-1996241/The-blind-during-the-Enlightenment

[67] Oxford University Press, A series of books titled Very Short Introductions. https://global.oup.com/academic/content/series/v/very-short-introductions-vsi/?cc=jp&lang=en&

Accessible Digital Mathematics Library

Oğuz Akkaya

Association of the Visually Impaired in Education

oguzakkaya19@gmail.com

Abstract

Accessible Digital Mathematics Library aims to meet the needs of all blind individuals in Turkey who want to learn and teach mathematics for accessible mathematics materials. This library, whose foundations were laid based on the need for materials in the preparation courses for university exams given within the scope of the Distance Education Academy Project, which has been carried out with its own resources by the Association of the Visually Impaired in Education since its establishment, includes all subjects in the high school mathematics curriculum, documents consisting of solved sample questions and subject review tests. Technological developments in recent years allow, on the one hand, everyone to write mathematical expressions in MS Word without the need for additional software, and on the other hand, screen readers to verbalize mathematical expressions if they are written correctly in Word. For this reason, the Digital Mathematics Library contains documents prepared in Word, compatible with screen readers. However, audio versions of all contents are also available so that blind individuals who do not have compatible technologies can benefit from these contents. In the library there are also video training set for mathematics teachers and audio training set for students with visual impairment. We continue our work on our project that we started in January 2022. This project is carried out by Association of the Visually Impaired in Education with the grant support of the Empower Emerging Market Foundation.

1 Introduction

In recent years, the inclusion model comes to the fore in the education of the visually impaired. In this system where sighted and blind students are educated together, in order for the blind to be successful, teachers must first have knowledge of special teaching methods and techniques. However, most mathematics teachers in our country do not have sufficient knowledge about these methods, techniques and materials. In addition, there is a serious prejudice that blind people can only be successful in verbal and language areas and cannot make progress in science. So much so that in the university preference guides published by ÖSYM, visually impaired students are advised to prefer social sciences and art programs as it is difficult for them to be successful in science and technology.

Another major obstacle to blind people learning mathematics is the lack of accessible books. Especially students at high school level in Turkey face this problem. In order to make science and technology content accessible for the visually impaired, teaching materials are developed around the world using Braille solutions and audio and tactile methods (Bell and Silverman, 2019). In our country, A small number of advanced mathematics books and university Preparatory documents that students can access at high school level are mostly audio books read by volunteers of associations. Failure to establish a valid reading standard in verbalizing mathematical expressions reduces the efficiency of audio books. In addition, such materials are insufficient for mathematics education and are used around the world to support other solutions. When solving a mathematical problem, writing down the mathematical expression it contains makes it easier to work on it and to double-check what is given and requested about the problem while executing the operations. In addition, seeing a mathematical expression, which is abstract in nature, directly in written form, instead of listening to its verbal description, makes it easier to visualize and concretize it. Therefore, the use of Braille printed books or electronic materials compatible with screen reader software and Braille displays is gaining importance. However, most mathematics books are not printed in Braille by the Ministry of Education; The printed ones contain many errors. On the other hand, e-books have been widely used by the blind for a long time since the production of e-books is fast, easy and low-cost compared to Braille printed books. However, although Covid-19 conditions imposed distance education as a basic method, blind students could not actively participate in the education processes in Turkey due to the lack of digital materials (Telli, Kalaç, and Erönal, 2020). Unfortunately, E-materials prepared in Turkish in the field of mathematics consist of a few subject review tests.

The most important features that make e-materials stand out are that they enable sighted and blind people to work on the same material, work compatible with Braille displays, and are suitable for distance education conditions.

2 Aim of the Project

Electronic mathematics documents, which in the past could only be created using some programming languages, can be easily prepared by daily users, with technological developments, without the need for extra cost and expertise by using equation editor of MS Word (Junqueira, 2006). In addition, recent developments in the field of screen reader software have enabled the visually impaired to access these created materials (Perkins School for the Blind, 2023). However, publishers do not find it appropriate to share their content in these formats due to concerns that they may be easily reproduced. On the other hand, teachers and students are not aware of these technological developments. The main purpose of the project is to eliminate the problems of visually impaired students and mathematics teachers who cannot receive qualified education in mathematics throughout the country. This general purpose has three basic components. These are:

1) To establish an online mathematics library with mathematics documents digitalized by using Word.

2) To provide students with the skills of reading and writing mathematics documents with screen reader software in MS Word.

3) To increase the awareness and skills of mathematics teachers about programs that allow preparing accessible materials (Microsoft Word).

3 Main Outputs of the Project

We continue our work on our project that we started in January 2022. This project is carried out by Association of the Visually Impaired in Education with the grant support of the Empower Emerging Market Foundation. The work we have done so far in line with our goals can be summarized in six items:

1) A website was established that includes accessible digital mathematics content and educational records. The website was designed within the framework of the universally accepted WCAG 2.1 accessibility criteria.

2) The website includes instructional documents covering the mathematics subjects in the high school curriculum, and subject review tests with and without solutions at 3 levels of easy, medium and difficult for each subject. We also included preparation documents for many national exams. Our content developer converted mathematics content into text in an accessible format and created these contents in audio formats.

3) A training set consisting of 4 videos was created for mathematics teachers, covering how to create mathematical documents compatible with screen reader software using the equation editor in MS Word and how visually impaired students can access such documents with screen reader software. This video training set can be accessed on our YouTube channel and website.

4) An audio training set consisting of three contents on reading and writing mathematical documents was prepared for visually impaired students using screen reader software. Visually impaired students who are members of our website can access this audio education set.

5) To verbalize mathematics documents, Turkish Mathematics Reading Guideline was prepared, taking into account the MathSpeak rules developed by Dr. Abraham Nemeth and the opinions of Turkish experts on the Turkish vocalization of mathematical expressions.

6) Workshops were held with 125 mathematics teachers in 5 different regions of our country. Through these workshops, we aimed to increase teachers' awareness, knowledge and skills, as well as to establish a network of volunteers who will donate content to our library.

Our aim in the video training set and workshops is to provide teachers with awareness, knowledge and skills so that they can prepare content that is accessible in Word, and compatible with screen readers. Even math teachers who know how to use the equation editor in Word make some small mistakes that cause some difficulties for screen readers. For example, they may write the verbal expression into the equation.

$Find\ the\ value\ of\ x\ by\ solving\ the\ equation: 2 \cdot (x-3)-(2-x)=x+10$

For example, they may write a single equation as two separate equations.

$\frac{2}{1-\frac{1}{3}}+\frac{1-\frac{1}{3}}{1-\frac{1}{2}}$ =?

At the and of the August 2023, via the online form sent to 100 visually impaired students who are members of the Accessible Digital Mathematics Library it is amed at collecting;

1. Opinions about the online platform where the library is located,
2. Their views on reaching mathematical content written in Word by Turkish screen readers,
3. Their opinions on the Turkish mathematics reading standards developed for the library.

The findings of this Study are collected under 4 themes.

Participants' opinions about the online platform were collected under the themes of "accessibility" and "personalization". Participants described the platform as user and disabled friendly. They also saw the design of the platform to allow personalization as a positive feature.

Participants' opinions about reaching mathematics content written in Word by Turkish screen readers were collected under the theme of "inclusivity". Participants find this method inclusive. However, they emphasize that technological developments are not yet comprehensive enough. Although reading mathematics becomes much easier with this method, writing mathematics still poses some difficulties for the blind.

The participants' opinions about the Turkish mathematics reading standards developed for the library were collected under the theme of "understandability". Participants stated that the audio format increased the understandability of mathematics questions. However, they stated that some vocalization rules affected reading speed.

4 Summary and Conclusion

In mathematics teaching, electronic materials that allow blind people to read and write mathematical expressions, provide facilities for distance education, and are compatible with screen readers and Braille displays come to the fore. Therefore, teachers' prejudices against visually impaired students will be eliminated, as they realize that visually impaired students can easily write and read mathematics at the same time and with the same tools as their peers. Teachers who learn the standard reading rules of screen reader software will have standard knowledge about the mathematical discourses they should pay attention to when teaching mathematics to their visually impaired students. The fact that the method they learn is easy for the daily user will increase teachers' motivation to design the documents they prepare in the future in an accessible manner. From the students' perspective; gaining awareness that they can access mathematics at the same time as their peers and teachers will lead to a positive change in their beliefs that they can achieve mathematics. They will set goals in the science field and easily prepare for mathematics lessons and exams thanks to the contents in the digital library. Undoubtedly, this transformation; it will be an important step for future-sighted physicists, mathematicians, engineers and people educated in other fields to have a say in our country and the world.

References

References

Bell, E. C., & Silverman, A. M. (2019). Access to math and science content for youth who are blind or visually impaired, retrieved from https://nfb.org/images/nfb/publications/jbir/jbir19/jbir090101.html , on October 1st 2023.

Junqueira, K. E. J. (2006). Student teachers and Microsoft Word's Equation Editor. *Interim: Interdisciplinary Journal*, *5*(1), 38-44.

Perkins School for the Blind (2023), Accessible Math Editor: Word, retrieved from https://www.perkins.org/resource/accessible-math-editor-word/, on October 1st 2023.

Telli, G., Kalaç, M. Ö., & Eronal, Y. (2020). Covid-19 mücadelesi kapsamında uzaktan eğitim sürecinde engelli öğrencilerin durumu sorunlar ve çözüm önerileri. Manisa: Manisa Celal Bayar Üniversitesi Rektörlük Basımevi.

A Tool for Improving Readability of Japanese PDF Files for the Print Disabled and People with Foreign Roots

Akio Fujiyoshi

Graduate School of Science and Engineering, Ibaraki University, Hitachi, Ibaraki, Japan
akio.fujiyoshi.cs@vc.ibaraki.ac.jp

Abstract

This study introduces a tool for improving readability of Japanese PDF files for the print disabled and people with foreign roots. Since the Japanese writing system is unique in its writing directions and usage of characters, there are many people who find difficulty reading Japanese documents. Those people include not only people with foreign roots but also native Japanese citizens. The tool presented here has ability to convert vertical Japanese text into horizontal text and to add “Furigana” to Kanji characters in the text in PDF files. Furthermore, it has a function to unify the color of text by making the back of the characters white.

1 Introduction

The Japanese writing system is unique in its writing directions and usage of characters. Regarding the writing directions, there are vertical (left to right) and horizontal (top to bottom) writing styles. Although the percentage of horizontal writing is gradually increasing among modern Japanese documents, vertical writing is still norms for newspapers, magazines, and novels. Concerning the characters, both Chinese logogram characters “Kanji” and domestic syllabary characters “Kana” are used. There are less than 100 kana characters, whereas there are 2,136 regular-use Kanji characters, specified officially by the Japanese Ministry of Education. Needless to say, this is very troublesome for people with foreign roots. In addition, there are native Japanese citizens who have difficulty reading vertically written text, and many dyslexic students suffer from reading and writing Kanji characters. After the Corona disaster, administrative services have been increasingly digitized and information from the government is often distributed in PDF.

This study introduces a software tool, named PDFremaker, for improving readability of Japanese PDF files for the print disabled and people with foreign roots. In DEIMS 2021, a software tool that can extract all information of components from a PDF file was presented [1]. For the purpose of confirmation of extraction, the tool was able to reconstruct the document in SVG format from all extracted information of the components of a PDF file. We came up with the idea that the extracted information can also be used to remake a PDF file, and in the remaking process, we can modify some part of the components of an original PDF file.

PDFremaker has ability to convert vertical Japanese text into horizontal text and add “Furigana” to Kanji characters in the text of a PDF file. Furigana is small Kana syllabary characters placed above or alongside Kanji characters to indicate their pronunciation. Furthermore, it has a function to unify the color of text by making the back of the characters white.

The tool is developed in Java using Apache PDFBox library [2]. Apache PDFBox library is an open source Java library for creation of new PDF documents and manipulation of existing documents, published under the Apache License v2.0. The tool is available on the website of author’s laboratory.

2 PDFremaker

We introduce a software tool, named PDFremaker, for improving readability of Japanese PDF files for the print disabled and people with foreign roots. This tool is developed by taking advantage of the software tool presented in DEIMS 2021 that can extract all information of components from a PDF file [1]. The extraction tool pursues the rendering process of a PDF document and can extract the precise coordinate information on the base coordinate system of the document. A new PDF file can be remade from the extracted information, and in the remaking process, we can modify some part of the components of an original PDF file.

2.1 Conversion of Vertical Text into Horizontal Text

Vertical writing style is traditional method of arranging text vertically from top to bottom in east Asian countries such as China, Japan, and Korea. With the influence of Western printing and typesetting, the percentage of horizontal writing is gradually increasing among modern Japanese documents. However, vertical writing is still norms for newspapers, magazines, and novels.

As shown in Figure 1, the tool can convert vertical Japanese text into horizontal text. This function is accomplished by rotating the characters in vertical text by 90 degrees at the same center positions. The page box is also rotated by 90 degrees in the opposite direction. Special handling was required for symbols of punctuations, long notes, Arabic numerals, and parentheses.

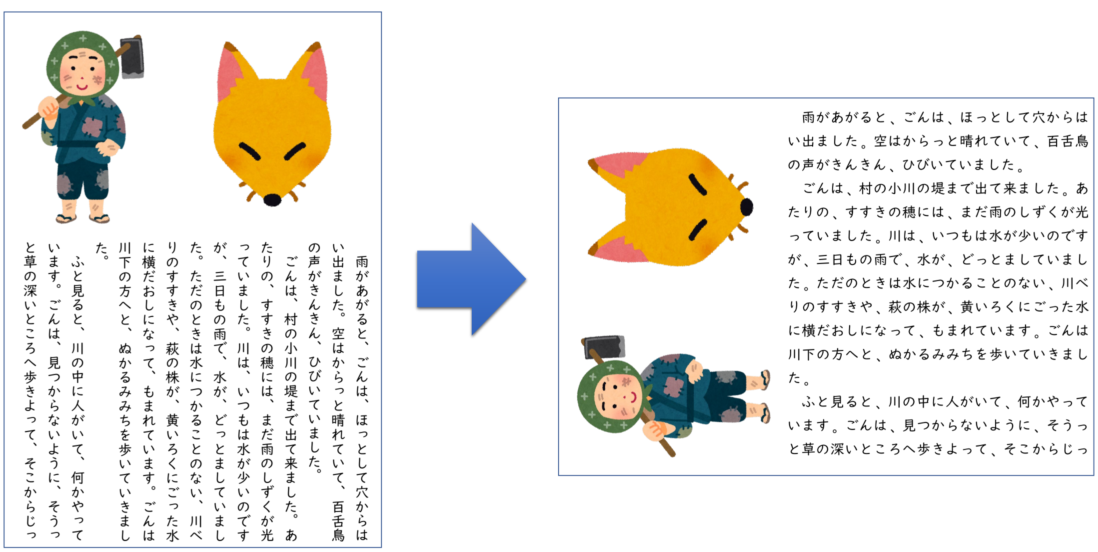


Figure 1: Convertion of Vertical Text into Horizontal Text.

2.2 Addition of Furigana to Kanji Characters

Furigana is small syllabary characters placed above or alongside Kanji characters to indicate their pronunciation. It is very useful for children and learners of Japanese. Furigana provides

Figure 2: Addition of Furigana to Kanji Characters: Original Document (upper) and Remade Document (lower).

a reading guide for Kanji characters, making the text more accessible and facilitating correct pronunciation.

As shown in Figure 2, the tool can add Furigana to Kanji characters in the text in a PDF file. In this example, some options are enabled for using universal design font [3] and unifying the color of text by making the back of the characters white.

A text file is generated in the same folder as a PDF file. By editing it, we can change Furigana assigned to Kanji characters.

3 Simple Processing Tools for PDF Files

PDFremaker becomes a member of simple processing tools for PDF files developed in the author's laboratory [4]. Other tools are "PDFcontentEraser", "PDFfontChanger" and "PDF-contentExtracter." PDFcontentEraser is a tool to remove a certain type of components in a PDF file. PDFfontChanger is a tool to change a selection of fonts in a document. As a default setting, all fonts are replaced by "Universal Design Font for Digital Textbooks" (UD font) [3] developed by Morisawa Inc.

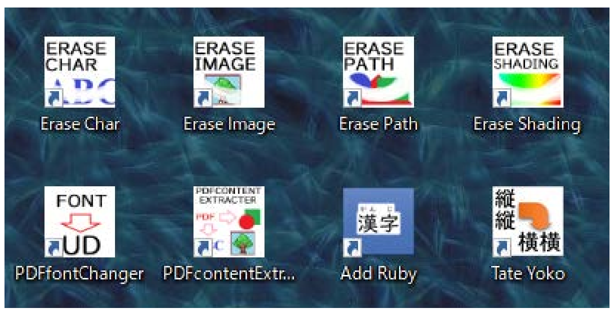


Figure 3: The Icons of the Simple Processing Tools.

The usage is very simple. When they are installed on a PC with Windows, icons are placed on the desktop as shown in Figure 3. A PDF file is processed by a drag-and-drop operation to these icons.

The simple processing tools for PDF files are available on the website of authors' laboratory: http://apricot.cis.ibaraki.ac.jp/PDFtools/.

References

[1] Shunsuke Nakamura, Kento Kohase, and Akio Fujiyoshi, Extracting Precise Coordinate Information of Components from E-Born PDF Files, Proceedings of DEIMS 2021, pp. 15-18, 2021/02

[2] Apache PDFBox, https://pdfbox.apache.org/.

[3] Morisawa Inc., Universal Design Fonts (in Japanese), https://www.morisawa.co.jp/fonts/udfont/.

[4] Shunsuke Nakamura, Kento Kohase, and Akio Fujiyoshi, A Series of Simple Processing Tools for PDF Files for People with Print Disabilities, Proceedings of the 17th International Conference on Computers Helping People with Special Needs (ICCHP 2020), LNCS 12376, pp. 314-320, 2020/09

Development of Assistive Learning Materials for Disabled Children, Which Make Online Video Contents Distributed by Zoos/Aquariums Be Barrier-Free

Yukiko Yamaguchi[1] and Akemi Matsumoto[2]
[1] Okayama College, Japan
yukiko@owc.ac.jp
[2] The Graduate School of Social Design, Japan
akemi.matsumoto@sentankyo.ac.jp

Abstract

This research aims to develop assistive learning materials to help visually disabled children recognize the main points of image-based information in online content, including videos, distributed by zoos or aquariums. At first, important/meaningful scenes of the video are picked up, and each scene is represented as a sequence of static pictures so that a story is advanced frame by frame. Next, only the animals are extracted from each picture. A stereo printer is used to produce tactile graphics for the extracted images of the animal, making it possible to feel the animal by touching, while all other visual components are printed normally in color. The assistive learning material can be used by both visually disabled and sighted children. In this paper, our current progress in developing the learning materials and future tasks is discussed.

(Keywords)
SDGs (Goals 4, 14, 15), Active learning, Barrier-free online content, Assistive learning materials for the visual disability, Zoos and aquariums, Inclusive education

1 Introduction

The Sustainable Development Goals (SDGs) adopted by the United Nations in 2015 are targets to achieve by 2030 that comprise 17 goals based on the philosophy of "leaving no one behind." The fourth of the 17 goals states "quality education for all" that insists strongly on the necessity to guarantee equal opportunities to learn, regardless of people with or without disabilities. From the perspective of quality education, the Ministry of Education, Culture, Sports, Science and Technology in Japan recommends the practice of active learning (independent, interactive and deep learning), which is clearly stated in the new curriculum guidelines; experiential and child-centered learning, the creation of environment for learning through the sharing of awareness, etc. are specifically recommended. From this perspective, visiting zoos and aquariums is one of important out-of-school activities for children. Exposure to a variety of living creatures allows children to gain scientific knowledge about the characteristics and behavior of living creatures; in addition, it is also expected to enhance environmental education through learning how to value biological diversity and nature. It is the important out-of-school activity that should also have educational effects on SDG 14, "Let's protect the abundance of the oceans," and SDG 15, "Let's protect the abundance of the land."

However, the COVID-19 pandemic that began at the end of 2019 has had a significant impact on children's educational environment. Although the treatment of COVID-19 in Japan has been reduced to the Category 5 level in May 2023, we should consider that children may continue to have difficulty finding opportunities for experiential learning. As will be discussed later, the animal welfare has become an issue for zoos and aquariums, and some of them are reconsidering activities that involve direct contact between humans and animals. In response to this situation, zoos and aquariums have focused on distributing online content, and as the pandemic persists, this content has grown more substantial.

This research aims to develop a prototype of assistive learning materials to help visually disabled children recognize the main points of image-based information in online content, including videos, distributed by zoos or aquariums.

2 Some Background

The online content often includes videos taken by zookeepers. Through these videos, children learn the characteristics and ecology of nature. Furthermore, they can develop a sense of familiarity and interest in animals, which may lead to future self-directed learning. These benefits can also be related to each other, depending on how videos are used. For example, a video content can be effective in education and serve as an active-learning material to deepen mutual learning through dialogue after watching the video. However, the problem here is that most online contents including videos cannot be fully used in its current state by children with visual disability. Even if explanations are embedded in audio, it is difficult for them to understand the content from audio information alone, and they cannot learn things in the same way as sighted children. Online content, which can be used from the home or school without having to travel to a specific location, is also easier to use for children with mobility difficulties. Making video contents barrier-free is definitely an important task. In addition, it is also difficult for visually disabled children to learn at the same level as sighted children through out-of-school activities that mainly involve observation at zoos and aquariums. This is a serious issue serving as a barrier to realize the basic principles of the SDGs.

Furthermore, from the perspective of animal welfare, some zoos have canceled activities that involve direct contact between children and animals; the use of video content is considered as an alternative. Thus, making the online contents barrier-free is an important issue. However, important and urgent questions remain on ways to convey necessary information to children who are unable (or find it difficult) to obtain visual information from the online contents, and how to establish educational environment where visually disabled and sighted children can learn from each other.

3 Method

As was mentioned, this study aimed to develop a prototype of an assistive learning material that will help visually disabled children recognize the main points of online content in videos distributed by zoos and aquariums. As the first step, we attempted to develop a learning material using the Okayama Creature School's "Growth Record of Red-crowned Cranes," which is distributed on YouTube by the Okayama Prefectural Nature Conservation Center [1]. This crane video was chosen as a subject because it was easy to understand the movement of the crane's head, neck, torso, legs, and body parts, and movements such as opening and closing its wings, standing, and sitting.

At first, important/meaningful scenes of the video are picked up, and each scene is represented as a sequence of static pictures so that a story is advanced frame by frame. After printing a scene from the video in regular color, we printed a tactile image with SINKA Corporation's EasyTactix to examine whether or not the characteristics and state of the animal could be realized by touching. EasyTactix is a printer that applies heat to special paper to "inflate" along carbon-toner lines/shading (a stereo image). The special paper can also be printed with an inkjet printer, and it is possible to produce printed materials that can be used by both sighted and visually disabled children.

If a stereo image of the entire video scene was printed as is, the plants or other things in the background of the scene would give rise to a difficulty to realize the animals themselves. Thus, the necessity to extract the animals before printing with EasyTactix has become clear. To extract the animals, we aimed to emphasize versatility and used Windows environment. We examined application software that is commonly used on PCs, and compared three types of preprocessing: Paint, Word/PowerPoint and Remove.bg. Although Word and PowerPoint are different apps, the tasks required in the two apps are the same. Remove.bg is free software although there is a certain limit in the number of times it can be used [2].

In terms of the three extraction methods, it has become clear that Paint is not suitable for our purpose. In an image extracted with Paint, thin lines such as The Red-crowned Cranes' legs could not inflate enough with EasyTactix, and it is impossible to accurately represent the animal entire body. In contrast, in Word/PowerPoint, the animal outlines tended to be easier to discern. Remove.bg allows us to erase the background and has the effect of highlighting the presence of the animals. Thus, we have concluded that preprocessing using a combination of Word/PowerPoint and Remove.bg would be appropriate.

His result was reported at a research meeting entitled "Various Problems concerning Information Accessibility" funded by Grants-in-Aid for Scientific Research in Japan [3]. At that meeting, we received some comments regarding the need for support materials that focus on animal movements, and other comments such as "I would like to know more about the living environment of animals." Based on these questions and advice received, we added further innovations and improvements to our prototype. Furthermore, to realize a highly versatile teaching-material-production method, we are also testing Photoshop, popular software for image processing and animal-extraction processing with Ichitaro/Hanako that are much popular in Japanese schools.

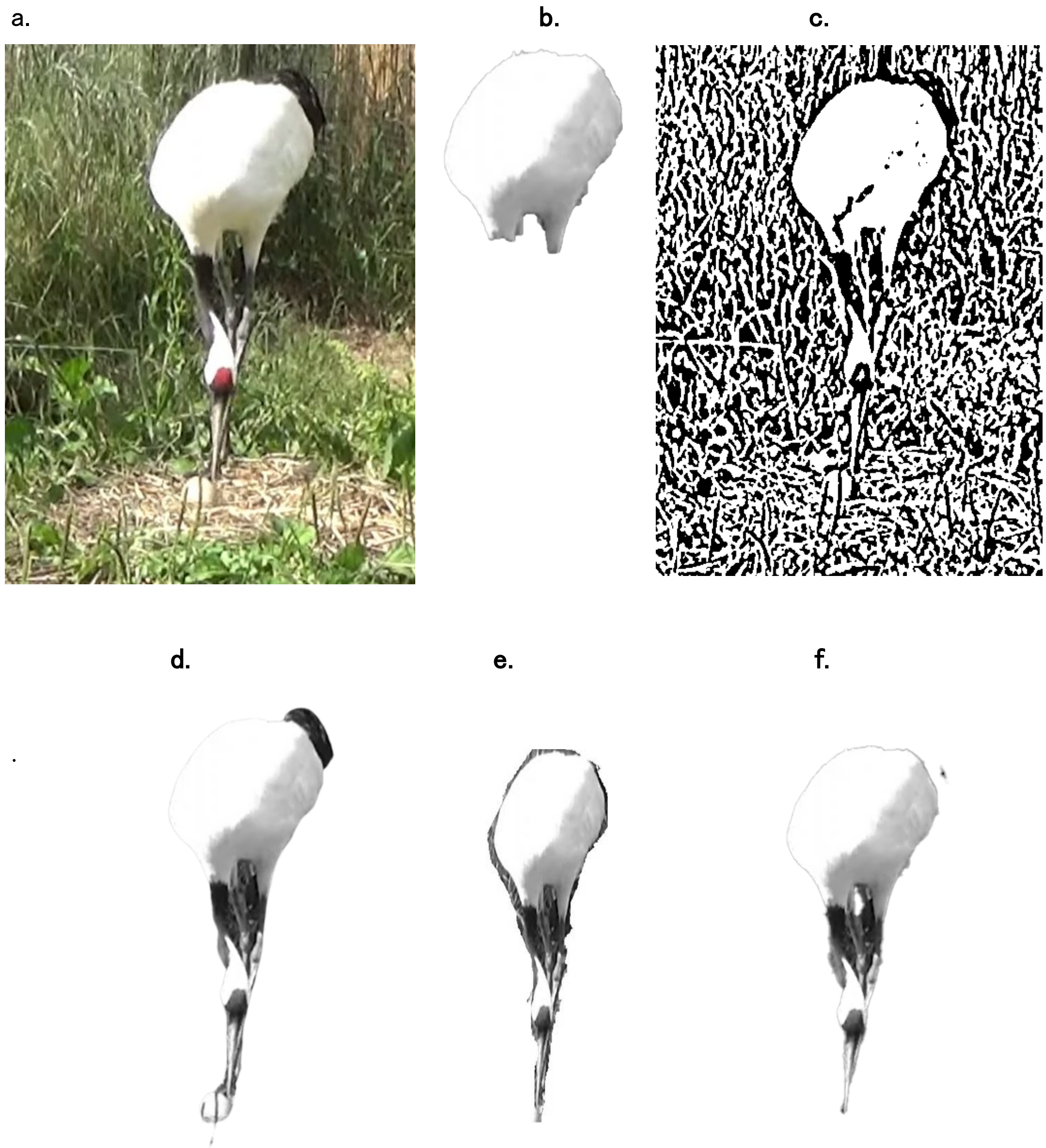


Fig.1 Results of examining extraction methods

We tried to extract animals from photos using several application software. Figures are a.original, b.Paint, c.Word/PowerPoint, d.Remove.bg., e. Ichitaro/Hanako and f. Photoshop respectively

4 Additional Discussions

Before closing due to COVID-19 pandemic, many zoos had so-called "petting zoo" corners where children could voluntarily interact with animals, for example, holding small animals such as guinea pigs; they were very popular at that time. Now the treatment of COVID-19 has been reduced to the equivalent of Category 5 level, but petting zoos are not necessarily resumed. In many cases, human-animal interaction is reconsidered from the standpoint of animal welfare. There are many zoos that do not allow petting because they consider that carelessly touching animals without understanding them should cause them undue stress. Thus, children may be unable to engage in experiential learning through touching. Interacting with animals provides them with the wealth of non-visual information, such as touching and smelling, and is an important learning opportunity for both visually disabled and sighted children. To carry out experiential learning with animals without jeopardizing animal welfare, understanding animals is definitely needed, and prior training for this purpose is required. Online videos constitute effective teaching material for pre-learning. If children have learned about the characteristics and nature of animals through videos and considered how to interact with them beforehand, through experiences at a petting zoo, they should be able to gain respect for life and consideration for others beyond scientific knowledge.

Nogeyama Zoological Park in Yokohama City has resumed direct interaction with guinea pigs and mice now after the pandemic has ended. From the perspective of animal welfare, they conduct video training before interacting with animals to teach them how to properly interact with animals. In addition, the prior training is not only meant to reduce the burden on animals but also to ensure the safety of those who handle them. For example, if you put your finger in front of a guinea pig's mouth, you run the risk that it thinks of the finger as it's food and bites it. Guinea pigs are usually kept in a cage, and you can touch them from above without taking them out of the cage. If you are careful by learning in advance, there is almost no risk of being bitten or injured. However, we heard of a case that a blind student was not allowed to touch guinea pigs for fear of being bitten even after receiving instructions. This suggests that there should exist the problem of information disparity; the content of prior learning is not accurately communicated. That is one of the reasons behind our currently working on the development of support materials focusing on guinea pigs which are the main animals in petting zoos.

At this workshop, we plan to report on the progress on how to produce highly versatile teaching materials and also show the materials that we have actually produced.

a.

b.

.

Fig.2 Image of Assistive Learning Materials for Disabled Children

In the Assistive Learning Materials, only the animals in the photos have irregularities(a.). The figure below shows only the irregularities(b.).

(Acknowledgment)
We appreciate the cooperation of Mr. Shigetoyo Ikemoto of the Okayama Prefectural Nature Conservation Center, Mrs. Aiko Saito of Nogeyama Zoological Park and Mr. Kozo Fujiwara, his assistant (visual support), with this project. We are supported by JSPS Grants-in-Aid for Scientific Research, Grant Number 22K12306, for which we are deeply grateful.

References

[1] Okayama Creature School "Growth record of cranes" (2021)
https://www.youtube.com/playlist?list=PLS1vOb39Ky6SjRnvvgLEOffXYV-YFpl34
[2] Remove.bg
https://www.Remove.bg/ja
[3] Yukiko Yamaguchi (2022)
"Development of Assistive Learning Materials for Disabled Children to Make Online Video Contents Be Barrier-Free," (Japanese only)
https://www.dropbox.com/s/y4tlh6stg1nv88y/2022%E5%B9%B4%E5%BA%A6_%E7%A0%94%E7%A9%B6%E9%9B%86%E4%BC%9A%E4%BA%88%E7%A8%BF%E9%9B%86.zip?dl=0

Help seeking from students with a VI in a mainstream classroom setting. How can digital tools support help seeking?

Mathieu Gaborit[1]*, Dominique Archambault[2], and Minna Puustinen[1]

[1] INS EI/GRHAPES, Suresnes, France
[2] Université Paris 8-Vincennes-Saint-Denis, France

Abstract

Secondary students with a visual impairment (VI) are mainly schooled in mainstream classrooms in France. Because of their visual limitations, they may find it hard to identify potential helpers. Thus, misunderstanding may occur such as: who the teacher is looking at? or, is the teacher questioning me when he says "yes, please?". These students may tend to ask help from their table companion rather than from the teacher, even for complex learning issues. The teacher, as well as the other companions, are sometimes kept apart from help interactions. So, how can digital tools support help interactions in classroom?

1 Background

Help seeking is a learning strategy based on social interactions [8, 7, 13]. In a classroom setting, students may request help to proceed with an academic task in which they find it hard to solve a problem by their own. But also, when students are asking for help, they let the teacher know about their difficulties. Thus, the teacher can regulate the lesson in accordance with the identified difficulties [12]. Research has shown that students may ask help from the teacher or from peers when in classroom [4]. Their choice in selecting the helper depends on how they consider the potential available helpers they request help from. For instance, students consider teachers more trustworthy for learning issues than peers, but teachers also are more associated with evaluation [15].

In digital environments, help may take a different shape. Different features related to help in digital environments have been compiled [6], notably on a timescale issue (for example, an answer to a help request may be postponed because the helper is offline), or regarding the availability of the answer (for example, if the answer is published on a forum, it can be available for several students [14]). In addition, one of the features refers to confidentiality. In some digital environments, students have to display their name, in some others, they can use an alias and benefit from privacy so that they can receive feedback without suffering public embarrassment [9]. In a live chat interface, students declare to be happy to take advantage of help support when studying distantly from the classroom [2]. Notably, they appreciate real time responses and staff accessibility. Academic help forums have also proved to be an interesting environment for helping students within academic tasks [14]. Indeed, students can post help requests on the forum and the requests are answered by teachers and/or other students. In the latter case, a reply from another students may be faster, and can later be confirmed, or corrected by the teacher who could also provide additional information. But above all, requests and answers are available for all the registered forum users so that they can take advantage of any help interaction they are interested in, even if the original requests are not theirs.

*Corresponding author: Mathieu Gaborit, gaboritmathieu@gmail.com

In France, more than 50 % of the secondary students with a VI are included in mainstream schools. Half of them daily use a digital tool in classroom, such as braille display, a suitable device or a specific software application [5]. They use their device for reading (Braille) and listening (audio). They also use the device to write their work from original paper instructions documents.

Students with a VI cannot always identify who they can get help from in the classroom, as the case may be, because, depending on their visual limitation, they may find it hard to localize their peers or the person they would ask help from [1]. In addition, teachers sometimes tend to use gestures to point and question students. In this case, students with a VI in an inclusive environment may not be able to identify who is questioned and who is raising their hand. Furthermore, when students with a VI raise their hand to ask for help, they cannot make sure they have been noticed by the teacher. For example, when the teacher turns their back to the students with a VI because they faces another student, students with a VI may not understand why the teacher does not react to their raising hand. In Gaborit [6], a video shows a student with a VI who does not receive an answer from the teacher while raising her hand and eventually gives up asking for help.

Seven French middle-school students with a VI were filmed during mathematics and biology classes [6]. Students used either a computer, or a tablet, or a braille notetaker. Their requests dealt with ICT as well as learning issues. The results of the research in natural context (*i.e.* in classroom during regular lessons [6]) show that students with a VI tend to ask for help from the teacher, a peer-tutor, a table companion [6]. Almost 75 % of all requests (220 over 295) were submitted to the table companion. Table companions are not always competent to answer complex issues. But also, they are not always available because they are busy with their own work.

An important issue when submitting a request to table companions is also that the teacher can be deprived of the student's difficulty which the request is about. Thus, the teacher may not be able to regulate the lesson in accordance with the students' needs.

2 Question

A great number of classrooms are equipped with a digital whiteboard, which may be used to display images, texts, instructions, evaluations. When students use tablets, they can be connected to the digital whiteboard [16] so that students' work can be displayed for collective reflection. For example, if a student faces with a difficulty in proceeding one task, his primary/intermediary work can be displayed for all and be used as a collective basis for reflection. May a help request be displayed the same way?

It seems that ICT could play a role in help seeking because ICT can provide the help seeker with confidentiality, and it can make help-seeking interactions available for all. In this regard, it would be interesting to consider ICT as a potential interface in which students could publicly post a help request which could be useful for all. It also would be easier for the students with a VI to make a request when they do not have to identify helpers and/or catch their attention. In addition, the digital interface could provide the teacher synchronized feedback of the students' difficulties enabling him to regulate the lesson content and rhythm.

This study shows that we need to start thinking about solutions in which ICT can support students' help-seeking in classroom. These solutions could be integrated in the Assistive Technology used by the students.

3 Discussion and proposition of a digitalised solution

Several issues have been identified in situations in which students with a VI may find it hard to seek for help. First, as far as the students with a VI are concerned, it may be difficult to select a helper because they sometimes cannot see enough to know who is present and where. When they manage to localize a potential helper, they may not be able to know if the helper is available to help them. This specific feature could explain (at least partly) why Gaborit [6] found out that 75 % of the help sought by seven students with a VI involved a table companion. On the one hand, the table companion may be a good choice because the help can be requested in a private way, without public embarrassment (*i.e.* the other students and the teacher do not know about the student's difficulty). On the other way, if the table companion occurs to be helper by default, the student in need will have to accept the given help, whether adapted or not. Table companions are not always competent to answer complex issues, though observation in natural classroom setting [6] show that table companions are responsive.

Second, as far as the teachers are concerned, a public request helps them regulate the lesson and lets them know about the student's problem. A public answer can also help all students, even those who did not actively participate in the help interaction. However, several research works [3, 6, 11] show that teachers are not always sensitised with some specific features in oral communication with blind individuals (such as not using sign deictics). On the one hand, Gaborit [6] observed that some teachers tended to physically point at students to question them without naming them. If so, they created confusion for the students with a VI. On the other hand, some students [6] raised their hand without being noticed by the teacher and finally gave up asking for help.

These observations led us to propose a specific IT solution in order to cope with the latter issues. A live chat style interface (let us name it the *Help Seeking Interface* – HSI) would be displayed on both the students' device and the the connected digital whiteboard. If students with a VI (but also sighted students) need help, they may write the request on the HSI from their device. The request would instantly appear on a pop-up window on the digital whiteboard so that the teacher is alerted of the incoming request. The request is publicly displayed, and the teacher may answer it or make the other students answer it. But also, the teacher can make the choice to delay the answer and help the student in a more opportune moment. Another solution would be to display the HSI on a secondary screen that is only visible by the teacher in order to not disturb the students' activity.

In this situation, the student with a VI does not find it hard to identify and localize a helper, including the teacher who is alerted of an incoming request for help by the HSI. As far as the teacher is concerned, they are aware of the student's difficulty, they can provide a real time response, they can publicly answer the question so that all the students can benefit from the help interaction. Regarding the confidentiality issue, settings in the HSI could provide the teacher and/or the students with optional anonymous and private modes. In the anonymous mode, the request appears on the digital whiteboard, but the inquirer is not identified. In the private mode, a pop-up on the digital whiteboard gives the teacher notice of an incoming request which they can read from their personal secondary screen. In both modes students do not suffer from potential public embarrassment and the teacher may direct the answer to the whole class. An additional advantage of such a solution is that the teacher could keep records of the students' difficulty by saving the history of the students' requests which have been posted on the HSI.

Such a solution entails specific needs from both the teachers and the students. Apart from the fact that the users must master the HSI, the interface needs to be (1) accessible and (2) usable by the students with a VI.

First, it would need to be compatible with Assistive Technologies, such as screen readers for instance. Second, the student would need to receive confirmations for their requests, both when sending and when reading. In this case, the teacher must have the choice with different type of responses which would appear on the student's device, such as "I can answer now"; "I will answer later"; "the question is irrelevant"; etc. It also would be interesting for the teacher to be able to answer in a private message (when he has the time for it) in order to answer the student's request without mobilising the whole class. Thus, an additional type of response would be "I will answer in a private message". Still, the moderation of such an interface during the lesson by the teacher should not disturb the teacher's progression in the lesson or mobilise them too much.

In practice, the HSI should be available as an application set up on the student's device and which would be automatically opened and connected to the teacher's computer through a local network when the device is on. The HSI on the student's device would display few quick commands such as a write field, a "send" button and a response field. Each request can be listed the one under the other in a chronological order and indicating the time the request was sent. The HSI on the teacher's device (digital whiteboard or secondary screen) could display information such as the request, the sender (if need be), the time, the answer and one tactile answer button for each of the pre-registered responses cited above.

4 Conclusion

It appears that help seeking from students with a VI in an inclusive academic setting can be beneficial for all students. Though, social environment can have an impact on students' help seeking behaviour (see for example [10]. Hence, digital tools may help classroom communication by supporting interactions. The HSI solution which is proposed in this paper must be considered as a communication tool and not an alternative to traditional forms of human teaching. However, learning technics and tools can differ from a teacher to another. So, some setting options must be considered so that such an interface can be used widely. The setting options presented in this paper must be developed to satisfy different forms of teaching.

References

[1] Thierry Bourgoin. Éducation physique et sportive. In Nathalie Lewi-Dumont, editor, *Enseigner à des élèves aveugles ou malvoyants*, pages 241–247. INS HEA, Réseau Canopé, 2016.

[2] Jaclyn Broadbent and Jason Lodge. Use of live chat in higher education to support self-regulated help seeking behaviours: a comparison of online and blended learner perspectives. *International Journal of Educational Technology in Higher Education*, 18(1):1–20, December 2021. Number: 1 Publisher: SpringerOpen.

[3] Laetitia Castillan. *Améliorer l'accessibilité des manuels scolaires pour les élèves déficients visuels*. phdthesis, Université Toulouse le Mirail - Toulouse II, June 2020.

[4] Kyle Davison, Lars-Erik Malmberg, and Kathy Sylva. Academic help-seeking interactions in the classroom: A microlongitudinal study. *British Journal of Educational Psychology*, 93(1):33–55, 2023. _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/bjep.12538.

[5] DEPP and SIES. Repères et références statistiques sur les enseignements, la formation et la recherche (RERS), 2021. https://www.education.gouv.fr/reperes-et-references-statistiques-2021-308228, accessed 2024/01/07. DEPP and SIES are statistical services of the French Ministries of National Education, Youth and Sports, and of Higher Education and Research.

[6] Mathieu Gaborit. *La demande d'aide en classe : Une analyse de données recueillies auprès de collégiens avec un trouble de la fonction visuelle et proposition théorique transversale.* phdthesis, Université Paris Nanterre, October 2023.

[7] Sharon Nelson-Le Gall. Help-seeking: An understudied problem-solving skill in children. *Developmental Review*, 1(3):224–246, September 1981.

[8] Stuart A. Karabenick, editor. *Strategic help seeking: Implications for learning and teaching.* Lawrence Erlbaum Associates Publishers, Mahwah, NJ, US, 1998.

[9] Stuart A. Karabenick and John R. Knapp. Effects of Computer Privacy on Help-Seeking1. *Journal of Applied Social Psychology*, 18(6):461–472, 1988. _eprint: https://onlinelibrary.wiley.com/-doi/pdf/10.1111/j.1559-1816.1988.tb00029.x.

[10] Stuart A. Karabenick and Richard S. Newman. Seeking help: Generalizable self-regulatory process and social-cultural barometer. In *Contemporary motivation research: From global to local perspectives*, pages 25–48. Hogrefe & Huber Publishers, Ashland, OH, US, 2009.

[11] Nathalie Lewi-Dumont. *Enseigner à des élèves aveugles ou malvoyants.* INS HEA, Réseau Canopé, 2016.

[12] Anaïs Loizon and Patrick Mayen. Le cours magistral en amphithéâtre : une situation d'enseignement perturbée par les instruments. *Distances et Médiations des Savoirs*, 3(9), March 2015. Publisher: CNED-Centre national d'enseignement à distance.

[13] Minna Puustinen. *La demande d'aide chez l'élève : Avancées conceptuelles, méthodologiques et nouvelles données.* Figures de l'interaction. L'Harmattan, Paris, France, 2013.

[14] Minna Puustinen, Josie Bernicot, Olga Volckaert-Legrier, and Michael Baker. Naturally occurring help-seeking exchanges on a homework help forum. *Computers & Education*, 81:89–101, February 2015.

[15] Adnan Qayyum. Student help-seeking attitudes and behaviors in a digital era. *International Journal of Educational Technology in Higher Education*, 15(1):1–16, December 2018. Number: 1 Publisher: SpringerOpen.

[16] François Villemonteix, Dany Hamon, Sandra Nogry, Arnauld Séjourné, Bruno Hubert, and Jean-Michel Gélis. Expérience tablettes tactiles à l'école primaire - ExTaTE. Research Report, Laboratoire EMA, September 2015.

Generating Image Description from Labels and Surrounding Text

Gopi Swamy Veerendra[1], M Balakrishnan[1], Volker Sorge[2], and Abhishek Baghel[1]

[1] Indian Institute of Technology Delhi, India
[2] University of Birmingham, United Kingdom

Abstract

Analysis of Science textbooks used widely in Indian schools showed that information surrounding the image often explain the key aspects of the image. This led us to explore an interesting approach to generate relevant image description. We first perform an OCR to extract the labels from the image and extract the corresponding caption provided for the image. We use these labels and caption to form a statement and search the text surrounding the image and extract the relevant sentences to the formed statement using sentence similarity. Study and experimentation with a large number of images have helped us identify the optimal settings of this search – in terms of the number of sentences around the image to be searched, which model of sentence embeddings to be used, and other optimizations. Finally, an image description is synthesized from the labels, caption and the extracted similar sentences from the surrounding text using the NLP summarizing tools. Initial experiments show that the technique has a potential for automatic image description.

1 Introduction and Motivation

One of the biggest impediments for the visually impaired to read and understand STEM material is their inability to see figures which describe experiments or provide illustrations. As part of the RAVI (Reading Accessibility for Visually Impaired) project, prior work by Chahal et al., 2018 [3] has shown that providing alternate (handwritten) text description for images greatly helps understanding for visually impaired. In that work they have designed an effective template to write such descriptions.

Table 1. Template sections and sources

	Template Section	Source of Content
1	Overview	Image caption
2	Number of Objects	From Image Processing
3	What are the Objects?	From the labels (e.g. Beaker, Burner, Test Tube)
4	Any specific placement?	From label positioning (Left to right, top to bottom, or none)
5	How are the Objects placed?	Relative positioning of the objects (e.g. x is at the top, y is on the left of x)
6	How are the Objects interacting?	This has to be extracted from the text surrounding the image. (e.g. beaker heated using the burner)

In this paper, we describe an approach that can eventually help automatically filling such template. We extract most of the relevant information required to fill the template automatically including image captions, image labels and their positions, and relevant text to the image from nearby text. We also generate a short summary of the relevant text, caption and labels using NLP (Natural Language Processing) tools. In future work, one can combine this information to automatically write text descriptions of images following the template. Since our project mainly deals with making school STEM books accessible to the visually impaired, our data set was formed by diagrams contained in the NCERT[1] science textbooks from classes 6^{th} to class 12^{th}. We also augmented the data set using some technical research papers.

2 Methodology

The process involves the following key steps:

1. Extraction of the caption of the diagram
2. Extraction of the labels and their positions
3. Extraction of text relevant to the diagram
4. Creating a summary description of the diagram

We are using epub files in this project because of structured data and clear images. All images are present in the Images folder of an unzipped epub file. The text folder have all HTML or XHTML files for each chapter, and this folder may contain one chapter or multiple chapters depending on the design of the textbook or technical paper. Figure 1 shows the directory folders we generally expect once we unzip an epub file.

[1]NCERT is the nodal agency that produces textbooks used by Schools affiliated to the Central Board of Secondary Education in India

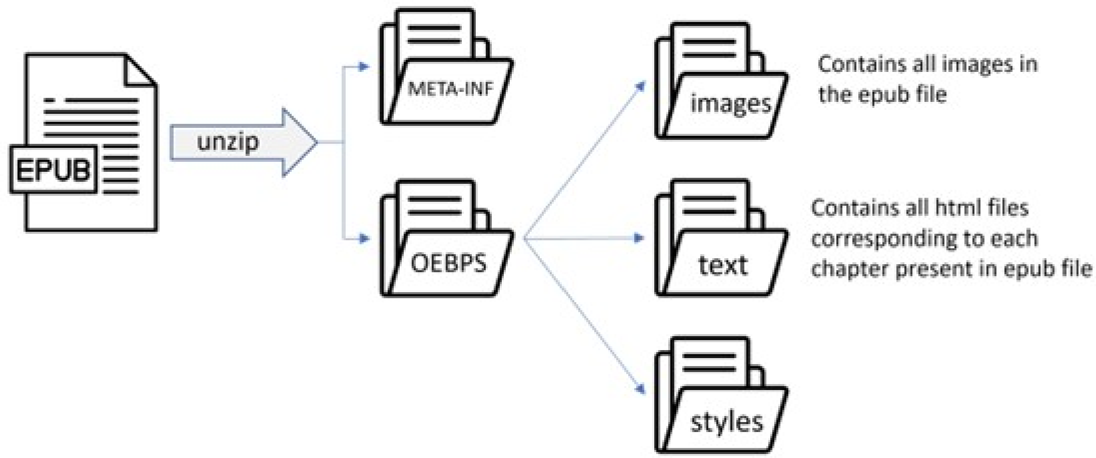


Figure 1: Folders directory when epub is unzipped to html or xhtml files

2.1 Caption and image extraction

A simple algorithm is used for extracting captions for the corresponding images. Sentences that begin with Fig (covering Fig. and Figure) and are within certain word lengths are considered to be the candidates. This have to be associated to a diagram to finally qualify as a caption. Diagrams are extracted using the “Src” tag on the html files.

2.2 Extracting labels and their position

To extract text as labels and their positions, we tested two solutions namely, Pytesseract python library and Google cloud vision API. Google cloud vision API performed better. In Google Cloud Vision API also there are two annotation features that support OCR.

- TEXT-DETECTION: detects and extracts text from any image. For example, a photograph might contain a street sign or traffic sign. The JSON includes the entire extracted string, as well as individual words, and their bounding boxes.

- DOCUMENT-TEXT-DETECTION: also extracts text from an image, but the response is optimized for dense text and documents. The JSON includes page, block, paragraph, word, and break information. In our testing, we saw that DOCUMENT-TEXT-DETECTION is performing much better compared to TEXT-DETECTION. As Google Cloud Vision API is a paid service based on number of images, we optimized it by storing the JSON object we get from google API call.

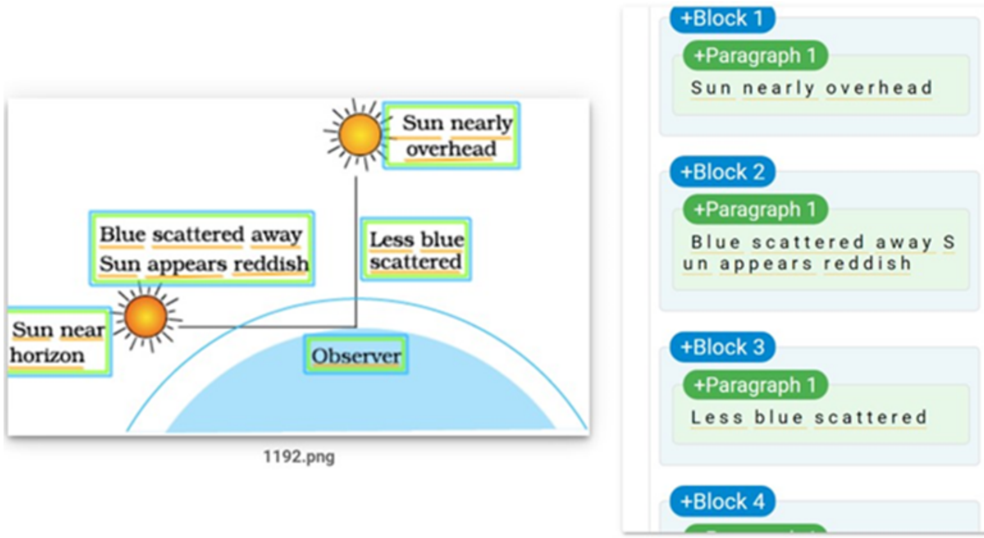


Figure 2: Visual depiction of labels and their positions from Google Cloud Vision API

2.3 Extracting relevant text

We extracted caption, corresponding image, labels and their positions in the image. Now in order get what the image is describing or referring to, we need to find relevant text from the nearby text around the image. The main challenge of finding relevant text from the nearby text is to find relevance between the image and the text surrounding it. Since we had information of caption and labels of the image, we use them to find the relevance between the image and text surrounding it. We append labels to caption to make a sentence i.e., (caption + "This figure contains " + labels separated by commas). Finally, the objective is to find relevance between the labels appended caption and sentences surrounding the image. The problem of finding relevance between sentences falls under NLP, specifically under Semantic textual similarity.

2.3.1 Sentence embeddings

Sentence Embeddings is the collective name for a set of techniques in NLP where sentences are mapped to d-dimensional vectors of real numbers i.e. sentence $S \to F(S) \in \mathbb{R}^d$. Sentence Transformers is a Python framework for state-of-the-art sentence, text and image embedding. The initial work is described in Sentence-BERT: Sentence embedding using Siamese BERT-Network by Reimers and Gurevych [4]. These embedding can then be compared e.g. with cosine-similarity to find sentences with a similar meaning. This can be useful for semantic textual similar, semantic search, or paraphrase mining. Here, we convert our sentences into sentence embedding using various models of Sentence Transformers to find relevant sentences between the labels appended caption and sentences surrounding the image. If S_1 and S_2 are similar then $F(S_1).F(S_2)$ is large or Cosine similarity ($cos\theta$) between $F(S_1)$ and $F(S_2)$ is close to 1. Figure 3 shows an example of how sentence embedding dot product values look like.

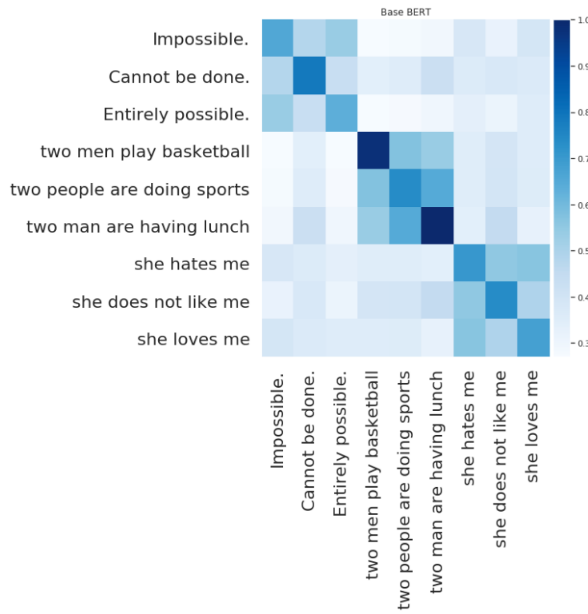


Figure 3: Similarity between sentence pairs encoded by sentence transformer model

We tested with various sentence embedding models from Hugging Face [2] like BERT (bert-base-nli-mean-tokens), all-MiniLM-L6-v2, paraphrase-multilingual-MiniLM-L12-v2, multi-qa-MiniLM-L6-cos-v1, MPnet (all-mpnet-base-v2).

2.3.2 Testing methodology

In our case, **False Negatives** are the sentences that are not predicted by the model as relevant sentences but are relevant sentences as per ground truth. **False positives** are the sentences that are predicted by the model as relevant sentences but are not relevant sentences as per ground truth. Figure 4 visualizes a general scenario of False Positives and False Negatives.

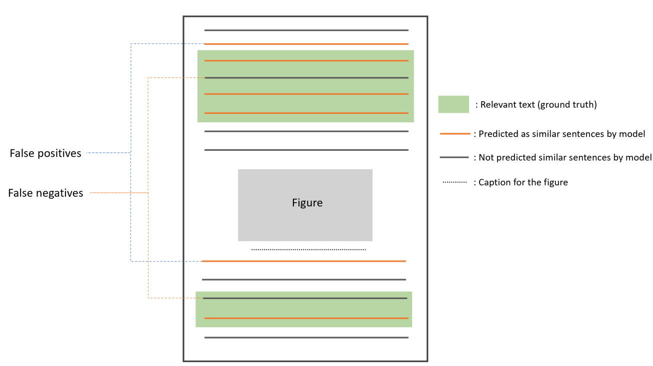


Figure 4: Visualizing false positives and false negatives in our scenario

Now we define False Negative Rate (FNR) and False Positive Rate (FPR) as follows:

$$False\ Negative\ Rate(FNR) = \frac{Number\ of\ False\ Negatives}{Number\ of\ relevant\ sentences\ as\ per\ ground\ truth}$$

$$False\ Positive\ Rate(FPR) = \frac{Number\ of\ False\ Positives}{Total\ number\ of\ sentences\ predicted\ as\ relevant\ by\ model}$$

We fix some threshold ϕ and all sentences whose cosine similarity to labels appended caption sentence crosses ϕ is predicted as relevant by the model. Then we calculate average FPR and FNR for varying threshold ϕ and plot a scatter plot for each model.

Figure 5 shows an example FNR-FPR scatter plot with varying cosine thresholds. Ideally, we expect FNR and FPR to be small. We will choose ϕ from the plots such that both FNR and FPR are minimized.

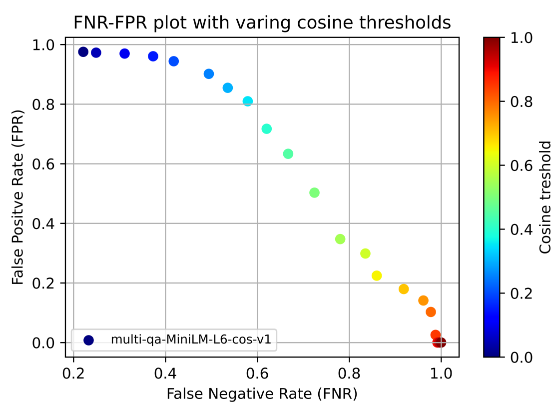


Figure 5: Example FNR-FPR scatter plot for a single model

To compare models, we plot similar scatter plots for each model into one plot as shown in figure 4.4. We can see in Figure 6 that **all-mpnet-base-v2, all-MiniLM-L6-v2,0 and multi-qa-MiniLM-L6-cos-v1** models are doing better compared to remaining models.

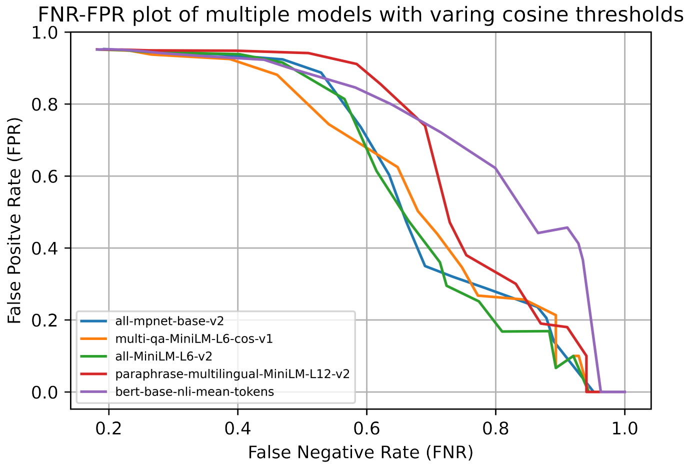


Figure 6: Example FNR-FPR plots for all models

2.3.3 Heuristics for improving performance

We experimented with various heuristics to improve FNR-FPR plots i.e. to bring down FNR and FPR values like smoothing cosines values, creating bounding box for searching relevant sentences, and predicting the relevant sentences not just above the threshold ϕ but other variations of using ϕ

Smoothing cosine similarity scores

Ideally, we expect relevant sentences are usually surrounded by other relevant sentences because all text generally in one area revolves around one topic. Hence, by locally smoothing cosine similarity scores i.e. convolution of cosine values with a smoothing curve, we can expect to reduce FPR-FNR rates. But there are various techniques available to locally smooth a curve.

- Convolution with Uniform curve with various window size

- Convolution with Gaussian curve with various window size

- Rolling max with various window size

Figure 7 shows whether smoothing improves the predictions.

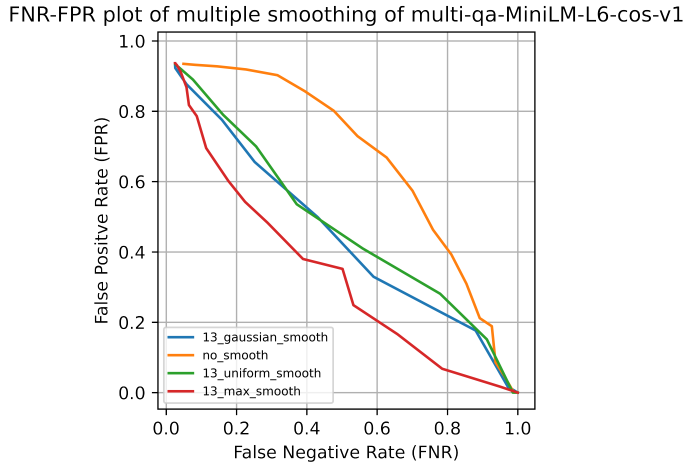


Figure 7: Comparison of various smoothing techniques for a single model

Note: Number before smoothing curve labels in figure 7 is window size for convolution

We can see that any locally smoothing of cosine similarity scores improve predictions. This trend is same for all other models.

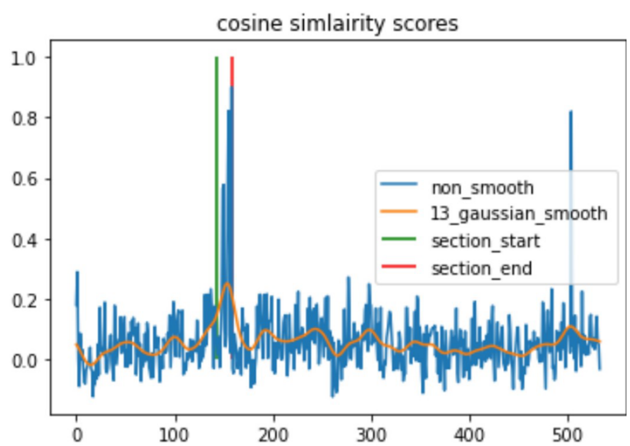


Figure 8: Example of a smoothed cosine scores

Why smoothing is improving the prediction?

We expect relevant sentences are usually surrounded by other relevant sentences but there are many cases where few sentences talk about similar things in a different scenario. For example, figure 8 shows there is a peak of cosine score outside the relevant section, but after smoothing, the peak outside the section was smoothed by the surrounding low cosine scores.

We found out that optimal smoothing for the cosine values is Gaussian smoothing with the window size of 7 values around it.

Various heuristics for threshold

Until now, we are finding optimal threshold from the FNR-FPR plots for predicting a sentence cosine similarity crosses that absolute threshold but we were not considering the noise, i.e. there may be many other sentences in the whole text that use the same words and may get classified as relevant sentences. To remove the noise in the cosine similarity values we introduced six heuristics:

- H_0 : Predict above λ
- H_1 : Predict all text between the first and last sentence above λ
- H_2 : Predict above $\mu + \lambda$
- H_3 : Predict above $\mu + \lambda * \sigma$
- H_4 : Predict above $\mu^* + \lambda$
- H_5 : Predict above $\mu^* + \lambda * \sigma^*$
- H_6 : Predict above $\mu + (\lambda * \mu)$

μ, σ : mean, standard deviation of all cosine scores in that file
μ^*, σ^* : mean, standard deviation of cosine scores inside the bounding box
λ : optimal threshold to find

Bounding box

From our experience whenever we encounter a figure, we expect that the relevant text will be within 1-2 pages around the figure. To capture this experience into our code, we will keep a bounding box(BB) around the figure and only within the bounding box sentences predicted as relevant would be considered. Sentences outside the bounding box even if predicted as relevant by the model are not considered. We will apply this Bounding box concept to H_2. Now, let's observe that all plots in figure 9 show improvement after the BB was applied compared to no BB. The same trend is observed for other heuristics. BB with width of 60 sentences, i.e. (**30_with_boundingbox**) 30 sentences on either side of the figure as bounding box, works best.

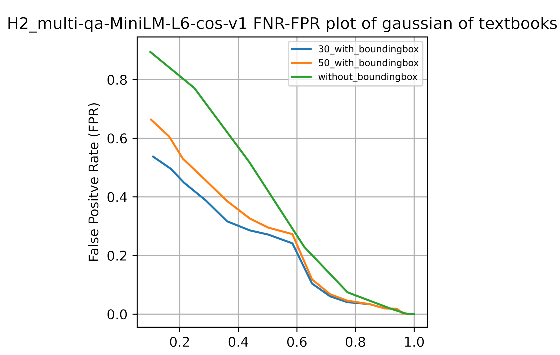


Figure 9: Effect of applying bounding box

Results of extracting similar sentences after optimal settings

We can observe that in Figure 10 the predicted relevant text roughly matching with the actual ground truth that was marked manually. This trend is observed in many other examples, which shows the settings for the sentence similarity model is working as expected.

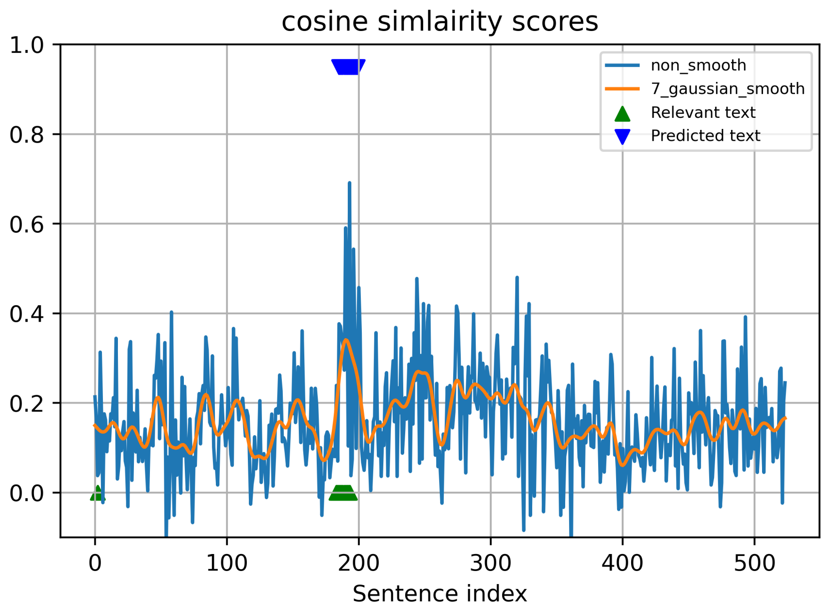
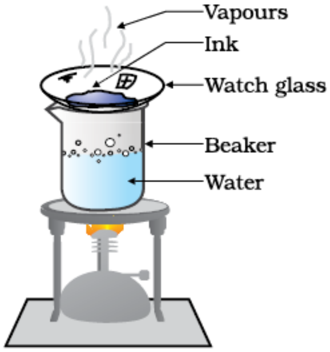


Figure 10: Example of cosine scores with predicted & actual relevant sentences

2.4 Creating summary description

Summarization is the task of producing a shorter version of a document while preserving its important information. Some models can extract text from the original input, while other models can generate entirely new text. Transformer models can be used to condense long documents into summaries, a task known as text summarization. Although there are multiple pre-trained summarization models, we tried particularly one model sshleifer/distilbart-cnn 12-6 [1]. We used this summarization tool to summarize the extracted information i.e caption + labels in the figure + extracted relevant sentences.



Summary with labels

Figure 2.5: Evaporation. this figure contains vapours ink, watch glass, beaker water. You will see that evaporation is taking place from the watch glass. What do you think has got evaporated? Is ink a single substance (pure) or is it a mixture?

Summary without labels

Fill half a beaker with water. Put a watch glass on the mouth of the beaker (Figure 2.5) Put a few drops of ink on the watch glass. Now start heating the ink. You will see that evaporation is taking place. Now answer: Is ink a single substance (pure) or is it a mixture?

Figure 11: A diagram description example

3 Conclusion

For a visually impaired image descriptions are a key to understanding the image. In most accessible documents, these image descriptions are written manually after the document is produced. Here we present a technique for automatically generating image descriptions using latest NLP tools. We have experimented with science school textbooks and results are promising. This is clearly a preliminary work and can be extended to other domains.

References

[1] sshleifer/distilbart-cnn-12-6 summarization tool. https://huggingface.co/sshleifer/distilbart-cnn-12-6, last viewed October 2023.

[2] Hugging face. https://huggingface.co/, last viewed October 2023, 1998–2011.

[3] Chahal, M. Belani, A. Bansal, N. Jadhav, and M. Balakrishnan. Template based approach for augmenting image descriptions. International Conference on Computers Helping People with Special Needs, pages 104–112. Springer, 2018, 2018.

[4] Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. https://arxiv.org/abs/1908.10084, 2019.

Color in STEM Diagrams: Solving an Accessibility Challenge

John Gardner and Dan Gardner
ViewPlus Technologies, Inc, Corvallis, OR USA
John.gardner@viewplus.com, dan.gardner@viewplus.com

Abstract

Diagrams in STEM (Science, Technology, Engineering, Math) and many other disciplines often use color as an important component of the information presented. Tactile versions of such diagrams can represent color as tactile patterns. This paper introduces a new, inexpensive, scalable technology for creating patterns that can automatically replace color blocks when diagrams are embossed with a ViewPlus embossing printer. ViewPlus users now have a number of choices of patterns keyed to color and the ability to customize these patterns or define entirely new ones. Both the promise and the limitations of displaying color as tactile patterns are discussed.

1 Introduction

It has been possible for years, in principle at least, to publish electronic STEM (Science, Technology, Engineering, Math) articles with text and formulas that are fully accessible by anyone (Gardner 2002). People with some types of disability may be unable to read the screen or paper copy, but they can read electronic publications if they have appropriate equipment. Examples of such equipment are screen readers that read text and equations aloud or show them on braille displays.
But this equipment does not provide access to STEM graphics. Currently the most common way to "make graphics accessible" is through a word description. But even excellent descriptions (not the usual case) are inadequate for many critical diagrams. A tactile version provides better access to good braille/tactile graphics readers (Edman 1992), and an audio-tactile version (Sorge 2016, Gardner 2021) can give excellent access to almost all people with visual impairments.

Color can be very important in STEM diagrams; it is commonly used to identify graphical items such as slices in pie charts, data in GIS (Geographic Information System) displays. Representing color in tactile graphics has been a long-time challenge. (Martiinez 2019) Researchers have experimented with representing color by tactile patterns (Shin 2020, Gardner 2023), by add-on tactile shapes (Ramsamy-Iranah 2016, Johnson 2022), and by various mixed dynamic methods (Ina 1996, Kanav Kahol 2006, Shin 2020). The goal of these projects has mostly been understanding human perception, not developing practical methods for representing color.

In this paper we describe practical, scalable methods being developed by ViewPlus for making color accessible in tactile and audio-tactile diagrams. The purpose of this paper is neither to evaluate the fundamental question of how best to make color accessible nor research into optimizing the representation of color by tactile patterns. The paper's purpose is to describe a new user-friendly technology that is, among other important uses, a powerful tool that will help answer these fundamental questions.

ViewPlus provides two ways to emboss colored regions in graphics. The default setting for embossing colored regions with a ViewPlus embosser has always been a 'tactile gray scale' where dark regions are embossed with tall dots and lighter regions with smaller dots. This is made possible because all ViewPlus embossers can create eight dot heights ranging from dot0 (no dot) to dot7 (tallest dot). For several years ViewPlus developers have experimented with both continuous and discrete methods for substituting patterns for colors. Tiger Software Suite (TSS) version 8.0 now allows users to define discrete pattern substitutions for colors. This paper describes this new option. Creating user-accessible methods for continuous substitutions (i.e. functions that define a pattern for any color) is more difficult and is still under development. Presently one continuous option is available to users as an experimental tool.

2 Embossing Options

2.1 The default: tactile gray scale

ViewPlus printer drivers use a standard method for expressing colors in gray scale intensity. Dot heights are assigned so that white and colors perceived visually as very light do not have any dots whereas regions that are black or very dark emboss with the (tallest) dot7. A table (https://viewplus.com/color-tables/TableOfDotHeightVSColor.html) gives a listing of all named HTML colors arranged into groups. The dot height for each color is shown along with the color name and three numerical ways to express colors in html and related applications. Software authors may use the color name, but they are not restricted to the named colors. Colors can be defined by any set of numerical values that are within the physically-allowed range. Note that if a graphic is being created for the sole purpose of making a tactile copy there is no need for color. So, the author may define dot height by expressing fill as gray. The table, Table of Dot Height vs Gray Scale, in the appendix and also at https://viewplus.com/color-tables/TableOfDotHeightVSGrayScale.html is a useful reference.

2.2 The Chessboard Option

In default mode, tactile images with large very dark regions can become distorted and tear easily. This is a particular problem for lightweight paper. For this reason, there is a Tiger Tab option in ViewPlus printer drivers to create a "Chessboard pattern" instead of embossing uniformly. If this option is selected, every second dot in fill regions is skipped and the paper is markedly stronger. Lines with missing even dots alternate with lines having missing odd dots, producing a chess/checker board pattern. It is a recommended option for drawings with large dark regions.

3 Color representation as patterns

The tactile gray scale representation is fine for many purposes, but it gives only information on intensity, not the color itself. Colors can be represented by tactile patterns. If a unique pattern were associated with each color, a blind person anywhere could identify a color by that pattern. There presently is no such color/pattern assignment standard. Creating such a standard would be a major undertaking. As a practical necessity, it requires a convenient technology for creating the large variety of patterns needed to develop and test such a standard. ViewPlus' aim is to create such a technology, to begin research on color-to-patterns, and to stimulate research by others on that topic.

Version 8 of the ViewPlus Tiger Software Suite introduced a new very convenient printer driver option that makes it possible for any user to substitute discrete patterns for color ranges. The new “Apply Patterns" printer driver option allows users to select the one 'automatic' pattern or a pattern file that defines patterns for HSL color ranges. The patterns and color ranges are collected in pattern files, and a number are available to users. Any such file may be modified or entirely new pattern files may be created by users.

4 Pattern Association with HSL Color Values

Color is the frequency of light, and most light perceived by sighted humans has a spectrum of frequencies. The normal human eye has three color receptors, detecting colors centered on frequencies we identify as red, green, and blue. Visual displays create color by mixing controlled intensities of red, green, and blue frequencies. The human brain does not distinguish this three-frequency mixture from a full spectrum. So color, as perceived by humans, is usually considered to have only these three components. The most common way of describing color is by the RGB (Red Green Blue) values. Each is typically represented either as three components in a range 0 to 255 or as three hexadecimal numbers, 0 to FF. HSL is another way of describing those three colors and is in many ways a more useful and intuitive description of color than RGB.

HSL stands for (Hue, Saturation, Luminosity). Hue represents the dominant color and is described by an angle from 0 to 359 degrees. Saturation and luminosity are described either by a value 0 to 255 or by a percentage of that value. Hue angle 0 is Red, 120 degrees is Green, and 240 degrees is Blue. Angles between these values represent mixtures of two of the primary colors, including angle 60 for Yellow (Red+Green), 180 for Cyan (Green+Blue) and 300 degrees for Magenta (Blue+Red).

The saturation value S defines the purity of those colors. S near 100% represents the pure Hue (one or two colors near maximum value with little mixture of the third primary color). So if S = 100%, Hue = 0 degrees would be pure intense red, Hue = 120 degrees would be pure intense Green, And Hue = 60 degrees would be pure intense Yellow, which is maximum Red and maximum Green with no Blue. S near 0 represents colors with substantial amounts of the third color, so a gray value (R, G, and B all equal) would have S = 0. For any case having all three colors exactly equal, H is irrelevant. Luminosity L is the average of the largest and smallest color component values. Note that L is less than 100% for all colors except for pure white - which has RGB values (255,255,255). And L is zero only for pure black, which has RGB = (0,0,0)

5 Printing and Creating Patterns

Users can select from a number of pattern files. All pattern files starting with “_” (underline character) have detailed explanations of the patterns. These explanations are in the pattern file itself at the end of the file. A ‘##’ denotes the break point between patterns and explanations. Users may alter existing patterns or create entirely new patterns if desired. There is a pattern editor/creator application for sighted people in TSS. Anybody, including blind people, can also alter or create patterns using any text editor. Pattern files and formats are still an active research topic, as we always attempt to make the process easier and more intuitive.

The current (Tiger 2.0) format file creation/modification is described in the final section and also on the ViewPlus web site https://viewplus.com. In principle, an indefinite number of patterns can be defined for any Hue range for different S (Saturation) and L (Luminosity) ranges. Tiger 2.0 patterns are defined in 30 °Hue ranges. The following table gives the Hue degree and the common name of the color at or near each 30 ° point. The figures following the table are a color wheel and the image when embossed.

Hue	Name
0	Red
30	Orange
60	Yellow
90	Chartreuse
120	Green
150	Spring Green
180	Cyan
210	Azure
240	Blue
270	Purple
300	Magenta
330	Rose

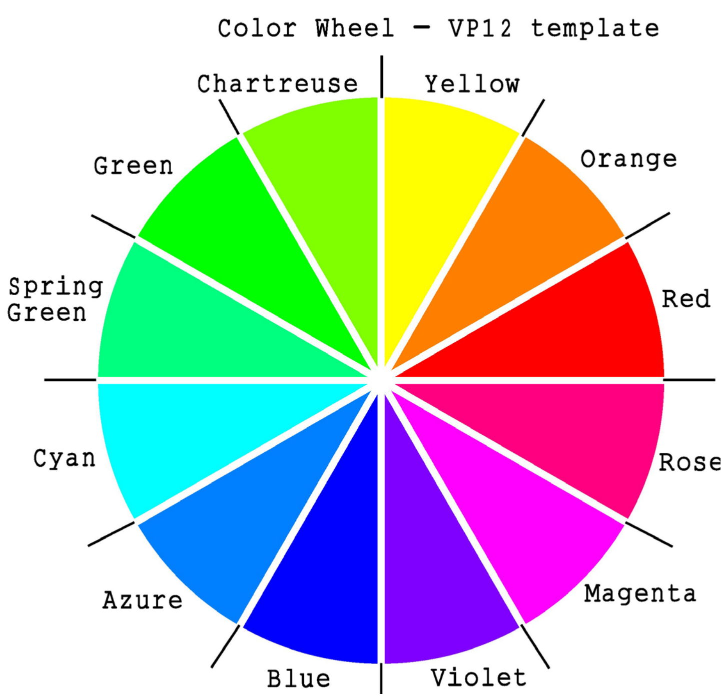


Figure 1: A color wheel with the three primary colors, red, green, and blue along with superpositions of 2 of the primaries, giving orange, yellow, chartreuse, spring-green, cyan, azure, purple, magenta, and rose.

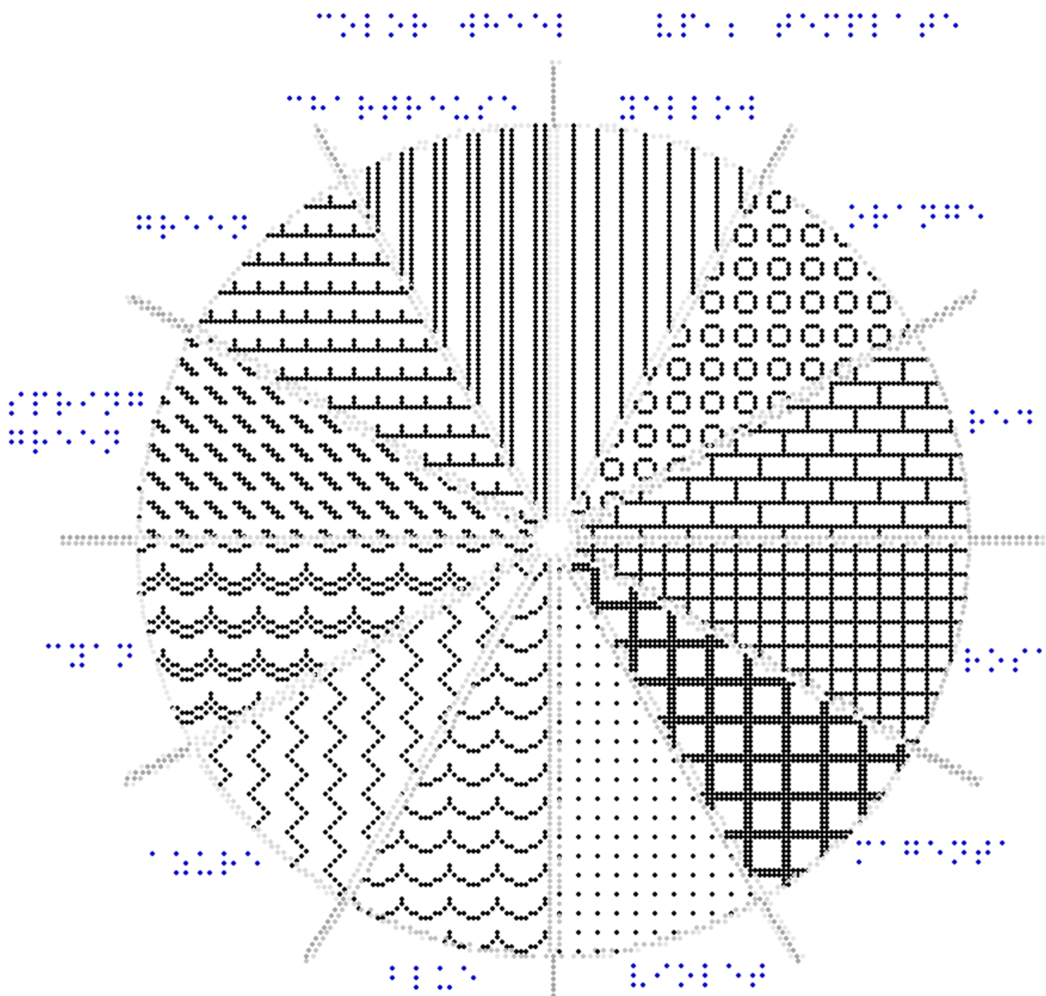


Figure 2: Embossed image of the color wheel shown in figure 1 when the VP12.dat pattern file is used.

6 Relating Patterns to Colors and Developing a Standard

The ViewPlus team has developed patterns with several philosophies. One guiding principle is to make it as easy as possible for a blind person to learn and remember the patterns associated with colors. It is helpful to note that modern usage of color for quantitative purposes is largely confined to seven colors: red, orange, yellow, green, blue, purple, and brown. So, it is important to create easily learned/remembered patterns for these seven cases. And if possible, define patterns for other colors that are related to those of one of the special seven. Hue angles for all but brown are included in the table of Hue points above. One can define many shades of brown, and several brown shades are included in some of the current tables.

One philosophy is to use patterns that have an intuitive relationship to the color it represents and is consequently easy to remember. Of course, what is intuitive to a person who can see or who has at one time had useful vision may not be intuitive to a congenitally blind person. But even most congenitally blind people know that grass is green; the sky and large bodies of water are blue; roses and bricks are red, etc.

Another philosophy is to use very simple easily-remembered patterns for red, blue, green, and then superpose them to make other colors. For example, equal intensities of red plus green is yellow. And red with green at half the red intensity (meaning half the dot height) is orange. Use of patterns for color is not a new concept, but realization of a practical inexpensive method for creating large numbers of patterns is new. Research and experience will teach us how to create good patterns.

STEM of course is not the only field of interest that will benefit from color accessibility, but it certainly is a very important one since color is extensively used quantitatively in virtually all STEM fields.

Color as patterns is also being used by ViewPlus as a way to attract interest and tactile learning among young blind people. Popular cartoon posters using color patterns of such characters as Bart Simpson with skateboard are give-away items by ViewPlus at meetings and trade shows. Many now hang on the bedroom walls of blind youth. Tactile coloring books are also proving to be extremely popular, providing stimulation for young blind children as well as bonding opportunities for blind children and sighted parents/grandparents or sighted children and blind parents/grandparents.

7 Limitations of Representing Color by Patterns

The examples quoted thus far in this paper use color in discrete blocks of constant color. We have not yet made controlled quantitative studies, but it is clear from casual observation that patterns in very small blocks are not easy to detect. The minimum size of such pattern blocks depends on the user's tactile ability, but even for the most capable experts, we estimate that a minimum block size of order half an inch (1.3 cm) is required. Careful research is needed to establish a minimum size for general use, and specifications for general-use materials should include this minimum.

It is also clear from casual observation that color blocks can be more easily detected if the edges are emphasized by lines. Original diagrams often do not have such boundary divisions, so conversion from the electronic file should include an optional edge-detection step that creates such a separation line. Edge detection options are included in the ViewPlus IVEO authoring software and will soon be included in ViewPlus printer drivers.

The focus of ViewPlus developments is embossing diagrams on paper or other embossable media. Depending on which model printer is being used, the minimum dot spacing varies from .050 to .060 inches with a dot placement precision up to 100 dpi. This resolution is comparable to the tactile sensitivity of the human finger. No other current technology can match the sharpness of points or lines embossed by ViewPlus printers. Swell paper does produce smooth lines, but the line widths and point sizes are significantly larger and increases the minimum detectable size of pattern blocks. Recently introduced multi-line braille/ tactile graphics displays (APH/Humanware 2023, Orbit Research 2023) have dots in square arrays and spacing of 0.1 inches (.25 cm) or more. So minimum block size for patterns on these displays will be significantly larger still.

Many STEM diagrams include color that is not present as large single blocks. Images of geologic strata and the color distribution during chemical reactions are among many important such examples. It is possible that with sufficient enlargement, patterns can still give adequate access to color in many such diagrams, but at this time accessibility of such diagrams remains a research question.

References

Edmond 1992: "Tactile Graphics", Edman, Polly, published by American Foundation for the Blind,1992

Gardner 2002: “Access by Blind Students and Professionals to Mainstream Math and Science” John A. Gardner, Proceedings of the 8th International Conference, ICCHP 2002, Linz, Austria, July 15-20, Miessenberger, K., Klaus, J., & Zagler, W., L. (eds) Lecture Notes in Computer Science 2398, 502-507

Gardner 2009: "Making Journals Accessible to the Visually Impaired – The Future is Near". John A. Gardner, Vladimir Bulatov, and Robert A. Kelly. Learned Publishing 22(4) 2009 pp. 314-319.

Gardner 2021: "Reading Graphics by Audio-Touch", John Gardner, The 4th International Workshop on "Digitization and E-Inclusion in Mathematics and Science 2021" (DEIMS2021), Zoom conference, February 18-19, 2021.

Gardner 2023: "Seeing Color as Patterns", John Gardner, 7th Biennial POSB Math & Science Institute for Instruction, held at the Kentucky Science Center, Louisville, Kentucky, February 27-28, 2023.

Humanware 2023: The Humanware Monarch multi-line braille/tactile graphics display was introduced in 2023. https://humanware.com

Ina1996: "Computer graphics for the blind", Ina, Satoshi, ACM SIGCAPH Computers and the Physically Handicapped (SIGACCESS-SIGCAPH), Issue 55•June 1996, pp 16–23

Johnson 2022: "Low-Cost Tactile Coloring Page Fabrication on a Cutting Machine: Assembly and user experiences of cardstock-layered tangible pictures", Johnson, Nicole, Yeh, Tom, and Cunningham, Ann, ASSETS '22: Proceedings of the 24th International ACM SIGACCESS Conference on Computers and Accessibility•October 2022, Article No.: 31, pp 1–5

Kanav Kahol 2006: "Learning and perceiving colors haptically", Kanav Kahol,Jamieson French,Laura Bratton,Sethuraman Panchanathan(Less, Assets '06: Proceedings of the 8th international ACM SIGACCESS conference on Computers and accessibility•October 2006, pp 173–180

Martiinez 2019: "Accessible Statistical Charts for People with Low Vision and Colour Vision Deficiency", Martiinez , Ruben Alcaraz, Turro, Mireia Ribera, and Saltiveri, Toni Granollers, Interacción '19: Proceedings of the XX International Conference on Human Computer Interaction•June 2019, Article No.: 28, pp 1–2

Orbit Research 2023 The Orbit Research Graphiti multi-height large graphics display is available from https://www.orbitresearch.com/

Ramsamy-Iranah 2016: "A comparison of three materials used for tactile symbols to communicate colour to children and young people with visual impairments", Ramsamy-Iranah, Sabrina, Maguire, Martin, Gardner, James, Rosunee, Satyadev, & Kistamah, Narinda, British Journal of Visual Impairment, 34(1), 54-71 (2016).

Shin 2020: "Please Touch Color: Tactile-Color Texture Design for The Visually Impaired", Shin, Jaeeun Shin, Cho, Jundong, and Lee, Sangwon

CHI EA '20: Extended Abstracts of the 2020 CHI Conference on Human Factors in Computing Systems, April 2020, pp 1–7

Sorge 2016: "Generating Audio-Tactile and Web-Accessible SVG Graphics", Volker Sorge and John A Gardner, Proceedings of Accessing Higher Ground, Nov 16, 2016, Westminster, Colorado.

Appendix

Table of Dot-Height vs Gray Scale Values (RGB > Lower Limit, RGB <= Upper Limit

Dot Height	**Lower Limit**	**Upper Limit**
0	(251,251,251)	(255,255,255)
1	(216,216,216)	(251,251,251)
2	(182,182,182)	(216,216,216)
3	(144,144,144)	(182,182,182)
4	(108,108,108)	(144,144,144)
5	(72,72,72)	(108,108,108)
6	(36,36,36)	(72,72,72)
7	(0,0,0)	(36,36,36)

.

Classification of Tactile Maps for Blind and Visually Impaired People

Tetsuya Watanabe[1]
[1] University of Niigata, Niigata, Japan
t2.nabe@eng.niigata-u.ac.jp

Abstract

Over the last six and a half years, we provided more than 128 tactile maps to the blind and visually impaired people. These maps were classified according to the range, contents, and purpose. The most common type were road maps, which were used to know the route and towns. The second most common were "regional maps" with a wider range than road maps. Others were park maps, intersection models with narrower ranges, train and bus route maps, building floor plans, world maps, and topographical models.

1 Introduction

In tactile maps for visually impaired people, roads, landmarks, and Braille letters are raised slightly (0.3-0.6 mm) above the paper surface so that they can touch and understand them. In order to easily create tactile maps, we have developed the tactile map automated creation system that draws road lines based on geographical data (Watanabe et al, 2011; Kurumadani & Watanabe, 2019) and a capsule paper tactile map creation support system that makes it easier to draw lines and symbols for tactile maps (Mori & Watanabe, 2019). Using these systems, we have been providing a service to provide tactile maps that visually impaired people request. The number of maps provided so far exceeds 250 types and 1,000 sheets (including the number of tactile maps distributed at exhibitions for the visually impaired). By classifying these maps, we will get the grasp of the demands of the visually impaired for tactile maps.

2 Classification of Tactile Maps

2.1 Classification

We started the tactile map providing service for visually impaired people in June 2010. Since we have already reported on the tactile maps provided from then until December 2016 (Watanabe & Yamaguchi, 2017), this paper classifies the tactile maps provided from 2017 to June 2023.

The 128 maps for which we kept records of request (some maps were not recorded, so the actual number of the maps created was more than this) were classified into each of the following categories: Road maps, park maps, intersection models, floor plans, and bus/train route maps are mostly for mobility, and regional maps, world maps, and topographic models, for understanding of the regions. The numbers of each maps produced are shown in Table 1.

Table 1: Classification of tactile maps and their numbers produced.

Type of the Map	Number of the Maps Created
Road map	50
Park map	10
Intersection model	4
Floor plan	15
Train/Bus route map	7
Regional map	23
World map	7
Topographic model	11
3D cityscape	1

2.1.1. Road Map

The type of maps produced the most was road maps, with 50, or 39% of the total 128 maps. The main purpose of this map is for mobility or walk, and for this reason, basically all roads within the map are drawn. Most roads are about 1 mm wide, and trunk roads are 1.5 to 2 times as thick (in the case of capsule paper). Tactile point symbols for landmarks are represented by circles, triangles, or squares, about 10 mm wide, with Braille (often abbreviated to two to three letters) near them. An example of a road map is shown in Figure 1. Tactile symbols needs more than 3 mm space around them to tactually separate from other symbols, and even two Braille characters require a space of about 12 mm in height and 15 mm in width. To place these symbols and Braille in a block surrounded by roads, blocks should be at least 15 mm to 30 mm wide, and so as to make them as wide as this the scale and range of a map is determined. The scale is not constant becase the density of the roads differ from place to place. Since the paper size for tactile maps are A4 or B4, the long sides of road maps are about 500 m to 1.2 km.

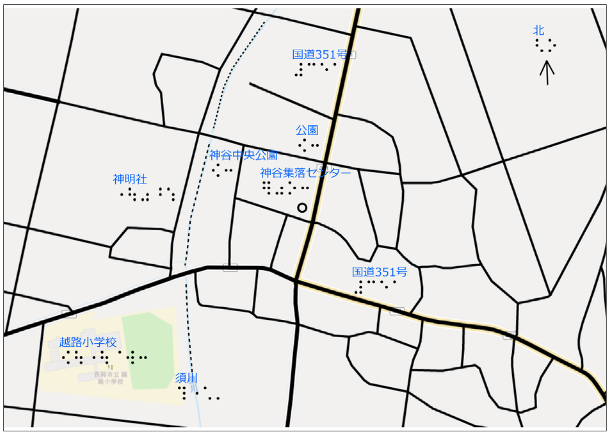


Figure 1: Example of a road map.

2.1.2. Park Map

Tactile maps of theme parks, parks, and universities are tentatively called as park maps (10 maps, 8%). What makes the site maps different from the road maps is the role and representation of the open spaces. While in road maps the space surrounded by the roads is assumed to be filled with buildings, in park maps, open spaces that have various roles such as lawns and hills need to be represented by surface symbols.

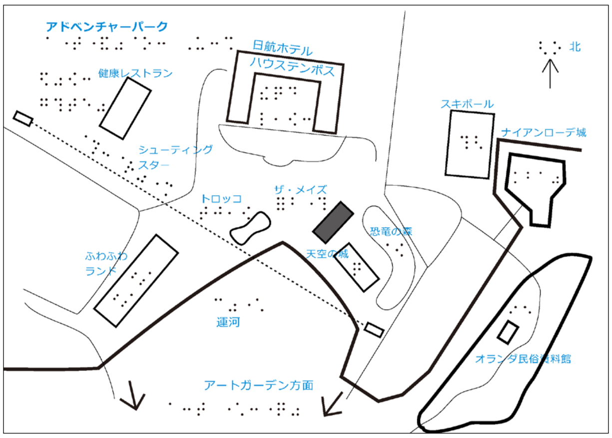


Figure 2: Example of a park map.

2.1.3. Intersection Model

For mobility purposes, four maps, or to say models, of intersections were produced. These models range from 30 m to a hundred or more meters, and are narrower than road maps. At this scale, roads should be represented with width. Sidewalks and roadways need to be separated. Open spaces are not filled with buildings. One way to convey these pieces of information is to draw lines along their boundaries, but this is not easy to understand different meaning of the lines. If the sidewalks are one level higher than the roadway and the buildings are one level higher than the sidewalks, it is easier to understand intuitively because it is kind of a miniature of the intersection. To produce the layer construction, we used and cut cardboards and styrofoam boards and pasted them manually.

Figure 3: Example of an intersection model.

2.1.4. Route Map of Trains and Buses

Seven railroad and bus route maps were produced. In these maps, railways are represented as single lines about 1 mm wide and stations, dots for simple stations and circles for transfer stations or terminals. A feature of these route maps is that the shapes of the routes are significantly deformed, topographically correct though, as you see them at stations and inside trains and buses. Since many station or stop symbols and the lines appear alternately, it may be difficult to follow the lines with a finger.

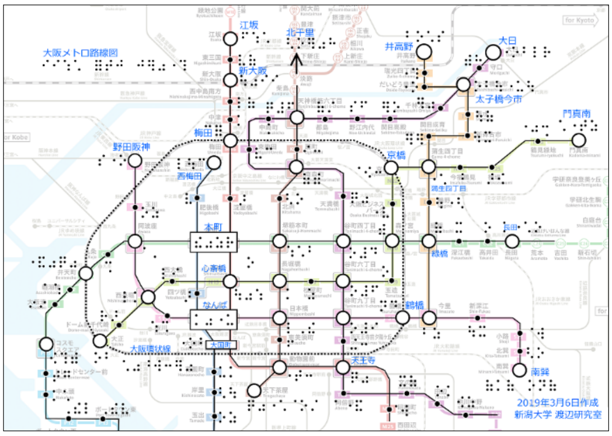


Figure 4: Example of a tube route map.

2.1.5. Floor Plan

Fifteen floor plans were produced (12%). Nine of them are for stations and other six are for shopping malls, municipal offices, and a luxury cruise ship. In station floor plans, the walls of the building and the platforms are drawn with lines, and the facilities such as stairs, escalators, elevators, and ticket gates are drawn with tactile point symbols. Restrooms are represented with its abbreviation, one or two letters, in Braille within the boundary line as their tactile symbols are not determined. In the maps, the tactile symbols are drawn larger than the actual size of the facility.

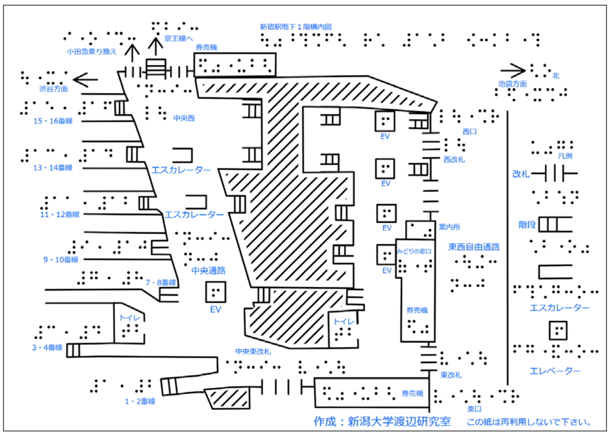


Figure 5: Example of a station floor plan.

2.1.6. Regional Map

The second most common type of the tactile maps produced is tentatively called a "regional map." We have created 23 maps (18%). The areas of these maps ranged from one entire city (around 20 km on the long side) to an entire island (around 20 km on the long side), a lake and its surroundings (around 60 km on the long side), and individual prefectures (around 200 km to 550 km on the long side). Since the purpose of these maps is to know the administrative boundaries of cities and prefectures and the location of towns, cities, and districts in relation to each other, lines of administrative boundaries and point symbols indicating the location of towns/cities/ districts are drawn. In a map of a relatively small area, such as a single city, it helps the map readers understand the location to include railroads and their stations, as well as trunk roads. But in maps of the entire prefecture, railroads and roads tend to be abbreviated as they are so close to each other and coastlines and cannot be distinguished tactually.

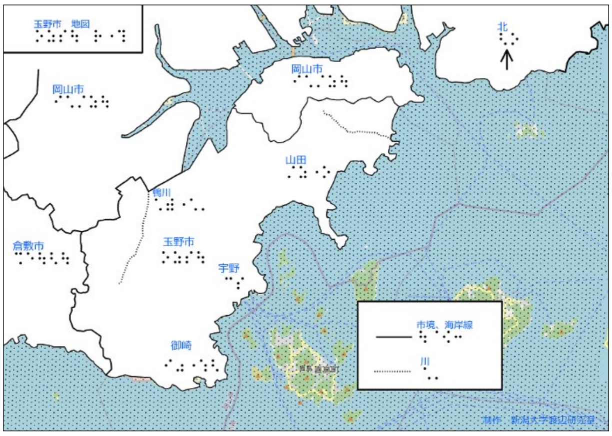


Figure 6: Example of a regional map.

2.1.7. World Map

World maps represents the borders of the countries. Some world maps include several countries and others include only one country at its center and part of the surrounding countries. In addition to borders, the capital and other major cities are represented by point symbols with their abbreviated Braille labels. World maps were created in response to requests from blind people who wanted to know the area of conflict, war, earthquake, or other newsworthy incidents. Although only seven world maps were created up to now, the number of prints sent were exceeding other maps that were made mostly for personal use. For example, the map of Ukraine, where the war has been going on since February 2022, was sent to more than 250 people.

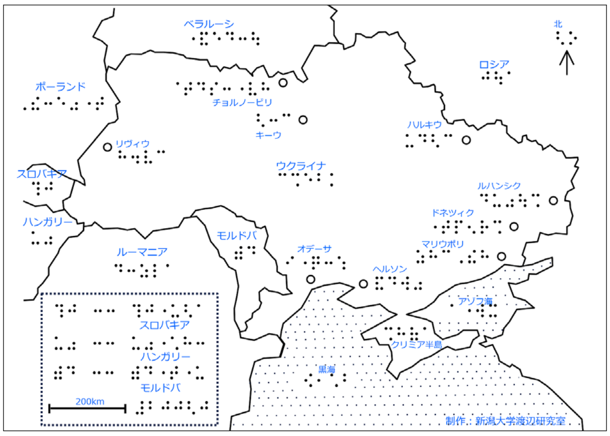


Figure 7: Example of a world map.

2.1.8. Topographic model

In accordance with the increase in natural disasters in recent years, requests for hazard maps are rising since 2019. Among different types of hazard maps, ordinary flood risk maps are drawn on the basis of the elevation of the land. Since the elevation of topographic models can be perceived by touch, these models can be used as hazard maps for visually impaired people. That is the reason why we created topographic models using a 3D printer on the requests for hazard maps. For this purpose, four models were produced so far. Seven more topographic models were produced with a 3D printer because it would be easier to understand the shapes of prefectures and islands than two-dimensional tactile diagrams.

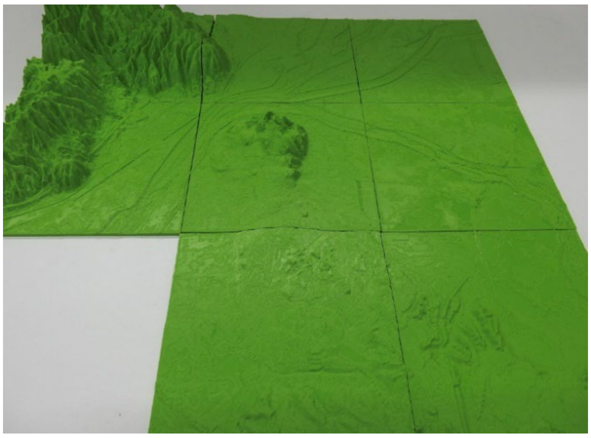


Figure 8: Example of Example of a terrain model.

2.2 Map Types and Range

Table 2 shows the types of maps and the approximate range each type represents. It may be a useful reference when considering what type of tactile map to create in response to requests from visually impaired people. However, this table was created on the basis of only a portion of the 128 tactile maps. So, we will investigate other tactile maps and improve its accuracy.

Table 2: Classification of tactile maps and their approximate ranges.

Type of the Map	Approximate Range
Intersection model	30 m to 100 m
Road map	500 m to 1.2 km
Train/Bus route map	2 km to 400 km
Regional map	20 km to 550 km
World map	100 km to 1000 km

3 Conclusion

We have classified the tactile maps ever produced by purpose and range, and explained the contents of each type. In the future, we will enhance this manuscript, including more diagrams and photos, to create a manual on how to produce tactile maps, and hold workshops on tactile maps using this manual. Increasing tactile map creators will lead to the high quality and sustainable tactile map providing service for visually impaired people.

Acknowledgements

The research and development of tactile maps including topographic models is partially funded by JST RISTEX (JPMJRX21I5) and KAKENHI (Grant-in-Aid for Scientific Research (B) 20H01705).

This is a short version of an English translation of a manuscript that was included in the proceeding of Human Interface Symposium 2023. After translation, major revisions were made.

References

Kurumadani, K., Watanabe, T. (2019). Development of a tactile map automated creation system using a high-definition tactile graphics printer. IEICE Technical Report, 119 (322), 125-130.

Mori, T., Watanabe, T. (2019). Development of a Web application for supporting the creation of tactile maps on swell paper. IEICE Technical Report, 119 (322), 119-124.

Watanabe, T., Yamaguchi, T., Watanabe, K., Akiyama, J., Minatani, K., Miyagi, M., Oouchi, S. (2011). Development and evaluation of TMACS, an automatic tactile map creation system for the visually impaired. IEICE Transaction D, J94-D (10), 1652-1663.

Watanabe, T., Yamaguchi, T. (2017). Six-and-a-half-year practice of tactile map creation service. Harnessing the Power of Technology to Improve Lives (P. Cudd and L. de Witte (Eds.)), IOS Press, 687-694.

Accessibility Features of Developmental Platforms: Towards Developing Accessible Mobile Applications with Cross-Platform, Research Challenges and Opportunities

Muhammad Shoaib[1*] Donal Fitzpatrick[2] and Ian Pitt[1]
[1]School of Computer Science and Information Technology, University College Cork, Ireland.
[2]Centre for Excellence in Universal Design, National Disability Authority, Ireland
*muhammad.shoaib@cs.ucc.ie

Abstract

Mobile application development is a process of designing and developing smartphone-based applications for several platforms i.e., Android, iOS, etc. Mobile application development platforms can be categorised as native and cross-platform, each offering resources, frameworks, and tools to developers so they can develop smartphone-based applications using them. Native platforms concentrate on specific applications designed expressly for a particular platform i.e., Android or iOS. Cross-platform allows developers to develop applications with a single codebase that can work across multiple platforms. Examples include Xamarin and Unity. In this study, we have compared the accessibility features offered by Xamarin and Unity with those offered by native platforms. We analyzed the accessibility features offered by the iOS and Android platforms to determined whether and to what extent Xamarin and Unity offer the same accessibility capabilities. This analysis shows that numerous functionalities are shared by native iOS and Android APIs, however, some of them are not included in Xamarin and Unity. They also do not provide the implementation of fundamental APIs. This will require additional work by developers to write platform-specific code in native APIs to access the unavailable APIs. We have provided a comparative analysis of these platforms. Important accessibility aspects are highlighted that may prove useful for researchers and developers who are working to create accessible apps.

1 Introduction and Background

Android's popularity has climbed sharply since Google released version 1.0 of the operating system on September 23, 2008. With the release of Android 13 in August 2022, Google claimed to have around 3.6 billion active users worldwide [1]. According to Statista's 2023 report, the Google Play Store, the official digital retailer for Android, holds a 71.47% global market share for mobile operating systems. iOS is the other operating system that competes with Android for users. Since its June 2007 release, iOS has increased its market share to 27.6% [2]. This is why developers are keen to create apps that can work on both iOS and Android [3]. Choosing the platforms on which a mobile app will operate is the first step in the development process. Next, using platform-specific toolkits like the Android SDK, developers must create tailored solutions (i.e., native apps) for each selected platform. An app that functions across two or more mobile platforms is known as a cross-platform mobile application. Xamarin and Unity are important cross-platform toolkits that make mobile applications easier and lower development and maintenance expenses[3].

* Masterminded EasyChair and created the first stable version of this document

Xamarin was first introduced in 2011 as a cross-platform app development framework, Microsoft later purchased it, giving it more legitimacy than before. It provides access to native APIs as well as the potential for code reuse and sharing with other platforms. It is one of the most widely used and favored mobile development platforms. This cross-platform uses the C# programming language to produce mobile applications that perform remarkably well across all mobile devices [4]. Unity is a cross-platform game engine that was introduced in June 2005. It provides an interface that offers access to all development tools in a single location and is called an Integrated Development Environment (IDE). Developers can drag and drop objects into interfaces with the help of a visual editor and can change their properties. Unity uses an editor of your choice when it comes to coding. Microsoft's Visual Studio, which integrates most of the time, is the most common choice. Unity uses a language known as C#. All of the scripting languages that Unity uses are object-oriented [5].

Mobile applications are increasingly becoming an integral part of our daily life and the accessibility factor of native and cross-platform apps is essential for disabled people. It means making these apps useful for as many individuals as possible. Applying fairness to all people by granting them the same opportunities regardless of their abilities by adding additional features like scalable texts and screen reader support. Previous studies showed that developers should design the new mobile application by considering the accessibility factor [6-9]. Based on WHO more than 1.3 billion people worldwide have a disability which means 16% of the world's population, or one in six people[10].

This article presents a detailed analysis of mobile application development platforms and discusses their strengths and weaknesses with regard to accessibility. We have demonstrated the functionality of these platforms by exploring their accessibility features, plugins and extensions. We have listed accessibility APIs, tools, and packages and compared their features. We examined how cross-platform mobile applications wrap native APIs. We demonstrate that two of the most popular mobile cross-platform toolkits, Xamarin and Unity, cover around half and one-third of the screen reader-related iOS and Android APIs, respectively. Finally, we present our findings and talk about how we were able to use them to develop accessible mobile applications.

2 Most Used Tools for Mobile Applications Development

The quote "the right tools for the right job" has been around forever, and it holds for software development just as much as it does for any other field. These days there are several mobile app development platforms available. Selecting the appropriate mobile development platform is essential for designing a proper solution to make it scalable, accessible and maintainable. The cost and time of creating a mobile app may be reduced using the right development platform [11]. This section analyzes all four developmental platforms (Android, iOS, Xamarin and Unity) and briefly describes them.

2.1 Comparative Analysis of Native and Cross-platform Toolkits

Mobile application development through native platforms is very common. Every native platform has its programming language, like Java and Kotlin, the official programming language for the Android platform. On the other hand, having a good grasp of languages like Swift and Objective-C is essential for creating iOS applications. Cross-platform mobile application development can run on multiple platforms and we can also achieve the performance of Native platforms by using them [12]. Here, we compared Native (Android and iOS) and Cross-platforms (Xamarin and Unity). Table 1 provides detailed information about the strengths and Weaknesses of these platforms.

Platforms Name	Strengths	Weaknesses
Android Studio	Provide a user-friendly development environment Have excellent support for Android devices Supports various programming languages, including Java, Kotlin, and C++ Provide support to drag-and-drop tools. Previewing the app on different device sizes Better debugging and testing support Support profiling and analyzing tools Support for automated testing frameworks	Resource-intensive Requiring high system requirements Requires prior programming knowledge The size of the app built on Android Studio can be larger compared to other platforms. Development and deployment of apps on Android Studio can be time-consuming Requiring considerable testing and optimization efforts Compatibility issues with some older devices
Xamarin	Write code in C# then compile it to the native language of any platform Provide access to native APIs and libraries for each platform Code Sharing between iOS, Android, and Windows apps Have pre-built UI Controls that can create native user interfaces quickly and easily. Integration with Visual Studio	Steep Learning Curve (requires knowledge of C# and native development environment) Limited Access to Latest Features because it relies on its own version Increased App Size because Xamarin requires additional native APIs Limited Community Support as compared with Android and IOS
Xcode	Integration with Apple Platforms Has robust support for Swift Programming Language Interface Builder allows developers to design interfaces by drag-and-drop Comprehensive Toolset which includes a source code editor, debugging tools, performance analyzers, asset management, version control integration, and testing frameworks Simulator and Emulator App Store Deployment Extensive Documentation and Community Support	Limited Platform Support not for Android or Windows Resource Intensive for large projects or using features like Interface Builder Steep Learning Curve(complex and extensive set of features for new developers) Interface Builder has Limitations in terms of flexibility and customization (manually written code) Code Completion and Indexing Issues in large projects Xcode IDE may occasionally experience stability issues or crashes.

Unity	Cross-Platform Development Visual Editor and Workflow offers drag-and-drop and+D68 visual scripting Asset Store and Community Strong 2D and 3D Capabilities Scripting and customization provide access to Unity's extensive API Provide support to performance and optimization tool Unity Collaborate allows teams to work together.	Performance Challenges because general-purpose design may result in overhead Steep Learning Curve because understanding the various components may take time Scripting Limitations because specific programming paradigms may be more challenging to implement Documentation and Support (some parts of the documentation may be outdated, incomplete, or lack detailed explanations) Asset Store Quality Control because some assets may lack documentation or have compatibility issues Large Build Sizes Software Updates(updating Unity to a new version can sometimes introduce issues)

Table 1: Provide information about the strengths and weaknesses of the development platforms.

In Table 2, we have mentioned the names of the platforms, types (open-source, not open source or offers free and paid licensing), names of the plugins and extensions, best-suited information (i.e., Native: Android and iOS, Cross-platform: iOS, Android, and Windows) and finally listed the accessibility features.

Platforms Name	**Type**	**Plugins and extensions**	**Best suited for**	**Accessibility features**
Android Studio	Open-source	Firebase Android ButterKnife Zelezny Android Material Design Icon Generator Genymotion Android WiFi ADB CodeGlance Android Lint	Native Android apps	Screen reader compatibility Text-to-speech and speech-to-text conversion Closed captioning High contrast mode Adjustable font size Customizable color schemes Voice recognition
Xamarin	Open-source	Xamarin.Forms Xamarin.Essentials Prism Syncfusion HockeyApp	Cross-platform development iOS, Android, and Windows. Native mobile apps with a shared codebase.	Support for Accessibility APIs Customizable User Interfaces Focus Navigation Accessibility Testing Accessibility Guidelines
Xcode	Not open source	Alcatraz Injection for Xcode SwiftLint ClangFormat XcodeGen KSImageNamed	Apple's platforms, including iOS, macOS, watchOS, and tvOS	VoiceOver Support Keyboard Navigation Voice Control Dynamic Type Support Color Vision Accessibility Accessibility Inspector

		Dash for Xcode GitHub for Xcode Reveal		Documentation Accessibility
Unity	Not open source offers both free and paid licensing.	Rider MonoDevelop-Unity ProBuilder Playmaker Odin Inspector VFX Graph	Cross-platform game development and interactive applications	Text-to-Speech (TTS) Support Screen Reader Compatibility Subtitle Support High Contrast Mode Keyboard and Gamepad Remapping UI Scaling and Resolution Independence

Table 2: Provides information about the type, plugins, extensions, best suitable for and accessibility features of the development platforms.

Table 3 summarizes information about the accessibility API, tools packages and their features (i.e., 1=Event Monitoring, 2=UI Navigation, 3=Content Descriptions, 4= Gesture Detection, 5= Text-to-Speech/Voice Over Support, 6= Event Filtering, 7= Accessibility Settings, 8= Hints and Lables, 9= Screen Reader and 10= Subtitles and Closed Captions).

Name	**Accessibility APIs, tools and Packages**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
Xamarin	AccessibilityDelegate AccessibilityService TalkBack Support Xamarin.Forms Accessibility API Xamarin.Android Accessibility Xamarin.Essentials Xamarin.iOS Accessibility APIs Custom Accessibility Implementations	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
Android Studio	AccessibilityService AccessibilityNodeInfo AccessibilityEvent AccessibilityManager ExploreByTouchHelper AccessibilityDelegate AccessibilityAction AccessibilityNodeProvider	Y	Y	Y	Y	Y	Y	Y	Y	Y	N
Xcode	UIAccessibility UIAccessibilityElement UIAccessibilityContainer UIAccessibilityCustomAction UIAccessibilityFocus UIAccessibilityIdentification	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

	UIAccessibilityTraits UIAccessibilityZoom NSAccessibility										
Unity	Unity's Accessibility Input System Text-to-Speech UIElements EventSystem Unity's UGUI Accessibility Inspector Unity Accessibility Package	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

Table 3: Provides information about the accessibility API, tools packages and their features.

3 NATIVE and Cross-platform screen reader APIs

The functionality and capabilities of the screen reader software on a particular platform are directly accessible through native screen reader APIs. These APIs allow programmers to interact with the screen reader, get data from the screen, and create accessible information for their apps. We thoroughly reviewed the screen reader APIs, taking into account our past knowledge as well as the iOS and Android programming development manuals and documentation. Based on the functionality exposed, we developed an API taxonomy. We have listed some API functionalities and their compatibility with the four platforms under consideration (native iOS, native Android, Unity, and Xamarin). APIs that support fundamental screen reader features are also shown in Table 4. We used the accessibility guidelines and best practices listed in Apple and Google's introductory accessibility documentation [13-16] to identify fundamental features.

3.1 Android Studio Accessibility APIs

AccessibilityService and AccessibilityNodeInfo are important accessibility APIs of Android Studio. AccessibilityService API allows programmers to develop services that interact directly with the screen reader to process the accessibility events. On the other hand, AccessibilityNodeInfo will enable developers to adjust accessibility information for their Android applications by providing information about the characteristics and state of UI elements.

3.2 iOS Accessibility APIs

UIAccessibility, UIAccessibilityElement and NSAccessibility are important accessibility APIs of XCode. UIAccessibility offers a collection of classes and protocols for improving iOS application accessibility, including VoiceOver support. These APIs allow developers to give UI elements accessibility labels, traits, hints, and other data. UIAccessibilityElement enables the creation of custom accessibility components and sharing this information through VoiceOver in iOS applications. NSAccessibility provides a complete set of classes and protocols for adding accessibility features to macOS apps. These APIs allow developers to handle accessibility events, handle the change in accessibility events and provide accessibility information on these events.

3.3 Xamarin Accessibility APIs

AccessibilityService, UIAccessibility and Xamarin.Forms are essential accessibility APIs of Xamarin. AccessibilityService enables programmers to build services that can handle accessibility events. With the help of this API, you may communicate with the screen reader for Android and can customize the UI elements. UIAccessibility offers a collection of classes and protocols for improving the accessibility of iOS applications. Developers can make UI elements more accessible with the VoiceOver screen reader by using these APIs to offer accessibility labels, traits, hints, and other information. Xamarin.Forms ensure accessibility support using the native accessibility APIs on each platform. Developers can use Xamarin to provide accessibility information, manage accessibility events, and create Xamarin forms with accessibility features and characteristics programs that use screen reader-friendly forms.

3.4 Unity Accessibility APIs

Unity offers no particular screen reader API. On the other hand, Unity provides some features that allow developers to integrate accessibility features into their Unity apps. The accessibility features can be implemented by making them compatible with screen readers on several platforms. Here are some strategies for integrating accessibility for screen readers in Unity: Text-to-Speech (TTS) Integration, Custom Accessibility Implementation and User Interface (UI) Navigation and Interaction. Table 4 indicates which APIs implement basic screen reader functionalities.

We considered the APIs mentioned above and organized them into the following five categories:

Accessibility focus: The ability for the user to select a view, hear a description of it, and then optionally activate it, is a fundamental way of mobile screen reader interaction. Once a view is selected, we can say this is currently in accessibility focus; only one view may be given the focus at once.

Text-to-speech: The screen reader speaks aloud the accompanying text description when a view is under the accessibility focus. The developers can define this description, which we refer to as text-to-speech.

Explicit TTS: Text-to-speech (TTS) software reads the text and speaks aloud. The programmers can also utilize TTS to read aloud content automatically.

Accessibility Hierarchy: The logical hierarchy structure defines the screen elements to enhance the accessibility of the screen contents. The developer has access to change this structure.

Other: Any other accessible features

ID	Category	API Functionality	XCode	Android	Xamarin	Unity
1	Accessibility Focus	Indicate which view received the accessibility focus	Y	Y	Y	Y
2		State the accessibility focus order.	Y	Y	Y	Y
3		Indicate action connected to accessibility focus-related events (Like toggle view focus)	Y	Y	N	N
4		Determine which view has the accessibility focus	Y	Y	N	Y
5	Text-to-speech	Indicate the position where TTS perform	Y	Y	L	L
6		Mention TTS with the help of a program	N	Y	N	Y
7		One view defines another view.	N	Y	N	N
8		Notice the change even without	N	Y	N	Y

		user interaction				
9	Explicit TTS	Inform when the screen reader finishes reading	Y	N	N	Y
10		Can change TTS features i.e., speech or pitch	Y	Y	N	N
11		Read a text with a non-screen reader TTS	Y	Y	Y	Y
12		Identify when reading performed by a non-screen reader	Y	Y	N	N
13		non-screen reader can change TTS features i.e., speech or pitch	Y	Y	Y	Y
14	Accessibility Hierarchy	Combine various views into a single element that is easily accessible.	N	Y	Y	Y
15		Divide a view into various accessibility components.	Y	Y	N	Y
16		Can access the parent element from the child	Y	Y	N	Y
17	Other	Detection of an active screen reader	Y	Y	N	N
18		Provide implementation on how to respond to a screen reader action	Y	N	Y	Y
19		Take action on the user's behalf.	N	Y	N	N
20		Get arbitrary elements when accessibility-related data you want to view.	N	Y	N	N

Table 4 indicates which APIs implement basic screen reader functionalities.

4 Discussion and Conclusion

The decision to develop Android, iOS, Xamarin, or Unity apps depends on several factors, including the project's requirements, the intended audience, and the developer's experience. Regarding accessibility features and screen reader APIs, every platform has its own advantages and disadvantages. Android can be fragmented but offers diversity and open-source freedom. While limiting customization, iOS promotes consistency and accessibility. Xamarin and Unity offer cross-platform development solutions, with Xamarin giving C# compatibility and Unity being open-source. Both support accessibility and have access to native functionality.

These platforms offer unique approaches to app development, which we have characterized by their strengths, weaknesses, and accessibility features. Android, an open-source platform, boasts many hardware options and incorporates inclusive features such as TalkBack and magnification gestures. Nevertheless, it contends with device fragmentation and potential delays in updates, which may affect the uniformity of the user experience. On the other hand, iOS is renowned for its accessibility-centric approach, delivering a tightly integrated ecosystem complete with VoiceOver and consistent updates,

ensuring high accessibility support. However, its closed nature may limit customization options, with the result that apps developed using these tools may present a steep learning curve for users.
Xamarin and Unity, serving as cross-platform development frameworks, provide advantages like code reusability and access to native features, enhancing efficiency. Xamarin, with its compatibility with C# and platform-specific accessibility APIs, empowers developers to create accessible apps. Developers require additional efforts to get familiar with the different platform-specific features to implement the Native APIs. On the other hand, Unity, being open-source, can harness native accessibility features but may encounter delays in adopting new platform features and require developers to acquaint themselves with its framework. Irrespective of the chosen platform or framework, placing accessibility at the forefront of app development is imperative to ensure that all users, including those with disabilities, can fully utilize and benefit from the created applications. Developers should conscientiously evaluate these factors in light of their project prerequisites and available resources when selecting.
Furthermore, native APIs provide a robust and deeply integrated accessibility solution for Android and iOS. Cross-platform solutions like Xamarin and Unity offer the benefit of code reusability and the potential to tap into native features. However, developers must be mindful of potential delays in adopting new platform features and the learning curve associated with cross-platform frameworks. In both platforms, the paramount consideration is accessibility in app development. This ensures that all users, including those with disabilities, can access and utilize apps effectively. Developers should select the best approach with their project requirements, user needs and available resources while keeping accessibility at the forefront during the development of the apps.

Acknowledgment

This publication has emanated from research conducted with the financial support of Science Foundation Ireland under Grant number 18/CRT/6222. For the purpose of Open Access, the author has applied a CC BY public copyright licence to any Author Accepted Manuscript version arising from this submission.

References

1. The Rise of Android: Why is Android Successful? (Source: https://www.bankmycell.com/blog/how-many-android-users-are-there) https://www.bankmycell.com/blog/how-many-android-users-are-there#:~:text=Android%20Users%3A%20Today%2C%20there%20are,mobile%20operating%20system%20market%20share.
2. Number of apps available in leading app stores as of 3rd quarter 2022 https://www.statista.com/statistics/276623/number-of-apps-available-in-leading-app-stores/
3. Lecomte, S., & Martinez, M. (2017). Towards the Quality Improvement of Cross-Platform Mobile Applications. In IEEE/ACM 4th International Conference on Mobile Software Engineering and Systems (MOBILESoft) (pp. pp. 184-188).
4. Toasa, R. M., Egas, P. F. B., Recalde, H., & Saltos, M. G. (2023, February). Mobile Development with Xamarin: Brief Literature, Visualizations and Important Issues. In International Conference on Information Technology & Systems (pp. 299-307). Cham: Springer International Publishing.
5. Bin Uzayr, S. (2022). Mastering Unity: A Beginner's Guide. CRC Press.
6. Balaji, V., & Kuppusamy, K. S. (2016, December). Accessibility analysis of e-governance oriented mobile applications. In 2016 International Conference on Accessibility to Digital World (ICADW) (pp. 141-144). IEEE.

7. Acosta-Vargas, P., Salvador-Acosta, B., Salvador-Ullauri, L., Villegas-Ch, W., & Gonzalez, M. (2021). Accessibility in native mobile applications for users with disabilities: A scoping review. Applied Sciences, 11(12), 5707.
8. Peter Korn, Evangelos Bekiaris, and Maria Gemou. 2009. Towards open access accessibility everywhere: The ÆGIS concept. In International Conference on Universal Access in Human-Computer Interaction. Springer, 535–543
9. Christoph Rieger, Daniel Lucrédio, Renata Pontin M Fortes, Herbert Kuchen, Felipe Dias, and Lianna Duarte. 2020. A model-driven approach to cross-platform development of accessible business apps. In Proceedings of the 35th Annual ACM Symposium on Applied Computing. 984–993
10. Disability https://www.who.int/en/news-room/fact-sheets/detail/disability-and-health
11. Charland, A., & Leroux, B. (2011). Mobile application development: web vs. native. Communications of the ACM, 54(5), 49-53.
12. Xanthopoulos, S., & Xinogalos, S. (2013). A comparative analysis of cross-platform development approaches for mobile applications. BCI '13, (pp. 213-220).
13. Apple. 2023. Apple Human Interface Guidelines. Accessibility section. https://developer.apple.com/design/human-interface-guidelines/accessibility
14. Google. 2023. Android Developers Documentation. Principles for improving app accessibility. https://developer.android.com/guide/topics/ui/accessibility/principles
15. Niccolò Cantù, Mattia Ducci, Dragan Ahmetovic, Cristian Bernareggi, and Sergio Mascetti. 2018. MathMelodies 2: a Mobile Assistive Application for People with Visual Impairments Developed with React Native. In ACM SIGACCESS Conference on Computers and Accessibility (ASSETS). ACM.
16. Andres Gonzalez and Loretta Guarino Reid. 2005. Platform-independent accessibility api: Accessible document object model. In Proceedings of the 2005 International Cross-Disciplinary Workshop on Web Accessibility (W4A). 63–71.

5th International Workshop on Digitization and E-Inclusion

in Mathematics and Science 2024

自主創造

日本大学

NIHON UNIVERSITY

Nihon University

Tokyo ,Japan

<u>Journal</u>

Title: Exclusion to Inclusion : A comparative study of Accessibility of Banks and Financial Institutions (BFi) of Bangladesh.

Authors:

1. Vashkar Bhattacharjee (Corresponding Author)

2. Mohammad Asif Sadique

Date of Submission:

16th October 2023

Title: Exclusion to Inclusion : A comparative study of Accessibility of Banks and Financial Institutions (BFi) of Bangladesh.

Abstract:

The development of a country's banking and financial institutions (BFIs) is indicative of its overall progress. Digital literacy, closely linked to the level of digital technology adoption by the populace, plays a pivotal role in a nation's development. In Bangladesh, significant strides have been made in the realm of digital technology, partly due to advancements in internet infrastructure. However, a key challenge lies in addressing the digital divide, particularly concerning individuals with disabilities. As per the "World Report on Disability" published by the World Health Organization (WHO), approximately 15% of the global population is living with some form of disability. In Bangladesh, it is estimated that 4.7 million population faces various disabilities. This research study undertakes a critical examination of two primary factors contributing to the digital divide:

(a) The accessibility of banking websites.

(b) The accessibility of financial and banking information available on these websites.

This research focuses on assessing the digital accessibility of websites belonging to 61 BFIs in Bangladesh., this study aimed to evaluate the websites from various angles, including their effectiveness as communication tools, information dissemination, service provision, accessibility, design, and the incorporation of interactive features catering to individuals with disabilities.

Keywords:

Bangladesh, digital divide, digital accessibility, banking websites, financial information, persons with disabilities, UNCRPD, World Health Organization, digital literacy.

Introduction:

In Bangladesh, the transformation of Banking and Financial Institutions (BFIs) into digital powerhouses has been nothing short of remarkable. Yet, in the age of rapid internet technology, the digital divide threatens to exclude underdeveloped and developing countries, with profound implications for individuals with disabilities. Globally, a staggering 3 billion people lack access to formal financial services. BFIs' websites have become the first impression for customers worldwide, as Gerry McGovern noted (December 5, 2013). These institutions are essential in Bangladesh, empowering businesses, and financial growth, including a significant portion of the population—persons with disabilities. For persons with disabilities, these websites are gateways to financial transactions, services, and critical information. Digital accessibility is the touchstone of global outreach, underscoring BFIs' commitment to inclusivity within Bangladesh's regulatory framework, overseen by the Bangladesh Bank.

.**The Bangladesh Bank Regulations on Accessibility**

Bangladesh Bank, as the central regulatory authority, has implemented guidelines and standards to ensure the accessibility of digital platforms and websites within the financial sector. These regulations emphasize user-friendly, barrier-free designs, promoting equitable access for individuals, including those with disabilities. Bangladesh Bank's efforts underscore the commitment to fostering financial inclusion and ensuring that everyone can participate fully in the nation's financial services.

The National Policy on Universal Electronic Accessibility in Bangladesh

The National Policy on Universal Electronic Accessibility in Bangladesh is designed to ensure equal access to electronic technology and information and communication technology (ICT) services for persons with disabilities (PWDs). Inspired by the United Nations Convention on the Rights of Persons with Disabilities (UNCRPD), this policy, approved by the Bangladeshi government, aims to eliminate discrimination, and promote inclusivity. Its primary objective is to provide PWDs, including those with physical and cognitive challenges, with unobstructed access to electronic devices and ICT services, with a focus on enhancing local language support. The policy envisions a future where electronic and ICT products and services are universally accessible, creating a barrier-free environment and reducing the need for special adaptations. This policy is expected to benefit persons with disabilities throughout the country, aligning with Bangladesh's commitment to upholding the rights and well-being of all including persons with disabilities.

The following strategies are envisaged for the implementation of the policy:

1. Creating awareness on design and accessibility of universal electronics.

2. Capacity building and infrastructure development.

3. Setting up of model electronics and ICT centers for providing training and demonstration to special educators of/and physically as well as mentally challenged persons.

4. Conducting research and development, use of innovation, ideas, technology, and so on, whether indigenous or outsourced from abroad.

5. Developing programs and schemes with greater emphasis for persons with disabilities.

6. Developing procurement guidelines for electronics and ICTs for accessibility and assistive needs.

Accessibility Rights of Persons with Disabilities

In Bangladesh, there are millions of individuals with disabilities, and a significant portion of them, roughly 87%, require banking and financial services. It is imperative that their accessibility rights are upheld to ensure financial inclusion, with the lack of accessible banking websites being a potential hindrance.

Table 1. Person with Disabilities by Sex and Residence.

Residence	Persons	Males	Females
Total	2.80 %	3.28%	2.32%
Rural	2.89%	3.10%	2.41%
Urban	2.45%	3.16%	2.43%

The UNCRPD, particularly Article 9.2 (g) and (h), emphasizes the facilitation of access to new information and communication technologies, including the Internet, for persons with disabilities (PWDs). This aligns with the global push for accessible mobile phones and services, as outlined by the International Telecommunication Union (ITU) and the global initiative for inclusive ICTs.

Despite the absence of specific provisions in Bangladesh's ICT Act, 2006, efforts have been made by advocacy groups to enhance ICT accessibility. This led to the development of national policies and guidelines for web accessibility, similar to Bangladesh's "guidelines for Bangladesh Government websites," aiming to ensure that websites are accessible to all citizens, including PWDs. Given the increasing Internet penetration in Bangladesh, banking, and financial institutions (BFIs) should recognize the significance of this large consumer base and the vital role that websites play in facilitating communication, especially for PWDs, who often rely on online platforms for cost-effective interaction.

Objectives

The main objective of the short research study was to assess and demonstrate the qualities of websites designed by BFIs in Bangladesh, in terms of information dissemination, accessibility, design, and interactive and participatory features. The study also aimed for measuring these features in all the websites selected for the study, using online validation methods already available on the Internet.

Research Questions

Does the BFI website have information dissemination features?

1. Does the BFI website follow the guidelines of UNCRPD with reference to accessibility, access to information?

2. Are the BFIs websites designed to meet the requirements of students with disabilities, such as availability of screen reader, font size increase, color change, alternate texts, and Sign Language?

3. Do the websites of BFIs in Bangladesh conform to the accessible design features of WCAG 2.1 ?

4. Are the websites of BFIs more interactive or participatory in nature for Persons with disabilities.

Method

This study focused on assessing the website accessibility of Bangladesh's banking and financial institutions (BFIs) in terms of design, formal and accessible features. The selection of websites for analysis was based on the list of 61 BFIs provided by the Bangladesh Bank, the country's central bank, as of the year 2023. The evaluation of website accessibility features took place on a specific date, May 18, 2023, designated as "Global Accessibility Awareness Day" in Bangladesh, signifying its significance for persons with disabilities (PWDs).

A total of 61 websites belonging to various BFIs in Bangladesh were meticulously examined. These BFIs were selected based on their accreditation by the Bangladesh Bank. Screenshots of the websites were also included as part of the sample for analysis. In this study, 61 websites of BFIs in Bangladesh were assessed for compliance with web accessibility standards, following parameters related to interactivity, accessibility, and information inputs.

The content analysis of website accessibility was carried out using online tools for web-based accessibility validation. The Universal Resource Locator (URL) of each BFI website in Bangladesh was entered into the address slot and tested for validation online. The results of these tests were recorded as the dataset for analysis. Online accessibility validation tools provided by the World Wide Web Consortium (W3C) were employed in this study, encompassing validator tools for "markup validation," "cascading style sheets validator," "mobile validator," "link check validator," and "unicorn validator.".

Sampling

The current research follows a survey methodology employing purposive sampling. The list of all Banking and Financial Institutions (BFIs) in Bangladesh was derived from the Bangladesh Bank's annual report, which provides a comprehensive profile of the country's banks and financial institutions as of the year 2023. Consequently, only the websites of BFIs operating in Bangladesh, as per the report, were selected as the sample for this study.

The sampling process began by visiting each selected BFI's website to explore various aspects related to the website's design, formal structure, functionality, and accessibility features. Subsequently, these websites were categorized into three distinct groups: (a) features belonging to private sector banks (PSBs), (b) those associated with public sector banks, and (c) features specific to foreign banks (FBs). This categorization ensured that each accessibility feature collected from the websites was assigned to one of these three groups, facilitating a comparative study of website accessibility across different types of BFIs in Bangladesh.

Sampling Limitations

While the purposive sampling method was employed for this research, it is important to acknowledge certain limitations associated with the approach.

Firstly, the study's reliance on the Bangladesh Bank's annual report from the year 2021-22 as the primary source for creating the sample may introduce selection bias. This report might not encompass newly established or smaller financial institutions that have emerged after the report's publication.

Secondly, the sampling process involved visiting the selected BFIs' websites to assess their accessibility features. The study's limitations include the potential exclusion of BFIs with inaccessible or poorly designed websites. This could lead to an underrepresentation of accessibility issues that exist in the banking and financial sector.

Lastly, since the research categorizes accessibility features into three groups based on the type of banks (private sector, public sector, and foreign banks), the study may not capture the nuances of accessibility challenges specific to each BFI. This categorization might oversimplify the complexities of web accessibility within individual institutions.

These limitations should be considered when interpreting the findings and their applicability to the broader landscape of banking and financial institutions in Bangladesh.

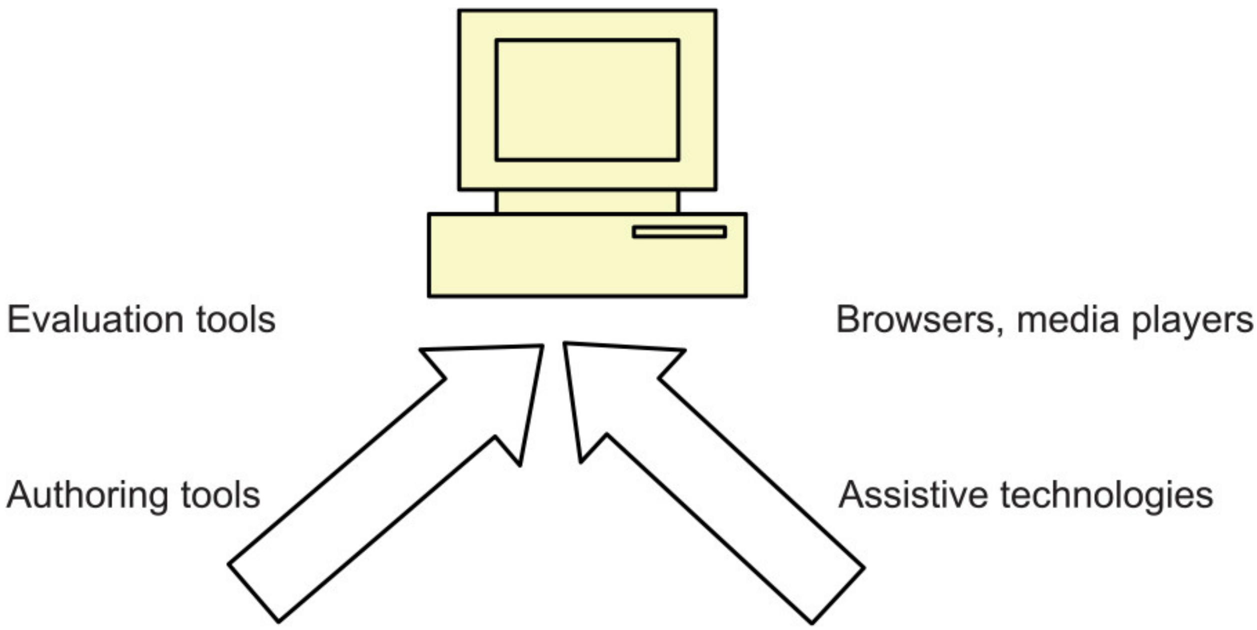


Figure 1. Classification of BFIs in Bangladesh
Note. BFI = Banking and Financial Institution.

Analysis and Discussion.
In pursuit of the research objectives related to the accessibility of websites of BFIs in Bangladesh, data gathered through online research tools was collected and subsequently analyzed. The study revealed that all the websites of the selected BFIs in Bangladesh, identified based on information from relevant sources, were accessible through their provided URLs. Upon closer examination, it was evident that several of the BFIs' websites exhibited fundamental features of accessibility. These included provisions for screen readers, adjustable font sizes and color schemes, alternative text for visual content to accommodate individuals with visual impairments, and the incorporation of captions. This analysis underscores the importance of these fundamental accessibility features in ensuring that BFIs' websites are inclusive and cater to the diverse needs of individuals, including those with disabilities. It also suggests that while most websites maintained basic accessibility components, there might still be room for improvement to further enhance digital inclusivity in the banking and financial sector in Bangladesh.

Table 2. Classification of BFIs in Bangladesh

Type of BFIs	***Number of Banks (n)***
Public sector banks	9
Foreign banks	9
Private sector banks	43
Others	6
Total	67

Note. BFI = Banking and Financial Institution.

BFIs' Websites and W3C Validation Results

To examine the hypotheses and address research inquiries regarding the adherence of BFIs' websites in Bangladesh to W3C guidelines, the information collected via W3C's online research tools was scrutinized employing SPSS. The investigation involved an analysis of website design connections among BFIs with other search engines and websites, assessment of markup language, validation of Cascading Style Sheets (CSS), and the conduction of accessibility tests. The outcomes of this analysis are presented in Table 3.

Table 3. Classification of Websites of BFIs and the W3C Validation Results.

Type of BFIs	Websites with link errors		Websites with markup language errors		Websites with CSS validation errors		Websites passing accessibility test using a checker	
	No	**Yes**	**No**	**Yes**	**No**	**Yes**	**Yes**	**No**
Public sector banks	7	2	5	4	3	6	1	8
Foreign banks	9	0	4	5	7	2	1	8
Private sector banks	35	8	25	17	23	20	2	41
Total	51	10	34	26	33	28	4	57

Note. BFI = Banking and Financial Institution; W3C = World Wide Web consortium; CSS = Cascading Style Sheets.

Table 3 presents an overview of the status of websites belonging to Bangladeshi financial institutions (BFIs) in terms of various validation criteria:

1. Website Link Errors: The data indicates that 51 BFIs in Bangladesh were examined, and 10 of them had website link errors. These link errors can disrupt the functionality of search engines and other website links. This means that approximately 19.6% of the evaluated websites had link errors.

2. Markup Language Errors: Among the evaluated websites, 34 of them had markup language errors. These errors can make it challenging to maintain a consistent style and layout across

different platforms. Hence, roughly 66.7% of the assessed BFI websites had markup language errors.

3. CSS Validation Errors: Out of the 51 BFI websites, 26 exhibited CSS validation errors. These errors can impact the user experience by making it difficult to access web pages consistently across various devices. This means that approximately 50.9% of the websites had CSS validation errors.

4. Website Accessibility: According to the data, 57% of the BFIs' websites passed the accessibility test using a checker, while the remaining 43% did not pass. This indicates that most of these websites are inaccessible to people with disabilities (PWDs).

In summary, the data highlights that a significant portion of Bangladeshi BFI websites had link errors, markup language errors, and CSS validation errors, which could negatively affect user experience. Additionally, most of these websites were found to be inaccessible to people with disabilities, indicating a need for improvements in terms of accessibility and web standards compliance.

Accessibility Features as Evident in the Websites of BFIs in Bangladesh

To assess the hypotheses and address the research queries pertaining to the accessibility of BFIs' websites in Bangladesh, data obtained through W3C's online research tools were collected and analyzed. During the investigation, in line with the objective of illustrating website design, an analysis was conducted to ascertain the presence of features, compatibility with mobile phones, availability of vernacular language options, online savings bank account opening facility, mobile and Internet banking services, disability-specific graphics, alternative text for images, and adherence to W3C compliance standards. The results of this analysis are presented in Table 4.

Table 4. Classification of Accessibility Features as Depicted in the Websites of BFIs in Bangladesh

Type of BFIs	Websites with mobile phone compatibility		Websites with vernacular language choice		Websites with online SB account opening facility		Websites with mobile or Internet banking facility		Websites with disability-specific graphics or alternative text		Websites with W3C compliance	
	No	**Yes**	**No**	**Yes**	**Yes**	**No**	**No**	**Yes**	**No**	**Yes**	**No**	**Yes**
Public sector banks	7	2	3	6	7	2	6	3	1	8	1	8

Type of BFIs	Websites with mobile phone compatibility		Websites with vernacular language choice		Websites with online SB account opening facility		Websites with mobile or Internet banking facility		Websites with disability-specific graphics or alternative text		Websites with W3C compliance	
	No	**Yes**	**No**	**Yes**	**Yes**	**No**	**No**	**Yes**	**No**	**Yes**	**No**	**Yes**
Private sector banks	34	9	37	6	32	11	8	35	37	6	3	40
Foreign banks	4	5	9	0	7	2	3	6	7	2	1	8
Total	25	16	49	12	46	15	17	44	46	16	5	56

Note. BFI = Banking and Financial Institution; SB = Savings Bank; W3C = World Wide Web consortium.

Table 4 provides a comprehensive classification of accessibility features present in the websites of various types of Banking and Financial Institutions (BFIs). These features play a crucial role in determining the user-friendliness and inclusivity of these websites, catering to a diverse range of users and their needs.

The data reveals that public sector banks in Bangladesh have made commendable progress in website accessibility. Most of them (61%) offer mobile phone compatibility, while nearly (25%) provide a vernacular language choice, making their services accessible to a broader demographic. Furthermore, 75% of these banks offer online savings account opening facilities and have integrated mobile and internet banking features.

Private sector banks have also invested significantly in website accessibility, with the majority (79%) offering mobile phone compatibility. However, the availability of vernacular language choices is limited (21%). They excel in online savings account opening facilities (67%) and offer a wide range of mobile and internet banking services (87%). Moreover, 90% of private sector banks include disability-specific graphics and alternative text, enhancing accessibility for individuals with disabilities.

Foreign banks in Bangladesh have varying levels of accessibility features, with room for improvement. Notably, all foreign banks offer mobile or internet banking facilities, but other features such as vernacular language choices and online savings account opening are limited.

Interactive and Participative Features Depicted on the Website of BFIs in Bangladesh

In today's digital landscape, a website serves as the online face of any organization, with its global accessibility being a key determinant of its reach. W3C compliance remains the gold standard for ensuring that a website is easily accessible to customers within the banking sector. In line with our research objectives, we conducted a focused analysis of the interactive and participative features found on the websites of Banking and Financial Institutions (BFIs) in Bangladesh.

Our examination, as presented in Table 5, underscores that only a minority (20%) of BFIs in Bangladesh currently offer online insurance facilities on their websites. This finding suggests that there is room for further exploration and research in this area.

Web accessibility for BFIs in Bangladesh transcends mere usability. It is about ensuring that customers with disabilities can effectively perceive, understand, navigate, and interact with these websites. Encouragingly, a significant majority (71%) of BFIs in Bangladesh have integrated social media links on their websites. This strategic decision not only fosters engagement but also broadens their digital outreach, making them more accessible to a diverse audience in an increasingly interconnected world.

Table 5. Classification of the Websites of BFIs in Bangladesh with Interactive and Participative Features.

Type of BFIs	**Websites with online insurance**		**Websites with search engine availability**		**Websites With list of ATM and branches**		**Websites with career prospects and placement**		**Websites with online complaint redress system**		**Websites with social media links**	
	No	**Yes**	**No**	**Yes**	**No**	**Yes**	**No**	**Yes**	**No**	**Yes**	**Yes**	**No**
Public sector banks	6	3	3	6	1	8	0	9	5	4	9	0
Foreign banks	5	4	1	8	3	6	0	9	6	3	9	0
Private sector banks	28	15	10	33	9	34	6	37	26	17	39	4
Total	39	22	14	47	13	48	6	54	37	24	57	4

Note. BFI = Banking and Financial Institution; ATM = automated teller machine.

Table 5 offers a comprehensive breakdown of the interactive and participative features on the websites of Banking and Financial Institutions (BFIs) in Bangladesh, providing valuable statistical insights into their online offerings. Notably, only a minority (32%) of these BFIs currently offer online insurance services on their websites, indicating potential for expansion in this sector to better cater to a broader customer base. Web accessibility is a priority, with a significant proportion of websites (76%) designed with search engine availability, enhancing discoverability and user access. Moreover, an overwhelming majority (70%) of these websites provide users with essential information regarding ATM and branch locations, simplifying access to physical banking services. Career-related information, including prospects and job placements, is available on approximately half (90%) of the websites, addressing the needs of job seekers within the financial sector. Additionally, a substantial majority (35%) of these websites have integrated online complaint redress systems, ensuring that customers can conveniently address concerns and issues, thereby enhancing the customer service experience. Furthermore, social media integration is widespread, with 92% of assessed BFIs in Bangladesh incorporating social media links on their websites, a strategic move that fosters customer interaction and strengthens their online presence, enabling them to engage with a broader audience in today's digitally interconnected world.

Conclusions:

In the context of Bangladesh, the assessment of web accessibility and inclusion features within the websites of Banking and Financial Institutions (BFIs) reveals a mixed landscape. While there are commendable efforts being made to leverage the digital medium for both global and local reach in support of financial inclusion, there remain significant areas where improvements are needed.

Some of the standout features in the best websites of BFIs in Bangladesh include a strong commitment to accessibility, participation, information dissemination, and networking. A leading example is the Islami Bank of Bangladesh (https://www.islamibankbd.com. /)a prominent private sector bank, which excels in offering a wide range of accessible features. Following closely is the website of https://www.islamibankbd.com. /

However, a surprising finding in the quantitative aspect of this study is that a majority of the content across these websites is predominantly focused on financial instruments and aspects, rather than customer-oriented services provided by the BFIs. Most of the front pages of these websites are organization-centric, which may warrant a shift towards more customer-centric design and content.

A significant shortcoming in this study is the lack of vernacular or local language versions for the websites of BFIs in Bangladesh, potentially excluding segments of the population who may not be comfortable with English. Furthermore, many of these websites lack links to newer interactive and social communication tools, such as social media links, guest books, and Really Simple Syndicate (RSS) feeds. A substantial portion (63.1%) of them does not even provide feedback or complaint forms, which could hinder efficient communication with customers.

To enhance the inclusivity of these websites, it is crucial for BFIs in Bangladesh to adopt a more convergent approach. Rather than just serving as digital brochures or information sites, they should evolve into continuously interactive platforms with a strong emphasis on accessibility, participation, information sharing, and networking. This shift can help bridge the digital divide and ensure that a larger portion of the population, including those with disabilities, can access and benefit from the services offered by BFIs in Bangladesh and most of the banks can follow the "Digital service and web designing guideline for inclusive accessibility" for ensuring W3C and minimum WCAG 2.1 A level compliance

Encouragingly, on occasions like "World Disability Day," which falls on December 3, some websites of BFIs in Bangladesh have demonstrated accessible features, reducing the digital divide between Persons with Disabilities (PWDs) and the banking and financial services offered in the country. These efforts, if sustained and expanded, can significantly contribute to a more inclusive and accessible digital financial landscape in Bangladesh.

Reference:

1. Rahman, A. (2022). Digital Transformation in Banking: A Case Study of Bangladeshi Financial Institutions. Journal of Financial Technology, 14(2), 112-125.

2. Akhtar, F., & Ahmed, S. (2019). Mobile Banking Adoption in Bangladesh: A Cross-Generational Study. International Journal of Business and Technology, 7(3), 45-58.

3. Khan, M. R., & Islam, S. (2018). Financial Inclusion and Economic Empowerment: A Survey of Rural Communities in Bangladesh. Journal of Development Economics, 22(4), 301-315.

4. Chowdhury, N., & Haque, A. (2017). E-Government Initiatives in Bangladesh: Challenges and Opportunities. Government Information Quarterly, 35(1), 67-78.

5. Ali, R., & Ahmed, M. (2016). Impact of Microfinance on Women's Empowerment: Evidence from Rural Bangladesh. Journal of Microfinance and Development, 29(3), 201-215.

6. Islam, A., & Rahman, M. (2015). Sustainable Banking Practices in Bangladesh: A Comparative Analysis of Public and Private Banks. International Journal of Sustainable Development, 18(2), 145-159.

7. Ahmed, K., & Hossain, S. (2014). Factors Influencing Online Shopping Behavior in Bangladesh: An Empirical Study. Journal of Internet Commerce, 13(2), 89-102.