

Technical Document Accessibility

by

Rynhardt Pieter Kruger



*Dissertation presented for the degree of PhD of Engineering
(Electronic) in the Faculty of Engineering at Stellenbosch
University*

Supervisor: Prof T. R. Niesler

Co-supervisor: Prof F. De Wet

December 2022

Declaration

By submitting this dissertation electronically, I declare that the entirety of the work contained therein is my own, original work, that I am the sole author thereof (save to the extent explicitly otherwise stated), that reproduction and publication thereof by Stellenbosch University will not infringe any third party rights and that I have not previously in its entirety or in part submitted it for obtaining any qualification.

Date: 2022/02/04

Copyright © 2022 Stellenbosch University
All rights reserved.

Abstract

Technical Document Accessibility

R. P. Kruger

*Department of Electric and Electronic Engineering,
University of Stellenbosch,
Private Bag X1, Matieland 7602, South Africa.*

Dissertation: PhD (Electron)

December 2022

Technologies such as text to speech and braille provide alternative representations of electronic text, allowing blind users to access most documents independently. However, non-textual graphical information in electronic documents, such as diagrams and equations, remain largely inaccessible to the blind population. In this study, we explore audio-visual sensory substitution as a means of rendering the information represented as diagrams and equations in electronic documents to blind readers. For diagram accessibility, we have extended the audio rendering approach used by the well-established vOICE algorithm to allow interactive and localised exploration by means of gestures and a touch screen. The effectiveness of this approach was evaluated in a set of user trials that required six sighted and six blind subjects to identify elements of scenes consisting of a number of geometrical shapes and emoticons. The results show that both groups of subjects were more successful at identifying shapes using the interactive algorithm than they were when using the baseline vOICE algorithm to a highly statistically significant degree. Furthermore, the results indicate that this improvement is greatest for the most complex scenes. Subsequently, we extended this approach to enable the reading of mathematical equations embedded in PDF documents, by incorporating a navigational system that draws on the navigational mechanisms often used to explore the virtual worlds of text adventure games. The relative spatial placement of the elements of an equation are represented as a virtual world, so that the reader can navigate from element to element. Textual elements are announced as synthesised speech while graphical elements are rendered using audio-visual sensory substitution. This approach was evaluated by eleven blind and fourteen sighted participants in a set of trials comprising the identification of several equations extracted from PDF documents. Considering

only completely correct responses, blind and sighted candidates were able to attain an accuracy of 74% and 83% respectively, with an overall average of 78%. When partial correctness is evaluated, these accuracies are even higher. We conclude that, by enabling interactive exploration by means of gestures, a spatial model represented as a virtual world, and audio-visual sensory substitution, currently inaccessible diagrams and mathematical content in technical documents can be made accessible to visually impaired readers without the need for specialised equipment.

Uittreksel

Tegniese Dokument Toeganklikheid

(“Technical Document Accessibility”)

R. P. Kruger

*Departement Elektriese en Elektroniese Ingenieurswese,
Universiteit van Stellenbosch,
Privaatsak X1, Matieland 7602, Suid Afrika.*

Proefskrif: PhD (Elektron)

Desember 2022

Tegnologieë soos teks na spraak en braille bied alternatiewe voorstellings van elektroniese teks, wat blinde gebruikers in staat stel om onafhanklik toegang tot meeste dokumente te verkry. Nie-tekstuele grafiese inligting in elektroniese dokumente, soos diagramme en vergelykings, bly egter grootliks ontoeganklik vir die blinde bevolking. In hierdie studie ondersoek ons audiovisuele sensoriese substitusie as 'n manier om die inligting wat as diagramme en vergelykings in elektroniese dokumente voorgestel word aan blinde lesers weer te gee. Vir diagramtoeganklikheid het ons die klankweergawe-benadering wat deur die goed gevestigde vOICE-algoritme gebruik word, uitgebrei om interaktiewe en gelokaliseerde verkenning deur middel van gebare en 'n raakskerm moontlik te maak. Die doeltreffendheid van hierdie benadering is geëvalueer in 'n stel gebruikersproewe wat ses siende en ses blinde proefpersone vereis het om elemente van tonele te identifiseer wat uit 'n aantal geometriese vorme en embleme bestaan. Die resultate toon dat beide groepe proefpersone meer suksesvol was met die identifisering van vorme deur die interaktiewe algoritme te gebruik as wat hulle was toe hulle die basislyn vOICE-algoritme gebruik het tot 'n hoogs statisties beduidende mate. Verder dui die resultate daarop dat hierdie verbetering die grootste is vir die mees komplekse tonele. Vervolgens het ons hierdie benadering uitgebrei om die lees van wiskundige vergelykings wat in PDF-dokumente voorkom moontlik te maak, deur 'n navigasiesistelsel in te sluit wat gebruik maak van die navigasie meganismes wat dikwels ingespan word om die virtuele wêreld van teksavontuurspeletjies te verken. Die relatiewe ruimtelike plasing van die elemente van 'n vergelyking word as 'n virtuele wêreld voorgestel, sodat die leser van element tot element kan navigeer. Tekstuele

elemente word as gesintetiseerde spraak aangekondig, terwyl grafiese elemente weergegee word deur oudiovisuele sensoriese substitusie te gebruik. Hierdie benadering is geëvalueer deur elf blinde en veertien siende deelnemers in 'n stel proewe wat die identifikasie van verskeie vergelykings behels het wat uit PDF-dokumente onttrek is. As slegs volkome korrekte antwoorde in ag geneem word, kon blinde en siende kandidate 'n akkuraatheid van 74% en 83% onderskeidelik behaal, met 'n algehele gemiddeld van 78%. Wanneer gedeeltelike korrektheid geëvalueer word, is hierdie akkuraatheid selfs hoër. Ons kom tot die gevolgtrekking dat, deur die insluiting van interaktiewe verkenning deur middel van gebare, 'n ruimtelike model wat as 'n virtuele wêreld voorgestel word, en oudiovisuele sensoriese substitusie, tans ontoeganklike diagramme en wiskundige inhoud in tegniese dokumente kan toeganklik gemaak word vir siggestremde lesers sonder die behoefte aan gespesialiseerde toerusting.

Acknowledgements

First I would like to thank my supervisors, Prof. Thomas Niesler and Prof. Febe De Wet. This work would not have been possible without your valuable advice and guidance. I would also like to thank the Council for Scientific and Industrial Research for the financial support which made this work possible. Thanks to my colleagues at the CSIR for all the stimulating conversations. Working with you on related projects helped me to place my own work in context. Thanks also to Prof. Martin Kidd from the Centre for Statistical Consultation at the University of Stellenbosch, for assisting with the analysis of the data. Furthermore, I would like to thank all the research participants who contributed to my study. Thank you for giving me your time, and also for your insightful feedback. Lastly, I would like to thank all my family and friends for your constant encouragement. Here I would like to mention my Mother and Desmond, and my siblings Lugene, Marnitz, Jumari, and Tyler, for your love and for continuing to believe in me. Also my very good friends Lilley and Willem, and the rest of the group known as the Wolfpack, for always being willing to listen, and for providing distractions when I required it. Every one of you played a very unique roll in providing me with the support I needed to complete this thesis.

Dedications

To my Mother, Engela Elliott. In 2019, during the course of this work, she became my live kidney donor.

Contents

Declaration	i
Abstract	ii
Uittreksel	iv
Acknowledgements	vi
Dedications	vii
Contents	viii
List of Figures	xi
List of Tables	xii
Nomenclature	xiii
Publications	xiv
1 Introduction	1
1.1 Linear Representations of Non-Textual Content	2
1.2 Direct Translation of Non-Textual Content	3
1.3 Scope	5
2 Background	7
2.1 The Need for Accessible Technical Documents	8
2.2 Workflow for Reading Accessible Electronic Documents	9
2.3 Accessibility of Documents	11
2.4 Summary and Conclusion	18
3 Accessibility of Graphical Material	19
3.1 Tactile Diagrams	20
3.2 Described Diagrams	25
3.3 Sensory Substitution	26

<i>CONTENTS</i>	ix
3.4 Domain Specific Diagram Explorers	36
3.5 Data Visualisation	37
3.6 Discussion	38
3.7 Summary and Conclusion	39
4 Approach to Accessibility of Graphical Material	41
4.1 Algorithms	42
4.2 Experimental Evaluation	45
4.3 Results	50
4.4 Discussion	52
4.5 Summary and Conclusion	53
5 Accessibility of Mathematical Material	54
5.1 Braille	54
5.2 Electronic Formats for Accessible Mathematical Content . . .	55
5.3 Challenges in Mathematics Accessibility	57
5.4 Inspiration from Accessible Games	64
5.5 Discussion	66
5.6 Summary and Conclusion	67
6 Approach to Accessibility of Mathematical Material	69
6.1 Proposed Approach	70
6.2 Example Session	77
6.3 Experimental Evaluation	84
6.4 Scoring	90
6.5 Results and Discussion	93
6.6 Informal Feedback	97
6.7 Summary and Conclusion	98
7 Conclusion	100
7.1 Integration	103
7.2 Further Applications of Our Approach	104
7.3 Further Extensions of Our Approach	107
7.4 Summary and Conclusion	109
Appendices	110
A Images Used in the Evaluation	111
B Sirin Tutorial	115
B.1 Introduction	115
B.2 Description of Sirin Output	115
B.3 Description of Sirin Modes	116
B.4 Example 1: Basic Shapes	117
B.5 Example 2: Basic Equation	117

*CONTENTS***x**

B.6 Example 3: Fractions	118
B.7 Example 4: Noticing where an exponent is	119
B.8 Example 5: a Square Root	119
B.9 Example 6: Equation with Large Brackets	120
B.10 Example 7: a Matrix	120
B.11 Further Examples	121
List of References	123

List of Figures

2.1	LaTeX source code for an example document.	10
2.2	Visual appearance of an example document.	11
2.3	An example document, as read out by a screen reader.	11
2.4	Eight-dot braille cell.	12
2.5	The Microsoft Inspect tool showing the accessibility properties of the edit menu in the Notepad application [1].	15
2.6	Image of a Brailliant B80 Braille Display [2].	17
3.1	Example of a semantic diagram: visual representation of a graph.	19
3.2	The graph of Figure 3.1, represented as a textual table of vertices and edges.	20
3.3	Example of a pictorial diagram which cannot be completely represented as text: a map of South Africa.	21
4.1	The three categories of shapes used in Stages 1, 2 and 3: Triangle and square at the top, circle at the bottom.	46
4.2	The four emoticons rendered in Stages 4 and 5: Skew mouth and smiley at the top, sad face and winking face at the bottom.	47
4.3	The percentage of correct responses for each algorithm, for each stage of the test. The means and 95% confidence intervals have been calculated using least squares means analysis.	51
6.1	Pseudocode of the algorithm used to construct the connections of the document object model (DOM).	72
6.2	Document Object Model for $y = \frac{x^2+4}{3}$	73
6.3	Number of perfectly correct responses per question for each stage.	93
6.4	Time taken by blind and sighted users to complete Stages 1 and 2 respectively. Vertical bars denote 95% confidence intervals.	96

List of Tables

2.1	Latin alphabet in braille.	12
4.1	The stages of the experimental evaluation, showing the number of test images presented using the vOICe and interactive algorithms respectively. The precise composition of each image is given in Appendix A.	48
4.2	Percentage of correct identifications per testing stage and algorithm type (vOICe and interactive) as well as average response times.	50
6.1	Equations for Stage 1 - Text Mode.	86
6.2	Equations for Stage 2 - Text Mode & Graphical Mode.	86
6.3	Some attributes of the test candidates.	88
6.4	Composition of the correctness score for the equation $y = \frac{x}{2} + x^2$	91
6.5	Example scoring table for the hypothetical case where the equation $y = \frac{x}{2} + x^2$ is transcribed as $y = x^2 + \sqrt{x}$	92
6.6	Number of completely correct responses per equation for both evaluation stages.	93
6.7	Correctness scores per equation for both evaluation stages	94

Nomenclature

Acronyms

- ARIA Accessible Rich Internet Applications
- STEM Science, Technology, Engineering, and Mathematics
- TTS Text to Speech
- WAI Web Accessibility Initiative
- WCAG Web Content Accessibility Guidelines

Definitions

- Accessibility The ability of disabled persons to equally access and benefit from products and services.
- Assistive Technology Technology which allows a disabled person to engage in an activity which would usually be inaccessible, for example a screen reader.
- Audio-Visual Sensory Substitution The mapping of visual to audio information, to allow substituting the sense of vision by the sense of hearing.
- Screen Reader Background software that monitors the state of the user interface and presents it as speech or braille, allowing blind and visually impaired users to operate a computer.
- Sensory Substitution The mapping of information which are usually received by one sense, to a form which can be understood by another.
- Tactile-Visual Sensory Substitution The mapping of visual to tactile information, to allow substituting the sense of vision by the sense of touch.
- Universal Design The practice of designing products and services which can be used by the widest possible audience, in the widest possible situations. Universal design includes accessibility.

Publications

The work in Chapter 4 of this thesis appears as a full-length peer-reviewed conference paper and has been presented at the 21st International Conference on Information Visualisation (IV2020), held in Melbourne, Australia in September 2020.

- Kruger, R; de Wet, F; Niesler, T.R. *Interactive image exploration for visually impaired readers using audio-augmented touch gestures*. Proceedings of the 24th International Conference on Information Visualisation (IV2020), Melbourne, Australia, pp. 544–549, 2020.

Furthermore, the work in Chapter 6 of this thesis is the subject of a paper that has been submitted to ACM Transactions on Accessible Computing, which is a respected, peer-reviewed and accredited journal in the field. A preprint of this paper is available at:

- Kruger, R; de Wet, F; Niesler, T.R. *Mathematical content browsing for print-disabled readers based on virtual-world exploration and audio-visual sensory substitution*. arXiv preprint, available from <https://arxiv.org/abs/2202.01639>.

Chapter 1

Introduction

The proliferation of electronic text has made it possible for blind and print-disabled readers to attain near-equal access to articles and other documents. This is achieved by the use of screen readers and other assistive technology, which are able to convey electronic text to a blind reader via alternative modalities like synthesised speech and electronic braille terminals. However, these alternatives are currently unable to convey graphical information like diagrams or print mathematics, which are by nature two-dimensional. Standards exist for adding descriptions to graphical material (referred to as alt text), but these standards are not always adhered to. Textual descriptions are also not sufficient when the graphical shape itself conveys meaning, for example, a geographical map or the specific shape of a sound wave. For these reasons, much technical writing, including contemporary published scientific research, remains largely inaccessible to the blind and visually impaired community.

The inaccessibility of graphical content has implications for the education of blind learners in fields such as mathematics, science, engineering, and technology [3]. For this reason, many professions within these fields remain difficult for a blind user to access, while some are completely inaccessible. These issues have been aggravated by the COVID-19 pandemic, where remote learning has become the only viable solution to maintain appropriate social distancing. The current approaches to provide effective access to graphical material for blind readers require the use of expensive hardware devices, which is impractical in a distance learning environment, due to their high cost and thus inaccessibility to individual users.

Accessibility to technical material can be achieved with two approaches. First, content can be presented in an alternative (usually linear) format which can be read using braille or synthesised speech (for example, textual descriptions or LaTeX notation). Second, by sensory substitution, which is the direct translation of the content into a form that can be understood by another sense (for example, tactile diagrams). The first approach is appropriate where the graphical content is a visual representation of semantic content that can be represented in an alternative form, for example, mathematical equations and

chemical formulas [4]. The second approach is particularly useful for providing access to pictorial content where the shape itself is important, for example, floor plans and geographical maps [5].

1.1 Linear Representations of Non-Textual Content

Several linear formats are currently in use for reading mathematical content. The best known example is arguably braille mathematics. Although different standards exist for braille mathematics, for example, Nemeth braille [6] and UK mathematics [7], they all make use of additional markup for providing context not as readily apparent in a linear form. For instance, mathematical braille codes provide markup for indicating the start and end of a fraction, which is immediately apparent when looking at a graphical representation of the equation but is harder to determine from a linear textual representation. Similar extra markup is also utilised by textual linear mathematical forms like LaTeX and ASCII Math. Although these textual formats are also readable by a blind or visually impaired reader, they were designed for typesetting. They are therefore not as easy to read in braille as a dedicated braille mathematics code, as well as being cumbersome to read using a screen reader.

Several solutions have been proposed for providing a more efficient reading experience of linear mathematical content with a screen reader. One of the first examples is AsTeR, described by Raman [8]. AsTeR converts mathematics and other content from LaTeX into synthesised speech, using different speech attributes to indicate context. For instance, AsTeR will read a numerator in a high pitch, while a denominator will be read in a low pitch. AsTeR also provides commands for moving forward and backward through the content, as well as exploring structures in more detail. With the development of the MathML standard for displaying mathematics on the web, similar speech browsers have been devised for exploring MathML. Design Science developed the Math Player extension for some web browsers, which provides a hierarchical explorer over MathML equations for screen reader users [9; 10]. Equations are represented as a tree structure, with sub-expressions represented as child nodes of their enclosing expressions. Similar explorers are also available as plugins for popular screen readers, like the Access8 extension for the NVDA screen reader [11]. An accessibility mode, including a hierarchical explorer, was also recently added to the MathJax library used for rendering TeX equations on the web [12; 13].

One disadvantage of a linear representation of spatial content like mathematics is the length of the linear form compared to the visual representation. Linear formats introduce extra markup to indicate context not readily apparent without spacial information. The result is more symbols in the equation

overall. As all symbols are part of a linear sequence, blind and visually impaired readers also cannot immediately determine the context of a term or symbol without reading forward or backward through the linear representation to find the contextual markup. This problem is partially addressed by the hierarchical structure utilised by many accessible mathematical browsers, although this approach requires extra navigational steps by the user.

An interesting approach for providing an efficient overview of lengthy structural content is to provide a method for spontaneous exploration. This can be achieved by using a braille display which allows the user to quickly move their fingers over the content. Unfortunately, most braille displays have a length of about 40 braille cells or units, which corresponds to 40 symbols or less. This limitation requires that the user has to scroll the display forward or backward to attain a full view of the content. Another approach for providing spontaneous exploration is to allow the user to immediately query elements along with their context. This can be achieved with natural queries, possibly via speech recognition [14]. This approach provides a user experience similar to that provided by command line explorers for nested structures.

Another challenge related to a specialised linear representation is that mathematical content in published works is often presented as image-only content, or vector-based formats like untagged PDF documents. Without access to the source document, these publications need to be converted back into a semantic format which can be represented as braille, LaTeX, or MathML. This conversion is most often a manual process, although pattern recognition approaches can be utilised for automated conversion. The InftyReader is a software package which accomplishes such automated conversion from graphical mathematical equations to a semantic form like LaTeX or MathML [15; 16]. However, a disadvantage of a pattern-based conversion process is that it can only recognise elements that were included in the training data. The output might also contain recognition errors. Although the developers of the Infty software report a successful recognition rate of above 95%, a blind reader would be unable to verify the correctness of the output without access to the visual representation.

1.2 Direct Translation of Non-Textual Content

An alternative approach for providing access to two-dimensional visual material is to offer a direct translation from visual data to a form that can be understood by a different modality of sensory perception. This approach is sometimes called **sensory substitution**, as it substitutes one sense by another for receiving information. Two main approaches that have been studied for non-visual accessibility are tactile perception and auditory feedback.

Sensory substitution has the attractive advantage that it can function independently from a semantic representation of the content. It can therefore be used to access material for which a semantic representation is limited, or not yet defined. Sensory substitution can also convey content that is pictorial in nature, that is, content for which the graphical shape itself conveys meaning.

1.2.1 Tactile Sensory Substitution

Perhaps the most well-known example of tactile-visual sensory substitution is the tactile diagram. Several methods are used to produce tactile diagrams. One method involves manually forming the diagram by pasting pieces of cardboard, lengths of string, braille labels, and other objects to a base. The base is then used to mould plastic sheets which ultimately contain the finished diagram. Tactile diagrams can also be produced with a graphics-capable braille embosser, like the embossers produced by ViewPlus [17]. When an embosser is used, the tactile diagram consists of braille dots of varying sizes and heights. It is thus possible to use differing heights to denote brightness. Braille text can also be embossed alongside the graphical content, to act as labels or markers.

Tactile diagrams will often require manual intervention in the conversion process, as the resolution of the tactile sense is much lower than that of the visual sense [18]. Tactile diagrams also require labels and other textual elements to be represented in braille, which takes up more space than the printed equivalent. With the increasing availability of digital diagrams containing semantic metadata, it has become possible to produce tactile diagrams requiring less manual intervention. This is accomplished by offering textual descriptions of graphical components alongside the tactile diagram, which is based on the metadata contained in the digital diagram. The company ViewPlus offers a product called Iveo, which enables a blind reader to study an embossed diagram by placing it on a touch pad [19]. Although the diagram might not be very effective in its tactile-only form (due to the lower resolution available with the tactile modality), the user is able to obtain a description about an element by selecting that element on the touch pad.

Disadvantages of tactile diagrams include the high cost associated with their production, both in terms of the manual process and the hardware that is required. Furthermore, the time required for tactile diagram production limits their utility in the research environment. Alternative hardware devices have been studied as more cost-effective solutions, but these devices are not readily available. Sensory substitution via the audio sense has the attractive property that no additional hardware is required besides the audio functionality built into most consumer devices.

1.2.2 Audio Sensory Substitution

Sensory substitution via the audio sense has been studied by Meijer [20], who described an algorithm called the vOICE (the capital letters meaning “Oh, I see”.) The vOICE translates images to sound by scanning an image left to right, generating a series of tone chords corresponding to the columns of pixels in the source image. Each chord consist of several sinusoids perceived as tones, where each tone indicates a bright pixel in the original image. The frequency and therefore the pitch of a tone is determined by the vertical position of the pixel in the source image, with higher pixels producing higher pitched tones. Chords are also placed in the stereo field to indicate horizontal position. Therefore, the left-most column of the image would be played to the left of the stereo field and the right-most column to the right [21]. To illustrate the vOICE algorithm, one could consider a source image consisting of a diagonal line from top-left to bottom-right. The resulting audio representation from the vOICE would consist of a descending tone starting in the left speaker and ending in the right. Alternative image to sound mappings have also been described by, for example, Capelle *et al.* [22], and Abboud *et al.* [23].

Although sensory substitution approaches to graphical accessibility have most often been employed for pictorial diagrams where the shape itself needs to be conveyed, it has also been studied as an alternative to the linear approach usually used to represent semantic spatial information like mathematical equations. DotsPlus is a system for representing mathematical content as tactile diagrams interlaced with braille symbols [24]. DotsPlus is intended to be produced directly from visual mathematics on a graphical braille embosser like the Tiger embosser from ViewPlus [17]. Many mathematical symbols are replaced by braille counterparts, while visual elements are directly translated to a tactile form. In this way, most of the spacial layout and graphical indicators of the original visual representation are preserved.

1.3 Scope

The aim of the work reported in this thesis is to explore the possibility of using sensory substitution to allow blind or visually-impaired users to access graphical content encountered in an electronic document. Our approach is to render the graphical content as audio. To achieve this, we extend the well-established vOICE algorithm, which renders an image as a sequence of tone chords. The standard implementation of the vOICE algorithm was designed to render an image in its entirety. However, more targeted exploration is necessary for understanding complex images like scientific diagrams and graphs. To this end we incorporate gestures and a touch screen to allow flexible and interactive exploration of the rendered scene. This allows an image that is too complex to be interpreted at once to be systematically explored.

We then further extend these ideas to allow the reading of mathematical equations. To do this, we draw on the methods often used in textual adventure games to represent and explore virtual worlds. We represent the elements of an equation as the rooms of a virtual world with a spatial layout reflecting the geometric structure of the equation. By allowing interactive exploration of this world using textual input and output as well as our extension of the vOICE algorithm, both textual and graphical aspects of mathematical content as well as their spatial arrangement are made accessible to blind and print-disabled readers without the requirement of additional markup.

In the following chapter, we provide an overview of general document accessibility to blind readers. Chapter 3 gives an overview of current approaches for graphics and diagram accessibility. In Chapter 4 we describe our proposed approach for providing direct access to graphical material via sensory substitution. Chapter 5 reviews the state of the art in terms of mathematical material in particular, while Chapter 6 describes the application of our approach to accessible equation reading. In each case, the proposed approach is evaluated by a set of user trials.

Chapter 2

Background

From an accessibility-based perspective, electronic technical material can be broadly categorised into two types of content. The first is textual content like the text of an article, which can already be accessed by screen readers and other assistive technology. The second is graphical content like diagrams and visual equations, which may be encoded as vector graphics or rasterised image data. This type of content is inaccessible to a screen reader unless additional accessibility annotations are included in the document [3]. It is important to note that the two types of content are often interlaced, for example, the symbols of an equation, or the labels of a diagram may be presented as electronic text. However, without the graphical content to provide context, the textual information of an equation or diagram can usually not be interpreted by a blind or visually impaired reader.

With the aid of modern web technologies like MathML [25] and MathJax [26], screen readers are increasingly able to access mathematical content. This is because these technologies are able to convey the semantic structure of equations to the screen reader, which is then able to present it as synthesized speech or braille. However, this approach requires that the document was published in a way that preserved the semantic information of the mathematical content. In practice, many documents are published with the mathematical content represented as images, which are inaccessible to a screen reader. These technologies are also currently limited to specific domains of graphical content for which a well-defined semantic structure can be inferred, like mathematical equations. Furthermore, most technical papers are still published as PDF documents without additional accessibility information. For these documents, screen readers are only able to access the textual content. The equations and diagrams, which are often required for a proper understanding of the work, remain inaccessible.

This chapter gives a general overview of current approaches that enable the accessibility of documents. Section 2.1 provides the context, and describes the need for accessible technical documents. Section 2.2 describes a typical work flow for accessing documents, and also gives an example of the degree

of accessibility that may be expected with untagged PDF. Section 2.3 gives a general overview of document accessibility, with a discussion on braille and audio as traditional vehicles of accessible content. This section also describes how electronic information is accessed by a print-disabled user, including the use of screen readers and other accessibility software.

Subsequent chapters will present the separate aspects of accessible technical documents in more detail. Chapter 3 describes approaches for accessibility to graphical information, which may be structured diagrams or diagrams which are completely pictorial in nature. Chapter 5 focuses specifically on mathematics accessibility, with regards to standards for mathematics in braille, approaches for accessing mathematics using screen readers in electronic documents, and the challenges with current approaches for mathematics accessibility.

2.1 The Need for Accessible Technical Documents

For a blind person to receive an adequate STEM (science, technology, engineering, and mathematics) education, non-visual material and teaching methods must be used [27; 28]. Traditionally, blind learners, like learners with other disabilities, were educated at special schools for the blind, which employed specially trained educators. However, these schools were often under-resourced, especially in terms of appropriately qualified science and mathematics teachers. Separate education also held social disadvantages for both disabled and non-disabled learners, as neither were able to interact with the other, which therefore contributed to social inequality. Inclusive education was instituted to allow disabled learners to learn alongside their non-disabled peers in mainstream schools [29]. Unfortunately, the inaccessibility of visual material, as well as inadequate training for educators in non-visual teaching methods, remains an impediment. This contributes to a low rate of blind students pursuing STEM subjects in tertiary education, and therefore a low representation in STEM related careers.

According to the National Federation of the Blind in the United States of America, blind people account for less than 5% of the entire STEM work force [30; 28]. This number is suspected to be much lower in developing countries like South Africa, where inclusive education is not yet fully implemented. Blind people also account for 43.6% of non-competitive areas of employment, compared to 19 other assessed disability types. It is therefore reasonable to assume that education factors significantly into the prospect for employment of a blind individual. Indeed, the number of blind people with a tertiary education has an employment rate twice that of those with just a high school diploma. While accessible STEM teaching methods in general are beyond the

scope of this study: we will focus on one key aspect, the accessibility of technical documents. We will therefore, in this chapter, examine specific forms of accessible documents.

In the ideal scenario, a blind person would receive material in a fully inclusive form. Also called born-accessible publications, these documents are designed to be delivered electronically, and can be converted into large print, audio, or braille as the user desires [31]. Graphical material like diagrams are either textually described, or (depending on the kind of diagram) embossed as hard copy tactile diagrams which can be explored by touch. However, documents are seldom available in a born-accessible form. Furthermore, not all blind learners have access to the hardware required to convert a document into braille, or diagrams into a tactile form.

To address the lack of born-accessible material, schools and universities have to convert documents from inaccessible formats themselves, or via a third party like an organisation for the blind. This conversion may take significant time, resulting in a student receiving material late. Furthermore, the skills necessary for a proper conversion is not always available, and in many cases students have to convert material themselves, which diverts their attention from their studies.

2.2 Workflow for Reading Accessible Electronic Documents

Of the current approaches to technical document accessibility, the best experience is arguably obtainable when technical papers are published using web technologies such as HTML with MathML. The World Wide Web Consortium maintains standards specifying how documents should be constructed for maximum accessibility to blind and other disabled readers, including the Web Content Accessibility Guidelines (WCAG) [32], and the Web Accessibility Initiative: Accessible Rich Internet Applications (WAI-ARIA) specification [33]. Version 2.0 of the WCAG is also an ISO standard [34]. When these standards are adhered to, modern screen readers can present the document to a blind user in a structured form, and offer navigational controls that allow the reader to move between sections and within structured data like tables. When MathML is used, blind users are able to read equations in sections, and to navigate among sub-expressions and terms [25; 9; 11]. For diagrams, the World Wide Web Consortium recommends the use of the scalable vector graphics (SVG) specification [35]. SVG allows parts of a diagram to be individually described to blind and print-disabled users.

Unfortunately, very few publications adhere to such standards. In addition, much research has been published before the introduction of such web-based workflows, and these documents would have to be converted to the richer,

more descriptive HTML and MathML formats. Such conversion cannot be achieved automatically. Therefore, the current situation is that most technical papers are offered as PDF documents without any accessibility markup. These documents consist of plain text interleaved with vector drawing instructions (to render equations and vector drawings) and embedded rasterised images (to render bitmap images). While these documents can be downloaded by and read using a screen reader, only the plain text portions of the document are accessible. Reading the text embedded in the diagrams or equations from these documents is of very limited use to a blind reader, as the text cannot be used to interpret the diagrams or equations without the graphical information to provide context.

To increase the accessibility of PDF documents, the PDF standard prescribes tags which can be added to the content [36]. PDF/UA, which is also an ISO standard, uses tags that are similar to HTML markup, and can be used to convey the same semantic information available with HTML documents [37; 38]. However, most technical papers continue to be published as untagged PDF documents. Even when PDF documents are tagged, they often do not include all the relevant semantic information. The PDF document of a technical paper may, for instance, include tags denoting tables and headings, but no tags denoting the semantic information of equations. Hence the current reality is that, although screen readers provide access to the textual material present in technical PDF documents, the information contained in equations and in figures remains overwhelmingly inaccessible.

2.2.1 Accessibility of an Example Document

To illustrate the accessibility of untagged PDF documents, consider the document produced by the LaTeX source code presented in Figure 2.1. The PDF will appear visually as presented in Figure 2.2. The text read by the screen reader will be the text appearing in Figure 2.3, read out line by line.

```
\documentclass[a4paper]{article}
\begin{document}
Consider the following equation:

$$y = \frac{x^3}{x+1}$$

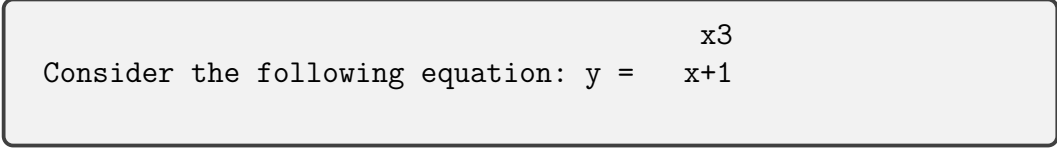
\end{document}
```

Figure 2.1: LaTeX source code for an example document.



Consider the following equation: $y = \frac{x^3}{x+1}$

Figure 2.2: Visual appearance of an example document.



Consider the following equation: $y = \frac{x^3}{x+1}$

Figure 2.3: An example document, as read out by a screen reader.

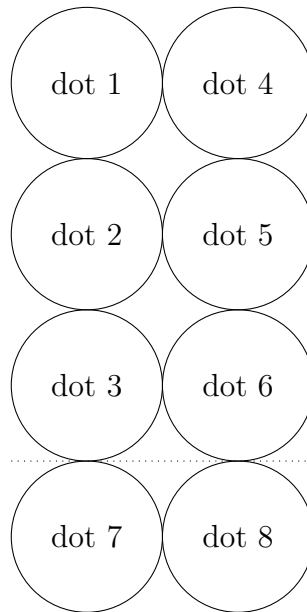
2.3 Accessibility of Documents

The current mainstream approaches to making documents accessible to blind and print-disabled users include braille and audio, as well as electronic documents. The latter are accessed by conversion to speech or braille, or a combination of both. This section gives a brief overview of these approaches with respect to documents in general, while subsequent chapters will describe how these approaches have been extended to allow access to graphical and mathematical content in particular.

2.3.1 Braille

Braille is a tactile writing system first developed by Louis Braille in 1824, and today used by blind people across the world [39; 40]. It consists of a series of horizontally consecutive cells, where each cell comprises a number of dots [41; 42]. For most braille scripts, including literary braille in most languages and music braille, a cell consist of up to six dots, arranged in a grid with two columns and three rows. The exact number of dots, as well as their placement within the grid, determine the letter or symbol being conveyed. However, in some braille scripts, including 8-dot computer braille, a cell may consist of up to eight dots [43]. In this case, an additional row of dots is added to the bottom of each cell. Figure 2.4 depicts such an eight-dot braille cell, also indicating how, for standard literary braille, only the top six dots are used.

Unlike printed text, no formatting or typeface can be applied to braille symbols. Braille is embossed using a specific size and diameter for each dot and cell. The specific braille standard is determined by the country in which the content is produced. An example of a braille standard is Specification 800, “Braille Books and Pamphlets,” National Library Service for the Blind and Physically Handicapped of the Library of Congress in the United States of America. It specifies that a cell should be 0.48 mm in height, and a braille dot 1.44 mm in diameter. The distance between centres of adjacent dots should

**Figure 2.4:** Eight-dot braille cell.

Braille:	•	⋮	••	⋮•	•	⋮•	⋮•	⋮•	••	⋮•
Print:	a	b	c	d	e	f	g	h	i	j
Braille:	⋮	⋮•	⋮•	⋮•	⋮•	⋮•	⋮•	⋮•	⋮•	⋮•
Print:	k	l	m	n	o	p	q	r	s	t
Braille:	⋮•	⋮•	⋮•	⋮•	⋮•	⋮•				
Print:	u	v	w	x	y	z				

Table 2.1: Latin alphabet in braille.

also be 2.34 mm for dots in the same cell, and 6.2 mm for dots in adjacent cells. The distance between dots in adjacent lines should be 1 cm [44]. The prescribed size ensures that a single braille cell can always fit underneath the tip of the finger.

Table 2.1 depicts the braille symbols representing the Latin alphabet. Note that the letters “a” to “j” appear similar to the letters “k” to “t”, except for the addition of one dot. A similar pattern can be observed for letters “u” to “z”, with the notable exception of “w”. At the time of braille’s invention, the letter “w” was not widely used in French (the language of Louis Braille), therefore it was added at a later time.

Braille cells do not always correspond uniquely to printed symbols. A cell may represent a letter or symbol, or it may represent a part of a letter or symbol. Special braille symbols may also indicate a context switch, for example the start or end of mathematical content, or indicate the grade of braille

being used. In contracted braille, common words and letter combinations are replaced with shortened forms. The braille grade denotes the level of contraction being used. Contracted braille was developed to increase the content that may fit within a fixed number of cells, and therefore content on a page or in a book. This is necessary since braille takes up a lot of space when compared with normal print. A typical example is the Bible, which comprises several volumes, and in most languages is approximately 1.5 meters high when completely converted to braille. Uncontracted braille is denoted as grade 1 braille, and is the form of braille usually taught to new braille readers. Grade 2 braille is standard contracted braille, and is the type of braille being read by most users. Some languages also provide grade 3 braille, which is similar to printed shorthand, and usually for personal use.

2.3.2 Audio

Before the proliferation of electronic documents, audio, along with braille, were the only mainstream formats for accessible documents. Audio is still an important format for accessible documents, although modern audio documents are usually augmented with standards like Daisy or Epub3 for improved navigation [45; 46; 47]. The Daisy and Epub3 standards allow listeners to navigate by chapter, section or word. However, due to the continuous nature of audio, navigation by symbol is not possible.

Some of the first phonograph recordings were spoken words [48]. However, due to the limited capacity of these early recording mediums, talking books first became viable in the 1930s on gramophone records. These books were (and often still are) usually recorded by volunteers for library services for the blind like the South African Library for the Blind, or the Royal National Institute of Blind People (RNIB) in the United Kingdom [49]. Audio documents were also often produced for individual users by assistants specifically hired on a personal basis, or by family members. As recording mediums improved, recorded books followed, becoming available on audio cassette and compact disc, and most recently in the Daisy and Epub3 electronic formats.

2.3.3 Electronic Documents

With the advent of automatic speech synthesis systems and electronic braille technology, blind and print-disabled users were able to gain access to electronic documents. Blind users will typically read an electronic document by means of a screen reader. A screen reader keeps track of textual and semantic information on the screen, and presents it using synthesised speech or refreshable braille terminals [50].

Screen readers are most often controlled via keyboard commands, although modern screen readers (specifically on mobile platforms) can also be controlled via touch screen gestures [51; 52; 53; 54]. Users are able to control which

content should be spoken or brailled, and can also move through content by character, word, sentence or line, using navigational commands provided by the screen reader. Therefore, unlike an audio document, users are able to check the spelling of a word, or to access a document in any order, even when speech is the only medium being used for output. In addition, the information can be filtered so that only information relevant to the current context is presented to the user.

Several techniques are employed by a typical screen reader to collect information about the user interface. For textual user interfaces like the command console in Unix or MS-DOS, the text can be read directly from the terminal or the VGA memory [55]. The situation is more complex for a graphical user interface (GUI) like Microsoft Windows or the X Window System. One method for accessing GUI information is to intercept the information data passed to the graphics driver or the X server [56; 57]. This data is then interpreted and an off-screen model of the GUI is formed. The information contained within this model can then be presented to the user. This approach is also known as screen scraping, because the information is interpreted directly from the layout on the screen.

However, screen scraping has a number of disadvantages. Primarily, the interpretation of information is highly dependent on the application rendering it, as applications may use widely differing screen layouts. For this reason, screen readers that employ screen scraping need to implement interpreters for each supported application. Furthermore, an update to an application may mean that the interpreter for that application must also be updated, as the screen layout might have changed. Finally, the on-screen layout of information is often not the most efficient representation of information for a print-disabled user. A prominent example of such a case is the world wide web, where the information is often presented in two dimensions, while print-disabled users may more effectively read the information linearly.

Newer versions of operating systems provide accessibility application programming interfaces (APIs), which offer more reliable information to screen readers [58; 59; 60]. GUI toolkits or applications subscribe to such an API, and emit events describing when the state of the content on the screen changes, including information about each introduced GUI component such as its role in the interface, the text it contains, and its relationship with other elements. Screen readers and other assistive technologies are able to query this API and receive information about the state of the GUI, organised as a hierarchy of objects representing the GUI (see Figure 2.5 which depicts such a hierarchy as presented by the Microsoft Accessibility Inspect Tool). This method is therefore more reliable since the information exposed to the screen reader can be shaped by the application developer. It should be noted that many screen readers still offer a function whereby the screen layout can be explored in its two-dimensional form, although only the text is typically presented in this view. This allows users to use applications where the accessibility standards

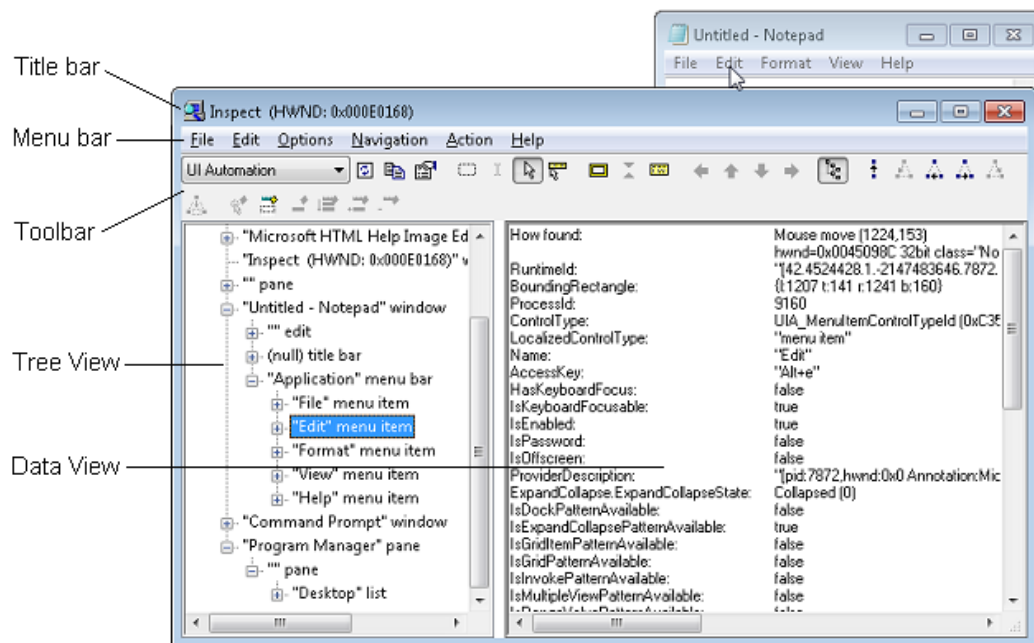


Figure 2.5: The Microsoft Inspect tool showing the accessibility properties of the edit menu in the Notepad application [1].

are poorly implemented. This functionality is also useful for blind application developers, as they are, to some extent, able to verify the visual layout of the interface under development.

Screen readers can be used to read most textual documents, since text can be rendered using synthesised speech or braille. This is eased by the accessibility APIs, as well as standardised character codes used to represent electronic text. Since these codes are separate from the visual representation, they can easily be represented by alternative means. Screen readers are also able to retrieve additional information about the text such as its font and style, as well as information about commonly used document objects like tables, by way of the accessibility API. However, screen readers cannot directly access content that is by nature graphical, for example, charts or diagrams.

For structured file formats like HTML or certain word processor formats, the screen reader will allow the user to navigate the document by jumping to specific sections or elements of a specific type. This is accomplished by navigating the structure of the document. A blind user might for instance move to the next heading within a web page, or the next paragraph in a document. This is necessary since synthesised speech and braille displays present information one line at a time, making it difficult for users of these technologies to efficiently scan for specific information.

Current accessibility standards require that graphical information be described as text which can be read out by screen readers or displayed in braille. An example is alt text that can be added to image elements on the world

wide web, as described in the Web Content Accessibility Guidelines [32], also an ISO standard [34]. These descriptions are traditionally added manually, although recently pattern recognition has been used to generate some descriptions automatically [61; 62]. Textual descriptions of graphical content can also be deduced from the accessibility APIs of some software packages, such as a description of charts in Microsoft Excel [63; 64]. However, a textual description on its own is not suitable for describing graphical content in which the visual shape itself conveys meaning. An example is a depiction of content where spatial relations are important, such as a geographical map or a floor plan. Graphical depictions are also often used to visualise large amounts of data in a succinct way, in which case a detailed description of the diagram would negate the advantage of the representation.

The two main output modalities used by screen readers are speech and braille, although sound icons (sometimes also referred to as earcons) are increasingly used as an alternative for short status indicators.

2.3.3.1 Synthesised Speech

Synthesised speech is one of the output methods used by blind computer users to read electronic text, the other being a refreshable braille display. Compared to braille, synthesised speech has the advantage that it can be implemented in software, requiring no additional hardware devices. The process of converting text as it is represented in computer memory to synthesised speech is known as text-to-speech (TTS) [65]. In its basic form, a TTS system consists of two parts. A front-end converts input text to a representation containing phonemic information, as well as intonation and other speech characteristics. This representation is symbolic information indicating the sounds necessary to form the given words. The back-end converts the phonemic representation to an audible form which can be played over the user's speakers or headphones. Systems based on machine learning approaches may not have a clear separation between front and back ends.

Several methods exist for synthesising speech from a phonemic representation. Concatenated systems synthesise speech by combining segments of audio from a database of prerecorded human speech [66; 67]. These systems have a natural sound, but are not very configurable. Newer approaches generate speech from machine learning models trained from human speech, for example Hidden Markov models or deep neural networks [68]. Other speech synthesising methods are not based on prerecorded speech, but instead use sound synthesis techniques to generate speech from basic sound units or noise. Some examples include spectral synthesis, which combines sinusoidal waves to create more complex sounds [69], and formant synthesis, which uses a noise generator and several tuned resonators to simulate the human vocal system [70]. Speech synthesizers based on technologies like spectral or formant synthesis are still widely used by screen reader users, due to their performance, multilin-



Figure 2.6: Image of a Brailleiant B80 Braille Display [2].

gual support, and configurability. Although text to speech approaches based on Hidden Markov models or deep neural networks are, in theory, just as configurable, this configurability is not often exposed to applications like screen readers. One of the most common configuration changes is the rate of speech, which can be used to read information at a slower or more commonly at a faster pace [71]. Different configurations of the synthesizer can also be used to indicate formatting and other changes in content and context. The Emacspeak system is one example, which uses different synthesizer profiles to indicate syntax and formatting in programming code and other documents [72].

The disadvantage of synthesised speech is that it must be perceived by the sense of hearing, which perceives the information sequentially. When reading braille with the sense of touch or looking at a page with the sense of sight, the user is able to select the information which will be absorbed, by moving the fingers or the eyes. The ears, on the other hand, are unable to selectively absorb information. Therefore, any information delivered by synthesised speech will be perceived by the user, regardless if it is currently important to the user or not. For this reason screen readers provide controls which allows the user to select the information that should be spoken.

2.3.3.2 Refreshable Braille Display

A refreshable braille display is a device that can render braille in real time to a blind user [73]. This is accomplished by the raising or lowering of retractable pins, based on instructions sent from a connected computer or other device. The device is thus able to display dynamic text, like the current time on a clock or the progress of a playing audio track. Figure 2.6 shows an example of a commercially-available refreshable braille display.

Most braille displays are only able to display one line of text at a time, although a few multi-line models have been developed in the past. One of the defining properties of a braille display is the number of braille cells it possesses. A braille display can have up to 85 cells, which enables it to display an entire line of terminal output. It may also be as small as 12 cells, for use with a mobile phone or other portable device. Each cell is made up of eight dots as opposed to the usual six. The additional two dots are used to display the cursor or

to indicate a change in format. As described in Section 2.3.1, a standard of braille, called eight-dot computer braille, has been developed which makes use of the additional two dots. Using this standard, each ASCII character can be displayed using a single braille cell.

2.4 Summary and Conclusion

This chapter has introduced the current state of the art for document accessibility. Although universal design and modern teaching practices have resulted in a more inclusive experience for print-disabled readers, the accessibility of published material remains an impediment. Digital publishing holds much potential for print-disabled readers, through the use of born accessible formats. Unfortunately, most technical work is still published as inaccessible PDF documents.

Braille is a script that can be read by touch, and is one of the traditional accessible mediums for blind readers. It consists of different combinations and arrangement of up to six or sometimes eight dots, which can be used to represent the alphabet, numbers, symbols, and other indicators. Through the use of special linear notation, complex material like mathematics or music can be conveyed in this way. Hardcopy braille can be bulky, and is therefore not very portable. Audio is another traditional accessible medium, and is typically produced by the aid of a human narrator reading the printed text. The Daisy and Epub standards can be used to augment the audio stream with navigational structure, but navigation at the symbol level is not possible.

Electronic formats can be read using a computer equipped with a screen reader. A screen reader is a software program that conveys a user interface via speech or braille. This is accomplished using either screen scraping or, more commonly, platform-specific APIs developed for accessibility purposes. Speech is produced using text to speech technology, and is sometimes sped up to allow quicker listening by experienced users. Braille is produced via hardware refreshable braille terminals, which can typically display one line at a time. These two mediums can only present text, and therefore all graphical information must first be converted to a textual description. The following chapter gives an overview of accessibility to graphical material, such as maps, graphs, and other diagrammatic content.

Chapter 3

Accessibility of Graphical Material

Diagrams can be loosely categorised into two classes: semantic and pictorial. We define semantic diagrams as those which serve as a visual representation of data or a concept, which is independent of the diagrammatic representation. Pictorial diagrams, on the other hand, are diagrams representing spatial information. Semantic diagrams can usually be represented using a symbolic alternative. A bar chart may, for example, be represented as a textual table, and a graph as a list of vertices and edges. However, a pictorial diagram, for example, the contours of a continent, would be very difficult to represent in a symbolic form. A symbolic representation would often either capture too little detail, or if adequate detail were captured, it would be too long to be useful. Figure 3.1 depicts a semantic diagram (a graph), while Figure 3.2 represents the same diagram as semantic textual information. Figure 3.3 is an example of a pictorial diagram, a map of South Africa.

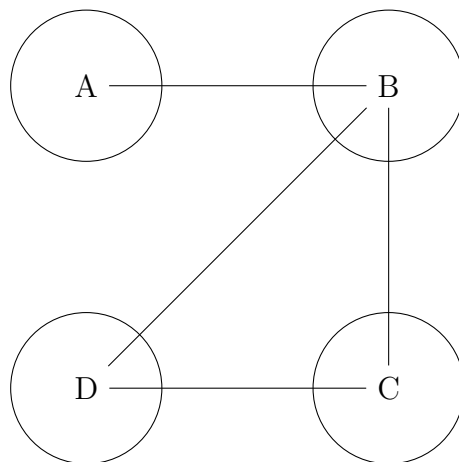


Figure 3.1: Example of a semantic diagram: visual representation of a graph.

A: B
B: A, C, D
C: B, D
D: B, C

Figure 3.2: The graph of Figure 3.1, represented as a textual table of vertices and edges.

While semantic diagrams can therefore be made accessible by providing the textual information to a screen reader, this information is usually not provided and cannot reliably be automatically inferred.

This chapter gives an overview of approaches to graphical accessibility. Section 3.1 describes tactile diagrams, which can be explored by touch. Section 3.2 explains the use of the “alt tag” and other mechanisms which allow textual descriptions to be associated with images in documents. Section 3.3 describes sensory substitution approaches which map graphical information to modalities perceivable by another sense. Section 3.4 provides some examples of systems that can be used to explore domain-specific diagrams with logical components. Section 3.5 discusses accessible data visualisation, which allows blind readers to visualise data in an accessible format. Finally, Section 3.6 introduces the approach to graphical accessibility proposed in this study, and how it addresses what we believe to be the limitations of the existing approaches.

3.1 Tactile Diagrams

Tactile diagrams are most often conveyed to blind users by encoding lines and shapes into the physical shape of a surface. They can be produced by hand using cardboard and other materials, but can also be produced by braille embossers equipped for this task [17], or printed as ink on material which expands when heat is subsequently applied [74]. A few braille displays are also available which can render tactile diagrams using a 2-dimensional array of pins, each which can move up and down independently [75]. Unlike usual braille displays, which can only display one line of braille at a time, these displays contain many more rows of pins, and the pins are often not separated into braille cells.

Tactile diagrams are often produced from diagrams in published textbooks and other study material by a process of manual editing by a trained translator before subsequent conversion into tactile form [18]. This editing is necessary, since the perception of touch is estimated to have a bandwidth 500 times lower than that of vision [76]. Therefore, perceiving information via touch necessitates different strategies for accessing and localising information, and



Figure 3.3: Example of a pictorial diagram which cannot be completely represented as text: a map of South Africa.

may be more vulnerable to systematic distortion [77]. Conversion involves the removal of unnecessary information, as well as the conversion of visual indicators like colour into tactile equivalents like different tactile patterns. Labels and other text are also replaced by braille, which may require resizing the diagram, since braille is longer than printed text.

3.1.1 Graphics-Capable Braille Embossers

The Tiger embosser, manufactured by ViewPlus, is an example of a braille embosser which is able to emboss tactile drawings using braille [17]. It contains a printing head with two separate pins, one to produce normal braille, and one to produce tactile drawings. The drawing pin can be programmatically raised

and lowered to seven settings, allowing the embosser to produce greyscaled pixels. This is useful for drawing one shape on top of another, or to indicate colour.

The Tiger embosser functions by taking standard printing instructions from the operating system, and then embossing the resulting pixel values using its printing pins. Pre-processing is often necessary to transform the diagram into a form that prints well in braille. Printing a diagram that is too bright, for example, would overuse the highest setting of the drawing pin, which could damage the paper. Many diagrams are also difficult to follow when printed unaltered in a tactile form, as the resolution of the tactile sense is much lower than the visual sense. ViewPlus provides special fonts to replace the printed text with braille on the diagram, and also functions to resize and edit the diagram to accommodate the different dimensions of the braille text.

3.1.2 Multiline Braille Displays

The Tactisplay Table by Tactisplay Corp is an example of a braille display for rendering tactile diagrams using several rows of separately moving pins [75]. Each pin can be set to several heights, which can be used to reproduce some greyscale values.

The Hyperbraille is a braille display capable of displaying both braille letters and tactile graphics [78; 79]. It comprises a grid of dots arranged equidistantly from each other. The grid contains 60 lines, each with 120 dots. Braille letters are formed by raising combinations of dots in groupings of six, with the dots between adjacent letters and lines lowered. Blind people need some practice to read with this display, since the lines are closer together than on a conventional braille page, while the cells are further apart.

The Hyperbraille Display is capable of displaying graphics using more than one translation method. One method involves treating each dot as a pixel, and thus forming images by raising the corresponding pins for each pixel of the image. Another method involves the use of shapes to denote different colours, with each shape comprising not more than four dots. The image is then formed by raising the specific shape for each coloured pixel arranged in a roster. The Hyperbraille Display contains controls for panning the data displayed on the display, and for zooming graphical content in or out. The display itself is also touch sensitive, allowing the user to execute commands by pressing on certain dots.

Prescher *et al.* tested the efficacy of the Hyperbraille when reading braille, compared to other braille reading modalities [80]. They found that the fastest braille reading rate was obtained with conventional braille paper, followed by traditional braille displays, with the Hyperbraille system having the lowest reading rate. They ascribe this finding to the way in which braille is displayed on the Hyperbraille system, by lowering dots between cells and lines.

One challenge to the general use of devices like those above by blind users is their significant cost. These devices are usually only available at large institutions like universities or schools, and rarely to individuals. They are therefore not readily available as aids for personal study or research, unless the user is funded by a large institution or has the independent means to obtain such a device. There are projects to develop low-cost alternatives to these devices. One example is the Graphiti multi-line braille display developed by Orbit Research [81]. However, also this device, which is not yet commercially available, will cost more than a moderately priced personal computer.

3.1.3 3D Printing

One solution to the high cost of tactile diagram production may be the use of 3D printing technology to produce tactile diagrams. As 3D printers are used by the general consumer, they have a much larger market than a device useful only to blind users. Hence it is probable that low-cost 3D printers will become available in the near future. However, although blind people are able to perform 3D printing independently [82], 3D printing technology shares a challenge with tactile diagram embossers, which is that the user is unable to verify the image or shape before it is printed without access to specialised hardware [83]. This is especially important, because the shape of a printed object influences its structural integrity and hence the accuracy with which it portrays the source image. Furthermore, the user would be unaware if the printed object differs from the source diagram.

3.1.4 Audio Augmented Tactile Diagrams

A limitation of tactile technology is the low resolution of human touch receptors. Tactile images must therefore be simplified before they can be conveyed to a blind reader. For diagrams containing a large amount of data, a larger tactile surface must be used, or the diagram must be converted by parts. This adds to the already significant cost of tactile conversion.

In addition, tactile diagrams cannot always convey the same information as sight, since the tactile sense is not trained to infer information implied by shading and other attributes specific to the visual sense. This is particularly relevant when 3D models are conveyed as tactile diagrams.

One solution to increase the effectiveness of low resolution tactile diagrams that has been reported on in the literature is the addition of meta-data that can be attached to regions of a tactile diagram. The user is then able to explore the tactile diagram with the sense of touch, but is also able to gain information on a specific region by a defined trigger gesture, such as a double tap on the surface. Additional information, like a description of the touched region, or the text of a label, is then conveyed to the user via synthesised speech or an attached braille display.

Such an approach was first introduced by Parkes [84], and also described by Löttsch [85], and Landau *et al* [86]. A modern example of this technology is the Iveo system from ViewPlus [87; 88]. The Iveo system consists of software paired with a touch pad. A diagram is loaded by the software, and the user is able to explore details of any element in the diagram using the touch pad. However, this requires the diagram elements to be tagged beforehand. The system is able to extract tags from SVG drawings, and the software also provides an interface for sighted translators to add tags to elements.

Low-vision users are able to read the diagram on the screen, and tap on any element that is too small for them to see clearly. Blind users must print the diagram using a ViewPlus embosser, and place the paper on the touch pad. The blind user is then able to explore the diagram by reading the tactile braille, and tapping on the touch pad where the braille is unclear or too dense. The user is also able to zoom into a specific element or area, in which case the blind user would have to emboss an additional image. However, the software does store every zoom position for later recall, allowing the blind user to reuse already embossed drawings.

A diagram can also be augmented by non-speech sounds. This approach was explored by Ramloll and Brewster [89], who described the TouchMelody system. TouchMelody augments tactile diagrams by adding a 3D soundscape generated from the user's finger motion on the diagram. The sound generation algorithm is based on observed behaviour of users exploring tactile diagrams, in particular the strategy of using one hand to observe a region, and the other to explore around it. Two motion sensors are fixed to the user's index fingers, the output of which is fed to the algorithm generating the soundscape. One finger is designated as the observer and therefore acts as the central point of the soundscape. The second finger acts as the explorer, and is used to derive a sound source in a virtual acoustic space, with the perceived position of the sound source based on the relative position of the second finger with regards to the first. The relative vertical position of the exploring finger is also mapped to pitch, resulting in a sound mapping similar to other pitch-based sonification approaches like the vOICe [20] and the Optophone [90]. Finally, different musical instruments are used to generate the sound, based on whether the exploring finger is situated above or below the observing finger. It must be borne in mind that the approaches described above require access to physical tactile diagrams, and therefore share some of the same limitations in terms of hardware cost and portability.

3.1.5 Tactile Diagrams on Consumer Devices

Various consumer devices have been considered as affordable alternatives for providing tactile graphical information to blind users. For example, a number of studies have considered force feedback devices like the PHANTOM haptic device [91; 92; 93; 94]. Force feedback devices consist of a probe which can be

grasped and manipulated by the user within one or more dimensions. A physical tactile space can be simulated by the application of force feedback, and therefore resistance, when the probe is moved to pre-defined locations. Many force feedback devices allow movement within two-dimensions, like joysticks commonly found on game controllers. Devices like the PHANToM allow movement within three dimensions, and can therefore be used to simulate surfaces with grooves or other features which can mimic a tactile diagram.

With the emergence of touch screens in mobile devices, the pairing of this touch input with the vibrating motor integrated into these devices as a means of haptic feedback has also been considered [95; 96; 97]. Other studies have used friction technology like the TeslaTouch [98]. The TeslaTouch consist of a layer on a touch screen, which is able to apply static (electrovibration) to the user's finger as it is moved over the screen. In this way, the friction of the user's finger movement can be controlled, which can be used to simulate the perception of tactile features.

It should be kept in mind that, unlike the Optacon and traditional tactile diagrams, all of these approaches provide only one stimulus to the user, and hence allows only one contact point for every area touched. It has been shown that multiple contact points are necessary for effective path tracing, as the user is then able to judge the curvature of the path and thereby judge the direction to move to continue tracing [99]. Actuators placed on the user's fingers have been proposed as an approach for allowing multiple stimuli [100]. However, similar to existing tactile technology, this approach requires using specially constructed hardware. This limitation can also be mitigated by using differing vibration patterns or multimodal feedback for distinct features of the diagram, but this requires a diagram with semantic metadata [96; 97]. Multimodal approaches are further discussed in Section 3.3.3. Audio is arguably a better approach for conveying multiple contact points simultaneously, since both the finger location and the shape of the image under the finger could be encoded in the audio signal, for example, by using a different frequency for each contact point.

3.2 Described Diagrams

Since text-to-speech is commonly used by blind and print-disabled users to read text, it has also been used to read textual elements in diagrams. One example, the Iveo system, has already been described in Section 3.1. The World Wide Web Consortium, as part of the web content accessibility guidelines, specifies the use of the "alt" tag to provide alternative text descriptions for images. Modern user interface toolkits like the Android accessibility API and the Apple accessibility API provide a similar functionality. When descriptions are added to graphical information using these specifications, screen reader users are able to read the descriptions via text-to-speech or in braille. However, graphical

information in documents and on the web are often not described.

Recently, pattern-based approaches have been studied which allow descriptions to be automatically generated for graphical information [62; 61]. However, descriptions of graphical information are often dependent on the context and the document in which they appear [101]. In addition, textual descriptions impose a linear representation on graphical content, which is inherently two-dimensional. Therefore, many of the advantages of the two-dimensional representation are lost, including effectively identifying relations, and comparing different parts of the diagram [102; 103]. Touch screen interaction may allow the exploration of relations of components within an image [104]. However, even when relations can be easily explored, textual descriptions are unable to convey pictorial information such as the shape of a sound wave, or a geometric shape with which the reader is not familiar.

The Web Accessibility Initiative: Accessible Rich Internet Applications (WAI-ARIA) specification also allows descriptions to be attached to parts of a diagram or interactive visualisation created with web-based technologies [33]. Screen reader users are then able to explore the components of a diagram or visualisation. However, in practice, web-based diagrams and visualisations are not developed in this way, and therefore remain inaccessible to screen reader users. Even when diagrams and visualisations are marked up, screen reader users often find them difficult to use [105].

Other work has considered using non-verbal audio as a means to convey graphical information. Much of this work has been performed under the umbrella of audio-visual sensory substitution. Sensory substitution is when one sense is substituted by another, by transforming the information intended for the original sense to a form perceivable by the substitute sense. In the case of audio-visual sensory substitution, visual information is mapped to a sonic representation, often by computing a soundscape from the pixels of the original image.

3.3 Sensory Substitution

Much of the research for making visual information accessible to blind readers has been performed under the umbrella of sensory substitution. Sensory substitution is the use of one sense to interpret information normally perceived by another. This is usually made possible by technology that converts the information destined for one sense into information that can be interpreted by another. It is therefore a viable means to provide access to specific sensory data for people who lack that particular sense. An example is the use of vibration to convey auditory information that can be understood by people who are deaf or hard of hearing [106]. Sensory substitution can also be used to convey sensory perceptions that are remote from the user, such as when performing robotic surgery [107].

By nature, sensory substitution approaches provide direct access to the graphical shape of the content. They are therefore most useful as solutions for accessing pictorial diagrams in a non-visual way. Many diagrams can be made accessible more efficiently by finding an alternative representation for the semantic content. However, a pictorial representation of a semantic diagram should also be accessible to blind users, for purposes of education. A blind learner should be able to explore the graphical representation of data, for instance, a bar chart or the sighted musical score, in order to allow effective communication with sighted colleagues. A blind mathematician or engineer should also be able to check their work (which might contain graphical representations of data) before it is published to a sighted audience. Finally, many articles, papers, and books are published with semantic diagrams representing data, but without the data the diagrams were created from.

3.3.1 Tactile-Visual Sensory Substitution

The following paragraphs describe work conducted under the banner of tactile-visual sensory substitution. In tactile-visual sensory substitution, visual information is mapped to tactile stimuli such as vibration or electrical impulses, to be interpreted by the sense of touch.

Optacon Tactile-visual sensory substitution devices have been extensively studied in the literature, with the first example being described by [108]. They proposed a system comprising a grid of vibrating actuators placed on the subject's back, with each actuator mapped to a pixel from the source image.

Vibrating actuators were also utilised by the commercially marketed Optacon described by Linvill *et al.* [109]. The Optacon consisted of a hardware probe incorporating photosensors which is moved across a line of printed text by the user. The hand-held probe is paired with a tactile screen comprising six columns of 26 vibrating pins, each driven by a piezoelectric actuator. As the probe is moved over the line of text, the corresponding actuator of each pixel from the probe is activated, resulting in a tactile image of the area under the probe. A switch on the device can be used to set whether dark or bright parts of the image are converted. Although originally designed to read printed text, some users have successfully used the Optacon to study some diagrammatic material, including circuit diagrams.

Brainport More recently, Bach-Y-Rita *et al.* have described a tactile visual sensory substitution device applied to the tongue, using a grid of electrodes to represent the source image [110; 111]. Called the Brainport, the device consists of two components. A small oral component is placed in the mouth, in contact with the top of the tongue. This resembles a square plate and may be fixed to a dental retainer or similar device. The underside of the oral component

contains a grid of electrodes, which are used to stimulate the nerve endings of the tongue. The oral component is connected with a cable to a base unit worn on the belt. The base unit performs preprocessing on the data sent to the oral component, and also supplies the power needed to drive the electrodes.

Much like the Optacon, the Brainport can be used as a sensory substitution device for vision replacement [111]. Images are received from a camera or other source, and scaled to the number of electrodes contained on the underside of the oral device. Each electrode is then activated according to the value of the corresponding pixel of the image. User evaluations were done with blind users of the Brainport. Subjects were able to perform simple tasks, like reaching for an object in the camera's view, or catching a ball rolled towards them. They were also able to navigate an indoor maze, without bumping into obstacles. However, subjects were unable to use the Brainport as a navigational aid outdoors near car traffic because the scenes were too complex to be interpreted.

Tactile Gloves Meers *et al.* introduced the electro-neural vision system (ENVS), a system for providing tactile-visual sensory substitution to the user's fingers by way of transcutaneous electro-neural stimulation [112]. The ENVS system consists of a pair of head-mounted stereo cameras, a software algorithm, a transcutaneous electro-neural stimulation (TENS) unit, and a pair of gloves fitted with electrodes at each finger tip. The ENVS functions by constructing a disparity depth map of the environment from the stereo cameras. The algorithm uses the depth map to calculate the intensity of electro-stimulation for each finger tip, which is then delivered to the electrodes in the gloves via the TENS unit. The user is expected to hold their fingers in front of their abdomen, and by feeling the stimulation of each finger, determine the approximate shape of the 3-dimensional environment.

The ENVS was unfortunately not evaluated by blind candidates, but was tested in the laboratory by five blindfolded sighted participants. Participants were able to find and navigate around obstacles. After familiarity with the laboratory environment, participants were also able to describe their location after being deliberately disoriented on a swivel chair which was spun around and moved about. The ENVS was designed for the identification of three-dimensional objects in the immediate environment, to aid with physical navigation. A similar approach employing a glove for tactile-visual sensory substitution was described by Manshad *et al.* for reading graphs [113]. Paired with a touch screen to track the positions of the user's fingers, the glove described by Manshad *et al.* contained an actuator on each finger, which vibrated with increasing intensity as the user approached the curve of the graph on the touch screen.

The FeelX system described by Soviak *et al.* also made use of a glove for tactile-visual sensory substitution [114], and was specifically evaluated for reading diagrammatic content. Soviak *et al.* developed a glove prototype

which could be fitted to the user's right hand. The glove was designed to move around on a table top, much like a typical computer mouse. Four of the fingers incorporated tactile braille cells, while the thumb could be used to stabilise the hand. User evaluations were performed by analysing the functionality of the glove when users were able to use one finger for exploration compared to four fingers. A comparison was also made between two different alignments of the haptic field, that is, the alignment of the braille cells relative to the grid of pixels representing the image. In the finger aligned condition, the haptic representation was always aligned with the user's finger, while in the field aligned condition the haptic representation was aligned with the alignment of the pixel grid. Users were asked to identify horizontal and vertical lines, triangles, rectangles, and circles using combinations of the two test cases, for a total of four permutations for each shape. According to the results, users were more accurate when identifying shapes using four fingers, for both alignments of the haptic field. Users had slightly higher accuracy when reading shapes with the field aligned haptic field, although this result was not statistically significant.

Feedback gathered by Soviak *et al.* after the user evaluations suggested that users thought the FeelX glove could be a useful alternative to current approaches for tactile diagrams, due to the long conversion time to produce traditional tactile diagrams. One of the most requested improvements of the FeelX system by candidates was the ability to spread the fingers wider to enable exploration of a larger area when desired. The initial design of the FeelX system limited lateral finger movement to about a centimetre.

Conclusions As with tactile diagrams, the main disadvantage of tactile-visual sensory substitution systems is its reliance on special purpose hardware. Special assistive hardware imposes a financial burden on the print-disabled user, and is also less portable than a medium that can be realised on already existing consumer devices. Audio sensory substitution is an example of such a medium as it only requires a digital to analogue converter for producing sound, which is available on virtually any consumer computing device.

3.3.2 Audio-Visual Sensory Substitution

In audio-visual sensory substitution, information regarding an image is encoded as sound, which a blind or print-disabled reader can interpret. The following paragraphs describe previous work based on this approach.

Optophone One of the first implementations of audio-visual sensory substitution was the Optophone, as described in [90]. This electronic device divided the light reflected from a vertical slice of the input image into discrete vertical segments. Moving from bottom to top, each segment was mapped to an

increasingly higher frequency with the help of a selenium photosensor. In this way, a chord was emitted for each vertical segment, where bright parts were sonified at a frequency corresponding to the vertical position while dark parts were silent. Users were expected to read letters by listening for the missing frequencies. Soon after the introduction of the Optophone, it was modified so that dark segments were sonified and bright segments silenced [115]. The optophone used a binary input representation: either a segment was considered dark and hence sonified, or it was considered light and not sonified. It could not convey a measure of intensity or greyscale.

The vOICe Almost 80 years later, a method similar to that employed by the Optophone was encoded as a computer algorithm [20]. This algorithm transforms a visual picture to a soundscape by mapping each pixel to a sinusoid, where the frequency (and thus the pitch) of the sinusoid is determined by the vertical position of the pixel (a higher position translates to a higher pitch), and brightness is translated to amplitude or loudness. Each column of the originating image is therefore transformed to a tone chord, and the chords are played in sequence from left to right, to produce the soundscape. In later versions of the vOICe, stereo audio was also taken advantage of, placing each column in the stereo field according to the position of the matching column in the source image [21]. Various experiments have been performed to evaluate the effectiveness of the vOICe algorithm, for example, by testing the ability of users to reach for an object [116], and to follow an obstacle course [117]. It has also been shown that reading letters with the vOICe activates the so-called word-form area of the brain [118].

The evaluation performed by Brown *et al.* [117] used 18 blindfolded sighted participants. The users were asked to complete tasks evaluating object recognition, as well as identifying object locations. The evaluation comprised three stages. In the first stage, users had to identify four simple shapes: a diamond, square, circle, and vertical rectangle. The shapes were presented to the users in the form of sounds representing the image rendered on a virtual rectangular canvas. Each shape could be either an outline or solid, large or small, and could appear at three possible vertical positions: top, middle, or bottom. This served as both a training stage and as part of the evaluation.

The second stage comprised the identification of real objects and their locations, when placed on a 3-by-3 grid on the surface of a table. The images were captured in real time by a camera that was either worn on the head of the subject or held in the hand. Common everyday physical objects were used, like a cup, a USB cable and a shoe. The third stage required users to negotiate an obstacle course, using only the vOICe as navigation aid. The course consisted of a hallway with four obstacles placed irregularly down the middle. Users were required to slalom between the obstacles, four times in each direction, for a total of eight attempts. The results of the study suggest that, for novice

users, the audio characteristics of the signal are more important than visual associations. They also found that holding the camera in the hand is more effective for object identification, while locating objects was easier with the camera mounted on the head.

Proulx *et al.* analysed the performance of the vOICe when integrated into a head-mounted device and used by 13 sighted but blindfolded subjects [116]. Of the 13 subjects, 10 wore a blindfold only for the duration of the evaluation, while three wore a blindfold also in the month before the evaluation. One subject from the group of 10 and one from the group of three were afforded the opportunity to familiarise themselves with the device during the month before the evaluation. A further subject from the group of 10 and another from the group of three was not given access to the device at all. These subjects were thus subjected to the test without any technological aid. The remaining subjects were presented with the device immediately before the evaluation, and therefore had no opportunity for prior familiarisation.

The test consisted of four sessions. In the first and second experiments, the users were asked to identify the position of a light-emitting diode (LED) among a series mounted on a panel placed in front of them. For the first experiment, 18 LEDs were arranged in a horizontal line, while for the second experiment they were arranged along the edges of a hexagon, with 164 possible locations. However, without informing the subjects, only six LEDs were in fact used for the second experiment, and each LED was used twice. During the third experiment, the task was to reach for, and grasp an object on a table. Different objects with different sizes were used, allowing the authors to evaluate the pose of the hand in anticipation of grasping the object.

Proulx *et al.* found that the subjects who were not given the device during the evaluation responded quickly but had a success rate corresponding to chance. Subjects who used the device during the evaluation performed more slowly, but exhibited higher accuracy. Furthermore, this accuracy improved during the course of the evaluation. Finally, subjects who had been allowed to familiarise themselves with the device in the weeks before the evaluation exhibited even higher accuracy and response times that improved during the evaluation to such an extent that they were almost on par with the group that had no access to the device at all.

Other evaluations performed with the vOICe include those by Striem-Amit *et al.* [118] who found that rendering short soundscapes of words to blind participants activates the so-called word form area of the brain. Although the article is written from a neurological point of view, it does illustrate that the vOICe can be applied to the task of reading text. The device developed by Cronly-Dillon *et al.* uses a similar conversion algorithm to the vOICe, but simplifies the source image into a line drawing before the conversion process, resulting in a more musical soundscape [119]. However, like the case with the Eye-Music algorithm described in the following section, the disadvantage of such an approach is that details of the original image may become lost.

Eye-Music The vOICE was also the basis for the Eye-Music algorithm described by Abboud *et al.* [23], which used musical instruments instead of sinusoidal components to represent pixels. The Eye-Music algorithm added the capability to represent colour in the rendered soundscape, by mapping each colour in the source image to a unique instrument.

It must be born in mind that the Eye-Music soundscape is more complex than the original vOICE soundscape, since in the case of the vOICE, each pixel in a column maps to exactly one frequency, while in the case of Eye-Music, each pixel maps to an instrument with its own unique timbre consisting of several harmonics. This attribute of the Eye-Music image to sound algorithm necessitates that the individual sounds are spaced wider apart with respect to frequency. Therefore, the Eye-Music algorithm maps the vertical height of pixels to frequency based on the pentatonic scale. However, a disadvantage of such a sound mapping is that gradual curves are more difficult to express. Consider, for example, the case of two ascending diagonal lines, the first unbroken, and the second dotted. With the vOICE, the first will sound like a tone with rising pitch, while the second will sound like beeps with increasing frequency. However, with the Eye-Music algorithm, they will sound similar.

The Eye-Music algorithm was evaluated in the context of a virtual environment [120]. Blind candidates were asked to choose between multiple doors based on the features of the doors in an initial task, and their surroundings in a subsequent task. Candidates were able to identify doors with an accuracy of 95% and 97% for both tasks respectively. Candidates were also observed when navigating a street with buildings and other landmarks, and guided through exploring the environment. Although the virtual environments were specifically produced for these experiments and therefore not as complicated as mainstream virtual environments, the results nevertheless suggest that an audio-visual sensory substitution algorithm like Eye-Music could be a valuable tool for enabling accessibility of virtual environments.

See ColOr The approach of mapping parts of an image to musical instruments based on colour was also utilised by the See ColOr system described by Gomez *et al.* [121]. See ColOr was specifically designed as an audio-visual sensory substitution algorithm for conveying the physical environment. It utilises a 3D camera to capture the environment, and is able to convey colour as well as depth. In addition to an image-to-sound mapping, Gomez *et al.* also explored the use of pattern matching to automatically identify objects, which is conveyed to the user via verbal announcements along with the sonified output.

See ColOr allows the user to explore the environment using two modes: In local mode, See ColOr allows blind users to receive continuous sonification of the environment, by mapping the camera view to 25 possible points in the stereo field, sonified using musical instruments to convey the colour of the image at the particular position. The distance of each particular point

(determined by the 3D camera) is conveyed by the duration of the emitted sound, with longer duration's denoting greater distances. In global mode, the user is able to explore the environment by moving their fingers over a tablet connected to the See ColOr system. Tapping a finger on a specific part of the tablet sonifies the camera view at the position corresponding to the position of the finger on the screen.

In addition to sensory substitution, See ColOr also utilises pattern matching to augment the sonification with verbal announcements of identified objects. According to the authors, this functionality increases the effectiveness of sensory substitution by allowing users to identify common objects much quicker than would be possible with a direct image-to-sound mapping.

Gomez *et al.* conducted four experiments with the See ColOr algorithm, to evaluate its effectiveness in conveying the physical environment. In the first experiment, users were asked to identify and turn towards a red object, while sitting on a swivel chair. Users were also required to walk to the object. A second experiment required users to move towards a wall without making physical contact. The third experiment tested users' capability to find a person within a room, while the fourth required users to find dropped objects without touching them. Experiments one and two focused on the image-to-sound mapping utilised by the See ColOr approach, while experiments three and four also tested the pattern-based object identification functionality.

Like the Eye-Music algorithm, See ColOr uses complex sounds to convey parts of an image, and is therefore less useful for conveying fine detail about a shape, as would be necessary for a diagram. However, such an approach has merit when used to convey the physical environment, which is the use case described by Gomez *et al.*. See ColOr also conveys distance and colour, which can be used to quickly identify areas of interest from a live camera view. Although colour is important when reading diagrams, distance information is not, and colour is only useful when the shape can also be explored. However, the use of verbal announcements to convey identified features is an approach that can offer valuable information to blind readers, also in the context of a diagram.

Audio Screen Audio screen [122] is an experimental extension for the popular NVDA screen reader [123] for Windows desktops. It allows users to sonify the area around the mouse pointer, or a finger on the touch screen, using the vOICE algorithm. Users can also place two fingers on the screen, in which case the rectangle bounded by the two fingers (as upper left and lower right corners) are sonified. The latter gesture is also available within the vOICE application for mobile devices running Android. However, to our knowledge, Audio Screen has not been formally evaluated in the literature. It should also be kept in mind that the parameters of the algorithm, as implemented by the Audio Screen extension, are independent of the location of the image content

being sonified on the screen. Put differently, sonifying a line at the bottom of the screen, and sonifying a line at the top, will sound the same.

Audio screen also includes a mode for sonifying graphical information based on colour. In this mode, the pitch of the tone is not determined by the vertical position of the corresponding pixel in the source image, but by the hue of the pixel's colour. The amplitude of the tone is determined by the brightness of the colour, while saturation is conveyed by mixing the tone with noise in an amount relative to the saturation. This mode is therefore less useful for reading shape information.

PSVA A different approach to sensory substitution was followed by Capelle *et al.* [22] in the design of the PSVA (prosthesis substituting vision for audition). The PSVA contains an artificial model simulating a retina. Pixels from a defined portion of an image are mapped to frequencies, all of which are played simultaneously. The algorithm allocates a higher pixel density towards the centre of the rendered part of the image than at the edges, thereby mimicking the function of the retina in the human eye. Starting from the bottom-left, each pixel in each line of the source image is mapped to a tone with increasing frequency, with the rightmost pixel in the line approximately one tone higher in pitch than the leftmost pixel. Moving from bottom to top, each pixel in a column is mapped to a tone that is one tone higher than the pixel below it.

Arno *et al.* evaluated the PSVA with six blind and six sighted but blind-folded participants [124]. PSVA was implemented as a computer algorithm and used to render the area around the mouse pointer. Users were asked to describe line drawings by moving the pointer around the drawing, using a pen on a graphics tablet to control the location of the pointer. The images used by Arno *et al.* consisted of single-pixel dots and lines with various orientations. Users were required to reproduce the rendered image by appropriately placing aluminium bars and circles on a surface in front of them. The results of this study indicate that all participants were able to identify graphical elements present in the drawings, but blind participants were able to explore the drawings more quickly. It was also found that the ability to explore interactively by moving the pointer increased the effectiveness of the algorithm.

3.3.3 Multimodal Approaches

A number of approaches have been described which utilise multiple output modalities, for example, vibration augmented with sound. An advantage of a multimodal approach is that different aspects of a diagram may be conveyed via separate output modalities, to allow easier identification of diagram features. However, such an approach requires the information source to be augmented with metadata beforehand, to indicate the different aspects of the diagram and how it should be presented.

Klatzky *et al* compared three approaches to touch-screen assisted diagram reading, namely vibration output, sound output, and a combination of the two [96]. In the first case, vibration is used to signal when a line is touched by the user's finger on the touch-screen. A specific vibration pattern is used to indicate the point where two lines meet. When using sound output, a sound is played when a line is touched, with stereo panning indicating the horizontal position of the finger on the screen and pitch the vertical position. A different sound is used to denote an intersection of lines.

The study found the main limitation of vibration output to be that the source of the stimulation is not well coordinated with the location being explored. The vibration is produced at a central point not spatially linked to the touch location, and thus cannot convey location or direction. This leads to the loss of contact with traced lines and other elements, as the user does not have advance warning of an approaching turning point or edge. To limit this problem, lines can be thickened, but this reduces resolution. Although audio as an output medium is also not directly linked to the input surface, it has the attractive property that the characteristics of the touched area can be encoded, for example, by using pitch to represent vertical position, as is done by the Optophone and the vOICe. Klatzky *et al.* conclude their study with a system that offers both vibratory feedback and sound icons to describe drawings consisting of continuous lines, dashed lines and dots. Users were allowed to explore the drawing using one finger, while receiving either vibration, audio, or a combination of the two as output. The results showed that users preferred audio output, or audio combined with vibration, over vibration alone.

Klatzky *et al.* advocated for more research to be performed using touch screens combined with sensory substitution, and emphasised that approaches should be as general as possible. Both Klatzky *et al.* and Brown *et al.* noted an increase in effectiveness when the focus, that is, the part of the image being rendered, is controllable by the user [117; 96]. One motivation for the research we describe in this work is the belief that more effective exploration of a displayed image can be achieved by allowing a greater degree of user interaction.

Some guidelines for representing touch screen-based graphics conveyed via a consumer vibratory actuator are described in [125], where three usability studies were conducted with 46 blind and visually impaired readers. The authors identified the following parameters: a line should be 1 mm in width, lines should be separated no more than 4 mm apart, and the minimum angular separation between oriented lines (the cord length) should be 4 mm. A prototype based on these findings was described in [97], which also incorporated sound feedback to indicate important features of the diagram. The prototype was evaluated by 22 blind and visually impaired readers, who were asked to explore diagrams using the proposed system, or as traditional embossed tactile diagrams. According to the results, readers were able to identify multimodal diagrams with comparable accuracy to embossed diagrams. However, sim-

ilar to the system described by Klatzky *et al*, diagrams are required to be augmented with metadata beforehand.

3.4 Domain Specific Diagram Explorers

A number of systems have been proposed allowing users to explore diagrams by separating the content into logical components. With this approach, users are able to explore relational data by navigating between the logical components of the diagram, similar to navigating through a graphical user interface. The TDUB system allows users to explore circuit diagrams, UML models, and architectural models by navigating through a tree structure containing a hierarchical view of the diagram's structure [126]. Users are also able to deduce the spacial locations of components by way of spacial audio queues.

Roth *et al.* suggested an extension to the SVG vector-based image format for adding logical tags to diagram components [127]. They also describe a system for interpreting drawings enriched with these logical tags. Their system uses text-to-speech to convey a component's general overview, type and shape, and haptic feedback to convey a component's location within the image.

The PLUMB system was designed as a solution allowing blind users to read, edit, and create relational graphs [128]. Users are able to use the PLUMB system with either the keyboard, or a drawing tablet with a pen. Using the drawing tablet with a pen as input modality, a blind user is able to trace the layout of the graph on the screen, receiving information about vertices and edges. However, Calder *et al.* note that this input modality, while effective for exploring the geometric shape of the graph, is not effective for finding and following connections between nodes. This is because following edges using this method is dependent on accurate hand movements by the user. The keyboard input modality allows users to explore nodes and edges with keystrokes, and receive information regarding the connections from each node. The keyboard modality is therefore more effective for exploring relations between nodes, by sacrificing geometric information.

A novel contribution of the PLUMB system for graph exploration is its search functionality. It allows users to search for specific nodes, by entering a search query. Users are then able to tap the pen anywhere on the screen, after which the desired node will be placed at the tapped point on the screen. It should be noted that the PLUMB system, by utilising a pen for geometric exploration, allows only one point of contact for path tracing. As it has been shown that multiple contact points increase the effectiveness of path tracing [99], it is expected that a geometric exploration modality for PLUMB which allows more contact points, may also increase the effectiveness for tracing relationships with this modality.

Component-based exploration systems like the ones described above require an image to be interpreted and tagged before it can be explored. This

interpretation can be done automatically, as is the case with the TDUB system [126], or requires manual tagging [127]. Automatic tagging is usually domain specific, and would require an extension to the program for each new diagram element, or class of diagrams. The TDUB system for example, can only interpret circuit diagrams, UML diagrams, and architectural plans [126]. For blind people to have access to diagrams without manual preparation or domain-specific automatic tagging, requires a method for gaining direct access to the shape of the diagram.

3.5 Data Visualisation

Data visualisation systems are different from accessible diagram explorers, in that the former operate directly on mathematical functions or raw data, while the latter operate on already created diagrams. This section describes approaches which produce accessible visualisations from data.

3.5.1 Orion TI-84 Talking Graphing Calculator

The Orion TI-84 talking graphing calculator is an accessible Texas Instruments TI-84 calculator developed as a joint project between American Printing House for the Blind (APH), and Orbit Research [129]. The physical unit consists of a standard TI-84 calculator and an additional component, located above the device, which includes speech synthesis and a keypad to control the speech and navigate through content in an accessible way.

The Orion TI-84 allows blind and visually impaired users to enter and manipulate equations while receiving speech feedback on typed keys and device output. It is also possible to produce audio renderings of graphs plotted by the device. Users are able to plot equations by using the usual controls provided by the TI-84 calculator, after which the graph would be rendered in audio form by the Orion component. Audio rendering consists of sinusoidal tones played in series, with each tone denoting a data point of the graph. The pitch or frequency of each tone denotes the value of a data point, i.e. its vertical position in the graph. White noise is added to a tone when the point is lower than the x-axis. Both the x- and y-axes are indicated with a tone when crossed, and grid lines can also be rendered. Users can explore the graph by using the provided navigational keys. Horizontal navigation is achieved by using the left and right navigational keys, and points are announced and sonified as the user moves across the graph. By interfacing with a personal computer, it is possible to produce braille tactile drawings of graphs if a graphics-capable embosser is available.

3.5.2 SAS Graphics Accelerator

The SAS graphics accelerator is an accessible data visualisation interface designed by SAS software [130]. It is integrated with the SAS studio, a mainstream environment for data analysis. The SAS graphics accelerator allows blind readers to sonify standard charts produced by SAS software, such as bar charts, line charts, heat maps, scatter plots, and bubble charts. This is accomplished with an algorithm similar to the general purpose image to sound mapping used by the vOICe. For each point on the x-axis, corresponding values on the y-axis are sonified using a piano note sound. The pitch of the note is determined by the value on the y-axis, with higher values producing notes with a higher pitch. The interface is also designed to make use of web accessibility technology, allowing the user's screen reader to announce values as they are explored. Users are able to listen to the chart in its entirety, or step through the chart with the cursor keys. Users can also obtain a textual description of the chart, or view the data as a standard HTML table.

3.5.3 Discussion

One drawback of data-visualisation tools like the Orion TI-84 Plus calculator or the SAS graphics accelerator is that they can only provide visualisations where the source data is available. Although many publishers now encourage authors to upload their raw data, many articles still only contain the graphs and figures provided and pre-rendered by the author. Data visualisation tools that require data or equation mark-up as input cannot visualise such already rendered graphs. They are also unable to render figures or depictions of data for which they are not programmed.

3.6 Discussion

For blind readers, effective access to diagrams is currently not possible without costly hardware equipment. A substitute is for authors to provide textual descriptions of diagrams, but, even when provided, textual descriptions fail to provide the relational view that diagrams are meant to depict. Studying the graphical depictions of functions and other mathematical systems is also an integral part of teaching mathematics and cannot adequately be conveyed by textual descriptions. Sonification may offer an attractive alternative to hardware approaches and textual descriptions, but existing research on audio-visual sensory substitution focuses on conveying images of the physical environment, and only a few studies have examined sonification as a direct reading aid for diagrams. Although sonification has been successfully applied to data visualisation, such approaches require access to original source data which is usually not available.

Another challenge which is not addressed by current access solutions for blind readers is that blind readers are unable to verify the visual representation of their own work. A significant requirement of producing technical and scientific content in a professional setting is the use of visual material like infographics, photos, and other images which are inserted into documents alongside textual content. Without a method for direct access to graphical information, blind and print-disabled users are unable to verify the adequacy and placement of these images. A similar situation arises when blind users construct other two-dimensional representations of data, like tables and charts. Although methods are available for constructing these representations in an accessible way, blind users are unable to judge the resulting visual layout of such objects, including whether they fit on the page and how much space they require relative to other content. For example, it is increasingly required to sign documents electronically, by inserting an image of a signature. Blind users are able to insert an image of their signature into a document, but without direct access to graphical information, they are unable to verify the alignment of the signature with the rest of the text. Finally, blind users working in the field of software development are currently unable to independently perform roles which require the development of user interfaces, as they are unable to verify the layout of user interfaces on the screen.

It is clear that a method is required for accessing graphical content without costly hardware equipment or human translation. Such a method should also aid blind readers in converting material to an established accessible format when an accessible version is not yet available. Finally, an accessible representation of two-dimensional content would enable blind scholars to study information that is constantly being updated, for which timely translations are not usually provided, and for which a textual description would be inadequate. Examples include weather data, astronomical images, and real-time maps.

3.7 Summary and Conclusion

One challenge of technical document accessibility that has not been addressed without the use of specialised equipment is the accessibility of graphs, charts and other graphical indicators that may occur alongside equations in a scientific or technical document. The current state of the art is to access these elements, which are most often in the form of vector-based drawings or rasterised images, using either a graphics-capable braille embosser, or a multiline braille display, both of which are beyond the financial means of most blind people. The W3C guidelines for the web and Epub3 books require that authors include textual descriptions of all graphical elements in the text, but this is often not done.

Sensory substitution is an approach for accessing the information destined for one sense by mapping it to a form perceivable by another. For graphi-

cal accessibility, the most relevant approaches are tactile-visual sensory substitution for accessing visual information by touch, and audio-visual sensory substitution which utilises the audio channel instead. Audio-visual sensory substitution may offer more portable and cost effective solutions for accessing graphical information, as no special hardware is required. However, most sensory substitution research has been performed in the context of access to the physical environment.

Another approach which may be implemented without specialised hardware is semantic diagram exploration. However, these approaches require diagrams to be specially prepared, and are often limited to specific domains. Finally, accessible data visualisation methods have been devised. These allow blind readers to generate accessible visualisations directly from data. However, these approaches require access to the raw data, which is rarely published.

Of the approaches discussed above, we believe that audio-visual sensory substitution holds the most promise for a portable cost-effective solution. We therefore propose extending an audio-visual sensory substitution algorithm by the addition of exploratory gestures, to allow exploration of diagrams by parts. The following chapter describes our proposed approach for diagram accessibility, a direct reading aid based on the vOICe sonification algorithm augmented with touch screen gestures.

Chapter 4

Approach to Accessibility of Graphical Material

It is evident from the literature that graphical information like diagrams remains largely inaccessible to blind readers. Current solutions to diagram inaccessibility are not widely in use, due to their inherent limitations. Tactile diagrams are arguably the most effective solution for facilitating accessible diagrams, but remain unavailable to most blind users due to the high financial cost. A manual production process also means that tactile diagrams are unsuitable for interactive diagrams, or diagrams that might be updated continuously. Finally, the physical nature of tactile diagrams and related technology results in low portability.

Although techniques have been previously discussed to allow exploration of diagrams using consumer equipment commonly available to blind users, these approaches are largely limited to very specific domains. Most of these approaches also require access to specially prepared diagram formats, or the original data from which the visual diagram was produced. Therefore, these solutions do not provide accessibility to diagrams published in typical scientific and technical papers, which are usually available as untagged (inaccessible) PDF documents.

We propose the use of audio-visual sensory substitution as a solution for universal diagram accessibility. Sensory substitution algorithms like the vOICE have previously been studied in the context of a continuous vision substitution solution to convey the physical environment to blind users. However, we believe that audio-visual sensory substitution lends itself well as a direct reading solution for diagram accessibility. Combining a sensory substitution algorithm like the vOICE with interactive exploration controls would allow blind users to study diagrams directly in their published visual form.

This chapter describes our approach for allowing interactive diagram exploration using an algorithm based on the vOICE. We developed an extension to the vOICE which utilises touch gestures to allow targeted exploration of rasterised image data. We also describe user trials comparing our approach

with the original vOICe algorithm as a baseline.

4.1 Algorithms

We will present an algorithm that extends the vOICe, as proposed by Meijer [20], by taking particular advantage of a touch screen to allow interactive image exploration. We will use a standard implementation of the vOICe algorithm as a baseline in our evaluation.

4.1.1 The vOICe algorithm

As introduced in Chapter 3, the vOICe converts an image to an audio signal by scanning through the image from left to right, one column of pixels at a time. Each column is rendered as a superposition of sinusoidal components (a chord), with pixels towards the top of the image corresponding to higher frequencies. The greyscale value of the pixel determines the amplitude of the component, so that pixels with an intensity of zero produce no sound at the corresponding frequency. The chords are then played one after the other, from left to right, with appropriate stereo panning.

The original implementation of the vOICe was designed to be flexible towards future development [20]. In particular, although the first implementation was realised on purposely-designed electronic hardware, the parameters of the algorithm were stored in writable memory. By default, the lowest and highest frequencies were 500Hz and 5KHz respectively, and the image resolution was assumed to be 64 by 64 pixels. Each pixel was assigned one of 16 discrete sinusoidal amplitudes.

Another attribute of the original algorithm that could be altered by the experimenter was the pixel-to-frequency mapping. The original board could store up to 250 individual mappings. Meijer examined both a linear and an exponential distribution of frequencies, between 500Hz and 5KHz [20]. An undesirable practical effect of a linear mapping was that it led to an audible buzz due to cross-talk between the oscillators. This effect was reduced by using the exponential mapping. The exponential mapping is presumably also more natural for the ear, as it corresponds roughly to the distribution in frequency resolution of the cochlea.

For this study, we use an exponential mapping for both the vOICe and the interactive algorithm, with a lowest frequency of 500Hz and a highest frequency of 5KHz. Although our implementations are able to produce a range of sound intensities, we evaluate only binary images. Therefore, a pixel is either sonified or it is silent. Unlike the original vOICe implementation, we also used a higher resolution for our test images, specifically 300 by 300 pixels.

4.1.2 Proposed Algorithm

We have developed a gesture enabled interactive algorithm, implemented on a touch screen, to address some limitations we believe exist with the current implementation of the vOICe when applied to the visualisation of non-textual document content. We have considered specifically the challenges of visualising line drawings, as may be present in published technical material. When using the vOICe as originally proposed, the user hears a rendition of the entire image, and it is not possible to directly control what is rendered. This makes it difficult to explore details of the image, such as the connections in a flow chart or the evolution of a graph. While more recent versions of the vOICe do allow the user to define the area being rendered by indicating the upper-left and lower-right corners on the touch screen, this feature was designed to focus the rendition of an everyday scene and is not appropriate for document reading. In particular, it is ill-suited to the tracing of lines or the finding of edges, as the scanning process results in a slow exploration. Furthermore, the position of an element is identified by its corresponding position in time within the sonification, and is not directly related to the user's interaction.

It should be borne in mind that the vOICe was intended as a continuous vision substitution algorithm, for use when exploring the physical environment with a live camera. In such a situation, large details are important, and can be quickly identified when the camera view is scanned from left to right. In addition, the changing camera view as the user moves or alters their body posture serves as a natural way of exploring the environment. The situation is different for document reading. Here the view is not expected to change. Instead, the user is expected to carefully explore various areas of the image interactively.

The vOICe emits a chord of sound for each column of the subject image, scanning from left to right. The user knows the position of each column from its position in time, and also from its placement in the stereo field. The algorithm used by the vOICe remains an integral part of the method we propose. However, we extend the algorithm with gestures, allowing the user to interactively explore details of the image.

Firstly, when one finger is placed on the screen, the system selects a short vertical segment of the image located directly under the user's finger. This segment is then rendered as sound using the vOICe algorithm. Secondly, when two fingers are placed on the screen, the segment along the line connecting these two locations is sonified as a chord. The upper and lower frequencies are determined by the vertical locations of the upper and lower finger respectively. Using this gesture, the user can determine both the length of the segment (by moving the fingers closer together or further apart) and the orientation of the segment (by rotating the fingers). For example, the user can emulate the vOICe by placing two fingers on the screen, one at the top, and one at the bottom, and moving both from left to right. However, because the start

and end points of the scanning path can be arbitrarily defined, horizontal and diagonal segments can, for example, also be sonified.

4.1.3 Illustration of the Algorithms

The following paragraphs will describe the operation of the proposed algorithm by means of three simple example images: a horizontal line, a diagonal line, and a right-triangle. Each image is centred on the display area.

For the horizontal line, the vOICe would render a single tone, with a steady pitch, moving from left to right in the stereo field. With our interactive algorithm, the user may also have the entire image sonified by issuing a keyboard command. In addition, they are able to explore the image in detail using touch gestures. If the user places a single finger on the screen, silence will be heard if the finger is not placed on the line. If the finger is moved to encounter the line, a tone will be emitted with a steady pitch indicating the vertical position on the screen and positioned in the stereo field to correspond to the horizontal location.

Using two fingers, a rendering similar to that produced by the vOICe can be achieved by dragging two fingers, positioned vertically above each other, from left to right across the screen. The temporal progression of the sound can be reversed by performing this gesture in the opposite direction.

Consider now a diagonal line, drawn from the upper-left to the lower-right of the screen. The vOICe would produce a descending tone, moving from left to right across the stereo field. With the interactive algorithm, dragging one finger across the screen will play a sound as soon as the line is encountered by the finger. If the finger is dragged from left to right, a short burst of a quickly descending tone will be heard as the line is crossed. Dragging two fingers placed vertically above each other across the screen from left to right will imitate the rendition that would be produced by the vOICe. Dragging two fingers placed horizontally from each other from the top to the bottom of the screen will produce a descending tone, while the same gesture performed from the bottom to the top will produce an ascending tone. The sound source will also shift from left to right in the stereo field for the former case, and from right to left for the latter.

Finally, consider a right-triangle, formed by a diagonal line from the upper-left to the lower-right of the screen, and lines along the left and bottom edges. The vOICe rendition would consist of a click (a brief chord containing all frequencies) followed by two simultaneous tones, one with a steady low pitch and one with a descending pitch. The interactive algorithm would produce silence if the user's finger were placed inside or outside the triangle. However, when the motion of the finger crosses one of the edges of the triangle, a tone is produced. Moving two vertically positioned fingers from left to right across the centre of the screen will play a descending tone, when the diagonal side of the triangle is crossed. When this diagonal is detected, The user can rotate

the two-finger gesture 45 degrees to the right, and then follow the descending line downwards. When the user reaches the lower right vertex of the triangle, a constant low tone will be heard indicating the bottom horizontal side of the triangle. At this point the user can return the two-fingered gesture to the vertical position and trace the horizontal line to the left where it eventually joins the vertical side of the triangle.

The three simple examples described above serve to illustrate the operation and functionality of our interactive algorithm. They do not, however, illustrate the strengths of the algorithm. Indeed, for basic geometric shapes such as these examples, the conventional implementation of the vOICE can be argued to be more effective, since the sounds generated are simple and easily recognised. The interactive algorithm is expected to show its advantage when considering more complex diagrams, consisting of multiple and interconnected elements. For such cases, the audio rendition produced by the vOICE will be much more complex and more difficult to comprehend. The gestures, however, allow elements of the diagram to be isolated and therefore individually identified. This will allow more complex figures to be systematically explored and analysed by a blind user, building up an understanding of the whole.

4.2 Experimental Evaluation

To evaluate the interactive algorithm, six blind and six sighted subjects were recruited and asked to identify a sequence of images presented to them as audio on a tablet computer. These images ranged from single simple geometric shapes to more complex compositions. Both the vOICE and the interactive algorithm were used to render the images. The objective was to determine whether the additional functionality offered by the interactive algorithm is in fact beneficial to the user. It was decided not to blindfold the sighted subjects during the test, since neither algorithm renders to the screen. The screen is used only as an input device.

Each test subject was familiarised with both algorithms before the testing process started. During these training sessions, users were required to practice recognising lines and simple shapes from the corresponding sound patterns produced by the vOICE. Subjects were also taught to use the gestures offered by the interactive algorithm. Finally, test subjects were familiarised with the shape of the emoticons that formed part of the test. This was necessary since most blind participants had never encountered these emoticons before.

4.2.1 Test Images

The test comprises five stages. The first three concern the identification of simple shapes while the last two involve the identification of emoticons. The simple shapes in question are squares, circles, ovals (horizontal and vertical),

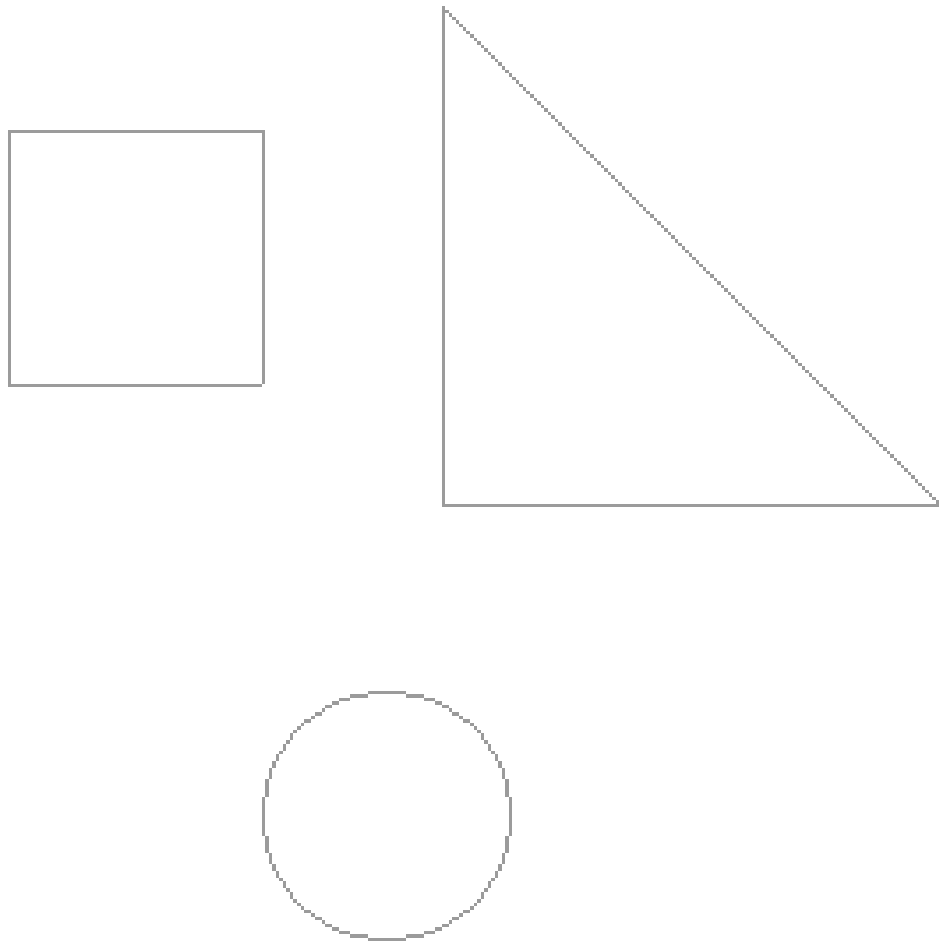


Figure 4.1: The three categories of shapes used in Stages 1, 2 and 3: Triangle and square at the top, circle at the bottom.

rectangles (horizontal and vertical), and triangles in four orientations. The emoticons are a smiley face, a sad face, a skew-mouthed face, and a face with a winking right eye.

Figure 4.1 shows the three shapes rendered in Stages 1 through 3. Note that the triangle can appear in one of four rotations, while the circle and the square might be stretched horizontally or vertically to produce an oval or a rectangle. Figure 4.2 shows the four emoticons rendered in Stages 4 and 5. The five stages of the experimental evaluation are presented in Table 4.1.

For each stage of testing, a number of images are synthesised. The precise composition of the images used in our test is described in Appendix A. Each test image is paired with either the vOICE or the gesture-enabled algorithm, resulting in a sequence of image-algorithm pairs. Users are required to listen

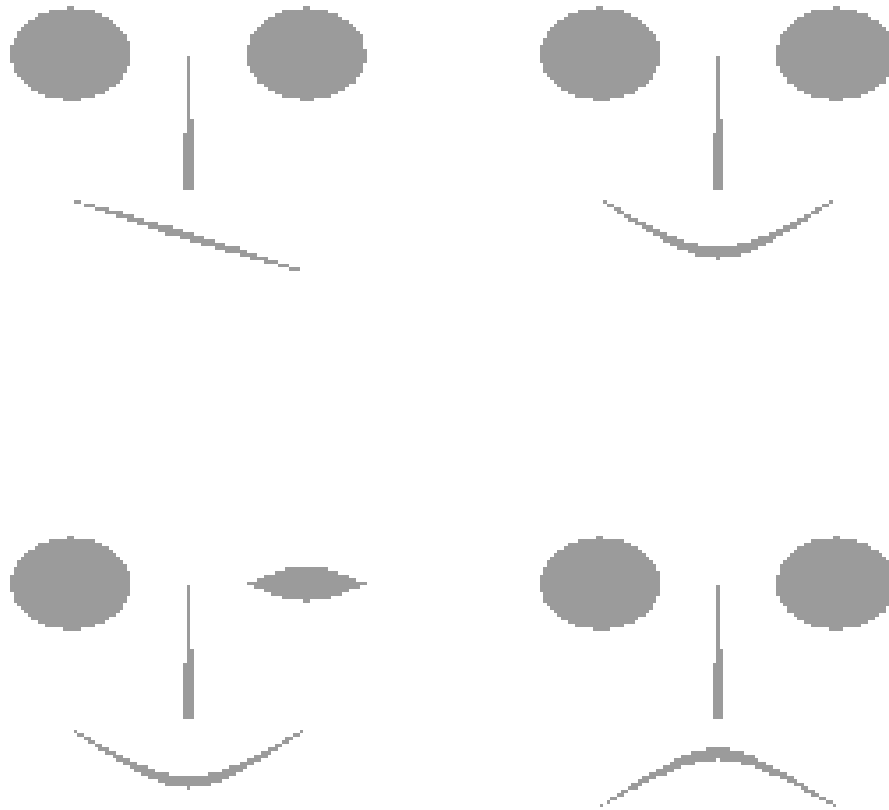


Figure 4.2: The four emoticons rendered in Stages 4 and 5: Skew mouth and smiley at the top, sad face and winking face at the bottom.

to the image as rendered by the algorithm, after which a menu containing a list of shapes is presented. Users must then identify all shapes that they recognise in the image using this menu.

A shape can only be selected once, even if it appears more than once in the image. This decision was taken as a compromise between the duration of the testing procedure, and the number of data points that can be collected. During initial experimentation, users were required to provide more detailed answers after images were rendered, but the duration of the test procedure proved too long for most test candidates. In its current form, the test procedure takes about an hour and a half to complete by most subjects.

The shapes for Stages 1 through 3 were generated algorithmically and written to the rendering buffer. These shapes were outlined but not filled.

Stage	Description	vOICe	Interactive	Total
1	Two simple shapes are rendered	2	2	4
2	Three simple shapes are rendered	3	3	6
3	Four simple shapes are rendered	3	3	6
4	A single emoticon is rendered	2	2	4
5	Four emoticons are rendered	5	5	10
All		15	15	30

Table 4.1: The stages of the experimental evaluation, showing the number of test images presented using the vOICe and interactive algorithms respectively. The precise composition of each image is given in Appendix A.

The emoticons for Stages 4 and 5 were pre-rendered using Tikz, and consisted of filled circles for the eyes, a filled triangle for the nose, and a narrow filled ellipse for the mouth. The mouth was either curved or angled to produce the possible mouth shapes, while one eye was narrowed horizontally to produce the winking face.

4.2.2 Test Procedure

The test was managed by software that renders the test images using one of the two described algorithms. After rendering each image, the system presents a menu containing the names of graphical objects that might be present in the image. Using this menu, the test subject must identify the components present in the image. Each component can be selected only once, even if the image contains several instances of this component. This was done to keep the testing procedure simple. Initial informal experimentation with more detailed test responses resulted in a procedure that was impractically long.

The test management software produces a log of each test subject's session indicating the selections made, the correct answer, as well as additional data such as the time elapsed since the shape was rendered, and in the case of the interactive algorithm, the gestures used to explore the image.

To perform the test, the user is presented with a tablet computer, a pair of headphones, and a wireless keyboard. The software that manages the test runs on the tablet computer. All messages are presented as synthesised speech.

The test process consists of the following steps.

1. The program introduces itself to the test subject.
2. The test subject is informed of the stage of testing (for example, identify shapes, identify emoticons, etc).
3. From a set of predetermined possible image and algorithm pairs, an image is selected and rendered with one of the two algorithms. In the case of the interactive algorithm, the subject is invited to explore the

image in more detail by using the touch screen. In the case of the vOICe, the scene is rendered.

4. A menu listing the possible components that may be present in the image is presented to the test subject. The menu also contains an option to listen to the image again, and to continue to the next question. The subject can move through the menu using the up and down cursor keys. The subject was required to identify the components present in the rendered image, but not their individual locations. This was done to reduce the complexity of taking the test. Earlier, informal experimentation requiring more detailed feedback was found to lead to an excessively long test and fatigue among the test subjects.
5. The algorithm, along with the correct answer, the subject's answer, the number of times the image was explored, and the duration of each attempt is recorded to a log file.
6. Items 2 through 5 are repeated for each algorithm, stage of testing and configuration of test images. Stages are selected in sequence, and algorithms and images are selected from a randomised set of image and algorithm pairs.

All subjects were given between 60 and 90 minutes of individual training before taking the test. During this training, subjects were taught to recognise images using the vOICe, as well as to use the exploration gestures provided by the extended algorithm. Particular attention was paid to the recognition of straight lines, and it was explained how multiple lines can form more complex shapes. Users were also familiarised with the shapes that occur in the test, as well as the four types of emoticons. The test was started when a user declared him/herself confident with using the two algorithms. All test subjects are shown the same test images and each test image is always rendered using the same algorithm.

4.2.3 Test Subjects

Evaluations were performed with 12 test subjects (six male and six female) between the ages of 20 and 45. Hence each of the images described in Table 4.1 was considered by 12 subjects, leading to a total of 360 responses, 180 for the vOICe and 180 for the interactive algorithm respectively. Half of the subjects were legally blind while the remainder were sighted.

Due to the inaccessibility of technical study material and other barriers, blind people remain underrepresented in STEM (science, technology, engineering, and mathematics) fields. Therefore, it was a challenge to identify suitable blind test subjects. One of the blind subjects was active in a STEM field,

while another was employed as an assistive technology specialist. The remaining four had no scientific or engineering background. In contrast, all six of the sighted subjects were from a scientific background.

4.3 Results

The blind subjects reported that they found the menu system easy to operate and use, as it was similar to TTS-based menus used by other assistive interfaces. All subjects, both sighted and blind, understood within a short time how the vOICe based image-to-sound mapping worked. Most subjects were also able to use the interactive algorithm within a few minutes of introduction, although confident use took some time.

Table 4.2 lists the results for each stage as attained by blind subjects, sighted subjects, and both groups combined. The table shows that, overall, the vOICe algorithm resulted in 13% correct responses, while the interactive algorithm resulted in 48% correct responses.

Stage	Blind subjects (6)		Sighted subjects (6)		All subjects (12)	
	vOICe	Interactive	vOICe	Interactive	vOICe	Interactive
1	0.0%	16.7%	0.0%	33.3%	0.0%	25.0%
2	5.6%	22.2%	5.6%	44.4%	5.6%	33.3%
3	22.2%	16.7%	0.0%	38.9%	11.1%	27.8%
4	66.7%	83.3%	50.0%	91.7%	58.3%	87.5%
5	10.0%	53.3%	0.0%	70.0%	5.0%	61.7%
Total	17.8%	38.9%	7.8%	56.7%	12.8%	47.8%
Average time (s)	62.0	159.4	66.6	141.8	64.3	150.6

Table 4.2: Percentage of correct identifications per testing stage and algorithm type (vOICe and interactive) as well as average response times.

Figure 4.3 summarises the percentage of correct responses per stage in the test. Note that the analysis presented in this figure is based on least squares means and hence deviates slightly from the results in Table 4.2. The figure clearly shows that the interactive algorithm exhibits higher accuracy during all five stages of testing, but this improvement is particularly large in Stage 5 (the identification of multiple emoticons). This is an encouraging result, since the composition of four emoticons in a single image can be considered to be the most complex among the images in the test.

In our testing procedure, an answer was considered correct when the user correctly identified all components in the rendered image. If the user identified a component that was not present, or failed to recognise any of the components correctly, the answer was regarded as incorrect. This means that the table above represents only answers for which an image was perfectly described. We decided on this approach in an attempt to reduce the cognitive load on the subjects, since the duration of the test was already more than an hour in most

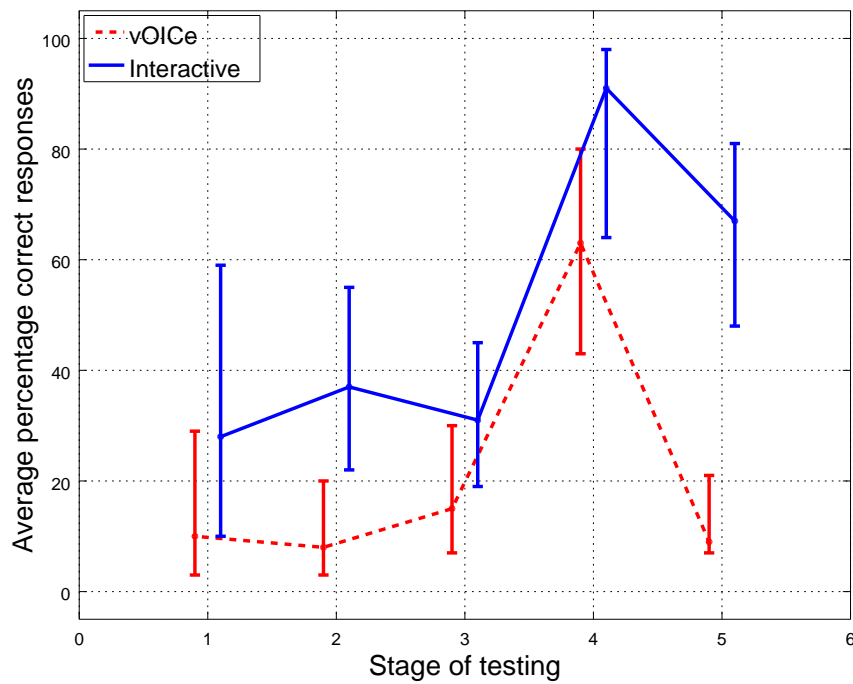


Figure 4.3: The percentage of correct responses for each algorithm, for each stage of the test. The means and 95% confidence intervals have been calculated using least squares means analysis.

cases. Due to this test structure, it was however not possible to unambiguously determine which shapes were most often confused.

4.4 Discussion

As shown by the table in the previous section, both the blind and sighted groups performed better when interpreting the rendition of graphical information using the interactive algorithm, compared with the rendition produced by the vOICE. This was confirmed significant ($p < 0.01$) using both a Chi-squared and a Fisher exact test with Rao-Scott adjustment to account for each subject contributing more than one response. This significance holds both overall as well as when considering blind and sighted subjects separately and suggests that the touch screen gestures provide a clear additional benefit when reading complex shapes.

During training, a number of blind subjects reported to experience occasional difficulty in interpreting the sound produced by the interactive algorithm. It was observed that these subjects were inadvertently rotating their fingers into a slightly diagonal position, while believing them to be vertically aligned. Since the interactive algorithm renders the pixels on the line between the two fingers continuously as the fingers are moved across the screen, such diagonal placement leads to a vertical line being rendered as a rising or falling pitch. Reminding subjects to pay careful attention to the alignment of their fingers usually improved the situation. Nevertheless, this aspect of tactile exploration on a touch screen by blind users merits further consideration.

Table 4.2 also reveals that the blind subjects were better than the sighted subjects at interpreting the renditions produced by the vOICE, in which exploration gestures were not allowed. Although both groups did better when using the interactive algorithm than when using the vOICE, the sighted users achieved higher scores with the interactive algorithm overall.

There are two possible explanations for this. First, the sighted subjects might have had an advantage since they could see their fingers move on the touch screen and thereby avoid, for example, the inadvertent rotation described above. Second, the sighted subjects might simply be more practised in interpreting two-dimensional visual information. For example, during the tests it was noticed that a number of the blind users had to be familiarised with the concept and shape of an emoticon since they had not encountered it before. This seems to support the second hypothesis. However, if the only advantage enjoyed by the sighted subjects was their much greater familiarity with the interpretation of two-dimensional shapes, one would expect them also to do better when recognising renditions produced by the vOICE. Therefore, we suspect that the sighted subjects benefited also from their ability to follow their fingers when using the interactive algorithm. Sighted users were able to more effectively visualise shapes by combining accurate knowledge of their finger

positions with their familiarity of the rendered shapes. In fact, two sighted users did report that they were able to visualise the shapes as if they had appeared in front of them. This was not reported by any of the blind subjects.

It should also be borne in mind that most of the sighted subjects were from a scientific background (five had completed an undergraduate degree in science, while the sixth was enrolled in an engineering degree). In contrast, the blind subjects were mostly from a humanities background, with the exception of one who has a masters degree in computer science. Although not shown in the table of results, the blind subject from the computer science background attained a score of 10 for the interactive algorithm, which was similar to the scores achieved by the sighted group. It is therefore also possible that the scientific background of the sighted subjects allowed them to infer the shapes more easily. However, this scientific knowledge did not seem to aid them in interpreting the renditions produced by the vOICe.

Feedback provided by the subjects indicated that they had most difficulty differentiating between the circle and ovals, the square and rectangles, as well as the smiley and winking face. The confusion between the smiley and winking face is understandable, since the two shapes are very similar, differing only in the right eye, which is narrower on the winking face. The confusion between the circle and ovals, as well as between the square and rectangles, may be explained by the fact that the overall soundscapes produced in both cases are similar to each other, differing only in the width of the frequency spectrum, as well as the duration in time. For example, both a square and rectangle consists of a click, followed by two simultaneous tones (one higher than the other), followed by another click. The difference lies in the duration between the first and final click, as well as the difference between the frequencies of the two tones.

4.5 Summary and Conclusion

In this chapter we have considered audio-visual sensory substitution as an aid for rendering diagrams to blind users. We have extended the audio rendering approach used by the well-established vOICe algorithm to allow interactive and localised exploration of an image by means of gestures and a touch screen. The effectiveness of our approach was evaluated by means of a set of user trials with six sighted and six blind subjects to identify the elements of scenes consisting of a number of geometrical shapes and emoticons. We found that both groups of subjects were more successful at identifying shapes using the interactive algorithm than with the baseline vOICe algorithm.

Our subsequent work focuses on the rendering as speech of any textual elements embedded in an image, such as the labels on a graph, or the symbols in an equation. We examine such an approach in Chapter 6 applied to the reading of equations embedded in PDF documents.

Chapter 5

Accessibility of Mathematical Material

In terms of technical document accessibility, two challenging areas can be identified: accessing purely graphical content like charts, graphs, and other diagrams, and accessing mathematical equations. In the previous chapters we have described the current state of the art in terms of access to diagrammatic content, and also proposed a solution for providing direct access to graphical content via audio-visual sensory substitution. This chapter describes background on accessible approaches¹ to producing and consuming mathematical equations. Section 5.1 gives an overview of braille mathematical notation systems. Section 5.2 describes systems for enabling access to mathematics in electronic documents. Section 5.3 highlights some of the challenges associated with current approaches to mathematics accessibility, including a comparison of linear and two-dimensional mathematical notation. As an alternative approach to conveying relational information, Section 5.4 gives an overview of textual and audio games, from which inspiration can be drawn to improve mathematical content accessibility. Finally, Section 5.5 explains how our proposed approach for graphical accessibility may also be applied to mathematical content.

5.1 Braille

Since a normal braille cell consists of six dots, a maximum of 64 symbols can be represented. For this reason, cells usually represent linear text characters. Therefore, separate contexts have been developed to represent content which may require special symbols, or need to convey additional information

¹Of the approaches discussed in this chapter, the author makes regular use of LaTeX (described in Section 5.2.1) to read and write mathematical content accessibly. The author also reads mathematical content using the MathJax accessibility component described in Section 5.3.2.3.

in a linear form. For example, contexts have been developed to represent musical scores and mathematics, since these printed representations are two-dimensional by nature, but must be displayed in a linear format in braille. In the case of mathematical content, special braille symbols are available to denote graphical indicators like the start or end of a fraction, roots, integrals, superscripts, and subscripts.

Because mathematical braille was developed fairly late in braille's history, several systems were independently developed in various countries, and many are still in use today. For example, the Nemeth code for representing mathematics is used in North America [6], the Marburg system is used in Germany, and the RNIB system is preferred in the United Kingdom [7]. In South Africa, a system adapted from the RNIB system was used until recently, when it was replaced by the unified braille code.

The unified English braille standard (UEB) is an attempt to unify braille across English speaking countries. It has also been applied to other languages, in which case it is just called the unified braille code or UBC. UEB includes a standard notation for mathematics similar to the British (RNIB) maths system, and hoped to become the standard mathematical code across the world. However, it has not been universally adopted. In the United States, for example, where UEB was mostly developed, a parallel system was developed by Andrew Nemeth (the designer of the Nemeth mathematical braille code), that includes most of UEB, but replaces the mathematical and technical specification with Nemeth braille.

5.2 Electronic Formats for Accessible Mathematical Content

With the advent of assistive technology for accessing computer systems, blind people have gained access to information in electronic document formats. This is accomplished by sending the text in the electronic document to a text-to-speech (TTS) system, or a hardware refreshable braille display. Because TTS systems and electronic braille displays can only render plain text, the information in electronic documents must be textual in nature, or converted to a plain text format before it is rendered to blind users. The easiest documents to access therefore contain plain text. The structural information of complex document formats like those used by popular word processors and in hypertext as found on the world wide web, can also be described in plain text. However, mathematical information is often contained within rasterised images or depicted with vector-based graphical instructions, and is therefore difficult to convert to text.

5.2.1 LaTeX

LaTeX is a plain text format designed for typesetting scientific material [131]. LaTeX is especially useful for blind users in STEM fields, since its textual nature means that it can be read in braille or by synthesised speech. Within LaTeX, mathematical content is written using plain text, and therefore mathematical equations in LaTeX are symbolic and linear. LaTeX can therefore be considered to use similar methods to braille to convey mathematical structure in a linear way.

Although LaTeX is in effect an accessible medium, and currently used by blind people to read and write mathematical content, it is difficult to read efficiently. This is due to the fact that LaTeX notation must necessarily be verbose, in order to convey all the structural elements required to produce a two-dimensional representation of the content. It should be noted that TeX, and by extension LaTeX, was originally intended as a system for producing and typesetting mathematical content, and not as a medium for studying mathematics or working with equations.

The accessibility of LaTeX content is also dependent on the method used by the blind person to read it. Most blind people will read LaTeX content via TTS, which makes it more difficult to explore the content in detail. This is due to the fact that TTS renders the text sequentially, while mathematical content should be explored in logical units often represented two-dimensionally on paper. Screen readers provide commands to read character by character, or word by word, but these textual units are ill-suited for the exploration of the logical units within an equation. Furthermore, the structural directives used by LaTeX do not follow the same separation pattern used by words in literary text. It is therefore more efficient to read LaTeX in braille, by using a braille display or embossing it with a braille embosser. However, it should be borne in mind that LaTeX content is significantly longer than the equivalent content in mathematical braille.

5.2.2 MathML

MathML is an XML-based format for representing mathematical content first introduced in 1998 by the World Wide Web Consortium [25]. It was devised specifically for embedding mathematical content in XML-based documents like HTML or docbook. It was later adopted as part of the HTML5 specification, and is also included as part of the Epub3 ebook format. MathML can be interpreted by assistive technologies and made accessible to blind readers. Examples of approaches for reading MathML are described in Section 5.3.2.

MathML allows for two different ways of specifying mathematical content: by presentation and by content. Presentation MathML is designed for displaying equations. Characters are specified by their Unicode values, and elements are available to denote layout and structure. Content MathML specifies the

semantic meaning of an equation, and contains elements denoting logical concepts, like the application of a function. In content MathML operators are also specifically identified, while in presentation MathML they are specified by the “mo” element, containing the Unicode character of the operator. Since the presentation and content are specified separately in MathML, equations can be interpreted in various ways depending on the viewer. Equations may be displayed on a screen, or may be interpreted by a screen reader or other assistive technology, or interpreted by an equation editor or solver.

MathML also allows semantic descriptions to be attached to equations. For example, an equation in MathML might have the original TeX source of the equation as a semantic element, or the code for solving the equation programatically.

Content MathML can be extended by a separate standard called OpenMath, maintained by the OpenMath society. OpenMath is an XML specification for specifying the semantics of equations, by referencing mathematical elements like operators and functions, which are specified in OpenMath dictionaries. OpenMath dictionaries can also be used to extend the set of symbols used in content MathML.

5.3 Challenges in Mathematics Accessibility

The previous section described some existing approaches that seek to make mathematical content accessible. Despite these options, most mathematical material in electronic documents remains inaccessible. This section describes some of the reasons for this state of affairs.

5.3.1 Linear Reading

One property shared by mathematical braille and LaTeX is that they are both linear writing systems. However, mathematical content is most commonly regarded as a two-dimensional writing system. Studying mathematics in a linear way presents several challenges to a blind or visually impaired person [4; 132]. The linear representation obscures structure that is easily apparent in visual form. In fact, it has been argued that proficiency in mathematics has a strong reliance on visualisation ability [133]. However, a linear system requires that the equation should be interpreted from left to right, and its structure parsed, before it can be mentally represented.

A linear representation of an equation is also longer than its equivalent print form, due to the fact that the structural information that is denoted by varying vertical placement must instead be denoted by symbolic notation. Depending on the specific linear code, it might also require more parentheses than its print counterpart, as the print counterpart can use spacing and vertical placement to group terms, which is not possible with the linear form. For example, a

fraction like $\frac{x+1}{y+2}$ requires two sets of parentheses in British mathematical braille notation, one set for the top, and one set for the bottom of the fraction. In print mathematics, the fraction line itself denotes the grouping of the terms. Nemeth braille code uses two symbols less than British mathematics for the above example, since it provides “fraction start” and “fraction end” indicators, but it is still two symbols longer than the equivalent print equation. Furthermore, as previously discussed, both LaTeX and braille often require more than one symbol to denote a single indicator. In the case of LaTeX, this is due to the fact that words are used as indicators, for example, “frac”, “int”, or “sqrt”. As six-dot braille can only denote a maximum of 64 different symbols with one braille cell, (one being the white-space), several cells are necessary to denote many symbols and indicators. Every additional symbol adds to the reader’s cognitive load when studying mathematics, and may distract from the equation under review. Finally, reconciling grouping terms such as nested parentheses requires more cognitive effort when reading an equation in linear form than when viewing its traditional graphical representation.

Several systems have been proposed as possible solutions to accessing linear content efficiently. One approach focuses on producing more efficient verbal descriptions of linear content, which may be easier to interpret when spoken by a TTS system. Another option is to provide a navigational system for efficiently navigating through mathematical equations. Often these possible approaches are combined, offering speech and braille in conjunction with a navigational system. The following section will survey prominent attempts to achieve this.

5.3.2 Accessible Mathematical Content Browsers

Due to the complex nature of mathematical expressions, a spoken representation of an equation is usually not adequate for a proper understanding of the content. This section describes specialised browsers, which allow blind readers to explore mathematical content by manageable chunks.

5.3.2.1 AsTeR

The Audio system for Technical Reading (AsTeR) was developed by Raman in 1994 as part of his PhD thesis [8]. AsTeR converts technical material written in LaTeX to an audio representation based on synthesised speech and other sonic cues. Although the original AsTeR system is not currently in use, some of its approaches are implemented as part of the Emacspeak accessibility extension for the popular Emacs text editor, developed by the same author [72]. The AsTeR implementation was designed to utilise a hardware speech synthesizer, the Dectalk Express. The Dectalk Express synthesizer was able to modify several parameters of the produced speech while synthesising, which AsTeR could take advantage of to convey mathematical structure. For example, mathemat-

ical content was spoken in a different voice from textual content, and within mathematical content, structural indicators were denoted by raising the pitch of the voice. Other voice attributes could be used to convey sub-expressions, references to other equations, and other contextual information. AsTeR also had the capability to utilise the stereo field to indicate spatial placement of elements.

AsTeR attempts to summarise equations by replacing sub-expressions with descriptive names. This allows the blind reader to get an overview of the equation before studying it in detail. AsTeR might, for example, read an equation as “x equals start fraction numerator one over denominator one end fraction”, where numerator one and denominator one are labels for sub-expressions included in the fraction.

5.3.2.2 Math Player

Math Player is a software component developed by the company Design Science [9; 10]. It is currently available as a plugin to popular web browsers, as well as Microsoft Office. By interfacing with screen readers on the Microsoft Windows platform, Math Player allows blind users to read and explore MathML content. When the user reads text on a web page or in a document, and a mathematical expression is encountered, Math Player provides a verbal description of the equation, using rules similar to those devised by Andrew Nemeth and the AsTeR system. Users are also able to explore the equation in more detail. This is accomplished by maintaining a tree-like structure of the equation, where the root denotes the whole equation, and each sub-expression is a child in the next level of the tree. The terms and symbols form the leaves of the tree. Users are able to explore the equation by using the cursor keys on the keyboard. Right and left move forward and backward within the current level of the tree, while up and down ascend and descend the tree structure.

5.3.2.3 MathJax

A hierarchical explorer similar to MathPlayer was recently added to the popular MathJax component for rendering mathematics on the web [26; 134; 13]. MathJax allows authors to include equations in HTML documents as LaTeX, which it renders as MathML (if supported via the browser), or by simulating mathematical structure using Cascading Style Sheets (CSS) and other web technologies. When activated, the hierarchical explorer allows blind users to read equations by expression or subexpression, using the cursor keys on the keyboard. Equations can also be rendered on an attached braille display, if available. Since the hierarchical explorer is part of the official MathJax distribution, all content published using MathJax is automatically accessible to blind readers.

5.3.3 Universality of Accessible Mathematical Systems

As described in Section 5.1, several systems have been developed to present mathematics in braille, used in different countries around the world. These systems are often incompatible, meaning that a blind person from one country might have to learn the mathematical code used in another country before being able to access material. Hence, even when an equation is available as braille, it may not be accessible to a blind reader. Furthermore, braille is a code used specifically by blind users. In order for a braille equation to be shared with sighted peers, it must first be converted to a printable format like LaTeX or MathML. As already pointed out, the reverse is also true. Thus, communication regarding mathematics can be difficult between sighted and blind individuals, as well as between blind individuals.

One method of overcoming these barriers is to devise algorithms that can convert between formats like MathML or LaTeX and one or several mathematical braille encodings, which can be read and edited by blind users, and converted back afterwards. Another proposed solution is to achieve the translation from printed mathematics to a tactile or accessible auditory form which is close enough to the printed form to allow easy communication between blind and sighted users.

5.3.3.1 The Lambda System

The Lambda system was developed as part of a project funded by the European Union, with the aim of increasing access to mathematical content for blind users [135]. It converts MathML content to a special code devised by the Lambda project, which can be read in braille, and is also easily verbalised by TTS systems. It should be noted that the code devised by the Lambda project does not correspond to any of the existing mathematical braille codes. Instead, it is a new system designed to take advantage of eight-dot braille.

The aim of the Lambda code is to have a one-to-one mapping between print mathematical symbols and eight-dot braille. This is easier with eight-dot braille than standard six-dot braille, since the former can represent up to 256 different symbols, rather than the 64 symbols possible with six-dot braille. Having a one-to-one mapping also makes it easier to convert an expression to speech, since each symbol can have a corresponding speech equivalent. For instance, the Lambda code provides symbols for the start and end of a fraction, the speech equivalent would simply be “start fraction” and “end fraction.”

The Lambda project also developed a text editor specifically for editing content written in the Lambda code. The editor is meant to serve as a general aid for blind users to do mathematics. It contains a comprehensive menu system with symbols listed in categories, to ease learning and allow symbols to be inserted into the content. Keyboard commands are also provided for many common symbols. The text editor is implemented as a character grid,

allowing the user to insert symbols anywhere on the screen, thereby preserving the layout of the equation.

The lambda project could not be evaluated with conventional methods due to the difficulty of finding blind users with appropriate mathematical knowledge. Instead, a number of workshops were held in European countries where the project was implemented. Users were encouraged to explore the program and were also provided with copies of the software to use at home and in the school environment. Follow-up sessions were arranged, often via telephone, at which feedback was gathered and used to improve the software. The authors found that users preferred to use braille over speech. This could be due to the fact that it is easier to review spontaneously in braille. However, another possibility is that users are often required to work in situations where speech could be distracting to their peers, for example, in a classroom situation.

The authors also observed a number of strategies devised by users to aid editing mathematics within the Lambda editor. Users would insert spaces on either side of a symbol or term, which would serve as emphasis. When the expression is later scanned by the fingers in braille, the spaces would make the symbol or term “stand out”. Users also preferred different methods of working on derived forms of expressions. Some users would duplicate an expression on a new line, referring back to the original on the line above, while some would duplicate an expression to the right, and others would copy an expression to a new window.

5.3.3.2 The Triangle System

Triangle is another translation system, designed to allow blind users to easily edit and manipulate mathematical equations [136]. The core of the Triangle system is a text editor paired with a special font. Characters have been provided to indicate the start or end of a fraction, an integral, or any other mathematical concept. Due to the special font, these characters are displayed on the screen using shapes which will make their function discernible to sighted readers. Blind users will get an equivalent verbal description when navigating over these characters, or when reading a line. Triangle can produce LaTeX as output, while a converter has also been designed to convert LaTeX into the Triangle format [137]. It is hoped that the Triangle method of displaying an equation might be more compact and easier to work with by blind users than the equivalent LaTeX.

5.3.3.3 The DotsPlus System

DotsPlus is an attempt to create a braille code that is easier to produce and closer to the original printed mathematics [24]. DotsPlus reproduces some aspects of the two-dimensional layout of the content in a tactile form, which might make it easier to be studied by blind and sighted people alike. Dot-

sPlus consists of a new set of symbols, based on eight-dot braille. Symbols are similar to existing symbols in six-dot braille, but designed to take up only one cell per symbol. DotsPlus also makes use of tactile graphics to indicate two-dimensional aspects of printed mathematics, like fractions, integrals, and other graphical indicators. DotsPlus can be produced by the Tiger embosser manufactured by ViewPlus. Technical documents can be produced from existing electronic documents, by replacing the text with a special braille font. ViewPlus provides software which can alter the layout of documents to accommodate the braille replacement text, which may take up more space than the original print text.

One drawback of DotsPlus is that it requires access to a braille embosser capable of producing tactile drawings, or a multiline electronic braille display. Both options are currently expensive to acquire and not available to most blind users not affiliated with a large institution like a university or school. DotsPlus is also difficult to write, since the mathematical content must be produced on a computer before it can be embossed, and would therefore require a different technology to enable the blind person to read the mathematics while it is being written. Hence, it is most suitable for reading typeset mathematics, and less so for manipulating mathematics.

5.3.3.4 The UMA System

The system for Universal Mathematics Accessibility (UMA) was developed as an inter-institutional collaboration project, with the aim of easing communication between blind and sighted mathematicians [138]. It consists of two components: a browser, and a conversion system. The browser allows blind people to explore equations using TTS, while the conversion system enables conversion of mathematical information between various braille codes and electronic formats like LaTeX and MathML.

Each supported mathematical information format is accommodated by a parser and writer which can convert to, or from, a common interchange format (CIF) developed by the authors. The CIF is based on the Open Math markup language. Open Math was chosen as it is a very detailed representation of mathematical information, therefore it should be able to represent any equation converted from the other supported markup languages.

The browser allows a blind user to explore equations by using the cursor and function keys on the keyboard, with output provided via TTS. The functionality is similar to projects like Math Player and Math Speak [9; 139]. The browser is implemented with a document object model (DOM) of the Open Math structure, and navigation of the equation takes place between the nodes of the DOM. The Open Math specification defines a textual description for each component of a mathematical equation, which is used by the browser to provide speech output during the reading process.

5.3.4 Accessibility of Source Material

One of the biggest challenges to mathematics accessibility is that the source material used to render the equations is usually inaccessible. For example, although an equation might have been generated using LaTeX, only the rendered result (usually in PDF format) is available to the reader, not the text-based source. The rendered equation must therefore be translated back into structured formats before it can be studied by a blind person in either braille or an auditory format. This is, for example, always the case for digitally scanned and rasterised formats like DJVU or TIFF, and also for the overwhelming majority of PDF documents. Further, most publishers, as well as online mathematical study tools, do not provide their equations as MathML, but only as already rendered images, in order to ensure the precise layout the author intended.

Although some documents are available as structured marked up formats like LaTeX and HTML with MathML, most scientific and technical material is only available in unstructured printer-ready formats like PDF and postscript. PDF and postscript documents consist of the document text interlaced with vector-based drawing commands, and rasterised images. Although later versions of the PDF standard do allow document structure to be layered over the content in an attempt to improve accessibility, this structure (known as PDF tags) makes little difference to the visual presentation of the document, and is therefore usually omitted. In addition, archival material is seldom converted to modern structured documents, and are usually digitised as image-only formats, or basic PDF documents.

Especially within academic publishing, articles are usually provided electronically as untagged (unstructured) PDF documents. This is due to the fact that documents are produced by independent authors using several systems, some which are unable to produce structured electronic documents. For example, LaTeX is used by many authors within scientific and technical disciplines. However, the latest versions of the commonly used LaTeX distributions are unable to produce tagged PDF documents. Although work is currently under-way to extend the LaTeX engine to allow the production of structured PDF [140], including MathML and other accessibility improvements [141], this will not increase the accessibility of documents that have already been published. It also remains to be seen whether such structured PDF will become the norm for scientific publication.

Some OCR systems have been developed to recognise mathematical content. The best known example is the InftyReader, developed by the Infty project, which is part of the Ideal Group [15]. InftyReader can recognise equations from scanned documents and TIFF image files. The most recent version can also recognise equations in unstructured PDF documents, by interpreting the vector drawing commands [16]. The InftyReader produces output in an XML format called IML, designed by the Infty project, which is designed to be markup system agnostic. IML can be converted to MathML or to LaTeX,

as desired by the user.

The Infty project also develops two other software tools. The Infty Editor was developed to allow sighted editors to verify and correct recognition mistakes made by the InftyReader. Blind users are unable to use this tool, since there is no way for a blind person to read the mathematical equation in the raw image before it is recognised by the InftyReader. Finally, Chatty Infty is a software tool, designed to enable blind users to read and edit the LaTeX produced by the InftyReader. Chatty Infty can thus be considered to be an accessible LaTeX editor.

Besides the relatively high cost of the system, the most significant disadvantage of InftyReader is the possibility of recognition errors. Since InftyReader is based on a trained model of mathematical symbols and structure, it is not infallible. Although the project reports a recognition rate of above 95%, results vary based on the quality of the source. The performance of a recognition system will also deteriorate when the author uses non-standard symbols or conventions. It should be kept in mind that unlike the case with textual OCR, recognition errors produced by a mathematical OCR solution may change the meaning of the produced output. It is therefore likely that recognition errors may go unnoticed by a blind user of the system. Without direct access to the source material, there is no way for a blind reader to verify the output.

5.4 Inspiration from Accessible Games

As we have seen, the spatial aspect of mathematics and other technical content is particularly difficult to represent accessibly. For this reason, a linear representation is most often employed. However, the virtual worlds in accessible games are another medium which also requires the representation of complex spatial information. Studying accessible games may therefore offer an alternative method for representing spatial information [142].

Games that are accessible to blind and visually impaired users can be broadly categorised into two types, based on the medium of output. First, there are textual games, the output of which can be accessed by a screen reader [143]. These games are also called text adventures or interactive fiction. Second, there are audio games, which use audio as the primary output modality [144].

5.4.1 Textual Games

The earliest example of a textual virtual world is Colossal Cave Adventure, originally developed by Will Crowther in 1975 and extended by Dan Woods [145]. Also known as Adventure or simply “advent” (from its program name), it allowed the player to explore a virtual representation of a cave system that was based on a physical location in Kentucky. The player interacts with the virtual

environment by issuing commands like “go east”, for movement, and “look” to describe the virtual space. The game responds by printing textual descriptions of the environment and the player’s actions.

An interface modality similar to colossal cave was later employed by commercial text adventures from the 1980s. One prominent example was the game “Zork”, produced by Infocom, which was a commercial release of an earlier implementation called *Dungeon* [146]. The 1980s also saw the introduction of *Muds*, which are text adventures designed for multiplaying [147]. These textual worlds are named after the first example in this genre, which was called “Multi User *Dungeon*” (after the *Dungeon* text adventure game.) *Muds* are considered to be the predecessors of modern massively multi-player online role playing games (MMORPGs).

The examples of textual virtual worlds described above all employ similar mechanisms to convey spatial layout [148]. Worlds are divided into locations (called rooms), which can conceptually represent any amount of physical space. Relations between rooms are described by means of exits, which are usually indicated by compass directions, such as “north” or “east”, or by “up” or “down”, and sometimes by arbitrary keywords like “path” (in which case the user would issue a command like “go path”). For the directional exits, most games allow the user to move via the exit by simply typing the keyword, such as “east” or “down”. Other relations are based on the descriptions of rooms or objects, such as a book that might be on a table, which may itself be in a room called “The lounge”. The user may then issue a command such as “look on table”, or “look in basket”.

5.4.2 Audio Games

Audio games utilise the audio channel as the primary output modality. The most basic example is arguably a game which requires the user to press keys associated with particular sound samples, like *Bopit* or *Simon* [149]. However, for this discussion we are specifically interested in audio games which convey spatial information. Spatial information can be conveyed as sound by utilising more than one audio channel, such as stereo sound.

Side scrollers are games which, in their visual form, present a one-dimensional or two-dimensional world. The user is able to navigate the world by scrolling either left or right, or sometimes up or down. These games are adapted to an audio-only medium by representing the game objects as sound sources placed in the stereo field [150]. Therefore, an object to the right of the player will produce a sound that is louder in the right channel, while an object to the left will sound from the left channel. Objects above or below the player can be conveyed by altering the pitch of a sound source.

The stereo field has also been utilised to represent three-dimensional virtual worlds. In this case, the direction to an object is conveyed to the user by means of the stereo field, while the distance is conveyed by the volume of the sound

source. The pitch of a sound source is sometimes altered to indicate whether an object is behind the player, since this is not possible using the stereo field alone. However, three-dimensional sound scenes are better represented by binaural audio, which can be simulated using head related transfer function (HRTF) filters [151]. These filters, when applied to a sound source, simulate the transformations of a sound usually caused by a listener's ear and head shape.

In addition, audio games usually provide a number of tools which the player can use to explore the environment [152]. These include simulated sonar (which produces a beep for every object located within a defined range of the listener), footstep sounds which echo in empty spaces, and speech synthesis that describes aspects of the world not easily conveyed by sounds. One common functionality is a command which, when triggered, describes an object in a specific direction relative to the player. The player may, for instance, request the name of the object in front of them, or the object to the left or the right side.

5.5 Discussion

Currently, linear methods remain the most effective way for blind readers to study mathematical content. However, a linear representation does not directly convey the vertical placement of elements in the source equation. This can be problematic if the author has used vertical placement in a novel way, or as part of a larger graphical depiction. Linear representations therefore also require a greater cognitive effort, since the reader must form the two-dimensional representation in their mind. For sighted readers, this has already been done, and hence they are free to focus on the mathematics itself.

As a two-dimensional representation of an equation can also be considered to be a diagram, a method for directly accessing graphical information should potentially also be useful for reading two-dimensional mathematics. Extending our approach from Section 4 by also conveying text via synthesized speech or braille, as well as the addition of navigational conventions based on accessible games, might provide the reader with enough context to interpret equations. Such a method would not solve all challenges regarding access to mathematics, but it should provide an additional representation of content for blind readers, and may therefore contribute to a more complete picture of the source material. Access to two-dimensional mathematics might also aid communication between blind and sighted peers. Finally and importantly, a large body of mathematical content is also not yet available in a linear form, and a two-dimensional way of accessing mathematics can potentially provide access to that material.

5.6 Summary and Conclusion

This chapter has provided an overview of the state of mathematical accessibility for blind readers. Blind readers access mathematical content via either braille or synthesised speech. Currently, there is no agreed international standard for accessing mathematical content and as a result, many countries have developed their own systems.

Many solutions have been developed to convert structured electronic mathematics, like LaTeX or MathML, into accessible formats like braille or speech. Most of these solutions are designed to output a particular braille code. A problem common to these systems, however, is that (with the exception of Dot-Plus) they are linear in nature. Accessing mathematics in a linear way holds various advantages to a blind person. For example, it easily integrates with systems for reading other content like literary text. A linear representation does, for example, not require an alternative representation of two-dimensional information like tactile drawings, which are costly to produce. It is also easier to adapt controls designed for text reading to allow the reading of linear mathematics. For example, most electronic braille displays are only able to display one line of braille at a time.

However, one major disadvantage of a linear system of reading maths is that the content tends to be more bulky than the corresponding printed alternative. It also does not immediately convey the structure of the equation to the user. When a sighted person looks at an equation, the structure is immediately apparent, and little parsing is necessary before they can start to manipulate it cognitively. In contrast, when presented with an equation in linear format, a blind user has to first convert the equation into a suitable mental form, by, for example, separating out fractions and terms. A third disadvantage of linear mathematics is that it impedes communication between blind readers and their sighted peers, as the linear structure of an equation might be subtly different from its two-dimensional equivalent.

Equations are also often displayed only as images, making them difficult to access without manual translation by a sighted translator. A few systems have been developed to automatically extract equations and even some graphs from images by means of pattern matching. But without access to the source image, a blind person has no way of verifying the correctness of the results delivered by these systems.

An example of an alternative approach to the presentation of relational information is that used by accessible virtual worlds such as textual and audio games. Textual games convey structures such as multiple rooms interconnected by exits to the user via textual descriptions, while audio games utilise sound to indicate the layout of the environment. This second approach can therefore be considered to be a type of audio-visual sensory substitution.

The direct translation made possible by sensory substitution approaches provides some attractive advantages when used to supplement linear represen-

tations of mathematical equations and other semantic spatial information.

First, a linear representation of the content is often not available, and in this case a direct translation approach offers a method for accessing content that would otherwise be inaccessible to a blind reader. Second, it can facilitate communication between blind and sighted colleagues, as blind people would have access to the same spatial representation as their sighted counterparts. Finally, it would allow blind authors to verify the visual representation of their work after typesetting it with a linear markup language like braille mathematics or LaTeX, as well as serving as an accessible way of verifying the correctness of linear representations automatically produced by software like InftyReader.

The following chapter describes an extension to the approach we introduced in Chapter 4, to allow the direct exploration of equations in previously inaccessible PDF documents.

Chapter 6

Approach to Accessibility of Mathematical Material

In its visual form, a mathematical equation can be considered as a type of diagram. Print mathematical equations rely strongly on two-dimensional placement to convey meaning. Many mathematical elements also convey meaning by the extent of their shapes, for example, the length of a fraction line, or the extent of a square root symbol.

Current mainstream approaches enabling accessibility to mathematical content for blind readers function by conveying the equations in an alternative (usually linear) form. This is possible since print mathematical equations are not pictorial in nature, that is, the graphical depiction is a representation of the semantic meaning. However, this approach requires access to mathematical content in a semantically rich format, such as the original source document or a born accessible format like Epub. Most published technical works are delivered as untagged (inaccessible) PDF documents, in which mathematical expressions are presented as image data interlaced with text. Such formats must be converted to a semantic form before they can be made accessible, usually via manual intervention. OCR solutions exist which can automate this process, but without access to the source material, blind readers are unable to verify the correctness of the recognition.

An alternative for enabling access to current published mathematical material is to offer a method which allows blind readers to access the pictorial representation of mathematical content directly. This would allow blind readers to read currently inaccessible mathematical material when no semantic format is available, and to verify the output of mathematical OCR software. Such an approach would also allow blind mathematical content producers to verify the visual representation of their work.

In Chapter 4, we have described an approach for making two-dimensional graphical content accessible to a blind reader. This is accomplished by an extension to the vOICE algorithm first proposed by Meijer [20]. Our extension allows a blind reader to read diagrams and other graphical material by way of

special gestures on a touch screen.

In this chapter we consider a second type of complex content, in particular two-dimensional print equations. We draw on the methods used to represent virtual worlds in text adventures in order to represent the spatial layout of the mathematical equation, and to allow it to be interactively navigated and explored by the reader. Textual elements of the equation, such as symbols, can be verbalised as synthesized speech or braille, mediums with which the user is already familiar. To render graphical elements that cannot be verbalised in this way, such as roots, fraction lines, and many types of brackets, we draw on the audio-visual sensory substitution algorithm we have already applied to diagrams in the previous chapters. By providing a flexible and interactive interface, the reader can gradually explore the equation and map out both the identity of its constituent parts as well as their spatial position relative to each other. By making use of synthesised speech or braille where possible, this exploration can proceed in the most efficient and already familiar manner for the reader.

6.1 Proposed Approach

Despite many initiatives and standards that aim to improve the accessibility of scientific and technical documents, the reality unfortunately remains that most such publications are only available in PDF format without any accessible markup. These documents consist of plain text interleaved with vector drawing instructions (to render equations and vector drawings) and embedded rasterised images (to render bitmap images). While such documents can be read using a screen reader, only the plain text portions are accessible. Reading the text embedded in the diagrams and equations of these documents is of very limited use to a blind reader, since the text on its own cannot be used to interpret the diagrams or equations without graphical information to provide the necessary context. Even when PDF documents are tagged, they usually do not include sufficient semantic information. The PDF document of a technical paper may, for instance, include tags denoting tables and headings, but no tags denoting the semantic information of equations. Hence the current reality is that, although screen readers provide access to the textual material present in technical PDF documents, the information contained in equations and in figures remains overwhelmingly inaccessible.

To address these limitations, we have developed a procedure to convey the graphical information contained in a mathematical equation by way of audio-visual sensory substitution, while continuing to allow access to symbols and other textual elements via synthesised speech or braille. Our approach is implemented as a non-visual content browser for graphical information that is interlaced with text, such as a mathematical equation. This browser draws on the techniques used to navigate and map virtual worlds in text adventure

games in order to represent the spatial relationships between elements of the equation. However, it also allows graphical information that cannot be identified using synthesised speech to be explored using an adaptation of the vOICe sensory substitution algorithm first described by Meijer [20] and subsequently adapted in our own work to allow the focused exploration of graphical content in electronic documents (described in Chapter 4). For the current study, we focus specifically on access to mathematical equations that are graphically represented in inaccessible PDF documents. However, our proposed approach is in principle also applicable to other two-dimensional representations of information, such as charts or infographics.

Our browser provides blind readers with two interface modes, both of which allow the two-dimensional structure of an equation to be explored. The elements of the equation are considered to be a kind of map of the virtual world being explored. The reader is always positioned at one of the elements of the equation, referred to as the **focus**. The focus is always a textual item that can, for example, be spoken as synthesised speech. Because the geometric arrangement of the equation is preserved, the reader is able to discover how items are arranged relative to each other.

Non-textual components of the equation, such as fraction lines, square roots and most brackets, are considered to be graphical information and their geometric shape is conveyed using the vOICe algorithm. Since these non-textual graphical parts of an equation are represented in PDF documents as graphical instructions for which the meaning cannot be easily inferred and therefore cannot be rendered as synthesised speech, they can never be in focus and are referred to as “not navigable”. Instead, they must be explored from the vantage of the surrounding navigable elements.

Text mode enables exploration using textual commands in a style similar to that used to explore the virtual worlds of text adventure games. There are commands to examine the focus, and also to move to elements adjacent to the focus. Output is provided in the form of verbose textual descriptions, containing information about the elements of the equation currently in focus and their spatial placement.

Graphical mode allows the user to explore the equation by using keys on the keyboard, as well as gestures on a touch screen, if one is available. This interface therefore does not rely on explicit textual commands to be typed in order to move around the equation. Textual content that is encountered is announced by the screen reader, while graphical content is rendered as audio interactively by the adapted vOICe algorithm.

Our approach requires two sources of data: a rasterised image of the document as it would appear on-screen, and the set of constituent textual elements with their locations and bounding rectangles. The textual elements and locations are obtained automatically by parsing the PDF document using a PDF parser like Poppler [153]. To facilitate navigation, a document model of the equation under consideration is constructed using all textual elements that are

```

for each textual element 1 in input_list {
  for each textual element 2 in input_list (skipping textual
    element 1) {
    if textual elements 1 and 2 are not within line-of-sight {
      continue;
    }
    let dir be the calculated direction from textual element 1 to 2
      (up, up right, etc);
    if textual element 1 has no connection of direction dir, or if
      element 2 is closer {
      set connection with direction dir from textual element 1 to
        textual element 2;
    }
  }
}
}

```

Figure 6.1: Pseudocode of the algorithm used to construct the connections of the document object model (DOM).

identified by the PDF parser.

6.1.1 Document Object Model

The document object model (DOM) consists of a graph constructed using the navigable elements that can be identified from the source equation. Non-navigable elements are not contained within the DOM, but are extracted from the rasterised rendering of the equation before sonification (described in Section 6.1.2). For each navigable element, a node is created containing as data the text of the element as well as its location and bounding rectangle. Thereafter, edges are added between neighbouring nodes. Each node may have up to 12 connected edges, representing four primary directions (left, up, right and down), each with three secondary variants such as up-left, up-centre, and up-right.

To compute the edges, a few simple rules are followed:

- A node may be connected to another node by only one edge of a certain kind.
- The possible neighbours of a node are prioritised by shortest distance.
- An edge between two nodes is only inserted if it does not cross any other nodes.

The pseudocode of the algorithm used to construct the edges of a DOM is given in Figure 6.1.

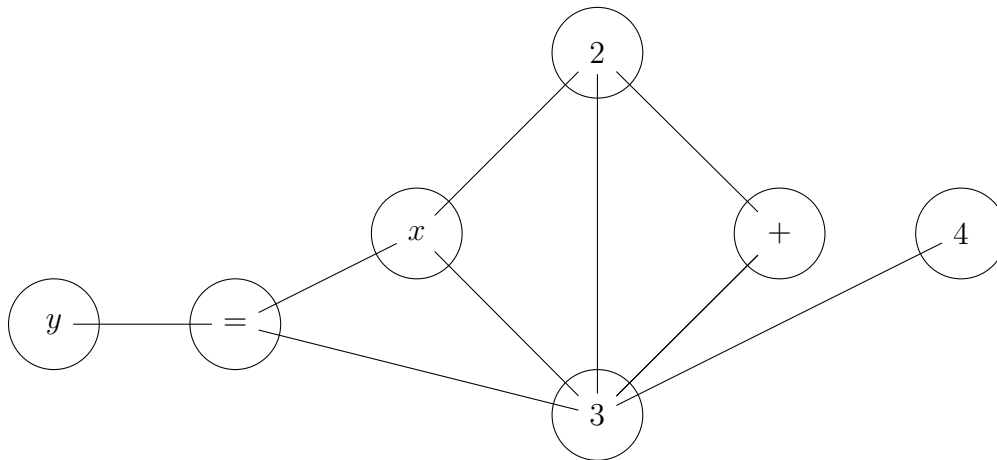


Figure 6.2: Document Object Model for $y = \frac{x^2+4}{3}$.

As an example, consider the equation $y = \frac{x^2+4}{3}$, for which the DOM is shown in Figure 6.2. The DOM contains only textual elements, purely graphical elements are sonified directly from the rasterised rendition of the equation. Our algorithm for producing a DOM is deterministic, therefore any given equation will always result in the same DOM.

To illustrate how the DOM depicted in Figure 6.2 was produced, let us consider the process for adding edges to nodes. To calculate the edges from y , we consider the direction from y to every other element, sorted by shortest distance. The closest node to " y " is "=", therefore we add an edge from " y " to "=" with the direction "right." Any other edges originating from " y " will bisect "=", therefore no further edges can be added to " y ". Let us now consider the process for calculating the edges of " x ". We add edges to the closest nodes, which are "=", "2", and "3", representing the directions "left", "right and up", and "down and right." If we consider an edge between " x " and "4", we find that such an edge would bisect "2", and is therefore not allowed. Similarly, an edge between " x " and " y " would bisect "=", and is therefore also not added. We use a similar process to calculate the edges from every other node in the DOM.

Note that non-navigable elements like fraction lines and the extent of square roots are not contained within the DOM, and are therefore not taken into account when edges are inserted. These elements are extracted from the rasterised rendering of the equation at the time of sonification.

6.1.2 Sonification and Navigation

Both text mode and graphical mode allow the user to move among the navigable elements of an equation. The focus, which is initially the top left navigable element of the equation, is shared between both interface modes. Since graphical elements of the equation are not navigable and can therefore never be the

focus, our approach is to allow users to interpret such graphical elements by means of audio-visual sensory substitution, using the surrounding navigable textual elements as context. From the current focus, sonification of graphical content can be requested in the four primary directions. For instance, if the current focus is the symbol x , the reader can request any graphical content directly above this x to be sonified.

Sonification of non-textual graphical content is accomplished by constructing a bounding box located in the desired direction relative to the focus element. The bounding box extends from the edge of the focus in the requested direction, up to the edge of any element within line of sight in the same direction, or the edge of the screen. For sonification above or below the focus, the left and right edges of the bounding box are aligned with the edges of the focus. However, for sonification to the left or right of the focus, the bounding box extends vertically to include lines with non-zero pixels. Therefore, sonifying graphical content to the left or right of the focus will usually play the entire graphical element, while sonifying above or below will only play the part directly in line with the focus. This allows the user to hear symbols like brackets in their entirety, but elements like fraction lines and the extents of square roots in discrete segments relative to navigable elements. The reason for this is two-fold:

1. Initial testing suggested that a bracket is difficult to interpret when this symbol is only partly sonified, while fraction lines and the extents of square roots are not. This is because brackets include much of their recognisable shape in the vertical dimension, which is rendered concurrently by the vOICE, while fraction lines contain information only in the horizontal direction which is rendered as time.
2. For brackets, it is sufficient for users to know whether they are present next to a navigable element or not, but for fraction lines and the extents of square roots, it may be important to know exactly which part is above or below a navigable element. The latter is particularly important when more than one graphical element is above or below a sequence of navigable elements. For example, consider the equation $y = \frac{3}{\sqrt{x+2}}$. The two elements x and 2 are both below the fraction line, but x is also below the extent of the square root. In order to correctly interpret the equation, the reader must be able to tell that x has two lines above it, but 2 has only one.

6.1.3 Text Mode

Text mode browsing views the constituent elements of an equation like locations in a text-based adventure game. The equation can be explored by issuing text commands to, for example, inspect the current focus, to move left, right,

up or down from the current focus, or to describe any graphical elements that surround the focus. Each command results in informative textual output that describes a particular aspect of the equation. This output is conveyed to the reader as synthesised speech or braille, facilitated by a screen reader. The structure and spirit of these textual commands are deliberately intended to loosely follow the conventions established by textual virtual worlds. To speed navigation, the responses to many commands include hyperlinks, which can be activated to trigger follow-on commands relevant to the context.

A typical command consists of a keyword describing the action to be performed, followed by an optional argument, for example, the name of an element or a direction. If a command operates on an element, but no element is specified, the command is assumed to refer to the focus. For example, the “play” command, when followed by the direction “right” will sonify the element immediately to the right of the focus. However, when no element is specified, the element currently in focus is sonified.

Movement is facilitated by commands describing the direction, such as “left” or “up”. A secondary direction may be specified to refine the movement. For example, the “right up” command will move the focus diagonally right and up from the current focus. The “look” command allows users to obtain a description of the current focus, or a specific element if provided as an argument. This command is also automatically issued when the navigation begins, to provide users with an immediate overview of the initial focus.

The following information is included in an element’s description:

- The textual content of the navigable element, for example, a mathematical symbol (like “ Ω ”) or the text string (like “exp”).
- The location of the element within the equation, as normalised horizontal and vertical Cartesian coordinates between the values of 0 and 100.
- A list of adjacent elements, with their respective relative directions. These are rendered as links which, when activated, move the focus to that element. An example of such output might be “From this element, the following elements can be reached: Up: “ x ”, Right, “2””
- A list describing whether non-navigable graphical content can be sonified in each possible direction. An example of such output might be, “There is additional graphical content left, up, and right”. The items in this list are hyperlinks which sonify the relevant graphical content when activated.
- A link which can be activated to sonify the navigable element being described.

In addition, when the “look” command is issued without arguments, the output also contains a list of items that are vertically aligned with the focus. These horizontally-sequential elements are often terms of an expression. Users

are also notified when the current line changes by a notification of the number of elements in the new line, for example, after a movement command.

6.1.4 Graphical Mode

As an alternative to the text-driven exploration described in the previous section, graphical mode allows the user to explore the elements of an equation using the keyboard cursor keys to trigger movement. As the user navigates to a new element, it is announced by the screen reader after which the shape of the element is sonified using the vOICe algorithm. The reader is also informed when the vertical position of the focus changes as a result of a horizontal movement, or when the horizontal position changes as a result of a vertical movement. For example, when the reader moves to the right, and the new focus is higher than it was previously, as it might be when moving into the numerator of a fraction, this is announced by the word “raised”. By learning to interpret these announcements, the reader can identify the spatial positions of the elements of an equation, such as exponents and fractions. When the user navigates to an adjacent element, and in the process passes over non-navigable graphical information, that graphical information is sonified before that of the newly focused element.

As described above, the document model allows each node to have up to 12 edges, denoting the four primary directions each with three secondary variants. However, since there are only four cursor keys on a standard keyboard, the same fine-grained navigation available in text mode is not possible. Instead, horizontal movement prioritises nodes in the upward direction, while vertical movement prioritises nodes in the leftward direction. This decision was based on initial user feedback which suggested a preference for reading the numerator of a fraction before the denominator when navigating horizontally, as well as a preference for first reading the left-most element in a term when navigating vertically. Although the navigation options in graphical mode may appear more limited than those available in text mode, and as a result sometimes require more steps to arrive at an intended destination, this is mitigated by the much greater speed at which cursor-key driven navigation is possible.

When a touch screen is available, graphical mode also allows the user to explore the content using gestures. Each keyboard command has an equivalent touch gesture, but a touch screen has the benefit of making a more spontaneous exploration of the content possible. As the user moves one finger across the screen, the entire column of the image situated beneath the point of contact is sonified as a tone chord using the vOICe algorithm. In addition, the portion of the column immediately under the fingertip is sonified with greater intensity. This is intended to allow targeted and interactive graphical exploration of the equation, while still providing context about the geometrically surrounding area. Finally, a two-finger gesture allows the line segment between the fingertips to act as a scanner, with the pixels along this line sonified as a tone

chord. By moving the fingertips closer together or further apart, or by rotating them around each other, the sonification of shapes with any orientation and size can be achieved. This interactive and localised image exploration is not possible with the classical implementation of the vOICe algorithm.

6.1.5 Integrated Browsing

When browsing, the user is initially presented with text mode. This mode appears similar to a typical computer command line interface, with output displayed above a field for input. The reader may switch to graphical mode at any time. The command line interface is then replaced with a canvas area in which the graphical form of the equation is rendered, and through which the reader can navigate using the cursor keys. Sonification is provided immediately as the focus changes, while textual notifications regarding the focused element are displayed in a status area and presented by the screen reader.

6.2 Example Session

To illustrate the implementation of our method, let us examine the practical exploration of the equation $y = \frac{x+2}{\sqrt{x-1}}$.

6.2.1 Text Mode

When the user first opens the software, a message is displayed containing information on how to access the help system, as well as a description of the user's initial location. In the following transcript, embedded links in the output are indicated in blue, and surrounded by square brackets. These links can be used to issue followup commands, as an alternative to entering commands via the keyboard.

Welcome to Sirin, the Sonified Image Reading INterface.

For help, type help.

You can also [\[view the documentation for command mode by clicking this link.\]](#)

Sirin loaded.

Textual content loaded

Image loaded. Width: 300 height: 140

You are currently viewing the document

The current line is:

[\[y\]](#) [\[=\]](#)

The current focus is:

y

Location: 11, 42

From this element, the following elements can be reached:

right: [=]

Use the [play] command to sonify this element.

Use the [play all] command to listen to the whole screen.

The description of the initial location contains a list of elements in the current line segment, in this case the “y” and “=.” A line segment is defined as a sequence of elements with the same vertical position. The user also received a detailed description of the focus (in this case, the element “y”), and the directions to nearby elements are indicated as well. Notice that the description contains the location of the focus, which consists of the screen coordinates normalised to a value between 0 and 100. Notice also that the description concludes with instructions for sonification.

The user is now able to issue commands to move the focus, or to query surrounding elements.

> right

Moved right

=

Location: 31, 42

From this element, the following elements can be reached:

right and down: [√]

left: [y]

right and up: [x]

Use the [play] command to sonify this element.

The user issued the command for moving the focus right, and received a description of the new focus (the “=”). Here, the list of nearby elements contains the direction and label of the previous focus, the “y.”

> right

Moved right and up

Entering line segment with 3 items

x

Location: 58, 21

From this element, the following elements can be reached:

right: [+]

down: [√]

left and down: [=]

There is extra graphical information [down].

Use the play command with a direction to play the extra graphical information.

Use the [play] command to sonify this element.

The “=” was connected to two elements in the right direction, the “ x ” right and up, and the square root symbol right and down. When the user issued the instruction for moving right from the “=”, the system prioritised the “ x ”. This strategy allows the user to read the numerator before the denominator. The user was also notified that the focus entered a new line segment, containing three items. Finally, the user was notified of extra graphical information below the focus.

```
> play down
played down from [x]
```

The “play” command allows the user to sonify a specified region on the screen. In the above case, the user chose to have the region below the “ x ” sonified. As the region corresponds to the fraction line, the software would have produced a constant tone indicating a horizontal line.

The user may now continue to explore the numerator, by navigating right as far as possible.

```
> right
Moved right
+
Location: 69, 21
From this element, the following elements can be reached:
right: [2]
down: [x]
left: [x]
There is extra graphical information [down].
Use the play command with a direction to play the extra graphical information.
Use the [play] command to sonify this element.
```

```
> play down
played down from [+]
```

```
> right
Moved right
2
Location: 79, 21
From this element, the following elements can be reached:
down: [-]
left: [+]
There is extra graphical information [down].
Use the play command with a direction to play the extra graphical information.
Use the [play] command to sonify this element.
```

```
> play down
played down from [2]
```

The “look” command can be used to examine a specific element, by providing the label of the desired element as an argument. The focus is also set to the provided element, allowing followup commands to continue from that location.

```
> look x
x
Location: 58, 21
From this element, the following elements can be reached:
right: [+]
down: [√]
left and down: [=]
There is extra graphical information [down].
Use the play command with a direction to play the extra graphical information.
Use the [play] command to sonify this element.
```

In the above example, the “look” command was used to examine the “ x .” The focus was also returned to the x .

```
> down
Moved down
Entering line segment with 4 items
√
Location: 53, 75
From this element, the following elements can be reached:
right: [x]
left and up: [=]
up: [x]
There is extra graphical information [up].
Use the play command with a direction to play the extra graphical information.
Use the [play] command to sonify this element.
```

```
> play up
played up from [√]
```

Here, the user moved the focus down to the square root symbol, and issued the command to sonify the region above it. This region again corresponds to the fraction line.

```
> right
Moved right
x
Location: 65, 75
From this element, the following elements can be reached:
```

```
right: [-]
```

```
left: [√]
```

```
up: [x]
```

There is extra graphical information [up].

Use the play command with a direction to play the extra graphical information.

Use the [play] command to sonify this element.

```
> play up
```

```
played up from [x]
```

Here the user issued the “right” command to navigate to the right, and the “play up” command to sonify the region above the “ x .” However, “ x ” is below both the fraction line as well as the line indicating the extent of the square root, therefore the user would have perceived two simultaneous constant tones (a chord).

```
> right
```

```
Moved right
```

```
-
```

```
Location: 76, 75
```

From this element, the following elements can be reached:

```
right: [1]
```

```
left: [x]
```

```
up: [2]
```

There is extra graphical information [up].

Use the play command with a direction to play the extra graphical information.

Use the [play] command to sonify this element.

```
> play up
```

```
played up from [-]
```

Once more the user issued the “right” command to move to the “-”, and the “play up” command to sonify the region above it. Here the software would have produced a single constant tone, indicating one horizontal line. The user may therefore have concluded that the “-” is outside the extent of the square root. The user may now explore the remaining elements constituting the denominator using additional “right” commands.

```
> look =
```

```
Entering line segment with 2 items
```

```
=
```

```
Location: 31, 42
```

From this element, the following elements can be reached:

```
right and down: [√]
```

```
left: [y]
```


right and up: [x]

Use the [play] command to sonify this element.

```
> right down
```

```
Moved right and down
```

```
Entering line segment with 4 items
```

```
√
```

```
Location: 53, 75
```

From this element, the following elements can be reached:

```
right: [x]
```

```
left and up: [=]
```

```
up: [x]
```

There is extra graphical information [up].

Use the play command with a direction to play the extra graphical information.

Use the [play] command to sonify this element.

When the user issued the command to examine the “=”, the focus was returned to it. The user then chose to explicitly move the focus right and down, therefore the focus was returned to the first element in the denominator of the fraction, the square root symbol.

```
> play all
```

```
played the screen
```

The “play all” command allows the user to receive a sonification of the entire equation. The sonification does not contain any announcement of textual elements, only the graphical structure of the equation is conveyed to the user, including both textual and graphical elements. However, this function may be used to determine the general structure of the equation, before it is explored in detail.

```
> play y
```

```
played [y]
```

When the label of an element is provided as argument to the “play” command, the region corresponding to that element is sonified. The “play” command can also be issued without arguments, in which case the region corresponding to the current focus is sonified.

6.2.2 Graphical Mode

To switch to graphical mode, the user activates a button provided on the text mode interface. A similar button is also available in graphical mode to return to text mode.

In graphical mode, the user may explore the equation using the cursor keys on the keyboard. As the user navigates among the elements of the equation, a

short textual description of the focus is announced by the user's screen reader, and also displayed in a status bar at the bottom of the window. Graphical mode shares the focus with text mode, therefore the user will start off on the element last described in text mode. The user can at any time request the item with focus to be announced by a key command.

The announcements of the program (also visible in the status bar of the interface) are denoted in the text as follows:

y

Given that the focus is initially on the “y”, the user may move to the “=” by pressing the right cursor key. The software will announce the new focus (the “=”), and also sonify the shape of the symbol, thereby conveying its shape. Pressing any of the other cursor keys will not move the focus, instead, the user will receive an announcement indicating that no items are available in that direction.

=

From the “=”, the user may press the right cursor key again, and the focus will change to the “x”, which is the numerator of the fraction. The software will announce the “x”, and also sonify its corresponding region. The software will also announce the word “raised”, indicating to the user that the new focus is situated higher on the screen than the previous focus. If the user waits for more than two seconds before pressing another key, the fraction line below the “x” will be sonified as a constant tone. Due to the spacing of the equation, the user will also perceive a short segment of the square root below the fraction line. The user may also request the sonification of the region below the current focus (using shift and the down cursor key), which will sonify the fraction line as well. Requesting a sonification of a region in any other direction will result in an announcement indicating that no graphical content in that direction can be found.

x, raised

By subsequent right cursor commands, the user may explore the remaining elements above the fraction line. Navigating left will return the focus to the “x.”

The user can move down to the denominator of the fraction by pressing the down cursor key. The software will sonify the fraction line, and immediately sonify the square root symbol, which is the left-most element in the denominator. The user will also receive an announcement of the square root symbol, as the utterance “square root.” The user may now issue the key command for sonifying the region above the focus, which will sonify the fraction line again.



Pressing the right cursor key will set the focus to the “ x ”, which is under the extent of the root. The user will receive an announcement of the “ x ”, as well as the sonification of its shape. In addition, if the user keeps the focus on the “ x ” for more than two seconds, both the fraction line and the extent of the root will be sonified as a two-tone chord.



To determine the extent of the root, the user may move to the right once more, which will set the focus to the “ $-$ ”. The user may wait for more than two seconds to hear the sonification of the graphical content above the “ $-$ ”, or decide to specifically request that region to be sonified by pressing shift in combination with the the up cursor key. In the case of this equation, the user will perceive a single constant tone, indicating one horizontal line. The user may therefore conclude that the region above the “ $-$ ” consists only of the fraction line, and not the extent of the root.



6.3 Experimental Evaluation

Our proposed method of interactive equation exploration was evaluated by 25 test subjects, 11 of whom were blind and 14 of whom were sighted. Before evaluation, subjects were provided with a training session in which they could familiarise themselves with the method of exploration and the text and keyboard commands. After training had been completed, each subject completed two evaluation phases. In the first, candidates were asked to consecutively identify a number of different equations using only text mode. In the second phase, candidates were permitted to use both text and graphical modes, and to switch freely between them. The exploration algorithm, as well as the training and testing procedures, were implemented using web technologies and were therefore accessible over the internet. This allowed remote evaluation, which also enabled us to comply with Covid19 safety protocols.

6.3.1 Equations Under Review

Given that our approach provides access to mathematical content by the selective and structured sonification of graphical (non-textual) structures, we identified a number of mathematical conventions which are difficult to read without access to this information. These mathematical structures cannot, for example, be correctly interpreted from the plain-text extracted from an

untagged PDF document, as would, for example, be presented by a screen reader.

The following five print mathematical conventions all depend on graphical structure for their correct interpretation, and are not identifiable when rendered as plain text. While these five conventions have been identified for the purposes of the evaluation we will present, the list is not intended to be exhaustive.

- Exponents. After conversion to plain text, exponents are at best presented over two lines, with the exponent in the first and the base in the second. Fractions are presented in a similar way, resulting in immediate ambiguity.
- Fractions. The fraction line is lost during the conversion to plain text. The numerator and denominator are usually presented on separate lines. This is similar to the presentation of exponents, as described above.
- Square roots. The radicand is indicated by a graphical line extending over the argument, which is lost during the conversion to plain text. Hence it is not possible to determine which terms fall under the root from a plain text representation. Depending on the PDF reader, the square root symbol itself may also be presented on a different line from the radicand.
- Large brackets. Depending on the source of the PDF, large brackets are often rendered using fonts with custom glyphs, for which the TTS system cannot identify a corresponding textual description. In print mathematics, the size of a bracket is also often used to convey the relationships between elements of the equation to the reader. Therefore, an accessible two-dimensional view requires the vertical extent of the brackets to be rendered.
- Matrices. After conversion to plain text, matrices are presented over several lines, which appear similar to the plain text representation of fractions and exponents. For reasons similar to those described above, the large square brackets delimiting a matrix are also usually absent from the plain text representation.

On the basis of these five types of mathematical content, we developed two sets of six equations with which to evaluate our approach. The first set was used to evaluate text mode interaction only, while the second was used to evaluate both text and graphical modes of interaction. In both cases, an attempt was made to keep the evaluation short, so as to minimise user fatigue. We also endeavoured to choose equations that are not identifiable from a plain text representation. To achieve this, we chose equations which contain at least

two of the graphical conventions in the list above. The following two tables list the equations we used for the two stages of our evaluation.

Equation	Exponents	Fractions	Roots	Brackets	Matrices
$y = \frac{x}{2} + x^2$	1	1	0	0	0
$y = \sqrt{x} + x^2$	1	0	1	0	0
$y = \frac{2}{\sqrt{x}}$	0	1	1	0	0
$y = \left(\frac{\sqrt{x}}{x-2}\right)^4$	1	1	1	2	0
$y = \left(\frac{x}{2} + 2\right)^2$	1	1	0	2	0
$\begin{bmatrix} 2 & 4 \\ 2 & 4 \end{bmatrix} \times \begin{bmatrix} 1 \\ 2 \end{bmatrix}$	0	0	0	4	2

Table 6.1: Equations for Stage 1 - Text Mode.

Equation	Exponents	Fractions	Roots	Brackets	Matrices
$y = x^2 + \frac{2}{x}$	1	1	0	0	0
$y = x^3 + \sqrt{x}$	1	0	1	0	0
$y = \frac{\sqrt{x}}{2}$	0	1	1	0	0
$y = \left(\frac{x\sqrt{x}}{x+2}\right)^5$	1	1	1	2	0
$y = \left(1 + \frac{2}{x}\right)^5$	1	1	0	2	0
$\begin{bmatrix} 1 & 2 & 3 \\ 2 & 3 & 4 \end{bmatrix} \times \begin{bmatrix} 1 & 2 \\ 2 & 3 \\ 3 & 4 \end{bmatrix}$	0	0	0	4	2

Table 6.2: Equations for Stage 2 - Text Mode & Graphical Mode.

The equations in these two tables were designed to illustrate the current inaccessibility of mathematical content in untagged PDF documents. For example, consider Equation 1 from Stage 1:

$$y = \frac{x}{2} + x^2$$

A current mainstream PDF reader extracts the following textual representation when viewing the equation:

```

x
2
y =+ x
2

```

We see that the equation is represented as text displayed over several lines as a result of the vertical placement of terms. It is, however, not possible to determine from this representation which elements are fractions and which are exponents. It is also not possible to determine the extent of the fraction line, and therefore which terms are part of the numerator and which are part of the denominator. For example, the following are all legitimate interpretations of the above textual representation.

$$y = \frac{2}{x} + \frac{2}{x}$$

$$y = \frac{2}{x} + x^2$$

$$y = \frac{2}{x+x^2}$$

$$y = x + \frac{2}{x^2}$$

As a second example, let us consider Equation 5 from Stage 2.

$$y = \left(\frac{x\sqrt{x}}{x+2} \right)^5$$

The plain text representation of this equation is as follows:

```

√ 5
xx
y =
x +2

```

This equation is therefore displayed over four lines, due to the spacing of the fraction and the exponent. However, the graphical indicators are absent, and therefore it is impossible for a blind reader to distinguish between the exponent and the fraction. If we assume that the equation does not contain brackets, the following interpretations are possible.

$$y = x + \sqrt{\frac{x}{x+2}}^5$$

$$y = x^x + \sqrt{\frac{x}{2}}^5$$

$$y = x^x + \frac{\sqrt{x}^5}{2}$$

$$y = \frac{x\sqrt{x}^5}{x+2}$$

The above examples are intended to illustrate the inaccessibility of equations in untagged PDF documents. The high ambiguity of the textual representation that can be extracted from such PDF documents means that this is not a viable means by which blind readers can access mathematical content. For this reason, and to avoid the unnecessary additional fatigue it would cause test subjects, we did not include this plain text format in our evaluation.

6.3.2 Test Candidates

Evaluation was performed by eleven blind and fourteen sighted human subjects. Subjects were recruited by word-of-mouth, and in the case of blind subjects also by means of mailing lists for blind STEM practitioners. The majority of sighted candidates were recruited locally from the Departments of Mathematics and Engineering at the University of Stellenbosch. However, the majority of blind candidates were recruited internationally. Candidates were between the ages of 18 and 70 and included students (undergraduate and postgraduate) as well as candidates who were not students but had an undergraduate or a postgraduate qualification. Both blind and sighted groups included candidates with mathematical and scientific backgrounds, but also candidates from fields outside STEM such as law and journalism. Table 6.3 lists some of the attributes of the test candidates.

	Blind	Sighted
Undergraduate student	1	1
Postgraduate student	1	7
Undergraduate qualification	4	5
Postgraduate qualification	8	10
STEM background	8	11

Table 6.3: Some attributes of the test candidates.

6.3.3 Training

Since mathematical material remains largely inaccessible to blind readers, it was a challenge to identify suitable blind test candidates. Even when willing individuals were identified, it was found that many did not read mathematics on a regular basis and therefore required an introduction to remind them of the key concepts, especially the typical graphical layout of equations that sighted readers are accustomed to. Hence this training phase, which was carried out prior to the evaluation, was essential.

An online tutorial was developed to re-familiarise participants with the key attributes of mathematical equations as well as to practice the use of the browsing software. The tutorial contained a step-by-step introduction and

explanation of the functionality of the proposed algorithm. This included the commands available in text mode, the commands and gestures available in graphical mode, and an explanation of the sound and synthesised speech output that could be expected while browsing. The tutorial also contained a section that allowed users to listen to sound renderings of example known graphical shapes, including a horizontal line, a diagonal line, and a square root. This was included to familiarise participants with the method of sonification, with which most were also not acquainted.

As many of the blind and visually impaired candidates had no prior experience with print mathematical notation, descriptions of the graphical symbols that form part of the evaluation were also included in the tutorial. These symbols included a fraction line, a square root, and both round and square brackets. Using the tutorial, users were able to practice exploring increasingly complex equations using both text and graphical modes. The tutorial also included a section that allowed users to test their ability to interpret a number of equations.

In addition to the tutorial, interactive training sessions were performed with candidates via online voice communications platforms. Users were guided through the initial questions of the tutorial, and had the opportunity to ask questions.

6.3.4 Testing Procedure

The evaluation consisted of two sets of six equations, as described in the previous section. The first set was used to evaluate the use of text mode for navigation, while the second was used to evaluate the mixed use of text and graphical modes for interaction, where subjects could switch freely between the two.

Like training, evaluation was designed to function over the internet. To achieve this, each equation was delivered as a web page and the browsing software was implemented as an interactive web-based application developed using the Rust and Javascript programming languages. The start page of the evaluation contained links to each equation, and also contained instructions for the candidate. Each link opened the software in a new browser window dedicated to the specific equation, and allowed the subject to explore it. After exploration, the browser window could simply be closed to return to the starting page.

Equations were typeset using LaTeX and rendered as a PDF to fill the screen, scaled to a horizontal resolution of 300 pixels, preserving aspect ratio. Although equations were rendered to a canvas, they were not visibly displayed, thus ensuring a similar testing environment for both sighted and blind participants. Textual symbols, along with their locations and bounding boxes, were extracted from the PDF documents using the Poppler [153] and PdfMinor [154] PDF extraction libraries. For each equation, a table of the

symbols extracted in this way, as well as an image containing the graphical rendering of the equation, were stored. Equations were rendered in isolation for the purposes of this study since the aim was to determine the effectiveness of the proposed browsing method. For an integrated solution, equations can be isolated and extracted from the PDF in which they are embedded, either automatically or based on a selection made by the human reader.

After candidates had completed the training tutorials and declared themselves comfortable using the browsing software, they could proceed to the evaluation stages. The first phase of the evaluation required the participants to explore and identify the first set of six equations using text mode only. Since the textual output was displayed on the screen, it could be announced by the screen reader to blind candidates. Most blind candidates used either the Jaws screen reader developed by Freedom Scientific, or the open source NVDA screen reader developed by NVAccess. However, at least one blind candidate used Linux with the Orca screen reader. The use of screen readers was not required for sighted subjects. In contrast to the blind subjects who were all familiar with the use of screen readers, the sighted subjects were not. Therefore, requiring the sighted subjects to use screen readers in order to enforce the same test procedure among all participants could negatively affect results in a way that is not indicative of the effectiveness of the methods under evaluation.

The second phase of the evaluation required the participants to explore and identify the second set of six equations, but allowed them to use both text and graphical exploration modes, and to switch among these two freely. For graphical mode, textual output consisted of short notifications which were displayed on a status bar readable by sighted participants and announced by the screen reader of blind participants.

To allow later analysis, a record of the actions performed and time spent during evaluation was kept for each test subject. Candidates were asked to write down, in a separate document and using full English sentences, what they believed each equation to be. Candidates were requested to be as explicit as possible in describing the structure of the equation, for example, noting the start and end of fractions, roots and exponents. Candidates who were familiar with LaTeX were invited to provide their answers in that notation.

6.4 Scoring

The transcriptions of the equations provided by the test subjects differed widely in style. Some were provided in LaTeX notation, others as ASCII maths, and many as full and descriptive English sentences. Due to this heterogeneous format, all responses were manually and individually assessed for correctness.

Two figures of merit, both expressed as percentages, were used to score the responses made by the test subjects. The first, which we will refer to as ‘completely correct’, is based on a binary score for each equation in the test indicating whether it was perfectly correct or not. However, even when an equation is not transcribed with perfect accuracy, the degree to which it is errorful can vary. Therefore, a second figure of merit, which we will refer to as the ‘correctness score’, was introduced. The correctness score is based on the number of symbols and graphical elements in the equation concerned, and is 100% only when all symbols and graphical elements are both correctly identified and correctly placed. The correctness score also had the effect of normalising the responses, allowing them to be compared.

As mentioned previously, the equations used in the evaluations were designed to contain at least two problematic elements each. Problematic elements are those which cannot be unambiguously identified from a plain text representation derived from an untagged PDF document. The correctness score is therefore based on six indicators: the five possible problematic elements and textual symbols. For each equation, we calculated the maximum score based on the number of times each of these elements was correctly identified, as well as the number of times it was correctly placed. For example, Table 6.4 illustrates the scoring of Equation 1 of Stage 1. The maximum possible score for this equation is the sum of the individual scores, a total of 18.

Symbols	Identified	7
	Placed	7
Exponents	Identified	1
	Placed	1
Fractions	Identified	1
	Placed	1
Roots	Identified	0
	Placed	0
Brackets	Identified	0
	Placed	0
Matrices	Identified	0
	Placed	0

Table 6.4: Composition of the correctness score for the equation $y = \frac{x}{2} + x^2$.

For each equation transcribed by a test subject, the correctness score is calculated by subtracting the number of inserted and deleted elements, as well as the number of incorrectly placed elements, from the maximum possible score. Table 6.5 illustrates the calculation of this score for a partially correct transcription of Equation 1 in Stage 1. Since the sum in the table is 10, this

must be subtracted from the score of 18 for a perfect transcription, yielding a correctness score of 44%.

		Correct Answer	Candidate's Answer	Errors
Textual Elements	Identified	7	6	1
	Placed	7	3	4
Exponents	Identified	1	1	0
	Placed	1	0	1
Fractions	Identified	1	0	1
	Placed	1	0	1
Roots	Identified	0	1	1
	Placed	0	1	1
Brackets	Identified	0	0	0
	Placed	0	0	0
Matrices	Identified	0	0	0
	Placed	0	0	0
Total		18	12	10

Table 6.5: Example scoring table for the hypothetical case where the equation $y = \frac{x}{2} + x^2$ is transcribed as $y = x^2 + \sqrt{x}$.

This type of scoring table can also be used to describe how specific elements impact on the final score of a particular equation.

- The original equation contains seven textual symbols, but the provided answer contains only six. We therefore count two errors: one for a symbol not identified, and one for a symbol not correctly positioned.
- Although an exponent of two was correctly identified, it was not placed in the correct position. Therefore we count one error. Furthermore, since the expression containing the exponent also comprises two textual elements, we need to subtract an additional two from the final score to account for the two displaced textual elements.
- The original equation contains a fraction, while the provided answer does not. We therefore count two errors: one for a fraction not identified, and one for a fraction incorrectly positioned. We also count one error for the textual elements.
- The provided answer contains a square root, while the original equation does not. We therefore count two errors, one for the square root incorrectly identified, and one for the square root incorrectly positioned.

This scoring procedure was used to evaluate the answers provided by the participants.

Equation	Stage 1			Stage 2		
	Blind	Sighted	Overall	Blind	Sighted	Overall
1	81.8%	78.6%	80.2%	81.8%	92.9%	87.3%
2	90.9%	78.6%	84.7%	100.0%	100.0%	100.0%
3	81.8%	78.6%	80.2%	90.9%	100.0%	95.5%
4	45.5%	50.0%	47.7%	54.5%	64.3%	59.4%
5	63.6%	78.6%	71.1%	72.7%	85.7%	79.2%
6	63.6%	92.9%	78.2%	54.5%	92.9%	73.7%
Average	71.2%	76.2%	73.7%	75.8%	89.3%	82.5%

Table 6.6: Number of completely correct responses per equation for both evaluation stages.

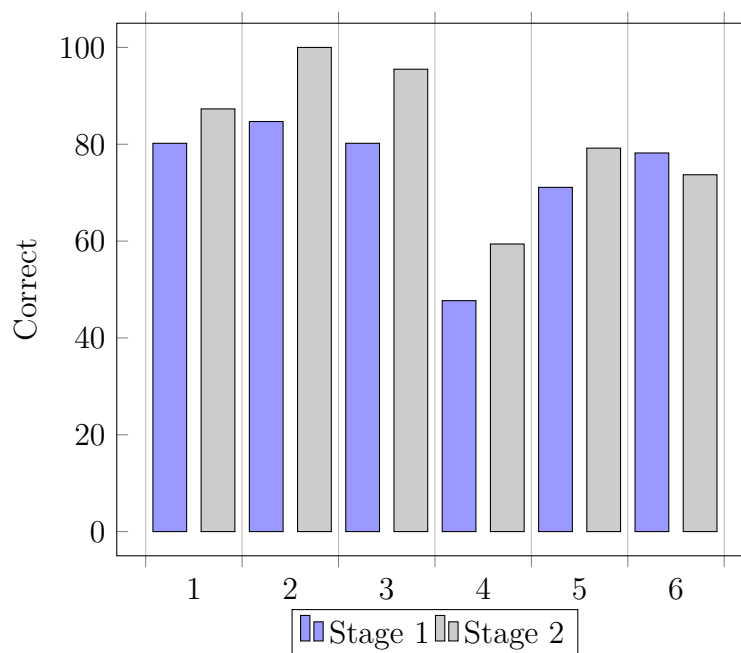


Figure 6.3: Number of perfectly correct responses per question for each stage.

6.5 Results and Discussion

Table 6.6 reports the number of times each of the six equations in Stages 1 and 2 were transcribed without any error by the 25 test subjects, while Table 6.7 reports the corresponding correctness scores. We see that, across all subjects and both stages of the evaluation, 78.1% of the responses were perfectly correct (73.5% and 82.7% respectively for blind and sighted subjects). The corresponding correctness score was 95.4% overall, with 93.3% and 97.6% for blind and sighted groups respectively.

Equation	Stage 1			Stage 2		
	Blind	Sighted	Overall	Blind	Sighted	Overall
1	96.0%	96.0%	96.0%	95.0%	98.4%	96.7%
2	98.0%	98.8%	98.4%	100.0%	100.0%	100.0%
3	95.5%	97.0%	96.2%	99.4%	100.0%	99.7%
4	88.5%	92.6%	90.5%	91.9%	95.7%	93.8%
5	90.5%	95.8%	93.1%	93.2%	98.9%	96.0%
6	88.8%	98.4%	93.6%	82.3%	99.4%	90.9%
Average	92.9%	96.4%	94.6%	93.6%	98.7%	96.2%

Table 6.7: Correctness scores per equation for both evaluation stages

Table 6.6 also shows that, for all except equation four, more than 70% of the equations were, on average, transcribed perfectly. Equation four was one of the most difficult equations in both stages, containing all elements identified in Section 6.3.1 except matrices. As expected, the correctness scores in Table 6.7 are higher than the corresponding completely correct scores in Table 6.6, and are all above 90%. This indicates that most elements in an equation were transcribed correctly both in terms of identity and geometric placement.

The tables also show that, even though sighted subjects were not able to see the equation they were exploring, they attained slightly better results on average than blind subjects for both stages of the evaluation. However, this difference was found to not be statistically significant ($p < 0.1$.) It must be borne in mind that, although the equation itself was not displayed on the screen, sighted subjects were able to see the textual output generated during browsing. Although blind subjects also had access to this information, since the text was synthesised as speech by the screen reader, arguably the ability to read the text allowed sighted candidates to more easily pick out relevant information. It should also be highlighted that two blind candidates attained a perfect score on all equations in both stages, which demonstrates the efficacy of our approach even when used entirely non-visually¹.

Another advantage that sighted subjects arguably have is their familiarity with visual mathematics. In performing the evaluation, it was discovered that most blind candidates were unfamiliar with the visual appearance of elements such as square roots, brackets, and matrices, and had to learn these during the training phase, in addition to the use of the browsing software. Sighted subjects, on the other hand, were all familiar with the mathematical notation. Furthermore, sighted subjects were able to form a two-dimensional representation of the equation they were exploring using pen and paper, while blind candidates had to resort to a linear representation like LaTeX or braille to

¹Although not formally analysed, the responses contributed by blind candidates with a STEM background did not significantly differ from those with other backgrounds. For example, of the three candidates without a STEM background, one attained a near-perfect score. We suspect that a general familiarity with technology was ultimately more important.

help them.

Tables 6.6 and 6.7 and Figure 6.3 also show that both blind and sighted groups attained higher accuracies in Stage 2, where they could use both text and graphical modes of exploration, than in Stage 1, in which only text mode could be used. This difference is statistically significant with $p < 0.02$. In Stage 1, every question had at least one incorrect response, while for Stage 2, Question 2 was correctly identified by all candidates. This may indicate that the graphical mode adds to users' understanding of equations. However, all sighted candidates, and ten of the eleven blind candidates also used text mode in Stage 2 for some of the equations, especially for Equations 4 and 6 (the matrix equation). It should also be remembered that candidates always completed Stage 1 before proceeding to Stage 2, which may also contribute to the higher accuracy attained in the latter.

As can be observed in Figure 6.4, sighted subjects were able to complete Stage 1 more quickly than blind subjects ($p < 0.01$), while in Stage 2 the average durations for the two groups were similar. We believe that the ability to use graphical mode and to switch between this and text mode provided the blind subjects with greater flexibility in how to construct an understanding of the equation structure. This additional flexibility was less useful to sighted subjects who could make better use of the verbose textual output provided by the browser.

Sighted candidates also more often used the links in text mode for follow up commands, as opposed to blind candidates, who preferred to type out the commands ($p < 0.01$). This may also have contributed to the observed time difference between the two groups.

If we examine the results in terms of the particular type of equation that subjects are asked to identify, we see that blind subjects attained lower scores than sighted subjects for Equation 6 in both phases ($p < 0.01$). Equation 6 included a matrix, and this was incorrectly identified by four blind subjects in Stage 1, and five blind subjects in Stage 2. However, the same three subjects contributed incorrect responses in both stages. This suggests that some blind candidates were not familiar with matrices, a hypothesis that is supported by the form of some of the incorrect responses, such as the placement of multiplication sign inside the matrix itself. In addition, one candidate omitted the matrix equation entirely in Stage 2. The highly two-dimensional structure of a matrix makes it especially difficult to interpret using the linear reading methods commonly available to blind readers. In addition, most braille displays can only display one line at a time, and can therefore also not express the two-dimensional nature of matrices.

Analysis of the answers submitted by blind subjects shows that most errors are due to missing textual elements. For instance, for the first equation of Stage 1, two subjects missed the "2" in the final exponent. For the matrix equations, all but two of the incorrect responses were due to missing textual elements. Across all blind subjects, missing symbols accounted for 53% and

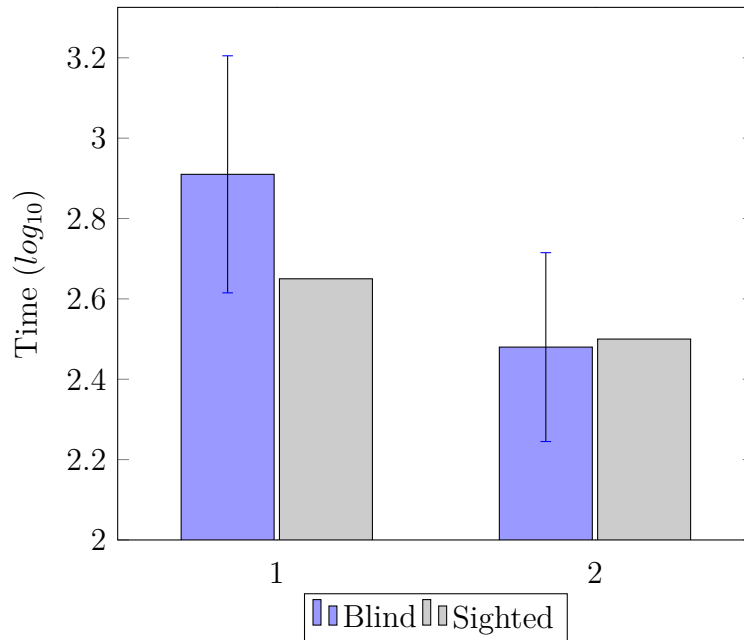


Figure 6.4: Time taken by blind and sighted users to complete Stages 1 and 2 respectively. Vertical bars denote 95% confidence intervals.

38% of incorrect answers for Stages 1 and 2 respectively. It was furthermore observed that subjects often did in fact visit all the elements of the equation during exploration, even though they did not include some of them in their answers. This might indicate that candidates missed the elements when writing their answers, rather than when reading them. Nevertheless, there were cases where subjects missed elements because they had in fact never visited them during exploration. Improving the browser to report whether all textual elements have been visited might go some way in reducing this type of error.

The second most prevalent type of error made by blind subjects was the incorrect placement of brackets. Brackets were either not placed in the correct position, or were not included within the submitted answer at all. As also seen for symbols, brackets were often omitted even though they had been deliberately sonified by the subject during exploration. Across all blind subjects, bracket errors accounted for 32% of incorrect responses for Stage 1, and 31% for Stage 2.

For sighted subjects, the most prevalent errors were bracket errors, followed by missing symbols. This finding is interesting, since unlike many symbols that are reflected in the textual output provided to the reader during exploration, brackets are not, and must therefore be rendered as audio. Hence, for brackets, the sighted subjects do not have the advantage of access to the rendered text. Across all sighted subjects, bracket errors accounted for 31% of errors, while missing symbols accounted for 24%.

The fourth equation was most often incorrectly identified by both groups

of candidates in both Stages 1 and 2. This equation was arguably one of the most difficult, containing fractions with more than one term in the numerator or denominator with the entire fraction enclosed within brackets. Most of the incorrect responses to this equation were due to bracket errors.

6.6 Informal Feedback

In addition to the quantitative results gathered during the two testing phases, informal feedback was gathered from the subjects during the training phase as well as after the two testing phases had been completed. Overall, the browsing approach was positively received, and most subjects were able to use the interface within an hour of beginning training. About half of the blind candidates successfully learnt to use the software by following the online tutorial independently. For these candidates, individual meetings were scheduled before the testing phases to gather feedback about their initial perceptions, as well as to answer any remaining questions.

Although candidates noted the similarity in functionality provided by the two exploration modes, most blind candidates expressed a clear preference for the graphical mode. The graphical mode was perceived to allow faster navigation and an improved grasp of the spatial layout of the equation. This might be because graphical mode is less verbose and uses quicker (single key) navigational commands. One blind candidate noted that the learning curve, and therefore the barrier to entry, was lower for graphical mode.

A number of blind candidates also mentioned that the included links provided in text mode speed up navigation. The links were found to be particularly useful when viewing the output in text mode as a history of the steps taken through the equation. Specifically, previous output allowed navigational steps to be retraced, and the embedded links could then be used to choose an alternative route through the equation.

Although most candidates had no prior experience with the vOICe algorithm, they found the sonification to be intuitive after practice. One blind candidate reported difficulty in distinguishing between multiple tones when played simultaneously (for example, two parallel horizontal lines), and specifically noted the case of a fraction with a square root in the denominator. However, the same candidate was able to identify all equations correctly.

Candidates were asked whether they thought textual elements should automatically be sonified on navigation in exploration mode (the current behaviour), or whether this should be automatic only for non-textual elements (such as fraction lines). One candidate expressed a clear preference for the latter, with sonification on request. However, several other candidates noted that the sonification contributed to their mental image of the equation, and that they preferred automatic sonification. Because the pitch of the sonification is based on the position of elements on the screen and therefore absolute, remov-

ing automatic sonification of textual elements would potentially increase the difficulty in judging whether a fraction line or other element is below or above the current focus. Sonifying the focus along with the graphical element which may be either above or below, allows users to judge the vertical alignment from the relative pitch.

Regardless of their preferred mode of visualisation, eight of the eleven blind candidates reported that the browser provided them with a clearer understanding of the spatial layout used in mathematical equations. In particular, one candidate noted that they had not previously had any understanding of the shape of print square roots, and that they were for the first time able to visualise the geometric shape of this symbol after using the browser. Another candidate suggested that more semantic information should be added to the textual output, even though they found the spatial information useful. It should be borne in mind that our approach was specifically designed to address the accessibility of content with no or insufficient accompanying semantic information, such as untagged PDF documents. Nevertheless, it might prove interesting in future to explore the incorporation of richer semantic textual output, by, for example, applying our methods to content in MathML or other semantically orientated markup.

6.7 Summary and Conclusion

We have introduced a browsing technique specifically aimed at making previously inaccessible mathematical equations, as typically found in untagged PDF documents, accessible to blind or low-vision readers. This browsing technique combines spatial navigational techniques commonly used in text-based adventure games with audio-visual sensory substitution for non-textual elements of the equation. Two modes of interaction have been implemented and evaluated: one purely text-based, and the other more directly based on the geometric layout of the equation and that accommodates gestures when a touch screen is available. During browsing, the reader can switch freely between these two modes of interaction. For audio-visual sensory substitution, we have adapted and extended the vOICE algorithm, first proposed by Meijer, to allow the interactive exploration of graphical elements in an equation, such as root signs, fraction lines and many types of brackets. Such elements are represented as graphics in PDF documents, and can therefore not easily be rendered as synthesised speech.

Evaluation of our approach by both blind and sighted test subjects showed that, after a training session in many cases not exceeding one hour, equations that would be inaccessible using current prevalent screen readers can be identified to a high degree of accuracy. With the exception of one equation in both stages of the evaluation, on average more than 70% of equations were perfectly identified. Indeed, it was possible for subjects from both the blind

and the sighted groups to obtain a perfect score. While sighted candidates expressed a preference for the text-mode interaction, blind candidates made extensive use of the graphical mode, but switched between the two when presented with more complex geometric arrangements such as matrices. Informal discussions with the blind candidates after the evaluation also revealed that, in some cases, additional insight into the layout conventions of mathematical equations has been gained. This indicates that the browsing approach we propose presents a means by which print-disabled readers can, without additional assistance, interactively explore and thereby decipher mathematical content that was, due to its inaccessible format, unfamiliar to them.

The most common type of error made by blind test subjects was the omission of symbols from the equation. This is an aspect we aim to address in ongoing work, since in some cases the reader can be alerted to the possibility that this type of error is being made. In addition, we aim to consider different input sources. Firstly, much older published material is available only in the form of rasterised images embedded in the PDF format, as may be obtained from a document scanner. By including optical character recognition (OCR), it may be possible to also make such content accessible using our approach. Secondly, we would like to consider other non-textual information present in scientific and technical documents, such as graphs and plots. The overall objective is to develop methods and tools that fill the accessibility gaps in electronic documents that are not currently addressed by screen reading software and that continue to impede access to blind and sight-disabled readers.

Chapter 7

Conclusion

This thesis has examined the potential contribution that audio-visual sensory substitution and interactive spatial exploration could make to facilitate the reading of technical material. The accessibility of technical documents requires particular consideration, since these documents contain graphical information (in the form of diagrams and equations), that is required for a proper understanding of the content.

Print-disabled readers access documents via braille or audio. Traditionally, material in these mediums are created by hand, but digital technology allows the automatic conversion of electronic text to accessible formats. This is achieved by the use of screen readers and other software programs which send the text to a hardware braille device (either an embosser, or a refreshable braille terminal), or to a text-to-speech (TTS) system, which is able to announce the text. A screen reader is able to access any textual information in a document, but graphical information such as diagrams and photos cannot be directly announced and must be accompanied by an alternative textual description.

Graphical material can be broadly classified into two types, depending on the function of the image. First, there are semantic diagrams, which depict a visual representation of a logical structure. The information presented by these diagrams can also be conveyed in an alternative (often linear) form, without the loss of information. Examples include equations, bar charts (which can be represented as a table), and graphs (a list of vertices and edges). Second, there are pictorial diagrams, for which the graphical representation itself conveys meaning. These images can be described, but access to the graphical depiction itself is required for a full understanding of the content. Examples include geospatial maps and the shape of sound waves.

Accessibility standards prescribe the use of alternative text (also called alt text) to describe graphical content in documents. Modern document formats such as HTML allow alternative text to be attached to any image element. Unfortunately, many documents that do not conform to these standards are still produced. Scientific material especially is often published as PDF doc-

uments without this accessibility markup (also known as "Untagged PDF"). When these documents are read using a screen reader or other accessibility software, only the plain text can be accessed. Furthermore, even when such accessibility annotations are included, they often do not fully describe pictorial material.

Currently, the state of the art for providing access to graphical information is via tactile diagrams. This requires the use of manual conversion, or specialised hardware devices. Hardware devices include braille embossers which can emboss tactile diagrams on paper, as well as multiline braille displays which can depict tactile diagrams immediately. However, hardware devices such as these are costly and therefore not easily available to the majority of blind consumers. Most diagrams also require manual preprocessing (since the tactile sense has a much lower resolution than the visual sense), which incurs further cost. Finally, manual conversion also requires significant time, and is therefore impractical for the research environment. Additional hardware devices have also been studied as ways to reduce the cost of tactile diagrams, including gloves with stimuli on the finger tips, joysticks inducing force feedback, and grids of electrodes or vibrating pins. However, these devices remain largely confined to the laboratory environment and have not been successful commercially.

Tactile diagram technology is an example of tactile-visual sensory substitution where the sense of vision is substituted by the sense of touch, allowing visual information to be perceived as tactile information. In this study, we were particularly interested in audio-visual sensory substitution, which maps visual information to audio. The advantage of audio-visual sensory substitution is that it requires no specialised hardware, since the necessary audio interfaces are included with virtually all modern computing platforms.

Several audio-visual sensory substitution algorithms have been studied in the literature, with the most well-known probably being the vOICe algorithm on which we based our research. The vOICe algorithm functions by converting a source image into a sequence of tone chords, where each chord represents a column in the source image and consists of tones which individually represent pixels from the corresponding column. The vOICe algorithm was originally developed as a real time vision substitution solution to convey live camera feeds, but may also serve as a solution for exploring static information such as diagrams.

First, we proposed an approach for accessible direct reading of diagrams. Our approach extends the vOICe algorithm by the addition of touch gestures which can be used to explore an image in detail using a device equipped with a touch screen. The approach can be controlled with two gestures. This mode of interaction allows vertical and diagonal lines to be traced, something which is not possible with the original vOICe implementation.

The proposed approach was evaluated with a set of user trials in which six blind and six sighted volunteers were recruited to identify several geometric

shapes and emoticons. Each type of stimulus was explored using both the standard implementation of the vOICe algorithm as well as the proposed approach. It was found that both blind and sighted candidates performed better when using the proposed approach ($p < 0.01$). Therefore, it can be concluded that the proposed interactive procedure using touch gestures increases the accuracy of diagram exploration, and is therefore a viable approach to allow graphical accessibility using audio-visual sensory substitution.

Adequate accessibility to technical documents also requires the reading of mathematics. Mathematical notation is two-dimensional but is usually conveyed to blind readers as a linear representation. This is due to the fact that the primary mediums by which blind readers access information, namely braille and audio, are linear in nature. Several linear representations for each medium are currently in use. Braille mathematics has no single standard. Rather, many countries have developed their own variants, and these are usually not compatible. Mathematics in electronic form can be represented as LaTeX or other textual notation, as well as MathML, an XML standard for representing mathematics maintained by the World Wide Web Consortium.

However, reading mathematics represented in a linear form poses several challenges. The conventional two-dimensional representation of mathematics efficiently conveys the relationships between the different components of an equation, but this is largely not so for linear representations. A linear representation is also longer than its two-dimensional equivalent, and may contain extra symbols to indicate context evident from the layout of the two-dimensional form. This disconnect between the linear and two-dimensional forms, as well as the myriad linear notation standards in use, impedes communication between blind and sighted mathematicians as well as between blind mathematicians themselves. Finally, most mathematical content is not available in a linear form and must be converted, often manually.

Several solutions have been proposed to address the challenges of accessible mathematics. Accessible mathematics browsers allow blind readers to navigate quickly among the parts of an equation. Translators between various formats allow blind readers to convert a given linear notation to a form with which they are familiar. However, these solutions require that the content is already available in an accessible format. Most mathematical content is still only available as raw images, or inaccessible PDF documents. Optical character recognition software that can recognise mathematics does exist, but without access to the source material, blind readers are unable to verify the accuracy of the automatically-generated representations. We proposed a solution based on our work in Chapter 4 to address these challenges. Using sonification to enable direct access to the source image, our aim was to allow blind readers to read mathematics in its two-dimensional form. This would enable access to mathematics in currently inaccessible sources as well as facilitate communication between blind and sighted mathematicians. Our proposed approach can also serve to complement existing approaches, by allowing a two-dimensional

overview of a complex equation.

Our approach provides two interface modes. The first enables the reader to examine equations by issuing textual commands, while the second supports exploration via the cursor keys on the keyboard, as well as gestures when a touch screen is available. The textual interface mode was designed to loosely follow conventions of textual virtual worlds, allowing commands like “look” to describe the context and directional commands such as “left” and “right” to navigate. Textual symbols are conveyed by speech synthesis, while graphical parts of the equation are sonified using the vOICE algorithm.

The approach was evaluated by eleven blind and fourteen sighted participants in a user trial that included the identification of twelve equations extracted from PDF documents. Overall, equations were identified completely correctly in 78% of cases (73% and 83% respectively for blind and sighted subjects). If partial correctness is considered, the performance is substantially higher. Feedback from the blind subjects indicated that the technique allows spatial information and graphical detail to be learnt that was hitherto unavailable to them, and that this was useful when exploring the equations.

The next section discusses ways in which our approach can be integrated into the existing accessible information workflow.

7.1 Integration

Much of the sensory substitution work described in the literature thus far only exists as prototypes, and has not been adopted in the mainstream. One reason for this is that print-disabled users are already required to utilise a number of technologies for various purposes, and adding another, especially given the financial cost and the learning curve of a sensory substitution solution, might not justify the investment of the funds and time. Therefore, a possible approach to ease adoption of our solution is to integrate it with the user’s already established software and workflow. This will enable users to gradually train themselves to utilise sensory substitution. Integration will also allow our solution to tie in with already established accessibility standards and best practices, allowing users to benefit from the advantages of both linear and spatial approaches to graphical accessibility.

Most touch screen capable screen readers allow the user to explore the content on the screen by moving one finger over the screen. This is also called “explore by touch”. Since our proposed solution allows the user to explore graphical content via the touch screen, it can be integrated with the touch screen support already established by screen readers. This could be accomplished in two ways. First, our one finger gestures could be added to the single finger explore by touch functionality already provided by touch screen capable screen readers. Then, as the user moves one finger around over the the screen, they will receive feedback about textual content, as well as the

graphical information. This functionality should be configurable by the user, with regards to the relative volume of the sonification output to the speech volume. The user should also be able to toggle the sonification functionality altogether, as it may be distracting in contexts where only the textual content is important.

A second way of integrating the methods we propose into screen readers is as an add-on or "plug-in" which the user may use to visualise the content of the current application. This approach is already taken by screen readers to visualise content which does not fit into the standard navigation strategy of typical user interfaces. One example is accessible mathematical content, for which screen readers provide a hierarchical explorer.

Our approach might also be offered as web-based technology, which would be cross-platform and not tied to a specific screen reader. Web technologies allow assistive functionality such as content browsers for complex content to be contained within the content itself, for example, as a Javascript component. An example of a technology offered using this approach is the MathJax accessibility mode, which allows blind readers to read mathematical content rendered using the MathJax Javascript library. As the accessibility mode of MathJax is included within the MathJax library itself, blind readers are able to accessibly explore the mathematical content regardless if their screen reader supports reading MathML content or not. In fact, we have already implemented parts of our approach as a web-based component, in order to perform the second evaluation remotely. This component, when offered as a Javascript library, could be included within a web page containing diagrams or equations, allowing the blind reader to explore the content using our system.

7.2 Further Applications of Our Approach

We have focused on simple vector diagrams and on mathematical equations in this thesis, since the accessibility of these is a primary impediment to sight-disabled readers of technical and scientific material. However, the proposed ideas might be useful also in other situations.

7.2.1 Accessible Maps

The integration of GPS receivers in mobile devices has allowed the development of navigation apps targeted at blind travellers. Traditionally, blind people navigate the physical environment by using mobility and orientation aids like white canes and guide dogs. However, navigational apps making use of geolocation services can allow a blind traveller to access information about the environment which cannot be determined by traditional mobility aids. This includes information which is usually only available in a visual form, like the information on sign posts. By using these apps, blind travellers can also deter-

mine points of interest, and receive turn by turn guidance to a predetermined location.

However, although turn by turn instructions are available to blind travellers through navigational apps, these instructions are often not optimal for travelling to an unfamiliar location. This is because these instructions are often generic, and best suited when they can be verified against the physical environment. A possible solution is to allow the blind user to explore geospatial maps before travelling [155]. Some implementations, like Apple Maps, include support for exploring maps in an accessible way, by using the touch exploration functionality of the screen reader. However, this support is mostly limited to textual information which can be announced via TTS or presented in braille. Our proposed approach for exploring graphical information may offer additional information when geospatial maps are explored, by allowing the blind user to determine graphical information like the angle of streets, and the relative size of buildings and other points of interest.

7.2.2 Aid to Optical Character Recognition

The development of optical character recognition has allowed blind readers to extract text from rasterised images. This has allowed blind readers to independently convert inaccessible documents like printed books, labels on product packages, and other printed sources of information to an accessible form which can be understood via text to speech or braille. One limitation of current OCR solutions is that blind readers are unable to verify the image before it is processed by the OCR algorithm. This is particularly important when text is recognised from a physical source like a book or product label, as the image is determined by the scanner or camera angle, lighting and other external factors. This is partially addressed by algorithms that give an indication of the detected edges of the captured source, but remains problematic when the lighting is poor, or the source is an unusual shape like product packaging.

Our approach could be employed as a means of verifying the captured image, before the OCR algorithm is carried out. Although blind readers may not be able to read the text using our approach alone, they may be able to determine the overall shape of the text, and thereby determine the image quality. In addition, if our approach is integrated into the OCR software, a blind reader may be able to determine whether text was cut off, or whether the image must be cropped or extended.

7.2.3 Accessible 3D Printing

3D printing technology produces tangible models of 3D structures. This holds benefits for conveying information of physical structures to blind users and may contribute particularly to the education of blind learners. Some organisations,

such as NASA, now offer 3D models of physical structures which can be printed and explored by blind users.

As for optical character recognition, a remaining limitation of accessible 3D printing is verification of the data. Blind users can download prepared 3D models and print them accessibly [82]. Blind users can also use non-visual methods for creating 3D models themselves, like textual command languages for describing 3D objects. However, blind users are unable to verify the model before it is sent to the 3D printer. Although our approach was designed for the exploration of two-dimensional information, it could be extended to the exploration of three-dimensions and offer interesting possibilities for the verification of 3D models in an accessible way. In addition, real time accessible exploration of 3D structures itself holds benefits for making virtual spaces accessible.

Exploring 3-dimensional information will require a further extension to our algorithm. We describe the possible form such an extension may take in Section 7.3.2.

7.2.4 Accessible Graphical User Interface Development

An area related to mathematics that was not considered in this study is the visualisation of graphical user interfaces for blind software developers. Currently, blind software developers find it difficult to do development of graphical user interfaces. Also called front-end development, this often requires the verification of the spatial layout of controls on the screen, as well as the size of text relative to boundaries. Although blind developers are able to develop the graphical interface from a purely logical perspective, the placement and scaling of controls and text on the screen usually requires entering coordinates explicitly, which can only be done if the result can be verified.

By using our approach, a blind developer would be able to determine the position and scale of controls on the screen, by using the touch screen exploration gestures. Our approach will also allow a blind developer to explore the general layout of text on the screen. Sonification alone would not be effective, as it is quite difficult to read text using this method. However, our approach combined with optical character recognition may enable a blind developer to determine the text within the bounds of a control, and thereby determine how the text should be laid out.

7.2.5 Training of Machine Learning Models

Pattern recognition has been successfully employed to improve access to information for blind readers. Current pattern recognition services can be used by blind people to gain information like the identification of common objects, faces, and colour. These services can be accessed on popular mobile platforms, and are also integrated into special purpose wearable technology such as smart glasses for blind users.

Like all products targeted towards a specific group, pattern recognition for blind users will increase in effectiveness when blind developers are involved in the research and development process. Blind users have a better understanding of the types of information they need to access. However, with no direct access to visual material, blind users have to develop models by using existing data sets, or by employing the assistance of sighted colleagues. A direct approach to graphical accessibility, such as the one we propose, would enable blind machine learning researchers to verify their data, and thereby gain a better understanding of the recognition models they are using, and how they should be altered.

7.3 Further Extensions of Our Approach

In the following, we describe a few immediate ways in which the methods we have developed could be extended.

7.3.1 Colour

One limitation of our current proposed approach to diagram accessibility is that it cannot convey colour information. It maps the grey-scaled value of each pixel to the amplitude or intensity of the corresponding sinusoid, while the vertical and horizontal positions are mapped to pitch and stereo placement respectively. However, many diagrams can only be effectively interpreted when the colour is conveyed. One example is a line chart conveying trends for several variables. These diagrams are usually produced by drawing each curve in a different colour, making it possible to distinguish the corresponding variables for each curve even when the curves intersect.

Our approach includes the ability to map parts of the source image to textual information. This capability is used to associate parts of the image with labels or symbols, and was evaluated for the case of conveying mathematical equations. This same method could be employed to convey colour information, by associating the colour of parts of the image to textual labels. In this way, for example, if a user drags their finger over the touch screen, they will receive the shape information as sonified output similar to the vOICE, while also receiving announcements about the colour of each region their finger encounters. However, since this functionality is already utilised for conveying textual information, using this functionality for conveying colour as well may contribute to increased cognitive load for the reader.

The Eye-Music algorithm offers an interesting approach as an alternative to verbal colour announcements, by mapping colours to different instrument timbres [23]. Although the Eye-Music algorithm on its own is not sufficient for conveying fine-grained shape information, it could serve as additional sonification specifically for colour information. For example, the user might receive

shape information via the sonification already used by our approach, while also hearing colour information via an algorithm based on Eye-Music. However, the intelligibility of the combined output of both algorithms is an area that requires further study.

7.3.2 Exploration of 3-Dimensional Data

As mentioned previously, one of the areas where our software may also prove useful is in the exploration of three-dimensional shapes. This would allow a blind user to verify the shape of a three-dimensional model before it is printed on a 3D printer. However, since our approach was specifically developed for exploring two-dimensional data, it might be difficult to apply our software to three-dimensional shapes in its current form. It is likely that a further extension will be required to allow adequate exploration of three-dimensional information. Such an extension would also allow blind readers to explore three-dimensional graphs and charts, which is also not currently possible with our software.

When physical models are not available, three-dimensional shapes can be embossed as tactile diagrams. However, tactile diagrams are ill-suited to depict three-dimensional information to blind readers. This is because visual pictures of three-dimensional shapes utilise shading and parallax to convey depth to the eye, but these approaches are not intuitive when interpreted by the sense of touch. It is also safe to assume that these approaches will have similar limitations when applied to an auditory medium like we have. Nevertheless, three-dimensional data can be conveyed via sound, as is demonstrated by its common use for that purpose in games and other media.

Spatial audio, also called 3D audio, is an approach whereby the position of a sound source is simulated in 3D space around the listener [151]. This can be accomplished by the use of multiple speakers in a surround sound set-up, or by calculating the resulting sound that should be received by each ear in a binaural set-up. In the latter case, head related transfer function (HRTF) filters are utilised to simulate the effect of a listener's head and outer ears on a sound perceived from a particular direction.

Our current approach to two-dimensional information accessibility utilises pitch (frequency) to convey vertical position, and stereo panning to convey horizontal position. We map brightness to volume (amplitude), which allows us to convey the grey-scaled value of pixels. However, to convey three-dimensional information, we also need to convey depth. By employing spatialised audio, we can convey depth by emitting our audio rendition of a pixel from a chosen position in three-dimensional space relative to the depth of the pixel.

Another area where an approach for conveying 3D information as audio may prove beneficial is the accessibility of 3D virtual environments. Accessibility to virtual environments like games is gaining more traction as companies seek to make their products accessible to a wider audience. Virtual environ-

ments have the advantage that 3D objects are already tagged with metadata which can be useful in the context of accessibility. Furthermore, accessibility to virtual environments also holds benefits for education and remote social interactions.

7.4 Summary and Conclusion

Much technical content remains inaccessible to blind and visually impaired readers, which has an impact on their education and careers. The existing approaches for presenting pictorial content require the use of costly hardware solutions which are not easily portable. Other spacial content can be represented in alternative linear forms which are accessible. However, the required additional mark-up is usually not provided, and even when it is, there is a possible loss of relational context.

This work described an approach based on audio-visual sensory substitution and spatial concepts employed to represent and navigate virtual worlds to make vector diagrams and mathematical equations accessible to print-disabled readers. Using touch gestures as well as keyboard or textual navigational commands, the reader is able to explore pictorial content and rendered mathematical equations without requiring specialised hardware.

Further development of this approach may include the integration with existing assistive technology, which would allow the reader to use both sensory substitution and traditional speech and braille methodologies to most effectively access information. Extending the approach to convey colour, as well as three-dimensional content, may also enable the application of this approach to as yet unexplored domains.

Appendices

Appendix A

Images Used in the Evaluation

The following provides a precise description of the images presented to the users during the experimental evaluation.

Stage 1

Image	Shape	Algorithm
1	Top: Horizontal rectangle, Bottom: Vertical rectangle	Interactive
2	Top: Horizontal oval, Bottom: Vertical oval	vOICe
3	Top: Right triangle with the right angle in the top left, Bottom: Right triangle, with the right angle in the top right	Interactive
4	Top: Right triangle, with the right angle at the bottom right, Bottom: Right triangle, with the right angle in the bottom left	vOICe

Stage 2

Image	Shape	Algorithm
1	Top: Square, and Vertical rectangle, Bottom: Horizontal oval	Interactive
2	Top: Circle, and Horizontal rectangle, Bottom: Vertical oval	vOICe
3	Top: Circle, Bottom: Circle, and Right triangle, with the right angle in the top left	vOICe
4	Top: Circle, and Circle, Bottom: Right triangle, with the right angle in the bottom right	Interactive
5	Top: Right triangle, with the right angle in the bottom left, and Right triangle, with the right angle in the top left, Bottom: Right triangle, with the right angle in the top right	vOICe
6	Top: Right triangle, with the right angle in the top right, Bottom: Right triangle, with the right angle in the bottom left, and Right triangle, with the right angle in the bottom right	Interactive

Stage 3

Image	Shape	Algorithm
1	Top: Right triangle, with the right angle in the top left, and Horizontal rectangle, Bottom: Vertical oval, and Circle	Interactive
2	Top: Right triangle, with the right angle in the top right, and Vertical rectangle, Bottom: Horizontal oval, and Square	vOICe
3	Top: Square, and Right triangle, with the right angle in the top left, Bottom: Right triangle, with the right angle in the top right, and Circle	vOICe
4	Top: Circle, and Right triangle, with the right angle in the bottom left, Bottom: Square, and Right triangle, with the right angle in the bottom right	Interactive
5	Top: Horizontal rectangle, and Vertical rectangle, Bottom: Square, and Square	vOICe
6	Top: Horizontal oval, and Vertical oval, Bottom: Circle, and Circle	Interactive

Stage 4

Image	Shape	Algorithm
1	Smiley	Interactive
2	Skew mouth	Interactive
3	Sad face	vOICe
4	Winking face	vOICe

Stage 5

Image	Shape	Algorithm
1	1 skew mouth, 2 smileys, and 1 sad face	vOICe
2	1 winking face, 2 smileys, and 1 sad face	Interactive
3	2 skew mouths, 1 smiley, and 1 sad face	vOICe
4	2 skew mouths, 1 smiley, and 1 sad face	Interactive
5	1 winking face, 1 smiley, and 2 sad faces	Interactive
6	1 winking face, 1 smiley, and 2 sad faces	vOICe
7	1 winking face, 1 smiley, and 2 sad faces	vOICe
8	2 skew mouths, 1 smiley, and 1 sad face	Interactive
9	1 skew mouth, and 3 smileys	Interactive
10	1 winking face, and 3 sad faces	vOICe

Appendix B

Sirin Tutorial

This appendix contains the text of an online tutorial which was used during the training phase of the approach described in chapter 6. Links are indicated in blue, and surrounded by square brackets. The online tutorial also contained interactive audio controls to play the example sound clips, but they have been omitted in this medium.

B.1 Introduction

Welcome to Sirin, the Sonified Image Reading INterface.

This document will give a brief introduction to the Sirin interface. Throughout this document, you will be guided through exploring basic images, as well as more complex content.

B.2 Description of Sirin Output

Sirin produces textual descriptions, as well as sound, to give you an idea of the content. Textual descriptions are used to convey text that may be present in the content, like symbols in an equation. Sound is used to convey shape information.

The sound produced by Sirin consists of one or more tones, where the pitch of a tone denotes vertical position in the image. The tones are also played from left to right in the stereo field to indicate horizontal placement. For instance, a horizontal line will consist of a tone with a constant pitch. A vertical line will consist of a short click, made up of several tones. A diagonal line from top-left to bottom-right will consist of a descending tone.

Here are a few examples of Sirin sounds for basic shapes:

B.2.1 Horizontal Line

A horizontal line at the bottom of the image.

[audio controls omitted]

B.2.2 Diagonal Line

A diagonal line from top-left to bottom-right.

[audio controls omitted]

B.2.3 Square Root

A short up-stroke starts a diagonal line down, followed by a diagonal line up, followed by a horizontal line.

[audio controls omitted]

B.3 Description of Sirin Modes

Sirin allows you to explore content with two modes, text mode, and graphical mode. When in text mode, press the "graphical mode" button to switch to graphical mode. Similarly, when in graphical mode, press the "text mode" button to switch to text mode.

In text mode, you can type commands to sonify the image. For example, type "play all" to play the whole screen. In graphical mode, you can listen to the image by using keystrokes on the keyboard. To play the whole screen in graphical mode, hold down the control key and press the down arrow. Make sure your screen reader is in focus mode or forms mode. You can view a complete list of commands in text mode by using the "help" command. The key commands and gestures for graphical mode can be accessed by pressing the f1 key in graphical mode.

If you are using a device with a touch screen, you can also use touch screen gestures to explore graphical content in graphical mode. Move one finger around on the screen to hear the column under your finger, with the part directly under your finger a bit louder than the rest. Use two fingers to explore a wider area. When you explore with two fingers, the line between the fingers acts as a scanner. Any content crossed by the line between your two fingers will be sonified. You will find that two fingers allow you to get a quick overview of graphical content. You can use one finger to explore a small area in more detail.

If you find that Sirin's gestures conflict with the gestures of your screen reader, you can put Sirin into self voicing mode. To do this, click the "self voicing" button when you are in graphical mode. Once in self voicing mode, you can turn off your screen reader, Sirin will use the text to speech functionality of the platform to make announcements. If the functionality is available on your platform, clicking the "self voicing" button will also make the program full screen. To disable self voicing mode press the "self voicing" button

again. Alternatively, if the program is full screen, press the escape key on the keyboard, or use the back functionality on a mobile device.

At any time you can access Sirin's built-in documentation, by clicking the link at the top of the page in text mode, or by pressing the f1 key in graphical mode. You can also [\[follow this link to read the documentation in a new tab\]](#).

B.4 Example 1: Basic Shapes

Use the following link to explore a basic image. It will open the Sirin interface in another tab. The screen will consist of a square, triangle, and a circle. Sirin will tell you that this image contains no text you can navigate to. However, you can listen to the whole image by using the "play all" command in text mode, or by holding down the control key and pressing down arrow in graphical mode. If your device has a touch screen, and you are in graphical mode, you can also use the touch screen gestures described in the previous section to explore the content. When you're done exploring, just close the tab. [\[Example 1: a square, a circle, and a triangle\]](#)

B.5 Example 2: Basic Equation

At the following link you will find a simple equation, y equals x to the power of 2. [\[Example 2: Basic Equation\]](#) Stay in text mode for now, and use the "look" command to examine the equation. Notice that Sirin tells you the current line, as well as the item in focus. The focus refers to the last item that you examined, it is like the focus of your screen reader, or the position of the cursor in a document. When you first open the equation, the focus should be on the item denoted as "y".

Now enter the command "right". Sirin will describe the "=". Sirin will also tell you that there are elements to the left and right. To the left is the "y", where you started. To the right is another element called "x".

Use right again to move to the "x". Sirin will describe the new item, and tell you that there are items to the left (the "="), and also an element called "2" to the right and up. Notice that the 2 is described as right and up from x. In print mathematics, an exponent is indicated by placing it to the right and a bit higher than its base number. Move right again with the "right" command, and Sirin should automatically move you right and up (since that's the only element to the right). If there were more than one item to the right, say an element right and up, and also another element right and down, you could have used commands "right up" and "right down" to move to a specific item. You can of course also use other directional commands like "left", "up", and "down" to move in other directions.

You can at any time use the "look" command to get an overview of the current focus and line. You can also jump to any element by typing its name after the look command, in other words you tell Sirin to "look" at a specific item. For instance, type "look y" to look at the very first item again.

Explore the previous equation again, but this time switch to graphical mode. Explore the equation with the arrow keys, just make sure your screen reader is in focus mode, or that the quick navigational keys are disabled. With NVDA, press the NVDA key and the space bar until you hear a sound like a printer. With Jaws, press jaws key and z. If you are using a device with a touch screen, you can also swipe left, right, up, or down with one finger on the touch screen to move around the equation. In addition, if you have a touch screen, you can use the gestures you practiced at the first example to explore the graphical structure of the equation. With this specific example, you should be able to read the whole equation with the left and right arrow keys. Notice that when you navigate to the 2, it is described as raised. When you navigate back to the x, it is described as lowered. [\[Example 2: Basic Equation\]](#)

B.6 Example 3: Fractions

The next page contains a different equation, y equals 2 over x . Because Sirin extracts an equation directly from a PDF document, some features of the equation is still in their original form. For example, a fraction in print consists of a numerator above a denominator, with a line between the two denoting that it is a fraction. Elements like the fraction line are unfortunately not represented in PDF documents by symbols, but by graphical objects. Although you can't navigate to the line with Sirin, you are able to hear where it is.

In text mode, use the "right" command until you reach the element called "2". Notice that Sirin tells you that the "2" has graphical content in the down direction. Now, type the command "play down". You should hear a constant tone. A constant tone indicates a horizontal line, in this case the line denoting the fraction.

Now, type the command "down". Sirin should move you over the fraction line to the "x". Again, notice that the description of the "x" also indicates that there is graphical content in the up direction. Use the "play up" command to listen to the fraction line again.

Something else to notice with this equation is that Sirin tells you when you enter a new line. Moving right from the equals for example, will tell you "one item in line". This indicates to you that you have reached a new line with only one item in it. Moving left, and therefore back to the equals, will tell you "entering line segment with 2 items". The two items in this case is the y and the equals. The 2 and the x are not considered part of the line, since the 2 is slightly higher than the y and equals, and the x slightly lower.

In graphical mode, you can use the up and down arrow keys to move between 2 and x, the top and bottom of the fraction. Notice how the line denoting the fraction is sonified each time you move between the two and the x.

If you keep the focus on either the x or the 2 for some time, you will hear Sirin automatically playing the fraction line.

You can also use keystrokes or gestures in graphical mode to sonify the graphical information next to an element. For example, focus on the x, and press shift and the up arrow to play the fraction line above the x. The equivalent gesture on a touch screen is to swipe up with two fingers. Similarly, shift and down arrow, or two fingers swipe down, will play the graphical information below an element. [\[Example 3: Equation with a Fraction\]](#)

B.7 Example 4: Noticing where an exponent is

The next equation, y equals x to the power 3 end power plus 4, has x raised to the power 3, but the 4 is outside the exponent. Notice that in text mode, the 3 is described as right and up from the x , and left and up from the 4. This indicates to you that the 3 is higher on the screen than the x and the 4.

If you move over the equation in graphical mode, you will hear the 3 described as "raised", but as you move either left to the x or right to the 4, you will hear "lowered", indicating that it is the end of the exponent. [\[Example 4, an Exponent with an End\]](#)

B.8 Example 5: a Square Root

The next equation is y equals square root of x . In print mathematics, a square root consists of a diagonal line down, followed by a diagonal line up, followed by a line extending over the expression the root is for.

You can use the commands for playing extra graphical information around an element to check whether an element is under the square root. Navigate through the equation in text mode by using the "right" command. You will first reach the square root. If you move right again, you will reach the x . If you then use the "play up" command, you will hear a line. That line is part of the square root, and indicates that the x is affected by the root.

In graphical mode, you can also hold down the shift key while pressing the up arrow, once you are focused on the x . This will play the extra graphical information up from the focus, in this case the line of the square root. You can accomplish the same with a two-finger swipe up on the touch screen.

It is a good idea when you encounter a square root, to check at each item after the square root whether it is under the root or not. You might for example

have two items after a square root, with one under the root, but the other one not. [\[Example 5: an Equation with a Square Root\]](#)

B.9 Example 6: Equation with Large Brackets

Sometimes, an equation contains large brackets or other delimiters. These elements are also not represented as symbols in PDF documents, but as graphical objects.

In text mode, Sirin will tell you when an element has extra graphical information to the left or right, such as a bracket. You can use the "play left" or "play right" command to listen to the bracket.

In graphical mode, Sirin allows one to listen to these symbols by pressing shift and either the left or right arrows. Sirin will also automatically play these elements in graphical mode, if you keep your focus on an element next to it for a while. Examine the equation on the following page: y equals big left bracket 2 plus start fraction x over 3 end fraction big right bracket to the power 9. [\[Example 6: an Equation with Large Brackets\]](#)

B.10 Example 7: a Matrix

A matrix is a collection of elements, arranged in a grid. Matrices are usually denoted by enclosing them in large square brackets. If you encounter many elements with no operators between them, it is probably a matrix. You can verify this by looking for the square brackets. This is best accomplished by using the gestures on the touch screen, but you can also use the "play left" or "play right" commands in text mode when you are next to a bracket, or the shift+left arrow and shift+right arrow commands in graphical mode.

A large square bracket, like those indicating a matrix, looks like an upright rectangle with one of its sides missing. If you listen to the left bracket with Sirin, you will hear a fat tone chord, followed by two tones (a high and low tone) played simultaneously. The right bracket will sound similar, but in reverse.

Examine the next equation: it contains a matrix with four elements (two rows and two columns) multiplied by a matrix with two elements (one column and two rows). In text it would be:

```
begin matrix
1, 2
3, 4
end matrix
times
begin matrix
1
```

2

end matrix

In text mode, focus on the element called "1", either by moving to it with the direction commands, or by typing "look 1". Now type "play left". You should hear the large left square bracket, indicating that this is a matrix.

In graphical mode, press left arrow and up arrow until you reach the 1, and press shift and left arrow. [\[Example 7: a Matrix\]](#)

B.11 Further Examples

The rest of this document contains some more equations you can use to practice reading with the Sirin interface. Each example has a link to Sirin loaded with the equation, followed by a textual description of the equation. If you wish, you can first try to identify the equation, and then verify your answer against the textual description.

[\[Example 8\]](#)

y equals square root of x end root plus 2.

[\[Example 9\]](#)

y equals square root of x to the power 3 end root.

[\[Example 10\]](#)

x equals big left bracket start fraction x over z end fraction plus 2 big right bracket to the power 2.

[\[Example 11\]](#)

y = x over the square root of 1 plus x end root end fraction

[\[Example 12\]](#)

```
begin matrix
1, 2, 3
4, 5, 6
7, 8, 9
end matrix
times
begin matrix
1
3
5
end matrix
```

[\[Example 13\]](#)

```
begin matrix
x over 2, x over 3
x over 4, x over 5
end matrix
times
```



```
begin matrix  
1 over 2  
2  
end matrix
```

List of References

- [1] Microsoft Corporation: Inspect tool for analysing accessibility properties. <https://docs.microsoft.com/en-us/windows/win32/winauto/inspect-objects>, 2021. Accessed: 23 August 2021.
- [2] HumanWare: Human brailliant b80 braille display. <https://store.humanware.com/int/brailliant-80.html>, 2005. Accessed: 23 August 2021.
- [3] Emerson, R.W. and Anderson, D.: What mathematical images are in a typical mathematics textbook? Implications for students with visual impairments. *Journal of Visual Impairment & Blindness*, vol. 112, no. 1, pp. 20–32, 2018.
- [4] Karshmer, A.I. and Bledsoe, C.: Access to mathematics by blind students. In: *International Conference on Computers for Handicapped Persons*, pp. 471–476. Springer, 2002.
- [5] King, A.R. and Evans, G.: *Re-presenting visual content for blind people*. University of Manchester, 2006.
- [6] Braille Authority of North America: Nemeth Code Book. https://nfb.org/Images/nfb/documents/pdf/nemeth_1972.pdf, 1972. Accessed: 12 May 2020.
- [7] Royal National Institute of Blind People: Braille mathematics notation. <https://www.ukaaf.org/wp-content/uploads/2015/05/Braille-Mathematics-Notation-PDF.pdf>, 2015. Accessed: 12 May 2020.
- [8] Raman, T.: AsTeR: Audio system for technical readings. *Information Technology and Disabilities*, vol. 1, no. 4, 1994.
- [9] Soiffer, N.: MathPlayer: web-based math accessibility. In: *Proceedings of the 7th International ACM SIGACCESS Conference on Computers and Accessibility*, pp. 204–205. 2005.
- [10] Soiffer, N.: MathPlayer v2. 1: web-based math accessibility. In: *Proceedings of the 9th international ACM SIGACCESS Conference on Computers and Accessibility*, pp. 257–258. 2007.
- [11] Tseng, W.: Access8Math NVDA Adon. <https://addons.nvda-project.org/addons/access8math.en.html>, 2021. Accessed: 2 June 2021.

- [12] Cervone, D., Krautzberger, P. and Sorge, V.: Employing semantic analysis for enhanced accessibility features in MathJax. In: *2016 13th IEEE Annual Consumer Communications & Networking Conference (CCNC)*, pp. 1129–1134. IEEE, 2016.
- [13] Cervone, D., Krautzberger, P. and Sorge, V.: Towards universal rendering in MathJax. In: *Proceedings of the 13th Web for All Conference*, pp. 1–4. 2016.
- [14] Murillo-Morales, T. and Miesenberger, K.: AUDiaL: A natural language interface to make statistical charts accessible to blind persons. In: *International Conference on Computers Helping People with Special Needs*, pp. 373–384. Springer, 2020.
- [15] Suzuki, M., Tamari, F., Fukuda, R., Uchida, S. and Kanahori, T.: INFTY: an integrated OCR system for mathematical documents. In: *Proceedings of the 2003 ACM Symposium on Document Engineering*, pp. 95–104. 2003.
- [16] Yamaguchi, K., Komada, T., Kawane, F. and Suzuki, M.: New features in math accessibility with INFTY software. In: *International Conference on Computers for Handicapped Persons*, pp. 892–899. Springer, 2008.
- [17] ViewPlus: ViewPlus Tactile Embossers. <https://viewplus.com/>, 2021. Accessed: 26 October 2021.
- [18] Miller, I., Pather, A., Milbury, J., Hasty, L., O’Day, A., Spence, D. and Osterhaus, S.: Guidelines and standards for tactile graphics. <http://www.brailleauthority.org/tg/web-manual/index.html>, 2010. Accessed: 22 November 2021.
- [19] View Plus: Ivey 3 hands-on learning system. <https://viewplus.com/product/iveo-3-hands-on-learning-system/>, 2005. Accessed: 1 November 2019.
- [20] Meijer, P.B.L.: An experimental system for auditory image representations. *IEEE Transactions on Biomedical Engineering*, vol. 39, no. 2, pp. 112–121, 1992.
- [21] Meijer, P.B.L.: Seeing with sound for the blind: Is it vision? In: *Tucson Conference on Consciousness, April 8-12*, p. 83. 2002.
- [22] Capelle, C., Trullemans, C., Arno, P. and Veraart, C.: A real-time experimental prototype for enhancement of vision rehabilitation using auditory substitution. *IEEE Transactions on Biomedical Engineering*, vol. 45, no. 10, pp. 1279–1293, Oct 1998. ISSN 0018-9294.
- [23] Abboud, S., Hanassy, S., Levy-Tzedek, S., Maidenbaum, S. and Amedi, A.: Eyemusic: Introducing a “visual” colorful experience for the blind using auditory sensory substitution. *Restorative Neurology and Neuroscience*, vol. 32, no. 2, pp. 247–257, 2014.

- [24] Gardner, J.A.: DotsPlus — better than braille? *ACM SIGCAPH Computers and the Physically Handicapped*, , no. 52-53, pp. 4–5, 1995.
- [25] World Wide Web Consortium: Mathematical Markup Language. <https://www.w3.org/TR/WD-math/>, 1998. Accessed: 26 October 2021.
- [26] MathJax Project: MathJax. <https://www.mathjax.org/>, 2009. Accessed: 26 October 2021.
- [27] Basham, J.D. and Marino, M.T.: Understanding STEM education and supporting students through universal design for learning. *Teaching exceptional children*, vol. 45, no. 4, pp. 8–15, 2013.
- [28] Villanueva, I. and Di Stefano, M.: Narrative inquiry on the teaching of STEM to blind high school students. *Education Sciences*, vol. 7, no. 4, p. 89, 2017.
- [29] Slee, R. and Allan, J.: Excluding the included: A reconsideration of inclusive education. *International Studies in Sociology of Education*, vol. 11, no. 2, pp. 173–192, 2001. <https://doi.org/10.1080/09620210100200073>. Available at: <https://doi.org/10.1080/09620210100200073>
- [30] National Federation of the Blind: Statistical facts about blindness in the United States. <https://nfb.org/blindness-statistics>, 2022. Accessed: 12 January 2022.
- [31] Turner, B.: Benetech global literacy services: Working towards a ‘born accessible’ world. *Learned Publishing*, vol. 31, no. 1, 2018.
- [32] World Wide Web Consortium: Web content accessibility guidelines (WCAG) 2.1. <https://www.w3.org/TR/WCAG21/>, 2018. Accessed: 26 October 2021.
- [33] World Wide Web Consortium: Accessible Rich Internet Applications (WAI-ARIA) 1.1. <https://www.w3.org/TR/wai-aria-1.1/>, 2017. Accessed: 26 October 2021.
- [34] for Standardization, I.O.: Information technology — W3C Web Content Accessibility Guidelines (WCAG) 2.0. Standard, International Organization for Standardization, Geneva, CH, October 2012.
- [35] World Wide Web Consortium: Scalable Vector Graphics (SVG) 2. <https://www.w3.org/TR/SVG2/>, 2018. Accessed: 26 October 2021.
- [36] Adobe: PDF Accessibility Overview. <https://www.adobe.com/accessibility/pdf/pdf-accessibility-overview.html>, 2022. Accessed: 19 October 2021.
- [37] for Standardization, I.O.: Document management applications — Electronic document file format enhancement for accessibility — Part 1: Use of ISO 32000-1 (PDF/UA-1). Standard, International Organization for Standardization, Geneva, CH, December 2014.

- [38] Drümmer, O.: PDF/UA (ISO 14289-1)–applying WCAG 2.0 principles to the world of PDF documents. In: *International Conference on Computers for Handicapped Persons*, pp. 587–594. Springer, 2012.
- [39] Jiménez, J., Olea, J., Torres, J., Alonso, I., Harder, D. and Fischer, K.: Biography of Louis Braille and invention of the braille alphabet. *Survey of ophthalmology*, vol. 54, no. 1, pp. 142–149, 2009.
- [40] Roth, G.A. and Fee, E.: The invention of Braille. *American Journal of Public Health*, vol. 101, no. 3, pp. 454–454, 2011.
- [41] Kent, D.: *What is Braille?* Enslow Publishing, LLC, 2012.
- [42] Jansson, G.: Haptics as a substitute for vision. *Assistive technology for visually impaired and blind people*, pp. 135–166, 2008.
- [43] Dixon, J.: Eight-dot braille. *A Position Statement of the Braille Authority of North America*, 2007.
- [44] National Library Service for the Blind and Physically Handicapped, Library of Congress: Specification 800: Braille books and pamphlets. http://www.loc.gov/nls/specs/800_march5_2008.pdf, 2008.
- [45] Engelen, J.: E-books and audiobooks: what about their accessibility? In: *International Conference on Computers for Handicapped Persons*, pp. 69–73. Springer, 2010.
- [46] Kahlisch, T.: DAISY: An opportunity to improve access to information for all. *Information services & use*, vol. 28, no. 2, pp. 151–158, 2008.
- [47] Garrish, M.: *What is EPUB 3?* ” O’Reilly Media, Inc.”, 2011.
- [48] Rubery, M.: *Audiobooks, literature, and sound studies*, vol. 31. Routledge, 2011.
- [49] Philips, D.: Talking books: The encounter of literature and technology in the audio book. *Convergence*, vol. 13, no. 3, pp. 293–306, 2007.
- [50] Evans, G. and Blenkhorn, P.: Screen readers and screen magnifiers. In: Hersh, M.A. and Johnson, M.A. (eds.), *Assistive Technology for Visually Impaired and Blind People*, pp. 449–495. Springer London, London, 2008. ISBN 978-1-84628-867-8.
- [51] Luthra, V. and Ghosh, S.: Understanding, evaluating and analyzing touch screen gestures for visually impaired users in mobile environment. In: *International Conference on Universal Access in Human-Computer Interaction*, pp. 25–36. Springer, 2015.
- [52] Carvalho, M.C.N., Dias, F.S., Reis, A.G.S. and Freire, A.P.: Accessibility and usability problems encountered on websites and applications in mobile devices by blind and normal-vision users. In: *Proceedings of the 33rd Annual ACM Symposium on Applied Computing*, pp. 2022–2029. 2018.

- [53] Google: Talkback documentation on touch gestures. <https://support.google.com/accessibility/android/answer/6151827?hl=en>, 2021. Accessed: 19 October 2021.
- [54] Apple: Voiceover documentation on touch gestures. <https://support.apple.com/en-za/guide/iphone/iph3e2e2281/ios>, 2021. Accessed: 19 October 2021.
- [55] King, A.: Screenreaders, magnifiers, and other ways of using computers. In: *Assistive Technology for Blindness and Low Vision*, pp. 265–288. CRC Press, 2012.
- [56] Edwards, W.K., Mynatt, E.D. and Stockton, K.: Access to graphical interfaces for blind users. *Interactions*, vol. 2, no. 1, pp. 54–67, 1995.
- [57] Blenkhorn, P. and Evans, G.: Architecture and requirements for a Windows screen reader. In: *IEE Seminar on Speech and Language Processing for Disabled and Elderly People (Ref. No. 2000/025)*, pp. 1–1. IET, 2000.
- [58] Android developers: Accessibility. <http://developer.android.com/guide/topics/ui/accessibility/index.html>, 2009. Accessed: 2 September 2021.
- [59] The Linux Foundation: atk/at-spi. <http://www.linuxfoundation.org/collaborate/workgroups/accessibility/atk/at-spi>, 2016. Accessed: 1 November 2019.
- [60] Microsoft Corporation: Microsoft Active Accessibility. <http://msdn.microsoft.com/en-us/library/ms971350.aspx>, 1997. Accessed: 1 November 2019.
- [61] Guinness, D., Cutrell, E. and Morris, M.R.: Caption crawler: Enabling reusable alternative text descriptions using reverse image search. In: *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, pp. 1–11. 2018.
- [62] Wu, S., Wieland, J., Farivar, O. and Schiller, J.: 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*, pp. 1180–1192. 2017.
- [63] Dinesh, S.G.M.V.B. and Balakrishnan, K.M.: Microsoft Excel charts™ accessibility: An affordable and effective solution. In: *The 3rd International Workshop on Digitization and E-Inclusion in Mathematics and Science 2016*, pp. 131–140. 2016.
- [64] Freedom Scientific: Jaws screen reader documentation. <https://support.freedomscientific.com/Products/Blindness/JAWSdocumentation>, 2021. Accessed: 6 July 2020.
- [65] Taylor, P.: *Text-to-speech synthesis*. Cambridge University Press, 2009.

- [66] Kayte, S.N., Mundada, M., Gaikwad, S. and Gawali, B.: Performance evaluation of speech synthesis techniques for English language. In: *Proceedings of the International Congress on Information and Communication Technology*, pp. 253–262. Springer, 2016.
- [67] Dutoit, T.: *An introduction to text-to-speech synthesis*, vol. 3. Springer Science & Business Media, 1997.
- [68] Ze, H., Senior, A. and Schuster, M.: Statistical parametric speech synthesis using deep neural networks. In: *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 7962–7966. IEEE, 2013.
- [69] Duddington, J.: eSpeak: Speech synthesizer. <http://espeak.sourceforge.net>, 1995. Accessed: 2 September 2021.
- [70] Klatt, D.H.: Review of text-to-speech conversion for English. *The Journal of the Acoustical Society of America*, vol. 82, no. 3, pp. 737–793, 1987.
- [71] Moos, A. and Trouvain, J.: Comprehension of ultra-fast speech – blind vs. “normally hearing” persons. In: *Proceedings of the 16th International Congress of Phonetic Sciences*, vol. 1, pp. 677–680. 2007.
- [72] Raman, T.: Emacspeak—a speech interface. In: *Proceedings of the SIGCHI conference on Human factors in computing systems*, pp. 66–71. 1996.
- [73] Hoffmann, R.: Speech, text and braille conversion technology. In: *Assistive Technology for Visually Impaired and Blind People*, pp. 497–554. Springer, 2008.
- [74] American Thermoform: Swell Touch Paper. <http://www.americanthermoform.com/product/swell-touch-paper/>, 2003. Accessed: 26 October 2021.
- [75] Tactisplay Corp: Tactisplay table. <http://www.tactisplay.com/product/tactisplay-table>, 2015. Accessed: 1 November 2019.
- [76] Loomis, J.M., Klatzky, R.L. and Giudice, N.A.: Sensory substitution of vision: Importance of perceptual and cognitive processing. In: *Assistive technology for blindness and low vision*, pp. 179–210. CRC press, 2018.
- [77] Hatwell, Y., Streri, A. and Gentaz, E.: *Touching for knowing: cognitive psychology of haptic manual perception*, vol. 53. John Benjamins Publishing, 2003.
- [78] Völkel, T., Weber, G. and Baumann, U.: Tactile graphics revised: the novel BrailleDis 9000 pin-matrix device with multi-touch input. In: *International Conference on Computers for Handicapped Persons*, pp. 835–842. Springer, 2008.

- [79] Prescher, D., Bornschein, J., Köhlmann, W. and Weber, G.: Touching graphical applications: bimanual tactile interaction on the hyperbraille pin-matrix display. *Universal Access in the Information Society*, vol. 17, no. 2, pp. 391–409, 2018.
- [80] Prescher, D., Nadig, O. and Weber, G.: Reading braille and tactile ink-print on a planar tactile display. In: *International Conference on Computers for Handicapped Persons*, pp. 482–489. Springer, 2010.
- [81] Orbit Research: Graphiti® – a Breakthrough in Non-Visual Access to All Forms of Graphical Information. <https://www.orbitresearch.com/product/graphiti/>, 2016. Accessed: 26 October 2021.
- [82] Götzelmann, T., Branz, L., Heidenreich, C. and Otto, M.: A personal computer-based approach for 3D printing accessible to blind people. In: *Proceedings of the 10th International Conference on PErvasive Technologies Related to Assistive Environments*, pp. 1–4. 2017.
- [83] Lieb, S., Rosenmeier, B., Thormählen, T. and Buettner, K.: Haptic and auditive mesh inspection for blind 3D modelers. In: *The 22nd International ACM SIGACCESS Conference on Computers and Accessibility, ASSETS '20*. Association for Computing Machinery, New York, NY, USA, 2020. ISBN 9781450371032.
Available at: <https://doi.org/10.1145/3373625.3417007>
- [84] Parkes, D.: Nomad: An audio-tactile tool for the acquisition, use and management of spatially distributed information by partially sighted and blind persons. In: *Proceedings of Second International Conference on Maps and Graphics for Visually Disabled People*, pp. 24–29. 1988.
- [85] Löttsch, J.: Computer-aided access to tactile graphics for the blind. In: *International Conference on Computers for Handicapped Persons*, pp. 575–581. Springer, 1994.
- [86] Landau, S., Russell, M., Gourgey, K., Erin, J.N. and Cowan, J.: Use of the talking tactile tablet in mathematics testing. *Journal of Visual Impairment & Blindness*, vol. 97, no. 2, pp. 85–96, 2003.
- [87] Gardner, J.A.: The progress in making figures universally accessible. In: *The 3rd International Workshop on Digitization and E-Inclusion in Mathematics and Science*. 2016.
- [88] Gardner, J.A.: Universally accessible figures. In: *International Conference on Computers Helping People with Special Needs*, pp. 417–420. Springer, 2016.
- [89] Ramloll, R. and Brewster, S.: A generic approach for augmenting tactile diagrams with spatial non-speech sounds. In: *CHI'02 Extended Abstracts on Human Factors in Computing Systems*, pp. 770–771. 2002.

- [90] Fournier D'Albe, E.E.: On a Type-Reading Optophone. *Proceedings of the Royal Society of London Series A*, vol. 90, pp. 373–375, July 1914.
- [91] Yu, W., Ramloll, R. and Brewster, S.: Haptic graphs for blind computer users. In: *International Workshop on Haptic Human-Computer Interaction*, pp. 41–51. Springer, 2000.
- [92] Schneider, J. and Strothotte, T.: Constructive exploration of spatial information by blind users. In: *Proceedings of the 4th International ACM Conference on Assistive Technologies*, pp. 188–192. 2000.
- [93] Tornil, B. and Baptiste-Jessel, N.: Use of force feedback pointing devices for blind users. In: *ERCIM Workshop on User Interfaces for All*, pp. 479–485. Springer, 2004.
- [94] Sjöström, C., Danielsson, H., Magnusson, C. and Rasmus-Gröhn, K.: Phantom-based haptic line graphics for blind persons. *Visual Impairment Research*, vol. 5, no. 1, pp. 13–32, 2003.
- [95] Poppinga, B., Magnusson, C., Pielot, M. and Rasmus-Gröhn, K.: Touchover map: audio-tactile exploration of interactive maps. In: *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services*, pp. 545–550. 2011.
- [96] Klatzky, R.L., Giudice, N.A., Bennett, C.R. and Loomis, J.M.: Touch-screen technology for the dynamic display of 2D spatial information without vision: promise and progress. *Multisensory Research*, vol. 27 5-6, pp. 359–378, 2014.
- [97] Gorlewicz, J.L., Tennison, J.L., Uesbeck, P.M., Richard, M.E., Palani, H.P., Stefik, A., Smith, D.W. and Giudice, N.A.: Design guidelines and recommendations for multimodal, touchscreen-based graphics. *ACM Transactions on Accessible Computing*, vol. 13, no. 3, August 2020. ISSN 1936-7228. Available at: <https://doi.org/10.1145/3403933>
- [98] Xu, C., Israr, A., Poupyrev, I., Bau, O. and Harrison, C.: Tactile display for the visually impaired using TeslaTouch. In: *CHI'11 Extended Abstracts on Human Factors in Computing Systems*, pp. 317–322. ACM, 2011.
- [99] Rosenbaum, D.A., Dawson, A.M. and Challis, J.H.: Haptic tracking permits bimanual independence. *Journal of Experimental Psychology: Human Perception and Performance*, vol. 32, no. 5, p. 1266, 2006.
- [100] Goncu, C. and Marriott, K.: GraVVITAS: generic multi-touch presentation of accessible graphics. In: *IFIP Conference on Human-Computer Interaction*, pp. 30–48. Springer, 2011.
- [101] Stangl, A., Verma, N., Fleischmann, K.R., Morris, M.R. and Gurari, D.: Going beyond one-size-fits-all image descriptions to satisfy the information wants of people who are blind or have low vision. In: *The 23rd International ACM SIGACCESS Conference on Computers and Accessibility*, ASSETS '21.

- Association for Computing Machinery, New York, NY, USA, 2021. ISBN 9781450383066.
Available at: <https://doi.org/10.1145/3441852.3471233>
- [102] Larkin, J.H. and Simon, H.A.: Why a diagram is (sometimes) worth ten thousand words. *Cognitive science*, vol. 11, no. 1, pp. 65–100, 1987.
- [103] Metatla, O., Bryan-Kinns, N. and Stockman, T.: Interactive hierarchy-based auditory displays for accessing and manipulating relational diagrams. *Journal on Multimodal User Interfaces*, vol. 5, no. 3, pp. 111–122, 2012.
- [104] Lee, J., Peng, Y.-H., Herskovitz, J. and Guo, A.: Image explorer: Multi-layered touch exploration to make images accessible. In: *The 23rd International ACM SIGACCESS Conference on Computers and Accessibility*, ASSETS '21. Association for Computing Machinery, New York, NY, USA, 2021. ISBN 9781450383066.
Available at: <https://doi.org/10.1145/3441852.3476548>
- [105] Sharif, A., Chintalapati, S.S., Wobbrock, J.O. and Reinecke, K.: Understanding screen-reader users' experiences with online data visualizations. In: *The 23rd International ACM SIGACCESS Conference on Computers and Accessibility*, ASSETS '21. Association for Computing Machinery, New York, NY, USA, 2021. ISBN 9781450383066.
Available at: <https://doi.org/10.1145/3441852.3471202>
- [106] Levänen, S., Jousmäki, V. and Hari, R.: Vibration-induced auditory-cortex activation in a congenitally deaf adult. *Current Biology*, vol. 8, no. 15, pp. 869–872, 1998.
- [107] Aviles-Rivero, A.I., Alsaleh, S.M., Philbeck, J., Raventos, S.P., Younes, N., Hahn, J.K. and Casals, A.: Sensory substitution for force feedback recovery: A perception experimental study. *ACM Transactions on Applied Perception*, vol. 15, no. 3, pp. 16:1–16:19, April 2018. ISSN 1544-3558.
Available at: <http://doi.acm.org/10.1145/3176642>
- [108] Bach-y-Rita, P., Collins, C.C., Saunders, F.A., White, B. and Scadden, L.: Vision substitution by tactile image projection. *Nature*, vol. 221, no. 5184, p. 963, 1969.
- [109] Linvill, J. and Bliss, J.: A direct translation reading aid for the blind. *Proceedings of the IEEE*, vol. 54, no. 1, pp. 40–51, Jan 1966. ISSN 0018-9219.
- [110] Bach-y-Rita, P., Kaczmarek, K.A., Tyler, M.E. and Garcia-Lara, J.: Form perception with a 49-point electrotactile stimulus array on the tongue: a technical note. *Journal of Rehabilitation Research and Development*, vol. 35, pp. 427–430, 1998.
- [111] Danilov, Y. and Tyler, M.: Brainport: an alternative input to the brain. *Journal of integrative neuroscience*, vol. 4, no. 04, pp. 537–550, 2005.

- [112] Meers, S. and Ward, K.: A vision system for providing 3D perception of the environment via transcutaneous electro-neural stimulation. In: *Proceedings. Eighth International Conference on Information Visualisation, 2004. IV 2004.*, pp. 546–552. IEEE, 2004.
- [113] Manshad, M.S. and Manshad, A.S.: Multimodal vision glove for touchscreens. In: *Proceedings of the 10th International ACM SIGACCESS Conference on Computers and Accessibility*, pp. 251–252. 2008.
- [114] Soviak, A., Borodin, A., Ashok, V., Borodin, Y., Puzis, Y. and Ramakrishnan, I.: Tactile accessibility: Does anyone need a haptic glove? In: *Proceedings of the 18th International ACM SIGACCESS Conference on Computers and Accessibility*, pp. 101–109. 2016.
- [115] Jameson, M.: The optophone: Its beginning and development. *Bulletin of Prosthetics Research*, pp. 25–28, 1966.
- [116] Proulx, M.J., Stoerig, P., Ludowig, E. and Knoll, I.: Seeing ‘where’ through the ears: effects of learning-by-doing and long-term sensory deprivation on localization based on image-to-sound substitution. *PloS One*, vol. 3, no. 3, p. e1840, 2008.
- [117] Brown, D., Macpherson, T. and Ward, J.: Seeing with sound? Exploring different characteristics of a visual-to-auditory sensory substitution device. *Perception*, vol. 40, no. 9, pp. 1120–1135, 2011.
- [118] Striem-Amit, E., Cohen, L., Dehaene, S. and Amedi, A.: Reading with sounds: sensory substitution selectively activates the visual word form area in the blind. *Neuron*, vol. 76, no. 3, pp. 640–652, 2012.
- [119] Cronly-Dillon, J., Persaud, K. and Gregory, R.: The perception of visual images encoded in musical form: a study in cross-modality information transfer. *Proceedings of the Royal Society of London B: Biological Sciences*, vol. 266, no. 1436, pp. 2427–2433, 1999. ISSN 0962-8452. <http://rspb.royalsocietypublishing.org/content/266/1436/2427.full.pdf>. Available at: <http://rspb.royalsocietypublishing.org/content/266/1436/2427>
- [120] Maidenbaum, S., Buchs, G., Abboud, S., Lavi-Rotbain, O. and Amedi, A.: Perception of graphical virtual environments by blind users via sensory substitution. *PloS one*, vol. 11, no. 2, p. e0147501, 2016.
- [121] Gomez, J.D., Bologna, G. and Pun, T.: See color: an extended sensory substitution device for the visually impaired. *Journal of Assistive Technologies*, 2014.
- [122] NVAccess: Audio screen NVDA addon. <https://www.nvaccess.org/audioScreen/>, 2021. Accessed: 17 June 2021.

- [123] NVAccess: NVDA Screen Reader. <https://nvda-project.org>, 2021. Accessed: 17 June 2021.
- [124] Arno, P., Vanlierde, A., Streel, E., Wanet-Defalque, M.-C., Sanabria-Bohorquez, S. and Veraart, C.: Auditory substitution of vision: pattern recognition by the blind. *Applied Cognitive Psychology*, vol. 15, no. 5, pp. 509–519, 2001.
- [125] Palani, H.P., Tennison, J.L., Giudice, G.B. and Giudice, N.A.: Touchscreen-based haptic information access for assisting blind and visually-impaired users: Perceptual parameters and design guidelines. In: *International Conference on Applied Human Factors and Ergonomics*, pp. 837–847. Springer, 2018.
- [126] Horstmann, M., Lorenz, M., Watkowski, A., Ioannidis, G., Herzog, O., King, A., Evans, D.G., Hagen, C., Schlieder, C., Burn, A.-M. *et al.*: Automated interpretation and accessible presentation of technical diagrams for blind people. *New Review of Hypermedia and Multimedia*, vol. 10, no. 2, pp. 141–163, 2004.
- [127] Roth, P., Pun, T. and Kronegg, J.: Rendering digital images accessible for blind computer users. In: *HCI International 2003, 10th International Conference on Human-Computer Interaction*. 2003.
- [128] Calder, M., Cohen, R.F., Lanzoni, J. and Xu, Y.: PLUMB: an interface for users who are blind to display, create, and modify graphs. In: *Proceedings of the 8th international ACM SIGACCESS Conference on Computers and Accessibility*, pp. 263–264. 2006.
- [129] Orbit Research: Orion ti-84 plus talking graphing calculator. <https://www.orbitresearch.com/product/orion-ti-84-plus/>, 2016. Accessed: 3 May 2020.
- [130] Software, S.: SAS Graphics Accelerator. <https://support.sas.com/software/products/graphics-accelerator/>, 2021. Accessed: 17 June 2021.
- [131] LaTeX Project: LaTeX - A document preparation system. <https://www.latex-project.org/>, 1998. Accessed: 26 October 2021.
- [132] Jayant, C.: A survey of math accessibility for blind persons and an investigation on text/math separation. 2006. Retrieved from: <https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.187.9287&rep=rep1&type=pdf>.
- [133] Amalric, M., Degenhien, I. and Dehaene, S.: On the role of visual experience in mathematical development: Evidence from blind mathematicians. *Developmental cognitive neuroscience*, vol. 30, pp. 314–323, 2018.
- [134] Cervone, D., Krautzberger, P. and Sorge, V.: New accessibility features in MathJax. *Journal on Technology and Persons with Disabilities*, 2016.

- [135] Edwards, A.D., McCartney, H. and Fogarolo, F.: Lambda: A multimodal approach to making mathematics accessible to blind students. In: *Proceedings of the 8th International ACM SIGACCESS Conference on Computers and Accessibility*, pp. 48–54. 2006.
- [136] Gardner, J.A., Lundquist, R. and Sahyun, S.: TRIANGLE: a tri-modal access program for reading, writing and doing math. In: *Proceedings of the CSUN International Conference on Technology and Persons with Disabilities, Los Angeles. Converging Technologies for Improving Human Performance (pre-publication on-line version)*, vol. 123. 1998.
- [137] Thompson, D.M.: LaTeX2Tri: Physics and mathematics for the blind or visually impaired. In: *Proc. 20th Conference on Technology and Persons with Disabilities*. Citeseer, 2005.
- [138] Karshmer, A.I., Gupta, G., Pontelli, E., Miesenberger, K., Ammalai, N., Gopal, D., Batusic, M., Stöger, B., Palmer, B. and Guo, H.-F.: UMA: a system for universal mathematics accessibility. In: *Proceedings of the 6th International ACM SIGACCESS Conference on Computers and Accessibility*, pp. 55–62. 2003.
- [139] Nazemi, A., Murray, I. and Mohammadi, N.: Mathspeak: An audio method for presenting mathematical formulae to blind students. In: *2012 5th International Conference on Human System Interactions*, pp. 48–52. IEEE, 2012.
- [140] Mittelbach, F. and Rowley, C.: LATEX Tagged PDF – A blueprint for a large project. *TUGboat*, vol. 41, no. 3, 2020.
- [141] Ahmetovic, D., Armano, T., Bernareggi, C., Berra, M., Capietto, A., Coriasco, S., Murru, N., Ruighi, A. and Taranto, E.: Axessibility: A latex package for mathematical formulae accessibility in pdf documents. In: *Proceedings of the 20th International ACM SIGACCESS Conference on Computers and Accessibility*, pp. 352–354. 2018.
- [142] Balan, O., Moldoveanu, A. and Moldoveanu, F.: Navigational audio games: an effective approach toward improving spatial contextual learning for blind people. *International Journal on Disability and Human Development*, vol. 14, no. 2, pp. 109–118, 2015.
- [143] Montfort, N. and Short, E.: Interactive fiction communities. *Dichtung Digital*, vol. 41, 2012.
- [144] Friberg, J. and Gärdenfors, D.: Audio games: new perspectives on game audio. In: *Proceedings of the 2004 ACM SIGCHI International Conference on Advances in computer entertainment technology*, pp. 148–154. 2004.
- [145] Jerz, D.G.: Somewhere nearby is Colossal Cave: Examining Will Crowther’s original “Adventure” in code and in Kentucky. *Digital Humanities Quarterly*, vol. 1, no. 2, 2007.

- [146] Lebling, P.D., Blank, M.S. and Anderson, T.A.: Special feature zork: a computerized fantasy simulation game. *Computer*, vol. 12, no. 04, pp. 51–59, 1979.
- [147] Bartle, R.A.: Multi-User Dungeons (MUDs). *The International Encyclopedia of Digital Communication and Society*, pp. 1–8, 2015.
- [148] Montfort, N.: *Twisty Little Passages: an approach to interactive fiction*. MIT Press, 2005.
- [149] Fizek, S., Woletz, J. and Beksa, J.: Playing with sound and gesture in digital audio games. In: *Mensch und Computer 2015–Workshopband*, pp. 423–430. De Gruyter, 2015.
- [150] Oren, M.A.: Speed sonic across the span: building a platform audio game. In: *CHI'07 Extended Abstracts on Human Factors in Computing Systems*, pp. 2231–2236. 2007.
- [151] Gardner, W.G.: 3D audio and acoustic environment modeling. *Wave Arts, Inc*, vol. 99, 1999.
- [152] Trewin, S., Hanson, V.L., Laff, M.R. and Cavender, A.: Powerup: an accessible virtual world. In: *Proceedings of the 10th International ACM SIGACCESS Conference on Computers and Accessibility*, pp. 177–184. 2008.
- [153] freedesktop.org: Poppler PDF rendering library. <https://poppler.freedesktop.org/>, 2005. Accessed: 10 December 2021.
- [154] Shinyama, Yusuke: Pdminer python pdf parser and analyzer. <https://www.unixuser.org/~euske/python/pdminer/index.html>, 2004. Accessed: 10 December 2021.
- [155] Ivanchev, M., Zinke, F. and Lucke, U.: Pre-journey visualization of travel routes for the blind on refreshable interactive tactile displays. In: *International Conference on Computers for Handicapped Persons*, pp. 81–88. Springer, 2014.