FingerReader: A Finger-Worn Assistive Augmentation

Roy Shilkrot, Jochen Huber, Roger Boldu, Pattie Maes and Suranga Nanayakkara

1 Introduction

Accessing printed reading material in an unstructured or unfamiliar environment is still a major challenge for people with visual impairments (VI). Whereas much of the printed material is not digitally accessible, many resort to using smartphone apps or simply asking for help from companions. Interviews with people with a VI [31] reveal that they struggle with focusing, aligning and even using reading assistive technology in settings such as in restaurants, on doctor appointments or reading mail items. In our experiments and interviews with persons with VI we validated these needs and problems and found a necessity for text-access technology that can overcome the hurdles of lighting, focus, aim and environment.

To this end we contributed the FingerReader, a finger-augmenting camera that looks at whatever the finger touches or points to. The major propositions of the FingerReader are: (i) using the finger for reading or pointing is well practiced within both sighted and non-sighted individuals, (ii) a finger-worn device creates a direct connection between the fingertip's strong tactile sensory capabilities and the directionality of the gesture, and finally (iii) camera and computer vision based algorithms can greatly benefit from the focused input as it is constrained to only what's underneath the finger or right in front of it.

The pointing gesture, flexing the index finger and pointing it at a thing, location or person, is a well practiced deictic gesture, rooted in the human gestural language and universally recognized across cultures and eras [19]. Pointing also carries many other symbolic meanings, such as signaling (e.g. in a classroom, or hailing a taxi),

R. Shilkrot (✉)
Computer Science Department, Stony Brook University, Stony Brook, NY, USA
e-mail: roys@cs.stonybrook.edu

J. Huber
Synaptics, Zug, Switzerland

R. Boldu · S. Nanayakkara
Singapore University of Technology and Design, Singapore, Singapore

P. Maes
Media Lab, Massachusetts Institute of Technology, Cambridge, MA, USA

J. Huber et al. (eds.), *Assistive Augmentation*, Cognitive Science and Technology, https://doi.org/10.1007/978-981-10-6404-3_9

or lexical meaning (e.g. the number one). The ubiquitousness of pointing makes it a strong candidate to augment technologically, since the entry barrier to performing the gesture is low, the social norms are lax, and it is well understood throughout society.

Using the pointing gesture for augmentation means capitalizing on the rich neural representation the index finger has in the somatosensory cortex of the brain. The tips of the index and middle fingers are the most highly dense areas of nerve endings, making them highly tuned to feeling an array of senses: tactile change in several frequencies, temperature, and pain [4]. The index finger is used by VI people to read braille, and by sighted people as a visual pointer for reading text in early learning stages. Utilizing this direct connection between the finger, fingertip and the brain, the FingerReader is a high sensory substitution modality between vision and tactility.

This article presents the comprehensive work performed on the FingerReader over the last 4 years towards its vision realization, beginning with the original FingerReader, on through the mobile version (MobiReader), the music reading version (MusicReader) and finally the latest development thrust to productization. We present the driving motivations, academic positioning and prior art, algorithmic components, design rationale and its implementation, as well as numerous user studies with visually impaired persons.

2 Related Work

Researchers in both academia and industry exhibited a keen interest in aiding people with VI to read printed text. The earliest evidence we found for a specialized assistive text-reading device for the blind is the Optophone, dating back to 1914 [6]. However the Optacon [16], a steerable miniature camera that controls a tactile display, is a more widely known device from the mid 20th century. Table 1 presents more contemporary methods of text-reading for the VI based on key features: adaptation for non-perfect imaging, type of text, User Interface (UI) suitable for VI and the evaluation method. Thereafter we discuss related work in three categories: wearable devices, handheld devices and readily available products.

Wearable devices. In a wearable form-factor, it is possible to use the body as a directing and focusing mechanism, relying on proprioception or the sense of touch, which are of utmost importance for people with VI. Yi and Tian [40] placed a camera on shade-glasses to recognize and synthesize text written on objects in front of them, and Hanif and Prevost's [10] did the same while adding a handheld device for tactile cues. Mattar et al. are using a head-worn camera [18], while Ezaki et al. developed a shoulder-mountable camera paired with a PDA [9]. Differing from these systems, we proposed using the finger as a guide [20], and supporting sequential acquisition of text rather than reading text blocks [30]. This concept has inspired other researchers in the community [33].

Table 1 Recent efforts in academia of text-reading solutions for the VI

Publication	Year	Interface	Type of text	Adaptation
Ezaki et al. [9]	2004	PDA	Signage	
Mattar et al. [18]	2005	Head-worn	Signage	Color, clutter
Hanif and Prevost [10]	2007	Glasses, tactile	Signage	
SYPOLE [22]	2007	PDA	Products, book cover	Warping, lighting
Pazio et al. [21]	2007		Signage	Slanted text
Yi and Tian [40]	2012	Glasses	Signage, products	Coloring
Shen and Coughlan [28]	2012	PDA, tactile	Signage	
Kane et al. [14]	2013	Stationery	Printed page	Warping
Stearns et al. [33]	2014	Finger-worn	Printed page	Warping
Shilkrot et al. [30]	2014	Finger-worn	Printed page	Slanting, lighting

Handheld and mobile devices. Mancas-Thillou, Gaudissart, Peters and Ferreira's SYPOLE consisted of a camera phone/PDA to recognize banknotes, barcodes and labels on various objects [22], and Shen and Coughlan presented a smartphone based sign reader that incorporates tactile vibration cues to help keep the text-region aligned [28]. The VizWiz mobile assistive application takes a different approach by offloading the computation to humans, although it enables far more complex features than simply reading text, it lacks real time response [3].

Assistive mobile text reading products. Mobile phone devices are very prolific in the community of blind users for their availability, connectivity and assistive operation modes, therefore many applications were built on top of them, however numerous specialized portable devices are also available. See Table 2 for a list of assistive products for reading text.

2.1 Finger Worn Cameras

The area of finger worn camera devices for interaction, not necessarily as assistive technology, is rapidly growing into a research agenda of it's own, albeit without notable consumer products yet in availability. The enduring work of Stetten et al. on FingerSight [12], first reported in 2006, tries to create an assistive finger-worn device to detect visual edges. The work of Nanayakkara and Shilkrot et al. spans a number of projects (not all cited here for brevity) into wearable assistive cameras to read text and recognize objects [20, 31], also occasionally serving as smartphone peripherals. Stearns et al. recently developed HandSight [33], which is geared directly at reading text with the finger. Other related work include the work of Rissanen et al. [25], which developed a smartphone peripheral camera for natural interaction with objects, and Yang et al., which created a miniature finger worn device that reacts to surface texture [36].

Table 2 Assistive mobile products for reading printed text

Name	Reference	Type
kNFB kReader	http://www.knfbreader.com	App
Text detective	http://blindsight.com	App
Text grabber	http://www.abbyy.com/textgrabber	App
StandScan	http://standscan.com	Device
SayText	http://www.docscannerapp.com/saytext	App
ZoomReader	http://mobile.aisquared.com	App
Prizmo	http://www.creaceed.com/iprizmo	App
LookTel	http://www.looktel.com	App
vOICe for Android	http://www.seeingwithsound.com	App
EyePal ROL	http://www.abisee.com	Device
OrCam	http://www.orcam.com	Device
Intel reader	http://reader.intel.com	Device
VizWiz	http://www.vizwiz.org	App
BeMyEyes	http://www.bemyeyes.org	App
TapTapSee	http://www.taptapseeapp.com	App

Camera-augmented fingers as an approach to assistive technology was also conceptualized earlier by designers without a technical implementation. Hedberg thought of the Thimble, a device to allow reading print and also braille [11], Lee designed the Reading Finger that reads barcodes [15], and both Munscher [1, 7] thought to use the finger as a point-and-shoot camera, literally.

The most relevant works in academia were already mentioned in [31], such as Kane et al.'s AccessLens [14], Yi's body of work [39] and Shen and Coughlan [28]. However other work not involving computer vision, such as El-Glaly's finger-reading iPad [8] and Yarrington's skimming algorithm [37], demonstrate the need to create an equilibrium between visual and non-visual readers by importing aspects of visual reading to assistive technology for VI persons.

3 FingerReader: A Wearable Reading Assistant

FingerReader supports persons with VI in reading printed text by scanning with the finger and uttering the words as synthesized speech. The device features hardware and software, including video processing algorithms, and two output modalities: tactile and auditory channels.

The design of the FingerReader is a continuation of our work on finger wearable devices for seamless interaction, namely the EyeRing [20, 30]. Exploring early design concepts with VI users revealed the need to have a small, portable device that supports free movement, requires minimal setup and utilizes real-time, distinctive multimodal response. Design explorations of form and function suggested such

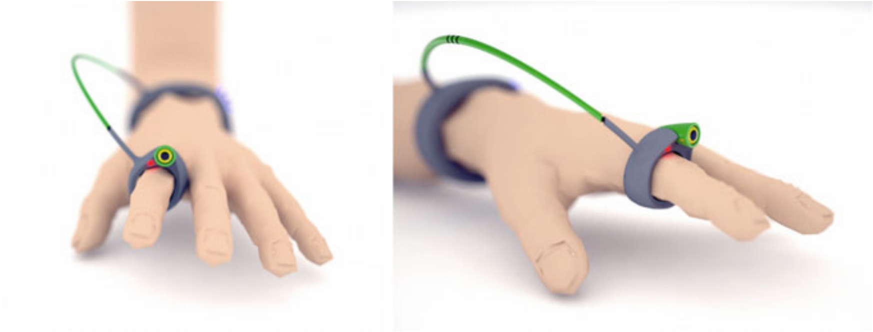
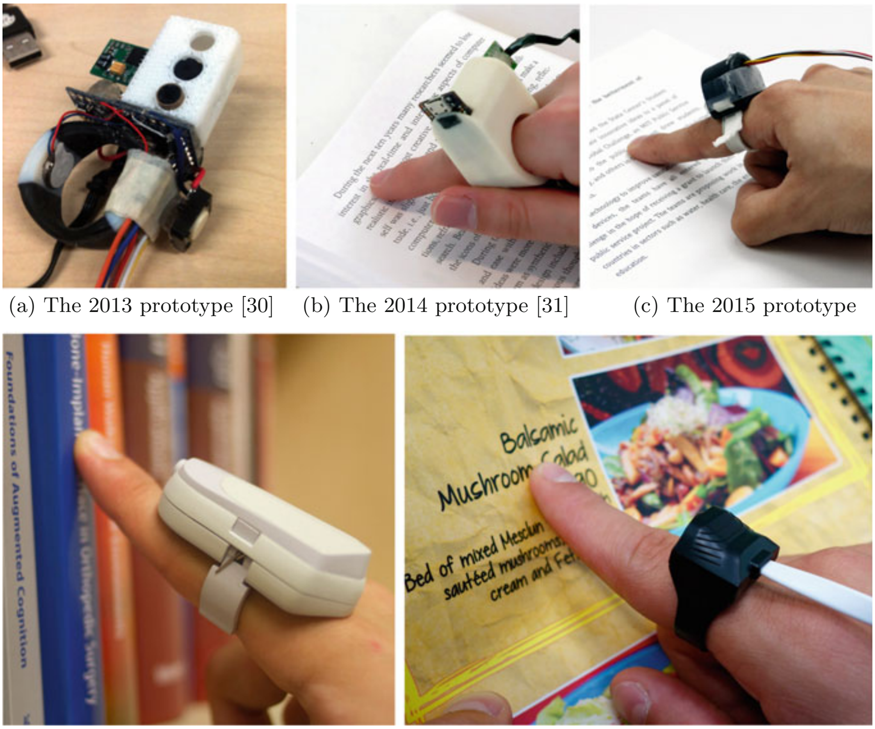


Fig. 1 Design vision. 3D modeling and rendering credits: Amit Zoran



(a) The 2013 prototype [30] (b) The 2014 prototype [31] (c) The 2015 prototype

(d) The 2016 prototype (e) The 2017 prototype

Fig. 2 Evolution of the FingerReader prototypes in the years 2013–2017

a device can be made aesthetically pleasing (Fig. 1), while keeping the camera in a fixed distance from the tip of the finger. The evolution of the laboratory prototypes (Fig. 2) led from a put-together mock up, to a fully enclosed device, decreasing in size and increasing comfort.

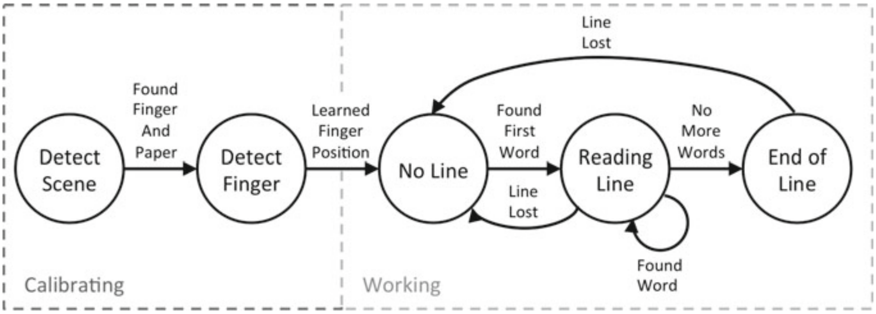


Fig. 3 Sequential text reading algorithm state machine

3.1 Text Reading Algorithm

The sequential text reading algorithm is comprised of a number of sub-algorithms concatenated in a state-machine (see Fig. 3), to accommodate for a continuous operation by a blind person. The first two states (*Detect Scene* and *Learn Finger*) are used for calibration for the higher level text extraction and tracking work states (*No Line, Line Found* and *End of Line*). Each state delivers timely audio cues to the users to inform them of the process. The algorithm in full detail can be reviewed in [31], therefore we only recount it in brief.

Scene and Finger Detection: The initial calibration step tries to ascertain whether the camera sees a finger on a contrasting paper. The input camera image is converted to the normalized-red channel: $nR = \frac{r}{r+g+b}$ that corresponds well with skin colors and ameliorates lighting effects. The image is matched to an example in a dataset of prerecorded typical images of fingers and papers. Once a stable match is achieved system deems the scene to be a well-placed finger on a paper. To detect the finger, an adaptive thresholding is performed and the top white pixel is considered a candidate fingertip point. During this process the user is instructed not to move, while our system collects samples of the fingertip location to build a location prior. The inlying fingertip detection guides a local horizontal *focus region*, located above the fingertip, within which the following states perform their operations. The focus region helps with efficiency in calculation and also reduces confusion for the line extraction algorithm with neighboring lines. See Fig. 4 for an illustration of this process.

Line Extraction: Within the focus region, we perform local adaptive binarization and selective contour extraction based on typical contour area sizes for characters. We pick the bottom point of each contour as the baseline point, and look for candidate lines by fitting line equations to triplets of baseline points, discarding extreme cases. We further prune by looking for supporting baseline points to the candidate lines based on distance from the line. We eliminate duplicate line candidates by binning and refine the equations based on their supporting points. We pick the highest scoring line as the detected text line. See Fig. 4 for an illustration.

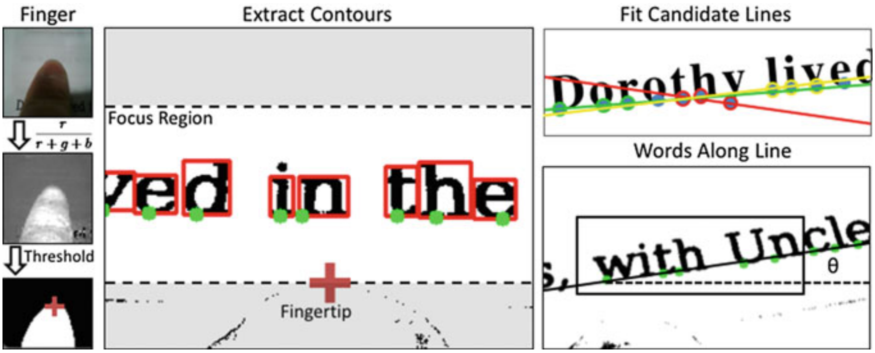


Fig. 4 Fingertip detection and text extraction

Word Extraction and Tracking: We employed the Tesseract OCR engine, set to only extract a single word and supply: the word, the bounding rectangle, and the detection confidence. Words with high confidence are retained, uttered out loud to the user, and further tracked using their bounding rectangle.

For tracking found words we use template matching over their image patches. Every successful match contributes to the bank of patches for that word. We constrain the template search region around the last position of the word while considering the predicted movement speed.

When the user veers from the scan line, detected using the line equation and the fingertip point, we trigger a gradually increasing auditory feedback. When the system cannot find more word blocks further along the scan line, it triggers an event and advances to the *End of Line* state.

3.2 Evaluation of the FingerReader

The central question that we sought to explore was how and whether FingerReader can provide effective access to print and reading support for VI users. Towards this end, we conducted a series of evaluations. First, we conducted a technical evaluation to assess whether the FingerReader is sufficiently accurate, and found that in perfect conditions the FingerReader's algorithm can recover over 93% of the words. In parallel, we performed an investigation of the usefulness of the different feedback cues with congenitally blind users. The results showed that participants preferred a tactile feedback compared to other cues (only audio or audio-and-tactile), and were able to recognized a gradual change in amplitude. One user reported that "*when [the audio] stops talking, you don't know if it's actually the correct spot because there's no continuous updates, so the vibration guides me much better.*"

We then used the results from these two fundamental investigations to conduct a qualitative evaluation of FingerReader's text access and reading support with 3 blind users. The methods, results and discussions thereof can again be read in depth in [31], and hereby we only highlight the major findings.

Qualitative User Study Findings We found that participants generally thought it was easy to access text with the FingerReader, however actual reading was considered less enjoyable and harder. Compared to other reading aids, participants were split between appreciating the immediacy of the FingerReader to the effectiveness of other aids.

- **Visual layout**: Restaurant menus and business cards were particularly challenging for the participants. Where some were specifically challenged by the multi-column layouts.
- **Audio feedback**: Some participants preferred an audio feedback to a tactile feedback, and mentioned that the choice of feedback could be better. One participant found it hard to navigate the different tones the FingerReader produced for line deviation and finger twisting/rotation.
- **Fatigue**: All participants reported that they would not use the FingerReader for longer reading sessions such as books, as it is too tiring. In this case, they would simply prefer an audio book or a scanned PDF that is read back, e.g. using ABBYY FineReader.
- **Serendipity**: Whenever any of the participants made the FingerReader read the very first correct word of the document, they smiled, laughed or showed other forms of excitement–every single time.
- **Efficiency over independence**: All participants mentioned that they want to read print fast and even "*when that means to ask their friends or a waiter around*". The FingerReader was marked with potential to help them towards independence, since they want to explore on their own rather than have others subjectively filter for them. However, efficiency in reading was consistently regarded as more important than independence.
- **Exploration impacts efficiency**: The former point underlines the potential of FingerReader-like devices for exploration of print, where efficiency is less of a requirement but getting access to it is. In other words, print exploration is only acceptable for documents where (1) efficiency does not matter, i.e. users have time to explore or (2) exploration leads to efficient text reading.
- **Layout navigation in an audio stream**: We found an indication that navigating text during the reading phase is comparable to the navigation in audio streams the device makes. The FingerReader recognizes words and reads them on a first-in, first-out principle at a fixed speed. Consequently, if the FingerReader detects a lot of words, it requires some time to read everything to the user. Stopping the finger movement to listen to the sound interrupted the interaction process and skewed the mental model of the blind user—the respective cognitive map of the document—specifically shaped through the text that is being read back.

On post-usage questionnaires, the overall experience with the FingerReader was rated as mediocre by all participants. They commented that this was mainly due to the synthesized voice being unpleasant and the steep learning curve.

4 MobiReader: A Mobile FingerReader

The second incarnation of the FingerReader was embodied in the MobiReader – a smartphone-based version of our software and a second iteration on the finger-worn camera design. Using a standard smartphone is key, since these are both prolific within the VI persons community and have ample computation power in recent generations. The MobiReader, designed as a peripheral device, could be made cheaper for using less components, and spare the user from purchasing a costly specialized device or even a new smartphone by simply adding external capabilities. Peripheral and smartphone-complementary devices are welcome in the VI community, a recent survey shows [38], as Bluetooth-coupled headsets and braille displays and keyboards are in wide use (Fig. 5).

To evaluate the MobiReader we designed a usability study with 10 VI persons in a lab setting, looking to estimate the potential success of the device as a mobile reading aid for printed material. Unlike former studies, here we contribute a quantitative assessment with a larger user base, and test the complete working system.

Our findings show that users were able to successfully extract an average of 74% of the words in a given piece of text when only provided with a feedback that told them how far away from the text line they were. The results demonstrate robustness in handling a range of standard font sizes, and that reading text within this range does not significantly hinder reading capability. The data also reveals insignificant

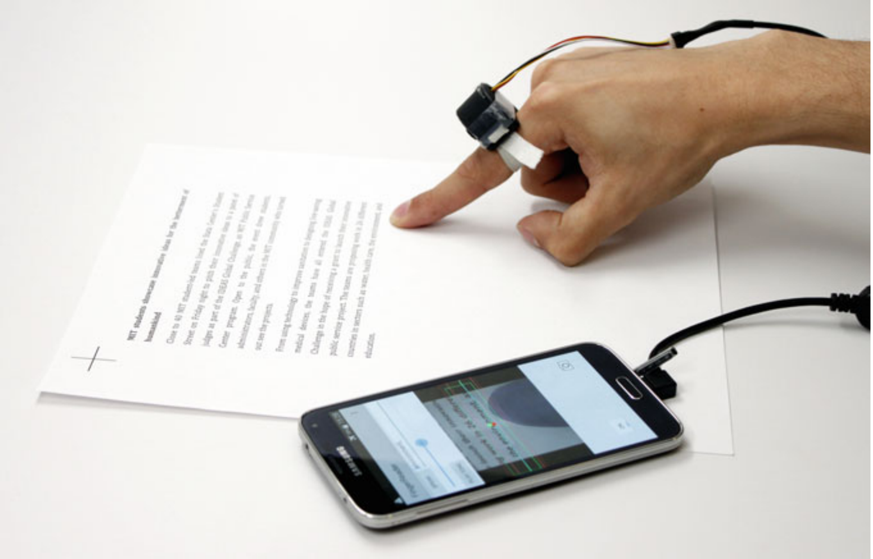


Fig. 5 The MobiReader camera peripheral

advantage for residual eyesight when using the MobiReader for reading, as some totally blind users actually had more success in reading than users with some residual vision.

The bulk of the details on the MobiReader, it's implementation and evaluation can be seen in [23, 29]. Hereby we describe the major differences the MobiReader made over the original FingerReader, as well as the results from further study into the proposed method of sequential text reading.

4.1 Improvements to the Device Hardware

Bearing resemblance to the FingerReader [31], the MobiReader is designed to be smaller and better adjustable to differently shaped fingers. The 3D-printed plastic case sports adjustable rubber straps and ergonomic design for adhering to the top of the finger. It also contains a considerably smaller camera module than that of the FingerReader, although not as small as the HandSight's NanEye [33]. The MobiReader, in contrast to FingerReader and HandSight, does not contain any vibration feedback capabilities and relies on audio cues alone, which allows it to be smaller and monolithic.

The camera module in use is analog; therefore a USB Video Class (UVC) video encoder is included with the system. The UVC interface allows the MobiReader to connect to practically any device with USB host capabilities and a modern operating system, smartphones included. This way the MobiReader could also be used as a peripheral by anyone carrying a smart device, e.g. a phone or an Android-enabled CCTV magnifier.

4.2 New Mobile User Interface on Android

We ported the implementation of the original computer vision algorithms for the Android platform in a new application (see Fig. 6), which also allows to control the hardware. Through the application, a variety of settings are available to the user that enables customizing the reading experience. Feedback settings can be adjusted: enabling and disabling candidate line feedback, distance, and angles, as well as customizing whether incoming words are read in their entirety or cut off when a new word is found. Speech rate, the speed at which words are read, can also be adjusted.

Android is an accessible operating system with built-in mechanisms to aid VI people to navigate the screens and interact with UI elements. With this new application the MobiReader is far easier for VI persons to adjust to their needs.

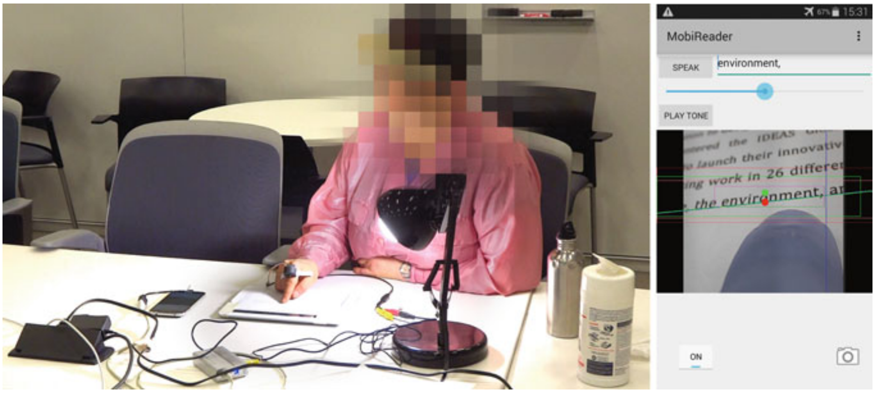


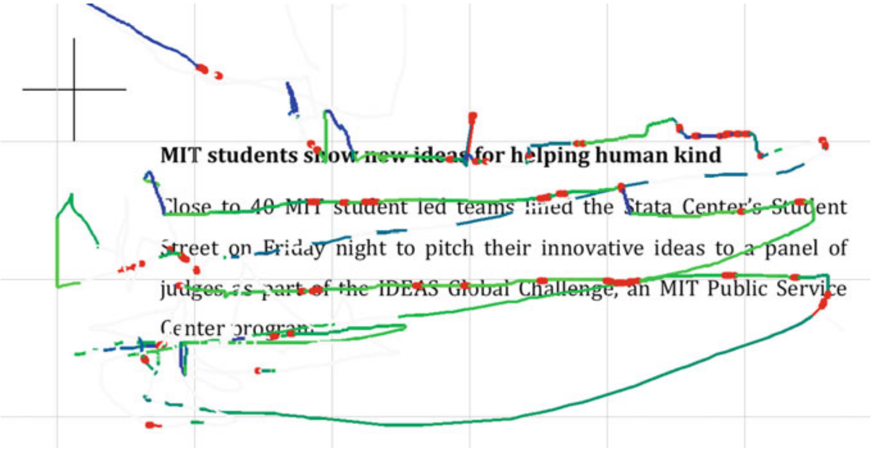
Fig. 6 *Left* P7 in midst reading with the MobiReader. *Right* Android app screen

4.3 Improvements to the Computer Vision Algorithm

The bulk of the algorithms used for the MobiReader are the ones used in [31], however our work contains a number of additional features and improvements. Existence of text (*No Text*/*Candidate Line* states) is determined by the number of qualifying character contours in the focus region, which is determined by the visible tip of the user's finger in the camera frame. If there are more than 2 qualifying characters that form a mutual baseline (tested by means of voting and fitting a line equation) the system transitions to *Candidate Line* state. In *Candidate Line* mode it will look for the first word on the candidate line via OCR.

The OCR engine, based on Tesseract [32], compensates for the distortion caused by the angle the finger takes with the paper. If the text is at an angle w.r.t the image, determined by the precomputed line equation, a 2D central rotation will correct it. Thereafter an intelligent trimming process will remove the whitespace surrounding the first word. We determine the first word by looking for large gaps in the x-axis projection of the words image patch (reducing the rectangular patch to a single row with the MAX operator on each column), similar to [33]. The trimmed patch is small enough to be quickly processed by Tesseract when set to the *Single Word* mode. OCR also does not occur on every frame, rather, only when new candidate words appear, greatly improving performance on our mobile processor.

The finger-tip detection algorithm of [31] was inefficient and expensive to execute in a mobile setting. We therefore introduced a coarse-to-fine method, where we start by analyzing an extremely downscaled (1% in number of pixels) version of the normalized-RGB image and later inspect the rough estimate in a small 100 × 60 pixel window to get a more precise reading. We also incorporate a standard Kalman filter to cope with noise in the measured fingertip point signal (see Fig. 11), which has a detrimental effect on the stability of the algorithms down the pipeline (Fig. 7).



MIT students show new ideas for helping human kind

Close to 40 MIT student led teams lined the Stata Center's Student Street on Friday night to pitch their innovative ideas to a panel of judges, as part of the IDEAS Global Challenge, an MIT Public Service Center program.

Fig. 7 Text-Feedback aligned reading session. The *green-blue* marking are a visualization of the *Distance* feedback the user received, while the *red* marking is when word were spoken out loud. *Greener* hues correspond to a closer distance to the baseline, and *blue* hues farther. The *middle line* of the text was missed, however the user, a completely blind person, was still able to recover 73% of the words in total and spend over 51% of the time in line-tracking mode

4.4 Evaluation of the MobiReader

Work on the FingerReader did not include quantitative measurements and did not test an end-to-end system. The primary contribution of the MobiReader was a quantitative assessment of the complete system, including its computer vision subsystem, as used by a larger group of visually impaired persons. We recruited 10 participants to undertake monitored usage tasks and interviewed them about their experience. In total 10 reading tasks were designed to contain text of different sized fonts, layout and two variations of the audio feedback.

The feedback condition was the independent variable in a within-subjects design: Distance (D) and Distance + Angle (D + A). In 'Distance' the user hears a continuous feedback of how far their fingertip is from the line, and in 'Distance + Angle' the users hears 'Distance' and also a continuous feedback of the angle their finger makes with the line. Both feedbacks were given as sine waves of different pitches (Distance: 540–740 Hz, Angle: 940–1140 Hz). Each feedback condition was crossed with the tasks (5 tasks for D and 5 tasks for D+A) and fully counterbalanced to remove order bias.

4.5 Findings from the Quantitative Study

For metrics we designed three measurable effects: (i) Consecutive Score, which measures the amount of correctly and consecutively extracted words, (ii) Total Words

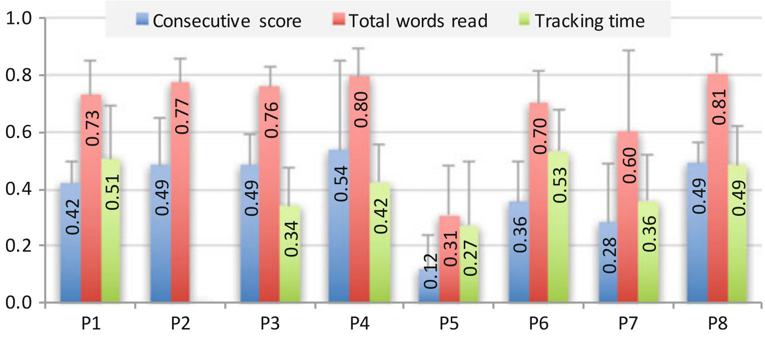


Fig. 8 Individual success in reading per participant

Read, which counts the number of correctly read words from the text without regard to sequence, as well as (iii) Tracking Time, which is simply the proportion of time the user spent in line-tracking mode versus line-searching mode.

On average our participants were able to correctly extract 68% (SD = 21%) of the words in the text, however some participants were able to extract up to 81% on average (see Fig. 8). The Distance feedback was somewhat better in helping users extract words from the text with 74% (SD = 18%) of the words on average, relative to 63% (SD = 22%) for Distance + Angle. Bigger font size only had a small positive effect w.r.t percent of extracting words (e.g. 72% for 11pt and 68% for 9 pt), but made a bigger impact in terms of the *Consecutive Score* (with 0.51 for 11 pt and 0.37 for 9 pt), which suggests, as one would expect, that larger font is easier to track.

As the *Consecutive Score* is not an absolute measurement, but rather a suggested model of the proficiency of a user in utilizing the MobiReader, it only can serve as a comparative measurement. As such, it does flush out the variance in users capabilities when it comes to feedback. Users not only extracted more words with only 'Distance' feedback turned on, they were also capable of extracting more consecutive words, with a score of 0.47 versus just 0.33 for 'Distance + Angle'.

The *Tracking time* measure provided little information as to how successful users were in reading, in spite of a correlation coefficient of 0.677 with *Total Words* and 0.526 with *Consecutive*. Interesting to note P6, the best participant in terms of time spent in the tracking modes (53% of the time), who was an Optacon user and understood very well the concept of text line tracking.

4.6 *Qualitative Feedback*

Open ended interviews with our participants revealed that all, save for one, did not appreciate the Angle feedback and were confused by multiplexing Distance + Angle (N = 8). Most users (N = 5) also mentioned the usage of the device causes excessive arm strain in keeping the finger and wrist straight and tense, as well as having to be very accurate and make very slight constrained movements (N = 3). Three users stated they would not use MobiReader to read long pieces of text, even though it was generally agreed that the device design was comfortable and small (N = 5). Some complained the overall reading process was slow (N = 3).

The prevailing reported strategy (N = 5) was to go top-to-bottom, i.e. finding the top line from the top of the page and working down to the next lines, as well as tracing backwards to the left to find the first word on the line; however backtracking was contested by some (N = 3).

Some users expressed dislike for the feedback in general, claiming the tone and increasing volume when straying from the line induced more panic than suggestion. At times this was reflected by large movements that could throw the user off the current line.

The evidence gathered in the MobiReader study largely corroborates the findings of the FingerReader studies. The auditory feedback was at times confusing, and users did far better with a simpler feedback, and fatigue was marked as a prevailing issue. While some users were very pleased with their newfound capability to explore text with their finger using a standard mobile device, there was still a general agreement that long pieces of text are better off read with an app or a dedicated device. We concluded that better algorithms for image analysis and text-tracking can alleviate some of the problems and create a smoother experience.

5 MusicReader: A Printed Music Sheet FingerReader

The latest evolution of the FingerReader's sequential reading algorithms is the MusicReader: a printed sheet music reading algorithm. While reading sheet music shares many traits of reading plain printed text, it also brings about many challenges to solve. Our work resulted in a unique Optical music recognition (OMR) algorithm that is able to sequentially trace and read back the note it encounters to a reader with VI. The motivation for our work came from interviews we held with musicians with VI, which revealed that in order to read printed music sheets they rely on human transcribers or scanning using specialized stationary equipment that often produces recognition errors. Printed music sheets for musicians with VI in accessible formats such as music braille, is generally considered expensive, rare and inefficient for reasons of portability. For musicians with VI participating in music classes or band sessions, this issue creates a barrier between them and their sighted colleagues, as they are not as independent. The MusicReader strives to enable music readers with VI to access non-instrumented paper musical notation sheets in a mobile context, and level the playing field with their sighted peers. Full details of the MusicReader's implementation and evaluations can be seen in [29], and hereby we highlight the main differences over the MobiReader and share only the key findings from the study with musicians with VI (Fig. 9).

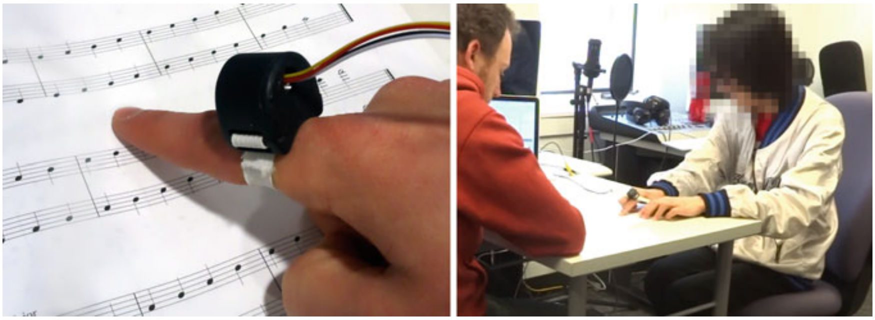


Fig. 9 The MusicReader. *Left* The finger-wearable camera design. *Right* A blind musician in a reading session

Needs Of VI Music Readers. Musicians with VI looking to learn new music mostly use braille music, learning by ears, or digitized music sheets in accessible digital formats. Braille music is a relatively prolific accessible format for encoding musical information based on the braille character set, however it presents a number of acute challenges. Braille music is expensive to produce and thus also to purchase, since it is a niche format for musical notation, which leads to a small offering of music translated to braille notation [13]. In addition, there are only few qualified braille music transcribers who can produce braille music,[1] and few teaching institutions for braille music exist. Braille music translation also results in very heavy and large "printed" books, which is another usability factor impeding accessibility. Although some VI musicians have outstanding aural skills that help them to learn new materials quickly by ear, it's not the case for every musician with VI, and there is information, such as finger markings, that entirely cannot be retrieved simply from listening.

These challenges with learn-by-ear and braille music lead some musicians with VI to learn new music by digitizing printed music sheets using OMR, however this too is not free of limitations. Operating a flatbed scanner requires experience using a screen-reader and the specific printed music digitization software (e.g. SharpEye[2] or SmartScore[3]). The physical setup for scanning is also important for a successful sightless operation, therefore it is usually done in a recognizable location, such as a specialized room or at home. But even in perfect scanning conditions, a properly scanned page will often result in errors in the OMR process. Furthermore, users with VI would not be able to recognize and correct these errors independently since they cannot refer to the original printed music sheet. Scanning in a different scenario, such as using a mobile phone, presents problems of aim, focus and alignment, but more importantly—such mobile music scanning applications for the VI are hardly in existence.

[1] The Library of Congress lists 70 braille music transcribers US-wide: http://www.loc.gov/nls/music/circular4.html.

[2] http://www.visiv.co.uk/.

[3] http://www.musitek.com/.

Problems with existing solutions are more acute in social situations such as a classroom or a band, where musicians with VI are expected to access music handouts in the same way their sighted peers do. Accessible workbooks and pre-digitized music do exist, however readers with VI are confined to this content and cannot spontaneously access printed material. Consequently, a music reading solution tailored for the VI to use in a mobile context could provide them with a way to better integrate into the learning and communal playing environment.

Existing Mobile OMR Solutions. The MusicReader is similar to Gocen [2], a music reading system where the user is allowed to scan a stave notation line using a handheld camera and hear the notes played in real time. However Gocen does not recognize any symbols other than full stemless notes, has no memory of notes outside of its immediate view, as well as it doesn't provide non-visual feedback on the scanning other than playing the note. Another related work is onNote by Yamamoto et al. [35], which uses the index finger as an access pointer to different parts of a paper-printed music sheet and changing the nature of playback with visual projected feedback. The sheets in onNote are scanned into the system beforehand and only matching to the existing database of pre-processed sheets can be performed. The advent of computationally capable smartphones enabled performing OMR on the phone itself within apps [17, 34], however these are not geared towards VI people and provide only visual feedback.

5.1 User Interface and Feedback

Similar to the feedback the MobiReader and FingerReader provide, the MusicReader has two main feedback components: scanning feedback via audio tones, and content (music notation) feedback via speech.

- **Speech Feedback**. Each note encountered in the scan is translated to duration and pitch [5]: "eighth", "quarter", or "half", followed by the pitch class in latin letters (CDEFGAB), and finally the octave ("3", "4" or "5"), for example "eighth-D4". Accidentals are uttered as class and pitch (e.g. "Flat-D4"), and symbols without pitch simply utters the word (e.g. "bar", "quarter rest").
- **Tonal Feedback**: The tonal feedback guides the user in scanning a line of stave notation music. The goal is to help the user keep the finger-camera pointing at roughly the middle of a line, via feedback that describes the distance from the center of the line. The major difference from before is that the feedback is binary: above the line (a high C note), and below the line (a low C note), instead of a continuous varying tone to describe the distance. This simplification greatly reduces cognitive load, as there is no tonal feedback when roughly centered so users can concentrate on the notes utterances. When the system cannot detect any line in the image it emits a quieter G note tone.

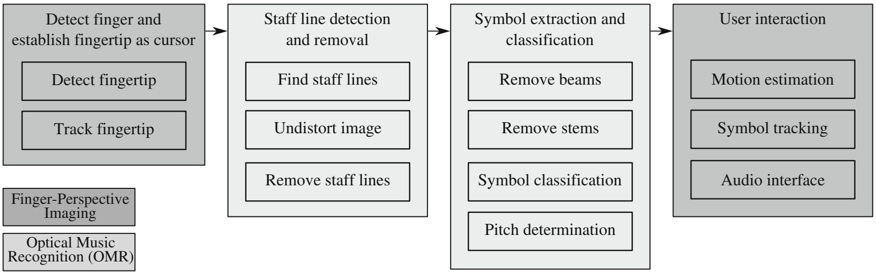


Fig. 10 The processing and interaction pipeline in our assistive music reading system

5.2 *Computer Vision Algorithm for Finger-Perspective OMR*

The MusicReader's computer vision system considers a unique approach for extracting musical information from printed sheets. A local view from the finger perspective is generally considered easier to computationally analyze, but it also introduces additional problems that do not exist in other OMR systems: using the finger as the cursor for the analysis, handling a moving view of the page, and providing feedback on the scanning operation itself rather than the content alone (the musical symbols). These issues augment the traditional requirements from an OMR pipeline, which are also included in our system: staff line detection and removal, segmentation, classification and more (see Fig. 10). With respect to Optical Character Recognition (OCR), OMR is considered harder as the symbols are often converged and multiplexed rather than clearly demarcated as text characters.

Staff Lines Detection, Removal and Tracking. The fingertip location in the image, calculated as was done in the MobiReader (see Fig. 11), allows us to process just a small region of interest where we look for the staff lines using [26]. Assuming the staff lines run from the left to the right extremities of the small region we pick the left-to-right lines that have the most black pixels along them, and then perform binning based on the line's intercept to finally converge to the 5 unique staff lines. To validate, we calculate the distance between neighboring lines as well as their angles, where a good line detection is when all measurements agree within a small variation. For further calculations we extract the staff line space (SLS) and staff line height (SLH) from the detected staff lines. In subsequent frames, we search for new staff lines only within a small region around the previously found lines, to speed up computation.

The staff lines impose a near-uniform 2D rotation of the image, although in some cases, depending on the perspective distortion, the lines disagree on the angle. Using the inverse 2D rotation roughly rectifies the symbols for proper classification (see Figs. 12c, and 13d).

Staff Line Removal. For removing the staff lines and keeping the musical symbols intact we use a Hidden Markov Model (HMM) (see Fig. 12). We sequentially scan the staff line to determine at each point whether it belongs to the staff line or a symbol. The hidden states we utilize are: {STAFF, SYMBOL, SYMBOL THIN, NOTHING}. Observations are based on counting the number of black pixels above and below the staff line, and the transition and emission matrices were manually

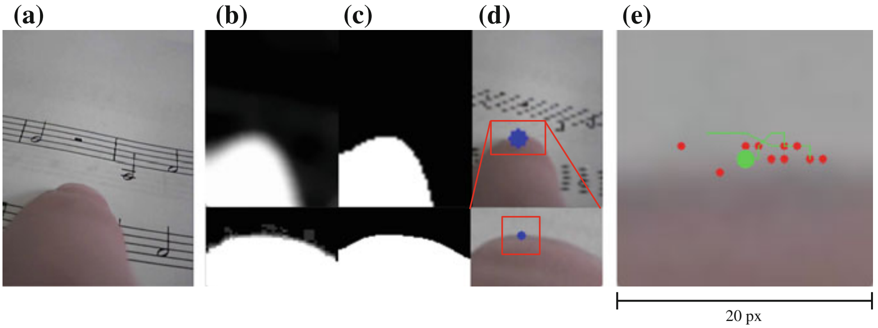


Fig. 11 Fingertip detection process. **a** Original image, Coarse-to-fine detection of the fingertip: **b** detected skin probably map, **c** binarized using Otsu method, **d** detected fingertip point, **e** Kalman filtering of the measurement point (*green*—filtered result, *red*—measurements with noise)

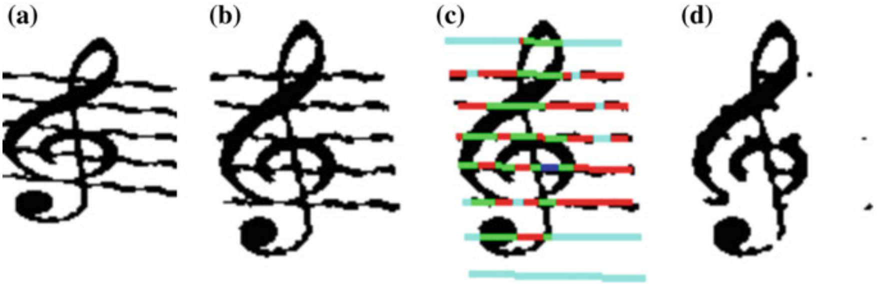


Fig. 12 Classification of staff lines for removal. **a** the original binarized image, **b** the undistorted view, **c** annotated staff lines from the HMM, and **d** the staff lines removed with minimal damage the symbol

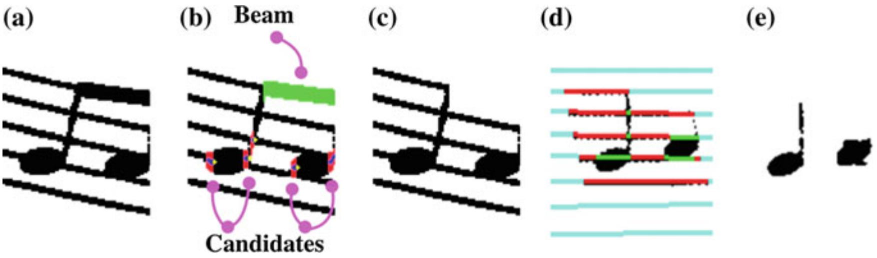


Fig. 13 **a** the binarized input region, **b** detecting consecutive segments of viable height, **c** removing the segments classified as beams, **d** staff line removal on the rectified image, **e** result with beam and staff lines removed, ready for note classification

trained from a number of annotated examples. To discover the annotation for a new staff line we calculate the observation sequence and running the Viterbi algorithm, which is then used to remove all pixels in the STAFF or NOTHING state according to the staff line height (SLH). We used [27] for the HMM implementation.

Beam and Stem Removal To correctly classify beamed groups of notes (e.g. connected eighth notes), we must remove the connecting beam. We detect vertical segments in the image that are likely to be part of a beam (based on their pixel length), and look for consecutive overlapping segments whom centers also converge on a line, since a beam is always a straight line. We finally remove the selected beam segments by painting over the pixels (see Fig. 13).

Many of the note symbols arrive at this point of the OMR pipeline with a stem (the vertical line going above or below the note head): half notes, quarter notes and also connected eighth notes after having their beam removed. However for a simple pitch and duration classification we keep only the note head, which we find from the analyzing the y-axis projection (reduce-sum operation on the rows).

Symbol and Pitch Classification. Once we obtain a clean symbol we use geometric features of the contour with a decision tree classifier to classify the symbol to its type (e.g. note head, accidental, bar line, etc.). Inspired by [24], we use the following features: width, height, area, ratio of black versus white pixels, and 7 Hu moments. The decision tree was trained with a dataset of 1170 manually classified note symbols from a training set of images.

To determine the pitch for note heads we consider the the center of mass and for incidentals (sharps and flats) we use the central point from bounding rectangle. If the symbol central point is within 15% of the SLS to one of the staff lines, we deem the symbol to lie on that line and assign an octave and pitch. See Fig. 14 for an illustration of this process.

5.3 Evaluation of the MusicReader with VI Musicians

To evaluate the performance and usefulness of the system we performed a controlled user study with VI musicians. The goal of the study was to assess the feasibility of the MusicReader to assist in reading a printed music sheet in an unstructured environment, simulating the real situation a person would wish to use the device. We recruited 5 participants from a pool of volunteer VI musicians, and an additional VI musician volunteered to act as a pilot user for the study.

Study participants used two printed music sheets in standard staff notation for reading. The melodies on the sheets were simple arrangements (no harmony) of the following standards: “Happy Birthday”, “Greensleeves”, “Over the Rainbow” and “Amazing Grace” (see Fig. 15). Participants had a chance to try and read the two sheets, and were questioned about whether they can recognize the melodies in them. The read notes and audio feedback events were recorded by the software on the PC along with timestamps, and were later analyzed as quantitative measures.

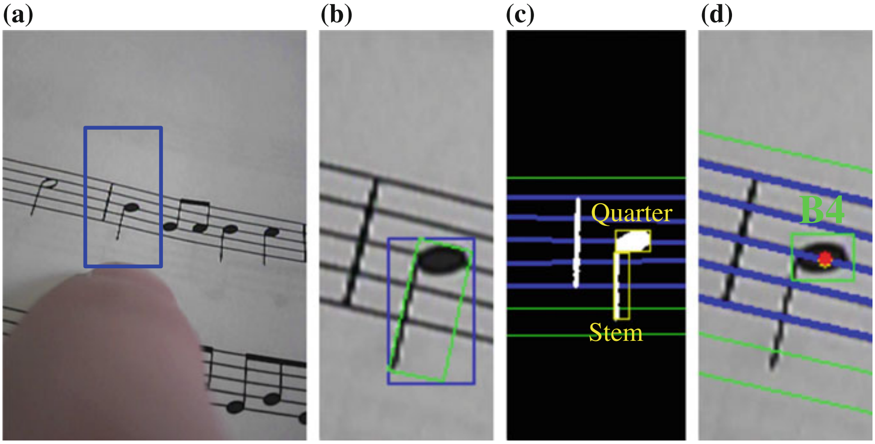


Fig. 14 Symbol pitch and type classification process: **a** original image with region of interest, **b** detected contour to classify, **c** the segmented contour of note-head and stem, **d** the classified note-head based on the position on the staff lines

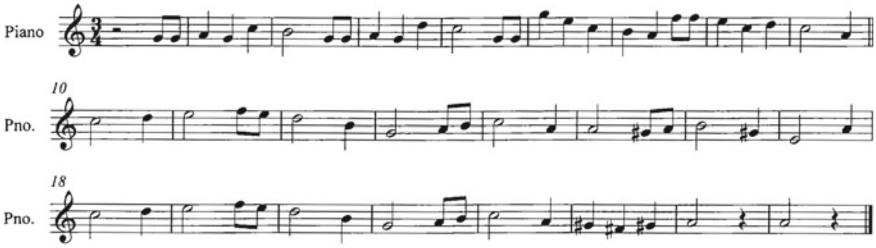


Fig. 15 One of the sheets the participants used for reading, with arrangements of "Happy Birthday" and "Greensleeves"

5.4 *Key Findings from the Quantitative and Qualitative Study*

The MusicReader is a novel approach in the domain of mobile assistive technology for musicians with VI, where most prior work did not attempt to tackle non-visual reading of printed music. As such, study participants had mixed comments about its utility, although there was a positive consensus about its potential.

For the quantitative part, the results show most users were able to cover roughly %35 of the notes from the test sheets in the allotted time for reading independently (20 min). All users were stopped by the examiner at the end of the time frame, therefore given additional time they would continue to read more notes. Thus the result on the proportion of notes read is severely skewed.

In the results of the qualitative part, which consisted of questionnaires and open interviews, participants did not think the MusicReader was easier than other music reading aids, although most reported that they do not know of similar aids. All participants save for one stated they would require an expert to help them operate the device but felt there weren't many things to learn.

Learning-by-ear While most of the participants in our experiment noted the MusicReader is an intriguing technology that would be useful when fully developed, all participants agreed that learning-by-ear is still the best tool they have to access music.

> I would love to be able to read music, but I still consider having aural skills, the ability to learn a piece by ear and play it back, a very important tool. It could be used in combination with reading, and I still think being able to read is a good thing.

On the other hand, our interviewees reported of numerous situations where learning-by-ear is impossible or impractical: band practice, working with a conductor, in the classroom and while teaching. In these situations, according to our participants, musicians with VI at a disadvantage even in spite of their technical abilities.

> Not being able to access printed music material knocks blind people out of a big segment of the market. I don't think I could go audition for the BSO [Boston Symphony Orchestra], even though I think I have the chops to at least play 3rd or 4th trumpet for the BSO, but they want you to be able to sight-read. Not to be able to work on-the-fly like that is a really big problem. [. . .] At the moment I would not be able to teach beginners that don't know how to read, but I can certainly teach them how to play. I feel like that's something that keeps me from teaching beginners privately.

Finger Positioning and Aiming. Most study participants had problems of aiming the device and maintaining the right angle for proper reading. This problem in the MusicReader is key as musicians with VI read with the specific goal of playing their instrument, and therefore can spare, at most, one hand for reading depending on the instrument they play.

> P3: I can't have my left hand off of the trumpet, I must hold it. [. . .] I have to have both hands on the instrument

Study participants were also concerned with getting a very quick and precise reading, and did not have much patience towards learning the hand positioning or maintaining it for long. We conclude that both the imaging hardware as well as the software may need to improve to overcome this problem. The camera lens could be of a wider angle, and the algorithms to find the fingertip and staff lines could have a much higher tolerance towards skewed views (Fig. 16).

> P2: [. . .] the finger needs to be in a very specific position, there should be a better way. The angle was not directly straight with the paper, and I can't see the paper.

Tonal Signals for Reading Music Some participants reported of an increased cognitive load when listening to the assisting tones while trying to mentally reconstruct the music only from the spoken names of the notes. In reading music with the MusicReader this issue of mental interference may be more severe than in reading printed text, which suggests tonal feedback may be less effective. Participants suggested we incorporate tactile feedback to circumvent this issue.

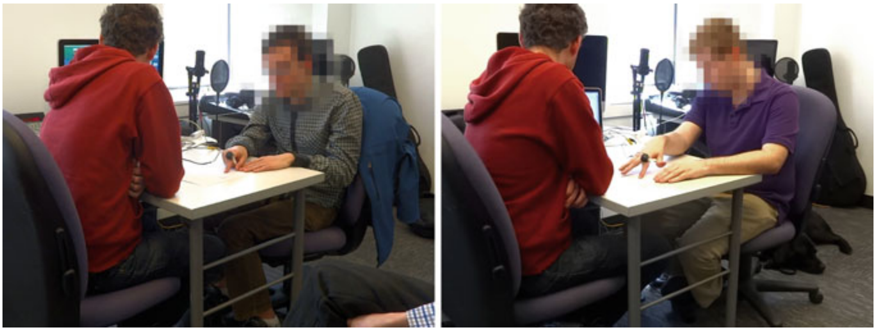


Fig. 16 User study participants in midst reading a music sheet

> P2: The notes are good for feedback, but if you're thinking about the music - that's confusing. Maybe it shouldn't be in the music range, not C,G and C if I am reading something in a C scale.

The findings from our study point both to the potential of the MusicReader as a mobile assistive technology and to the usability obstacles of such an approach. Reading printed music for musicians with VI is not a special case of reading printed text but rather a new problem class. In many situations music is read with the goal of immediately playing it, often in a group setting with other musicians, which requires a fast response, high accuracy and less than ideal reading conditions. Reading music also requires the reader to mentally reconstruct the music, which can interfere with any audio feedback from the system. Nevertheless, some elements of reading music are similar to reading text such as locating oneself in the page.

6 Latest Design Iterations of the FingerReader Hardware

Since the early lab prototypes of the FingerReader (Fig. 2), efforts were devoted to improve the usability of the FingerReader device. Further iterations on the camera electronics have reduced the size considerably, and went hand in hand with industrial design iterations on the casing and materials. Finally, the new device has a much smaller form factor and higher comfort (see Fig. 2e). The onboard miniature camera operates at VGA resolution (640 × 480 pixels) and 30 frames-per-second, and connects via standard USB to a PC, smartphone or smartwatch. The latest version of the FingerReader design was produced in order of thousands, and many devices were distributed to users and organizations pending a large-scale user study.

7 Conclusion

The FingerReader is a unique assistive augmentation interface for reading by pointing. Over the last 4 years we led many design and development iterations, demonstrations and evaluations to assess the FingerReader's feasibility as an assistive technology for visually impaired persons. Results of numerous user studies with the target audience—persons with VI—show clear potential for FingerReader to be used in exploring printed documents. However, there is an obvious need to improve on a number of fronts: the computer vision algorithms must improve to allow for more intuitive usage with less guidance, the feedback mechanisms must match the application as well as support and not obstruct the content.

Acknowledgements We would like to acknowledge the people who were directly involved in the ideation, creation and evaluation of the FingerReader: Connie Liu, Sophia Wu, Marcelo Polanco, Michael Chang, Sabrine Iqbal, Amit Zoran, K.P. Yao. We would like to acknowledge the help of the VIBUG group in MIT for testing and improving the FingerReader.

References

1. Ubi-Camera (March 2012). http://www.gizmodo.in/gadgets/Finger-Camera-Lets-You-Frame-a-Shot-Like-a-Pompous-Director/articleshow/19139922.cms
2. Baba T, Kikukawa Y, Yoshiike T, Suzuki T, Shoji R, Kushiyama K, Aoki M (2012) Gocen: a handwritten notational interface for musical performance and learning music. In: SIGGRAPH 2012 emerging technologies. ACM
3. Bigham JP, Jayant C, Ji H, Little G, Miller A, Miller RC, Miller R, Tatarowicz A, White B, White S, Yeh T (2010) VizWiz: nearly real-time answers to visual questions. In: Proceedings of the 23nd annual ACM symposium on user interface software and technology, UIST '10, New York, NY, USA. ACM, pp 333–342
4. Byrne JH, Dafny N (1997) Neuroscience online: an electronic textbook for the neurosciences. The University of Texas Medical School at Houston, Department of Neurobiology and Anatomy
5. Crombie D, Dijkstra S, Schut E, Lindsay N (2002) Spoken music: enhancing access to music for the print disabled. In: Computers helping people with special needs. Lecture notes in computer science, vol 2398. Springer, Berlin, pp 667–674
6. d'Albe EEF (1914) On a type-reading optophone. Proc R Soc Lond A 90(619):373–375
7. David M (Oct 2007) Every stalker's dream: camera ring
8. El-Glaly YN, Quek F, Smith-Jackson TL, Dhillon G (2012) It is not a talking book;: it is more like really reading a book! In: Proceedings of the 14th international ACM SIGACCESS conference on computers and accessibility, ASSETS '12, New York, NY, USA. ACM, pp 277–278
9. Ezaki N, Bulacu M, Schomaker L (2004) Text detection from natural scene images: towards a system for visually impaired persons. Proc ICPR 2:683–686
10. Hanif SM, Prevost L (2007) Texture based text detection in natural scene images—a help to blind and visually impaired persons. In: CVHI
11. Hedberg E, Bennett Z (Dec 2010) Thimble—there's a thing for that
12. Horvath S, Galeotti J, Wu B, Klatzky R, Siegel M, Stetten G (2014) FingerSight: fingertip haptic sensing of the visual environment. IEEE J Trans Eng Health Med 2:1–9

13. Jacko VA, Choi JH, Carballo A, Charlson B, Moore JE (2015) A new synthesis of sound and tactile music code instruction in a pilot online braille music curriculum. J Vis Impair Blindness (Online) 109(2):153
14. Kane SK, Frey B, Wobbrock JO (2013) Access lens: a gesture-based screen reader for real-world documents. In: Proceedings of the SIGCHI conference on human factors in computing systems, CHI '13, New York, NY, USA. ACM, pp 347–350
15. Lee H (Sept 2011) Finger reader
16. Linvill JG, Bliss JC (1966) A direct translation reading aid for the blind. Proc IEEE 54(1):40–51
17. Luangnapa N, Silpavarangkura T, Nukoolkit C, Mongkolnam P (2012) Optical music recognition on android platform. In: Advances in information technology. Communications in computer and information science, vol 344. Springer, Berlin, pp 106–115
18. Mattar MA, Hanson AR, Learned-Miller EG (June 2005) Sign classification using local and meta-features. In: CVPR—workshops. IEEE, p 26
19. McNeill D (2000) Language and gesture, vol 2. Cambridge University Press
20. Nanayakkara S, Shilkrot R, Yeo KP, Maes P (2013) EyeRing: a finger-worn input device for seamless interactions with our surroundings. In: Proceedings of the 4th augmented human international conference, AH '13, New York, NY, USA. ACM, pp 13–20
21. Pazio M, Niedzwiecki M, Kowalik R, Lebiedz J (2007) Text detection system for the blind. In: 15th European signal processing conference EUSIPCO, pp 272–276
22. Peters J-P, Thillou C, Ferreira S (2004) Embedded reading device for blind people: a user-centered design. In: Procedings of the ISIT. IEEE, pp 217–222
23. Polanco MRII (2015) Mobireader: a wearable, assistive smartphone peripheral for reading text. Master's thesis, Massachusetts Institute of Technology
24. Rebelo A, Fujinaga I, Paszkiewicz F, Marcal ARS, Guedes C, Cardoso JS (2012) Optical music recognition: state-of-the-art and open issues. Int J Multimedia Inf Retrieval 1(3):173–190
25. Rissanen MJ, Fernando ONN, Iroshan H, Vu S, Pang N, Foo S (2013) Ubiquitous shortcuts: mnemonics by just taking photos. CHI '13 extended abstracts on human factors in computing systems, CHI EA '13. New York, NY, USA. ACM, pp 1641–1646
26. Roach JW, Tatem JE (1988) Using domain knowledge in low-level visual processing to interpret handwritten music: an experiment. Pattern Recognit 21(1):33–44
27. Sand A, Pedersen CNS, Mailund T, Brask AT (2010) HMMlib: a C++ library for general hidden Markov models exploiting modern CPUs. In: 2010 ninth international workshop on parallel and distributed methods. IEEE, pp 126–134
28. Shen H, Coughlan JM (2012) Towards a real-time system for finding and reading signs for visually impaired users. In: Computers helping people with special needs. Springer, pp 41–47
29. Shilkrot R (2015) Digital digits: designing assistive finger augmentation devices. PhD thesis, Massachusetts Institute of Technology
30. Shilkrot R, Huber J, Liu C, Maes P, Nanayakkara SC (2014) FingerReader: a wearable device to support text reading on the go. In: CHI EA. ACM, pp 2359–2364
31. Shilkrot R, Huber J, Meng Ee W, Maes P, Nanayakkara SC (2015) Fingerreader: a wearable device to explore printed text on the go. In: Proceedings of the 33rd annual ACM conference on human factors in computing systems, CHI '15, New York, NY, USA. ACM, pp 2363–2372
32. Smith R (2007) An overview of the tesseract OCR engine. In: ICDAR, pp 629–633
33. Stearns L, Du R, Oh U, Wang Y, Findlater L, Chellappa R, Froehlich JE (Sept 2014) The design and preliminary evaluation of a finger-mounted camera and feedback system to enable reading of printed text for the blind
34. Viktor L (Nov 2014) iSeeNotes—sheet music OCR!
35. Yamamoto Y, Uchiyama H, Kakehi Y (2011) onNote: playing printed music scores as a musical instrument. In: Proceedings of UIST. ACM, pp 413–422
36. Yang XD, Grossman T, Wigdor D, Fitzmaurice G (2012). Magic finger: always-available input through finger instrumentation. In: Proceedings of the 25th annual ACM symposium on user interface software and technology, UIST '12, New York, NY, USA. ACM, pp 147–156

37. Yarrington D, McCoy K (2008) Creating an automatic question answering text skimming system for non-visual readers. In: Proceedings of the 10th international ACM SIGACCESS conference on computers and accessibility, Assets '08, New York, NY, USA. ACM, pp 279–280
38. Ye H, Malu M, Oh U, Findlater L (2014) Current and future mobile and wearable device use by people with visual impairments. In: Proceedings of the SIGCHI conference on human factors in computing systems, CHI '14, New York, NY, USA. ACM, pp 3123–3132
39. Yi C (2010) Text locating in scene images for reading and navigation aids for visually impaired persons. In: Proceedings of the 12th international ACM SIGACCESS conference on computers and accessibility, ASSETS '10, New York, NY, USA. ACM, pp 325–326
40. Yi C, Tian Y (2012) Assistive text reading from complex background for blind persons. In: Camera-based document analysis and recognition. Springer, pp 15–28