UNIVERSITY OF CALIFORNIA
SANTA CRUZ

INCREASING PRINTED DOCUMENT ACCESSIBILITY WITH GUIDED IMAGE ACQUISITION

DOCTOR OF PHILOSOPHY

in

COMPUTER ENGINEERING

by

Michael P. Cutter

June 2015

The Dissertation of Michael P. Cutter
is approved:

Professor Roberto Manduchi, Chair

Professor Sri Kurniawan

Professor James Davis

Dean Tyrus Miller
Vice Provost and Dean of Graduate Studies

Table of Contents

List of Figures

List of Tables

Abstract

Increasing printed document accessibility with guided image acquisition

by

Michael P. Cutter

Printed text accessibility is an issue that impacts mobility, scholastic achievement, and thus career growth. Our research addresses how to make printed text accessible to a blind person.

We start this dissertation with a formal definition of the range of positions and orientations from where an OCR readable document image can be acquired from. Initially, we conducted a naive user studies with a wizard-of-oz system that relied on fiducial markers printed on a document to facilitate visual odometry in order to study if blindfolded people could capture compliant document images with the help of our system. We then moved on to blind participants with an experiment meant to simultaneously assess the range of problems with document image acquisition and test different interaction modalities statistical effect on the time to capture an image.

Finally we devised a computer vision algorithm, without fiducials, capable of verifying document image compliance and providing instructions for acquisition. We tested this algorithm with participants in a redesigned counterbalanced repeated measures experiment. Our analysis revealed that guidance significantly reduces the amount of time necessary to capture a compliant image. Our participants expressed positive comments on the system, and generally felt that their proficiency at taking OCR-readable images had increased by interacting with the system.

Chapter 1

Introduction

Optical Character Recognition (OCR) is a technique which transforms an image of handwritten or typed glyphs into a sequence of recognized characters. The accuracy rates of constrained OCR problems such as the MNIST hand written digit data currently exceed 99% [33]. This level of accuracy allows applications of OCR into our daily lives such as automatic license plate readers on the golden gate bridge. OCR is one of the most widely accepted success stories from the pattern recognition community and has enabled a new frontier of document and printed text accessibility.

A visionary in the OCR community, Ray Kurzweil, realized that OCR combined with CCD flatbed scanner and text-to-speech technology could be a breakthrough technology for the blind. In 1976 he held a press conference with National Federation of the Blind and released a finished product, the Kurzweil Reading Machine [6]. These document to speech solutions that combine flatbed scanners, OCR, and text-to-speech synthesizers have made meaningful affect in blind people's lives [57]. However, CCD flatbed scanning solutions require nearly a full table-top of hardware. Blind people need to be able to access text on the go. This desire for mobile OCR was also first answered by Kurzweil with the Kurzweil National Federation of the Blind (KNFB) Mobile reader [7], which was the first powerful mobile OCR solution for the blind and visually impaired. This solution ran on the Nokia N82 phone. Along with its capability to read printed text documents it also had an optional feedback system. This system provides instructions to the user if all four corners of the page are visible amongst other signals. However, this system was not sold as software, but as an entire package costing nearly 2000 dollars. This cost, along with insurance companies' reluctance to fund assistive technology, proved prohibitive for wide-scale adoption of the KNFB Mobile reader.

There is increasing interest in mobile applications that allow a blind person to access printed information such as restaurant menus, bills, signs on a door or on a wall, class handouts, etc. The computational power of modern smartphones, combined with high quality onboard cameras, is enabling the development of OCR-based, low-cost apps that have great potential for benefitting the blind community. The fact that these software systems run on mainstream platforms (Android and iOS), rather than on customized devices, is an important bonus, since the latter are often expensive, lack support, and, like many assistive technology tools, are sometimes not well accepted due to the associated "stigma". Moreover, the fact that mobile devices are becoming accessible without sight (witness the great success of the iPhone among the blind community thanks to the Voiceover utility) bodes well for wide acceptance and adoption of accessibility apps.

Unfortunately, even the best OCR algorithm fails if the text in the image is cropped, has low resolution, is blurred, or badly lit. For sighted users, this is not a problem: one just needs to look at the scene through the screen, moving and orienting the phone until the desired text document is correctly framed and exposed before taking a shot. Blind users, however, cannot see the screen, and thus have very reduced ability to take an OCR-readable picture of a document. Some recent apps facilitate acquisition of OCR-readable images for people who are blind as well as for the general public (e.g., the Bank of America deposit app). These apps typically decide when to take a snapshot based on the presence of text in the image (TextDetective, kNFBReader) or upon detection of the four corners of the document (Prizmo). However, the former approach cannot ensure that the text in the document is detected in its entirety (text may be cropped; Fig. 1.1 (a)) or that the text is OCR-readable (it may be imaged at too low resolution; Fig. 1.1 (b)). In addition, requiring all four corners to be visible may be too restrictive (Fig. 4.1 (b)) or may fail if there is low contrast (Fig. 4.1 (c)). If a non-OCR-readable image is taken, OCR may produce no text at all or garbled/incomplete text, requiring a second snapshot to be taken from a different position. Often times, multiple pictures may have to be taken before successful OCR reading. A different strategy may be to use a stand to support the smartphone (e.g., the Samsung Optical Scan Stand, which works only for the Galaxy Core Advance model). However, this requires carrying one more accessory, limiting the appeal of this otherwise well designed device.

In order for OCR results to be semantically meaningful the entire composition of the document must be visible in the viewing frustum of the camera. Further to maximize OCR quality resolution should be maximized. Camera captured images of documents often

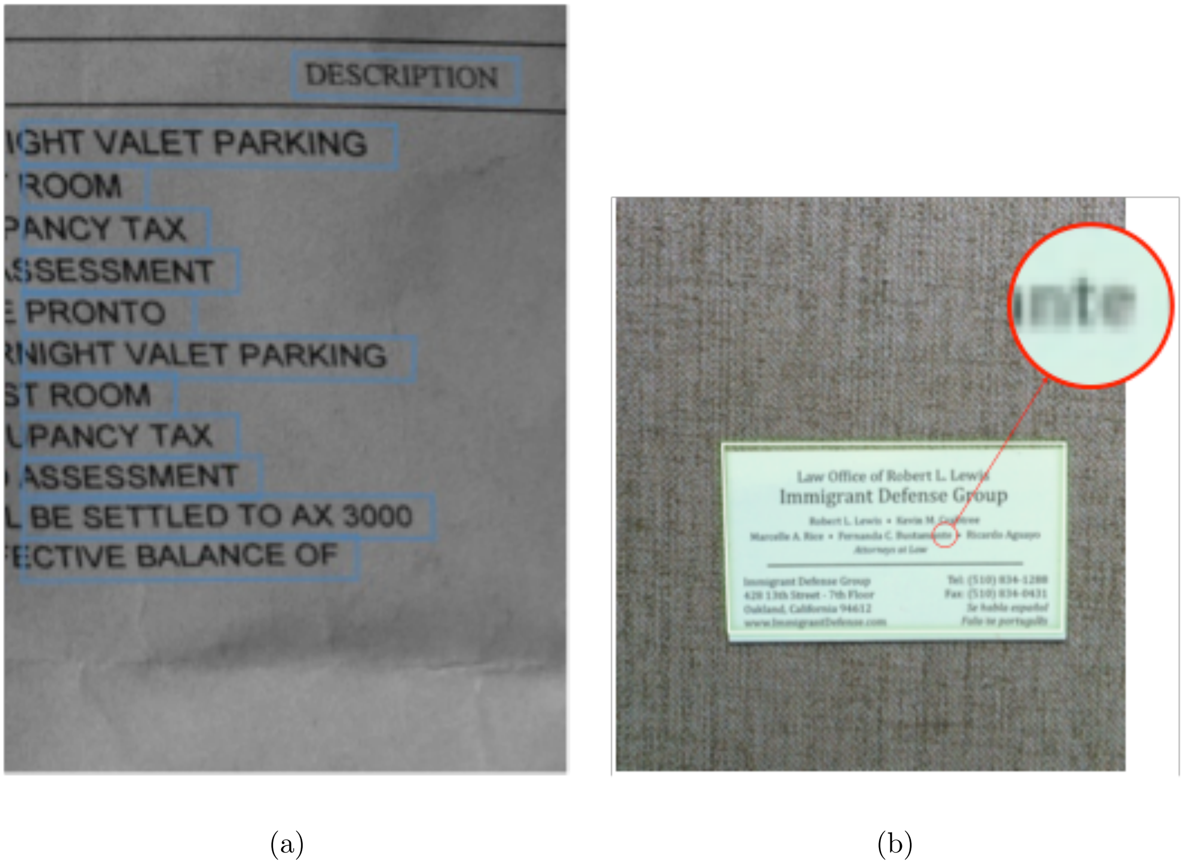


Figure 1.1: Non-OCR-readable images (detail). (a) Text lines were successfully identified by the TextDetective app (blue rectangles), but parts of the lines are not visible. (b) The business card was correctly framed (yellow rectangle) by the Prizmo app, but the resolution is too low for OCR (see zoomed-in inset).

suffer from perspective distortion, motion blur, and uneven lighting. In order for OCR results to be semantically meaningful the entire composition of the document must be visible in the viewing frustum of the camera and captured at sufficient resolution. Perspective distortion should also be minimized as the further the viewing angle changes from planar the smaller the minimum resolvable distance becomes. Therefore, meaningful OCR results are more likely when the entire document fills the majority of the image plane and when the image is taken from overhead. Our software supplies real-time feedback that helps someone capture an image of a document as if it was scanned. Further, our application is intended to work on any document and to be made widely available for free on the Apple iOS appstore.

Our research addresses how best to guide a blind or visually impaired photographer in real-time to take an image in which most of the document is visible and recognizable. We aim to evaluate if an automatic guidance system is able to ensure that images taken by visually impaired photographers meet these requirements for complete and meaningful OCR results.

1.1 Related work in printed text accessibility and assisted image acquisition

Document scanners coupled with OCR and text-to-speech have been used successfully by many blind people to access printed text [56]. Kane et al. [46] developed an augmented reality digital desk assistive environment which allows blind people to interact with complex paper documents. Their acquisition technology is a mounted desktop camera, which captures a live stream of images. The largest contour in the image is assumed to be the document and processed by optical character recognition. One of their user interface contributions is an "edge menu", inspired by the author's previous work [47]. The edge menu displays an alphabetical list of detected words. By clicking on an word on the list translational guidance is spoken to the image coordinate where the word was detected.

In recent years, a number of mobile OCR applications have been introduced to the market, to enable quick text access "on the go". The KNFB Mobile reader [7] and Blindsight's Text Detective [11] iPhone app are perhaps the best known such systems. The

KNFB reader, which runs on the Nokia N82 phone, generates an optional “field of view report” via synthetic speech a few seconds after a picture has been taken of a document. This report contains information about the angle of the camera relative to the page and about whether all corners are visible or some text is cut off. By carefully holding the phone in position after the first picture has been taken, the user may be able to re-position the camera, if needed, so as to take a better framed picture. In practice, after taking a snapshot with KNFB, one has to wait for OCR to be completed before realizing that the shot was not compliant. Since multiple shots are normally needed, the whole process may be intolerably slow (possibly several minutes). However, KNFB just released an iOS version of their application which might remedy some of the latency issues. Unlike the KNFB Reader, Text Detective lets the user move the phone over the document, processing images continuously as they are taken by the phone’s camera. As soon as an image is found containing text-like patterns, the phone vibrates briefly and the OCR process (which takes a few seconds) is started. This “opportunistic” approach is made possible by a fast text detection algorithm that is used to select promising images to be passed to the more computationally intensive OCR. However, their text spotter does not measure compliance: it will take a picture as soon as some text-like pattern is seen, possibly resulting in truncated lines etc. Only after OCR processing will the user find out that the shot was not compliant and that the hovering operation needs to be restarted. Often multiple hovering-OCR iterations are necessary, resulting in a long acquisition time. A similar opportunistic strategy is taken by an iPhone app named Prizmo [5], which processes each input image to find the edges of a rectangular document.

None of these smartphone OCR applications ensure that a blind user will be able to take a well-framed image of the document. In order to help a person take a good picture of a document, the use of mechanical stands has been proposed (e.g. the Optical Scan Stand tool that is available for the Galaxy Core Advance handset). These devices may be very useful for fixed-size documents, but do not allow the user to reduce or increase the distance to the document, which is often necessary to account for small font size or large document size.

The difficulty of taking good pictures without sight represents a hurdle not only for mobile OCR, but also for other applications of camera-based information access, as well as for recreational photography. For example, Bigham et al. [20] used simple computer vision techniques along with crowdsourcing to help a blind user point the camera correctly to an object (for example, to better identify it or to get closer to it). Brady et al [22] analyzed the type of objects blind people take photos of in a crowdsourcing answer–seeking scenario. Their analysis also includes photo quality assessment. They found that 46% of the questions asked by their recent power users regarded reading some text. The use of remote sight operators, who can look at the image taken by a blind person and provide advice on how to orient the camera to take a better picture, was also considered by Kutiyanawala et al. [49] in a tele-assistance system for shopping. Sight on Call, an NIH-funded project at Blindsight, Inc., is building an on-demand service that relies on a distributed workforce to provide assistance to old and/or visually impaired users.

Zhong et al. [78] developed a key-frame selection algorithm, to be used in combination with a cloud-based visual search engine designed to help blind people identify objects.

Experimental results showed that automatic key-frame selection from a video led to higher success rate compared to when users themselves decide when to take a snapshot.

EasySnap and PortraitFramer are mobile applications developed by Jayant et al. [44], that give feedback to a blind photographer about the scene light, or about the presence and location in the picture of an object or of a person. The use of real-time feedback to help a blind person document transit accessibility by taking pictures of the scene was also studied by Vazquez and Steinfeld [72]. In this scenario, there is no clearly defined "target" (e.g., a face) that could be used to guide framing. Instead, a general–purpose saliency map is used to select a region of interest. A camera-based system for barcode access, equipped with a guidance mechanism that suggests how to move the camera in order to precisely center a detected barcode, was developed by Tekin and Coughlan [70]. The process of taking a precisely framed picture of a document for OCR processing could potentially be facilitated by stitching together multiple pictures, each containing a partial view of the document, into a panoramic image (or mosaic) of the whole document, as suggested by Zandifar et al. [76]. A similar mechanism was used by Zhong et al. [79] in their RegionSpeak system to facilitate exploration of a spatial layout.

A different approach to text reading was recently proposed by Shilkrot et al. [68], who designed a finger-mounted camera that can be used to scan a text line. They explored continuous tone and haptic feedback to alert the user that they have reached the end of a textblock; or have veered too far from the textline. There is also a commercial product worn as glasses called OrCam [13] that provides real time OCR by users pointing their head and finger at the block of text they wish to be read. However, neither of these approaches ensure that the user has captured the entire document and both require that the person buy a dedicated piece of technology. Although these systems have shown potential, we believe that use of a commodity hardware such as a smartphone may have wider appeal than a specialized assistive technology device.

Chapter 2

Compliant pose space of document

Abstract

Increasing printed document accessibility with guided image acquisition

by

Michael P. Cutter

Here we present an evaluation of an ideal document acquisition guidance system. Guidance is provided to help someone take a picture of a document capable of Optical Character Recognition (OCR). Our method infers the pose of the camera by detecting a pattern of fiduciary markers on a printed page. The guidance system offers a corrective trajectory based on the current pose, by optimizing the requirements for complete OCR. We evaluate the effectiveness of our software by measuring the quality of the image captured when we vary the experimental setting. After completing a user study with eight participants, we found that our guidance system is effective at helping the user position the phone in such a way that a compliant image is captured. This is based on an evaluation of a one way analysis of variance comparing the percentage of successful trials in each experimental setting. Negative Helmert Contrast is applied in order to tolerate only one ordering of experimental settings: no guidance (control), confirmation, and full guidance with confirmation.

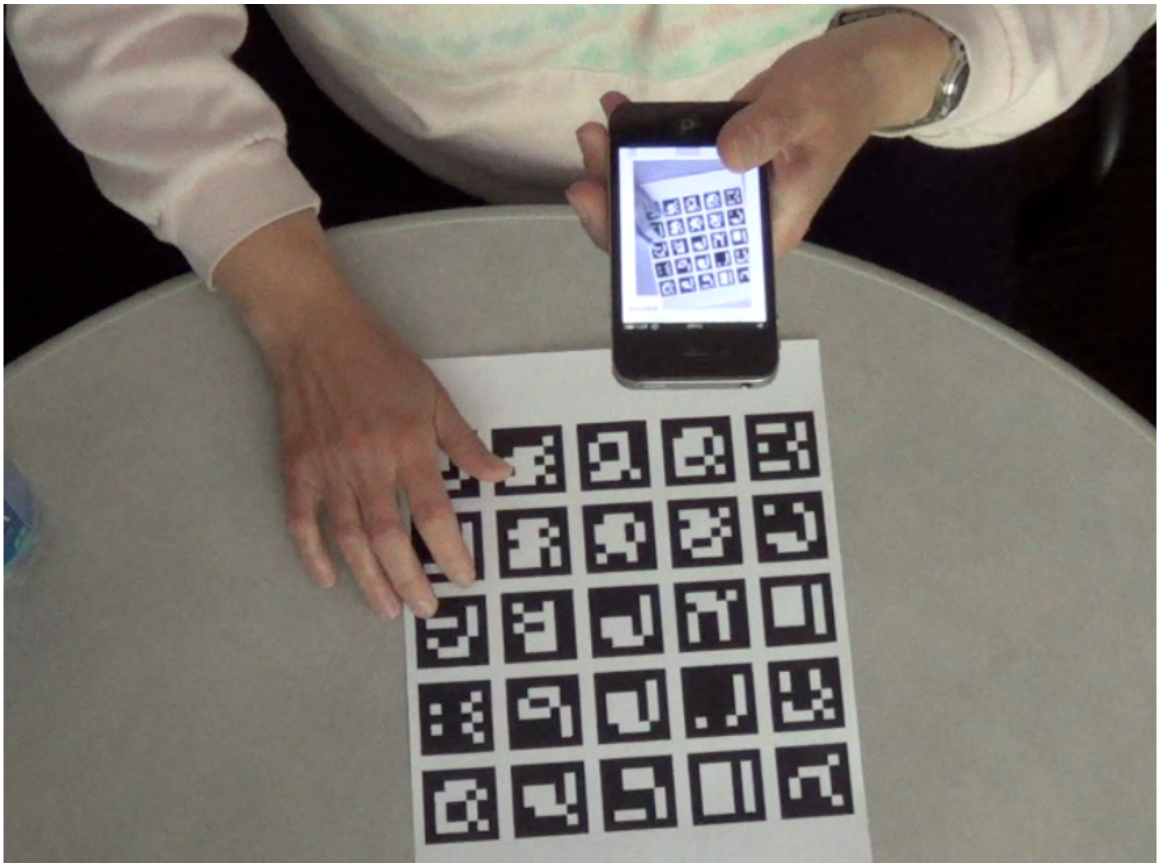


Figure 2.1: A participant positioning the iPhone over the document printed with the ArUco fiducials.

2.1 Introduction

Our research addresses how best to guide a blind or visually impaired photographer to take an image where most the entire document is visible and recognizable. We aim to evaluate if an automatic guidance system is necessary, and whether such a system will be able to ensure that images taken by visually impaired photographers meet these requirements for complete and meaningful OCR results.

In Section 2.2 we derive the theoretical aspects of our model. Next we discuss the design of the experiment and the metrics in Section 2.3 and in Section 2.4 we address our hypotheses with an evaluation of the user study. Finally, we motivate future work in the final Section 2.5.

2.2 Method

We have conducted an experiment to assess the potential of a document image acquisition guidance. Our contribution is a principled approach that maximizes the chance of a document being recognized by OCR. This method requires that the position of the camera of the phone be known with respect to the document. We determine pose optically for this experiment by placing fiducial marker on a document. We used an open source augmented reality package, ArUco [12], for fiducial marker generation and detection. ArUco is implemented in the OpenCV library [21].

Clearly, this an idealized case and a regular document does not contain fiducial markers that can be used for pose estimation. However, we note that previous studies have already considered adding tags to documents for augmented reality purposes such as in work by Guimbretière [41], Paper Augmented Digital Documents and therefore it is conceivable that are system could be employed in tagged documents. Other work by Nakai et al. [60, 43] address locating the position of a camera with 'locally likely arrangement hashing' [59] affine invariant features by posing the task as a retrieval problem. Liang et al. [53] work, Camera-Based Document Image Mosaicing, could also play a role in document image acquisition assistance.

Once the fiducial markers are detected, we know the correspondence between the 2D location of the marker on the image plane and the 3D-position of the markers on the document. Then we can use these correspondence to solve for the pose of the camera with respect to the page. Generally the problem of solving for pose given 2D image points to 3D world points on a plane (the document image) is known as the perspective-n-point problem [64]. An efficient non iterative solution, Efficient Perspective-n-Point Camera Pose Estimation [52], to the perspective-n-point is used for this purpose.

2.2.1 Compliant Space

Our real time guidance algorithm verbalizes instructions as a function of the current pose to guide the user to acquire an image of a document that can be completely read by OCR. We define this type of image as a compliant image. There exist a set of possible phone positions from where compliant images of the document can be acquired. We call this the Compliant Space, a bounded tetrahedron in 3D world coordinates.

The Compliant Space is formally defined as a set of poses where compliant images can be captured. A compliant image must satisfy the following conditions (1) when all four corners of the page are visible and (2) a small letter printed anywhere on the page maps to a sufficient number of pixels for accurate OCR. An example of Compliant Space with these assumptions is visible in Figure 2.1. The mathematical definition of these two requirements (1) in viewing frustum and (2) minimum reading distance are discussed next. Note that capital letters are used to denote 3D coordinates and matrices and lowercase letters are for 2D coordinates. Vectors and matrices are bold-faced. The four corners of the page are each denoted $\mathbf{\Pi}_i$ and π_i, in the world and image plane respectively. In the following section we will explain how the Compliant Space is derived.

In Viewing Frustum

The camera's viewing frustum is a function of its focal length, principal point, and pose, $\mathbf{P}$. Pose can be decomposed into a translation, $\mathbf{T}$, and a rotation matrix, $\mathbf{R}$, from the page to the camera. We assume the camera is calibrated [77] its unique intrinsic matrix $\mathbf{K}$ is therefore known. We assume no radial distortion. We set the reference frame centered in the middle of the document. All points on document are on the z plane origin. We can calculate the viewing frustum in world coordinates relative to the document by back-projecting the image plane corner points to 3D rays.

First we can solve for the camera location in world coordinates $\mathbf{C}_w = -\mathbf{R}^{-1} \cdot \mathbf{T}$ and for pixel $\mathbf{p}$ homogeneous (x,y,1) we can solve for the visible 3D location in world coordinates X_w, Y_w, Z_w. The set of points define the viewing frustum.

$$(X_w, Y_w, Z_w) = \mathbf{C}_w + \lambda \mathbf{R}^{-1} \mathbf{K}^{-1} \mathbf{p}$$

Since all the points on the page are on a plane where the z coordinate is equal to zero we can calculate λ and solve for X_w Y_w. In other words we can recover the 3D position of a pixel coordinate of the marker.

$$\lambda = \frac{Z_w - C_{w,z}}{r_3}$$

$$\text{where } (r_1, r_2, r_3)^T = \mathbf{R}^{-1} \mathbf{K}^{-1} \mathbf{p} \text{ [58]}$$

We can find the bounds of the viewing frustum in world (meter) coordinates using the following relation. Without loss of generality a corner of the image plane can be denoted as $\mathbf{p}_i$.

$$(X_w^i, Y_w^i) = \mathbf{C}_w - \frac{C_{w,z}}{r_3}\mathbf{R}^{-1}\mathbf{K}^{-1}\mathbf{p}_i \tag{2.1}$$

This functions maps the pixel corners, $\mathbf{p}_i$ to visible world coordinates (X_w^i, Y_w^i). With this relation the four corners of the image plane define the viewing frustum in world coordinates. Then by testing if all four corners of the page, $\mathbf{\Pi}_i \in n$, are within the viewing frustum we can determine if the entire page is visible.

Minimum reading distance

OCR requires a sufficient resolution for accurate recognition. For scanned documents the rule is that a document should be scanned at least at 300 dpi [54]. For camera captured images we will focus on the generally accepted rule of thumb that a text line should map to at least 12 pixels [76]. According to typographic standards for Latin script the height of a 'x', x-height, is a standard measure from the baseline to average height of a letter. A lowercase 'x' in 12 point Arial font (a font approved by the American with Disabilities Act [14]) has a height 4.23 millimeters which forms a constraint that anywhere a 'x' could be printed must map to 12 pixels in the image plane. We use this equality to define the Compliant Space.

A document can be recognized by OCR if an 'x' printed on each corner $\mathbf{\Pi}_i$ (location of corner of page in world coordinates) maps to 12 pixels of the image plane. This can be calculated by projecting two 3D points into rays for each corner; 3D point $\mathbf{\Pi}_i^U$ is the upper point of the theoretical 'x' and $\mathbf{\Pi}_i^L$ is the lower point of the theoretical 'x'. Recall we have already calculated Pose, $\mathbf{P}$, where $\mathbf{P} = [\mathbf{R} \mid \mathbf{T}]$. $\mathbf{P}$ is a 4 by 4 matrix where the fourth row is (0 0 0 1). For this computation it is convenient to map to homogeneous coordinates to compute the projective transform with matrix multiplication. In order to compute 3D to 2D projection we matrix multiply by camera intrinsic $\mathbf{K}$ and camera extrinsic $\mathbf{P}$. In the following equations. Without loss of generality we can find $\pi_i^{U_x}$ from $\Pi_i^{U_x}$ with the following relation. Recall that $\Pi_i^{U_z} = 0$ as all points on the page are along the $Z = 0$ plane.

$$(q_1, q_2, q_3)^T = \mathbf{K} \cdot \mathbf{P} \cdot (\Pi_i^{U_x}, \Pi_i^{U_y}, \Pi_i^{U_z}, 0)^T$$

$$\pi_i^{U_x} = \frac{q_1}{q_3}$$

Therefore we can compute the x-height in pixels of a 12 point Arial 'x' that is theoretically located at each $\boldsymbol{\Pi}_i$ of the page.

$$\text{x-height}_i = \sqrt{(\pi_i^{U_x} - \pi_i^{L_x})^2 + (\pi_i^{U_y} - \pi_i^{L_y})^2} \tag{2.2}$$

If $\forall i \in n$ such that x-height$_i \geq 12$ pixels is satisfied for all four corners then minimum reading distance is observed.

Guidance Algorithm

We approximated the problem of finding the shortest path to the Compliant Space by instead finding the shortest path to a 3D-line within the Compliant Space. We call this 3D-line the Reduced Compliant Space Center Line, L, which is between $(0, 0, .28)$ and $(0, 0, .42)$ in world coordinates (meters). This approximation is sufficient and does not lead to non-compliant images because for any pose we can determine compliance in real-time through conditions (1) and (2).

The guidance algorithm computes the shortest path from the current position, $\mathbf{C}_w$, to the closest point on L by projecting the point $\mathbf{C}_w$ onto the 3D-line L. The closest

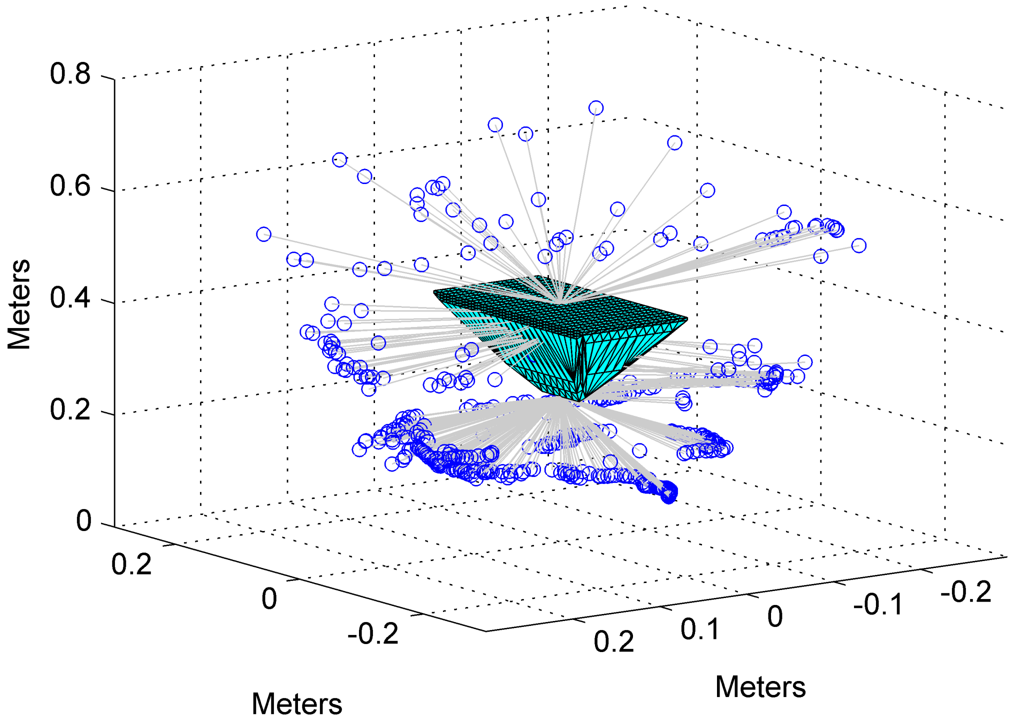


Figure 2.2: Plot of a camera trajectory in 3D space. The compliant space is shown as the green tetrahedron. Each blue circle represents a camera position. At each detection an approximate shortest path (see section 2.2.1) can be found from camera position $\mathbf{C}_w$ to the closest point $\mathbf{O}_w$ on the Reduced Compliant Space Center Line.

point on L to $\mathbf{C}_w$ is called $\mathbf{O}_w$. The software then verbalizes the two axes in need of the most correction. The instructions come in centimeter units such as “move the phone up 15 centimeters and forward 9 centimeters”. Additionally, the software notifies the user if they are holding the phone at a significant tilt or angle relative to the page. This is important because the Compliant Space assumes the phone is held directly over the page.

2.3 Evaluation

A blind person trying to take a snap shot of a document to be recognized by OCR typically has difficulties taking a compliant image of any document. Our goal is to the verify whether a guidance system as described in the previous section can facilitate a blind person in taking compliant images. To perform this, we establish two metrics to assess the ability of a person without sight to acquire a compliant image of a document.

We can extract measures for each user per trial and aggregate them per Experimental Setting (ES). The most important measure is the percentage of successful trials, which is equal to $\frac{1}{8} \cdot$ Count of Successful Trials. It is also interesting to measure the distance from the closest point in the Reduced Compliant Space Center Line the user reached. This is the euclidean distance from current camera position to the point $\mathbf{O}_w$, which is equal to $\|\mathbf{C}_w - \mathbf{O}_w\|$.

2.3.1 Design of experiment

We used convenience sampling to recruit all eight of our participants. They are all adults who consented to being in the study. All participants are not blind and are therefore blindfolded, all but one participant was male. We understand that there will be differences between how blindfolded and blind users react to our system, therefore a follow up user study with the target community will soon commence.

The experiment began by handing the blindfolded participant a backpack that contained the document with fiducial markers. Then each participant heard the following instruction.

"Inside this backpack is a single piece of paper. Please take it out and feel for the side with a sticker. This side has the fiducial markers and should remain face up so it is visible to the camera. The sticker should be on the top left of the piece of paper with respect to you."

The participant is then handed a phone and told to tap the screen to begin the trial. The phone vibrates to alert the user that the trial has begun. The participant has ninety seconds to complete. After the timeout the trial ends unsuccessfully.

Next we explain each of the Experimental Settings (ES) used in the experiment.

- **ES-Control** Provides no guidance. The user clicks the volume button when they believe the phone is in the correct position.

- **ES-Confirmation** There is no guidance. However, as soon as the phone has captured a compliant image the software alerts the subject and ends the trial.

- **ES-Confirmation+Guidance** In this setting the user receives continuous guidance as described in section 2.2.1. As soon as the phone has captured a compliant image the software alerts the subject and ends the trial.

2.3.2 Prototype software

We implemented a real time marker recognition and guidance prototype iPhone application to obtain data to answer our hypotheses. The application is programmed to run each experimental setting for eight trials, the application logs the pose of the phone relative to the marker at approximately 3 hertz.

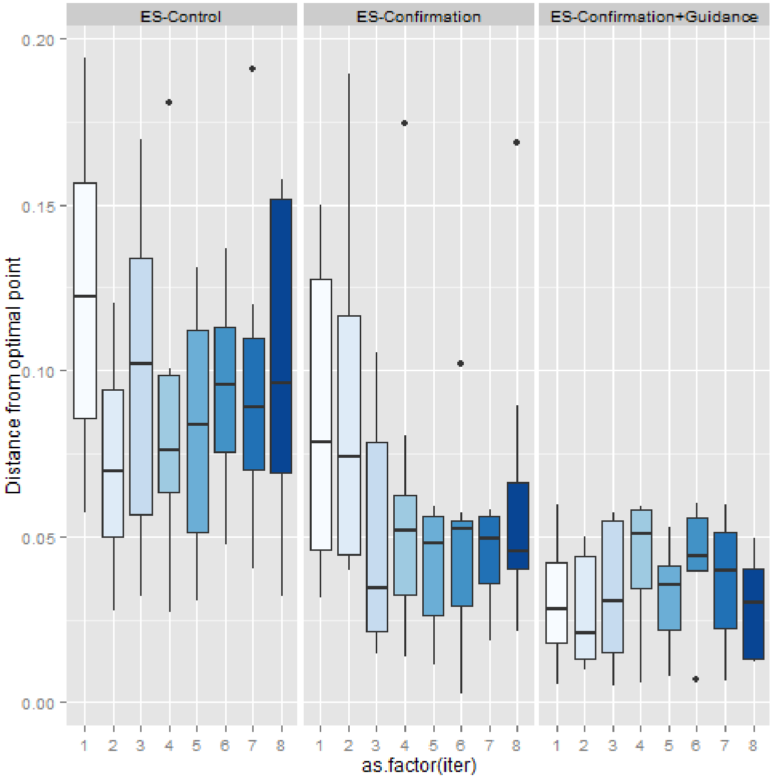


Figure 2.3: The above plot is a facet box plot. The main facets are the experimental settings. Each experimental setting consists of eight trials differentiated by shading (best viewed in color). We can see a trend of small distances as users become more familiar with our system. These are per trial measures.

2.4 Results and Discussion

In this section we will summarize our findings based on the experiment conducted with our prototype software. Our hypothesis is that our guidance system will help people take a greater percentage of compliant images than they would be capable of without guidance. Starting with data exploration we can see a clear trend of lower distances from the optimal position by examining Figure 2.3.

In order to answer our hypothesis we turn to our measure of the percentage of successful trials. Since we knew we had a limited number of study participants we used negative Helmert Contrast [26] to design a tractable experiment that can tolerate only one orderings of experimental settings. Helmert Contrast compares the mean of a experimental setting with the means of previous experimental settings, and is often used in a medical context were an investigator is trying to discover the right dose of medicine. With this methodology we first to establish a baseline ES-control to get a measure of where the user is before intervention. Next, the user is given ES-just-confirmation and finally the user is provided eight trials of ES-full-guidance. This experiment design will allow us to answer whether or not guidance improved the user's ability to take a compliant image of the document. However, this experimental design does not rule out the effect of reordering experimental setting and therefore it is future work to redo the study with a Latin Square design.

Improvement trends are visible by examining Figure 2.4, were we can see a large improvement from the control in the subsequent trials. In addition, in order to statistically test for improvement we conducted the following analysis.

From the one way Analysis of Variance (see Table 2.1) it is clear that between the ES-Confirmation+Guidance there is a significant improvement in the percent of trials that a compliant image is achieved. Between the ES-Confirmation and ES-Control there is a small enough p-value to indicate a trend but not small enough to be significant at the 95% confidence level.

2.4.1 User Experience Report

After completing the experiment we asked each of the users to describe their experience. The general consensus is that the ES-Confirmation+Guidance was preferable

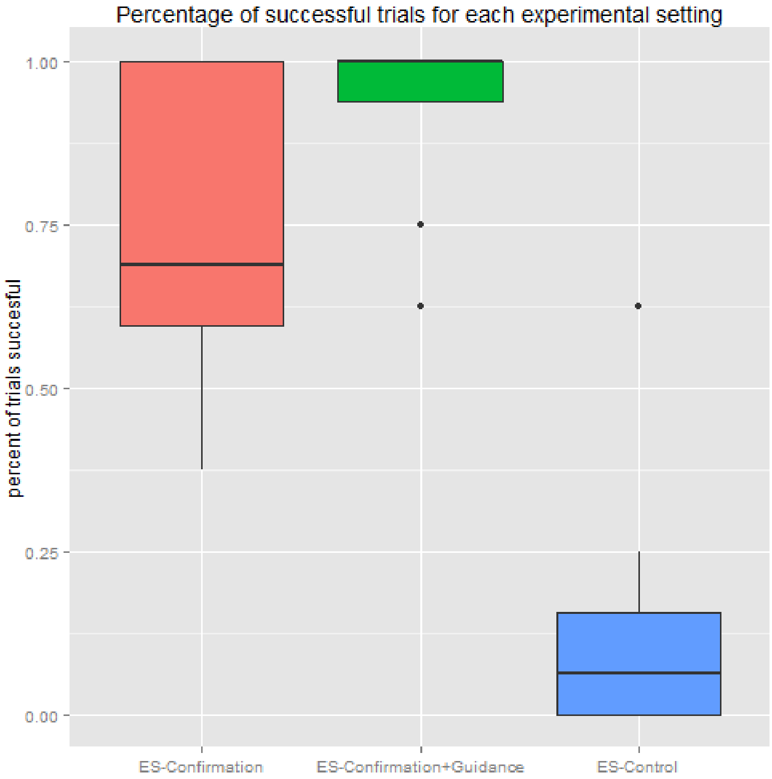


Figure 2.4: This box plot shows users percentage of successful trials for each experimental setting.

	Df	Sum Sq	Mean Sq	F value	Pr(>F)
group	2	2.66	1.33	31.04	0.0000
group: C1	1	0.14	0.14	3.28	0.0845
group: C2	1	2.52	2.52	58.79	0.0000
Residuals	21	0.90	0.04		

Table 2.1: This table is the summary of the one way ANOVA for the percent of succesful trials for each of the ES. 'Group:C1 is measuring ES-Confirmation compared to the ES-Control. group:C2 is comparing ES-Confirmation+Guidance to ES-Confirmation and ES-Control. The latter is significant at the 95 percent level.

over ES-Control or ES-Confirmation. This is also supported by the average user's increase in accuracy after intervention.

Users reported frustration with one of the guidance features, which alerts the user if the phone was in the compliant space while the marker was not in the viewing frustum (Equation 2.1). The message states "Make phone aligned and level" to indicate an improper orientation of the phone with respect to the document. However, this message is too vague since the user is unsure which of the three axes are improperly aligned. A superior version of the software would indicate if the phone is not level or if the phone is rotated around the optical axis.

A learning curve is noticeable as the participants gained a sense of where the acceptable space is located. Some users would spend the first few trials patiently probing possible positions to find the Compliant Space in ES-Just-Confirmation. However, once the user found the correct position they were much faster in returning to the Compliant Space in subsequent trials. Users typically became frustrated if they did not find the Compliant

Space before the timeout occurred. Once they found the Compliant Space for the first time this frustration appeared to subside.

2.5 Conclusion

We demonstrated that that our guidance system significantly improves compliant document image acquisition on camera phones. Our experiment provides evidence to support further research and development in adding non-obtrusive fiducial markers to documents. In addition, future work on an improved guidance system will take into account post-hoc user feedback before beginning. If successful this will provide far better document accessibility for the blind and visually impaired.

Acknowledgment

The project described was supported by Grant Number 1R21EY021643-01 from NEI/NIH.

2.6 Transition: compliant pose space to user study one

Experiments with sighted, blindfolded participants using a system discussed in this chapter were conducted to validate the feasibility of such an approach. This study used a naive guidance algorithm. In fact, sighted people are likely to develop, through daily experience with vision-mediated camera handling, mechanisms and skills that are very different from those available to blind people, and thus cannot, even when blindfolded, be considered representative of blind users for the tasks considered in the experiments. In the next chapter we present a re-designed guidance algorithm, new experiments, a better evaluation criteria, and only considered blind participants.

Chapter 3

Towards Mobile OCR: Understanding the problems with document image acquisition without sight

Abstract

Increasing printed document accessibility with guided image acquisition

by

Michael P. Cutter

The advent of mobile OCR (optical character recognition) applications on regular smartphones holds great promise for enabling blind people to access printed information. Unfortunately, these systems suffer from a problem: in order for OCR output to be meaningful, a well-framed image of the document needs to be taken, something that is difficult to do without sight. This contribution presents an experimental investigation of how blind people position and orient a camera phone while acquiring document images. We developed experimental software to investigate if verbal guidance aids in the acquisition of OCR-readable images without sight. We report on our participant's feedback and performance before and after assistance from our software.

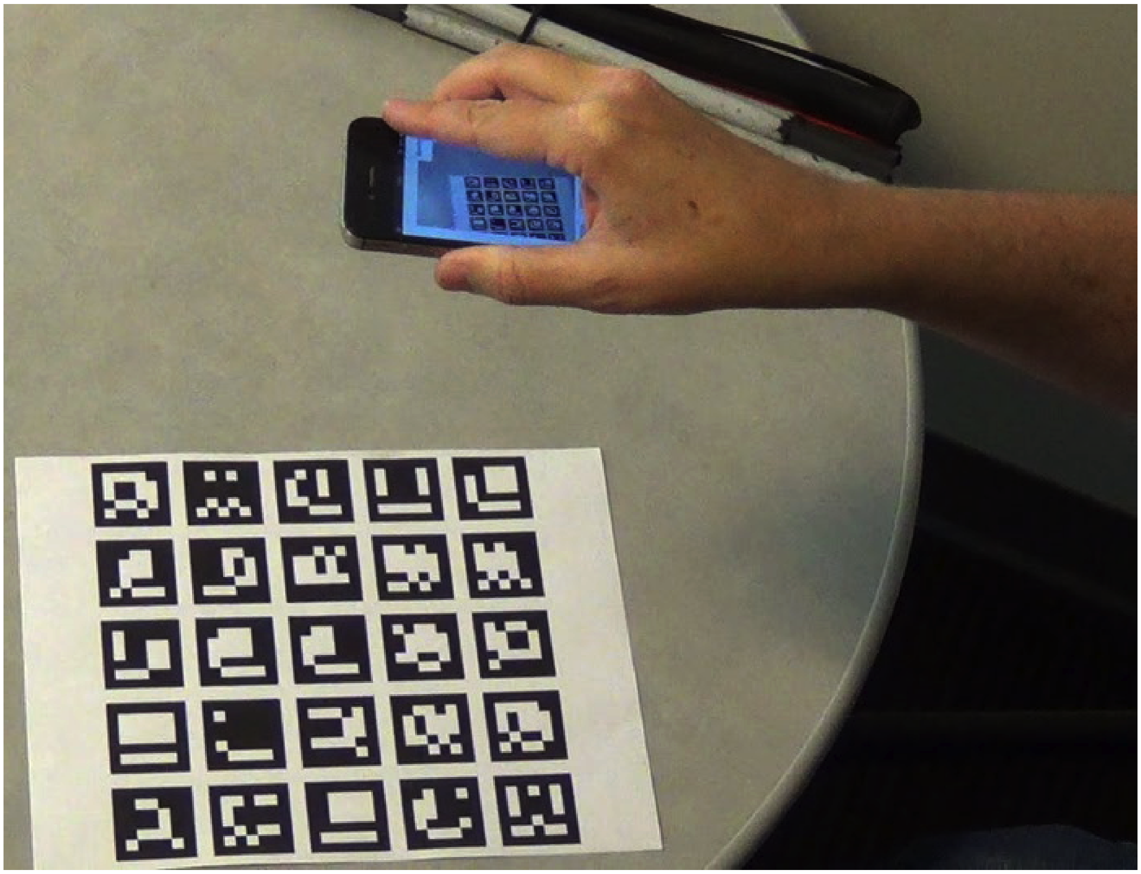


Figure 3.1: A participant positioning an iPhone over a document printed with ArUco fiducials.

3.1 Introduction

This contribution presents an experimental study with 12 blind participants. We first investigate how they hold and position the camera during image acquisition. Then they use our software that provides feedback while he or she tries to take an OCR-readable picture of a document. This experimental tool offers two modalities of usage. The first modality provides real-time confirmation when the user has reached a *compliant pose*, that is, when he or she has moved the camera to a position from which an OCR-readable image of the whole document can be taken. The second modality utters spoken directions to the user about where to move the phone next in order to increase the likelihood of reaching a compliant pose. After trying out our software we measure how they hold the camera again without assistance from our system.

In its current implementation, this tool is, in fact, only an experimental device. In order to function, it requires that the document be printed with special fiducials, and cannot be used on regular printed documents. More precisely, this study investigates how a blind user may become more proficient at taking images of a document *while* using this software, as well as how blind persons may use this tool to train themselves to take better pictures of a document *without* using the software. Still, this investigation has value in that it explores the "best case scenario" of a perfectly functioning device; similar functionalities on regular printed documents are not out of reach of modern computer vision technology.

This experimental study addresses two important research issues, concerning: (1) the ability of blind people to correctly position a smartphone in order to obtain an OCR-readable picture of a document; (2) the potential for increasing the success of this task by means of system-generated feedback. These results will hopefully inspire more research into mechanisms that could enable more efficient use of mobile OCR applications, and thus allow better access to printed information for blind users.

3.2 Method

3.2.1 Overview and Rationale

Our goal in this work was to shed light on the process by which a blind person can operate a hand-held camera (embedded in a smartphone) to access text data printed on a document. We assume that the user can rely on OCR software capable of decoding printed text provided that: (1) the entirety of the page is visible, and (2) text is imaged with a certain minimum size. Furthermore, we assume that the OCR software is able to decode text at any orientation and even with noticeable perspective distortion (due e.g. to camera slant), as these factors can be corrected by proper image processing. In these conditions, text access can be obtained as long as the user is able to take a proper (*compliant*) picture of the document, in a way that is precisely formalized in a later section.

The main questions driving our investigation are:

1. *How difficult is it to take a compliant picture of a document without sight?* To the best of our knowledge, there are no published studies about the ability of blind people to maneuver a camera in order to take a readable picture of a document. Our research seeks to establish a baseline against which any proposed assistive technology for mobile OCR can be compared.

2. *Could this process be facilitated by proper system-generated feedback?* We considered two different approaches to provide feedback to the user. In the first approach, the system continuously takes images (frames) and analyzes each image to verify whether the imaged document is readable; as soon as a compliant (readable) image is taken, the user is notified and the process is stopped. In the second approach, the system additionally provides instructions to the user about where to move the phone to increase the likelihood of a compliant picture being taken.

To address these questions, we developed the necessary experimental software tools and designed experiments. We decided to emulate an "ideal" OCR software and feedback mechanism by means of an image processing system based on augmented reality (AR) markers. Rather than dealing with a regular printed document, our participants interacted with a sheet of paper on which a number of AR markers (*fiducials*) were printed. Based on the image taken by the iPhone camera of these fiducials, the system quickly and robustly identifies its own position and orientation (collectively called *pose*) with respect to the document. This information is sufficient to establish whether the image of a "real" document of known size taken from the same camera pose would be OCR-readable (i.e., the pose is compliant), and to provide feedback and guidance to the user. This almost-Wizard-of-OZ mechanism allows us to abstract from the actual OCR software employed and to concentrate on the user interaction component of the system, under the assumption of an "ideal" image processing software. Using this tool, we can ascertain whether feedback mechanisms have potential for improving the user experience with mobile OCR without sight, which would justify further research in this direction; additionally, this system allows us to investigate the most promising strategies to present feedback to the user.

3.2.2 Population

Given that the technology considered in this work is meant solely for blind users, we only recruited blind participants for our experiments. We did not consider sighted blindfolded participants as this could bias the results (since, as observed in multiple contexts, sighted users – even when blindfolded – and blind users often perform very differently under the same task.) Twelve blind participants (four females and eight males) were recruited through announcements on newsletters and word of mouth. All but one participant had at most some residual light perception. The participant who had some residual vision left had acuity of 20/3800 in one eye; the other eye had no vision (prosthetic). In order to remove any possibility that the little residual vision could bias results, this participant was blindfolded during the test. The participants were of age between 18 and 65, with a median age of 53. Causes of blindness were: acquired retinitis pigmentosa (RP) (2), congenital RP (2), retinal damage at birth (2), trauma involving both eyes (2), traumatic brain injury (1), retinoblastoma (1), Leber's congenital amaurosis (1), and cornea damage caused by Stevens-Johnson Syndrome (1). Of these participants, two were congenitally blind, two became blind at age three, and all others lost their sight after the age of ten. Two of the participants had lost their sight less than five years prior to the experiment. Seven participants were regular iPhone users, and four participants had tried mobile OCR systems before (but were not regular users of this technology).

3.2.3 The Compliant Pose Space of a Document

A *compliant* picture of a document is a picture that contains all of the text in the document, at enough resolution that it can be read by OCR. More precisely, a picture of a letter-sized (8.5" by 11") document is considered compliant for the sake of this study if: (1) all four corners of the printable area are visible, where in our case the printable area has top and bottom margins of 1.5" and left and right margins of 0.5"; and (2) a small letter placed anywhere in the printable area is seen in the picture at enough resolution that it can be read accurately by OCR. A "small letter" could be, for example, a lowercase 'x' character typed in 12 point Arial font, which has height of 4.23 mm. By "accurately readable by OCR" , we mean that the height of the letter in the image should be of at least 12 pixels [76]. This is based on the readability constraint discussed in [29] calculated at 8MP photo resolution of the iPhone. Thus, a compliant image of a document is such that the whole content can be read via OCR. Note that we define compliance only in geometric terms: factors such as bad illumination or blur certainly contribute to the quality of OCR reading, but are neither considered in this definition nor in this study.

We define *compliant pose* as the pose (3-D location + camera orientation, with respect to a reference system fixed with the document) of a camera that takes a compliant picture. Note that the compliance of a pose depends on the camera's optical/imaging characteristics (intrinsic parameters [42]). For example, a pose that is compliant using a wide field-of-view lens may be non-compliant using a longer lens (because the document may not be seen in its entirety in the second case). Likewise, a compliant pose for a narrow field-of-view lens may be non-compliant for a shorter lens due to reduced angular resolution.

For a given camera, the set of all compliant poses form the *compliant pose space*. The compliant pose space of a document can be computed based on geometry. In addition, given a non-compliant pose, one could predict whether moving the camera in a certain direction and rotating it around a certain axis will result in a compliant pose. This information may be used in a guidance mechanism to provide hints to the user about how to move the camera in order to take an OCR-readable image. Of course, this assumes that the camera pose can be somehow computed – a difficult problem in itself. Several techniques are available for image-based pose estimation, ranging from stereo triangulation (when a system with two cameras is available), to structure from motion/SLAM, to methods that use fiducials printed on the page at known locations.

In our study, we used printed fiducials for camera pose estimation. In fact, in our experiments we give away completely with textual information, and use a document containing solely well-calibrated fiducials instead (see Fig. 3.1). This approach is justified by the fact that the goal of this investigation is to study the mechanisms that can facilitate reaching a compliant pose and thus obtaining an OCR-readable image of the document. In this way, we are able to separate the *technical* difficulties of pose estimation from the *human factors* that pertain to holding a camera and taking a compliant picture.

3.2.4 Interaction Modalities

We considered three different interaction modalities in our study. Each modality represents a mechanism by which the user may try to take a compliant picture of a document using a smartphone. The three considered modalities are described below.

Snapshot

In the *snapshot* modality, the user simply takes a snapshot of the document (e.g. by pressing a button or tapping the screen), from a position and orientation that, in his or her judgment, results in a compliant picture. No feedback is provided by the system, except to confirm (via synthetic speech) that a snapshot action was registered.

Hovering: Just Confirmation

In this case, the user moves the camera over the document ("hovering") while the system takes and processes pictures continuously. As soon as a compliant picture is detected, the system notifies the user and the process is stopped. The user is not required to take any action (such as pressing a button) besides moving the camera around the position that he or she expects to be the most appropriate for a compliant picture.

Hovering: Guidance

This represents a more interactive version of the “hovering” modality. The system continuously takes and processes pictures, and in addition produces hints (in the form of short synthetic speech sentences) advising the user about where to move the camera next in order to increase the likelihood of reaching a compliant pose.

3.2.5 Apparatus

Pose Estimation

The application developed for this experiment runs on an iPhone 4S (with a 4:3 aspect ratio and video resolution of 640x480). To compute the camera pose from a picture of the printed fiducials, we use the ArUco [12] Augmented Reality library, implemented with OpenCV [21]. A letter-size sheet is printed with ArUco’s fiducial patterns in known locations (see Fig. 3.1). The software detects the location of the fiducials in the camera’s field of view and computes the pose of the camera (previously calibrated off-line). Only a single fiducial is necessary for pose estimation, but accuracy is increased when multiple fiducially are seen. The software is able to process 20 images per second on average, although in practice the effective frame rate is smaller due to other concurrent processing on the phone. Given the camera’s pose (computed with respect to a reference system centered at the paper sheet), one can obtain the homography (perspective transformation [42]) that maps points in the paper sheet to pixels. This information is used to compute compliance of the current pose, based on the criteria discussed above (visibility of all corners of the document’s printable area, minimum resolution). Note that pose compliance detection (along with proper user confirmation) is all that is needed for the *hovering: just confirmation* modality. The *guidance* modality requires further processing and a more complex user interface, as explained below.

Guidance

The goal of the *guidance* mechanism is to give clear instructions as to where to move the camera to reach a compliant pose. This algorithm produces a *correction vector* that takes the camera to a compliant pose if the same orientation is maintained. The correction vector links the current camera position with the closest point in the *compliant segment* (see Fig. 3.2), which is the set of points on a line through the center of the sheet, parallel to the optical axis of the camera, such that each point in the segment is a compliant camera location under the current orientation. The compliant segment for a given camera orientation is defined by two endpoints, $\mathbf{p}_1$ and $\mathbf{p}_2$, where $\mathbf{p}_2$ is higher (with respect to the document) than $\mathbf{p}_1$.

However, if the slant of the camera with respect to the sheet normal is too large (*non–compliant orientation*), the compliant segment for the current camera orientation may contain no points, meaning that, in order to reach a compliant pose, the camera needs to be rotated.

Correction information is communicated to the user through synthetic speech. Synthetic speech capabilities are provided by the Flite [9] library. Each short sentence contains directions along at most two Cartesian axes, and precisely those in need of the largest correction (e.g., "Move up 5 and forward 3" or "Move left 4"). We felt that specifying three vector coordinates (e.g., "Move up 5, forward 3 and left 8") would generate exceedingly long sentences and possibly become confusing. Units are expressed in centimeters, and the reference system is fixed with respect to the paper sheet (not the user). This could create a conflict if the user construes the direction as if in reference to his or her body; however, we noted that most participants kept the paper sheet aligned with their body, reducing the risk of conflicting frames of reference. Note that the camera pose is monitored in real time, and directions are produced continuously (with a minimum gap of 1 second between two sentences).

If a non–compliant orientation is detected, the system utters the sentence "Reset orientation", which prompts the user to re-orient the phone, ideally bringing the phone parallel to the document. Upon detection of a compliant pose, the system utters the sentence "Pose compliant", terminating the trial.

Our strategy for determining the correction vector was inspired by a similar algorithm originally proposed by Cutter and Manduchi [29]. Their algorithm does not consider camera orientation: it always produces a correction vector that would bring the camera to a compliant pose *under the assumption that the sheet is seen front-to-parallel.* With the system used in their study, the heights of $\mathbf{p}_1$ and $\mathbf{p}_2$ are of 28 cm and 42 cm, respectively and centered at the orgin. In practice, this means that the correction vector is potentially incorrect as soon as the iPhone is not held parallel to the sheet (i.e. at non-null *off-axis angles*). As shown in Fig. 3.6, off-axis angles of 10 degrees or more are to be expected, which highlights the need for explicit orientation reasoning as in the new algorithm proposed here. With our algorithm $\mathbf{p}_1$ and $\mathbf{p}_2$ and set dynamically given the current orientation

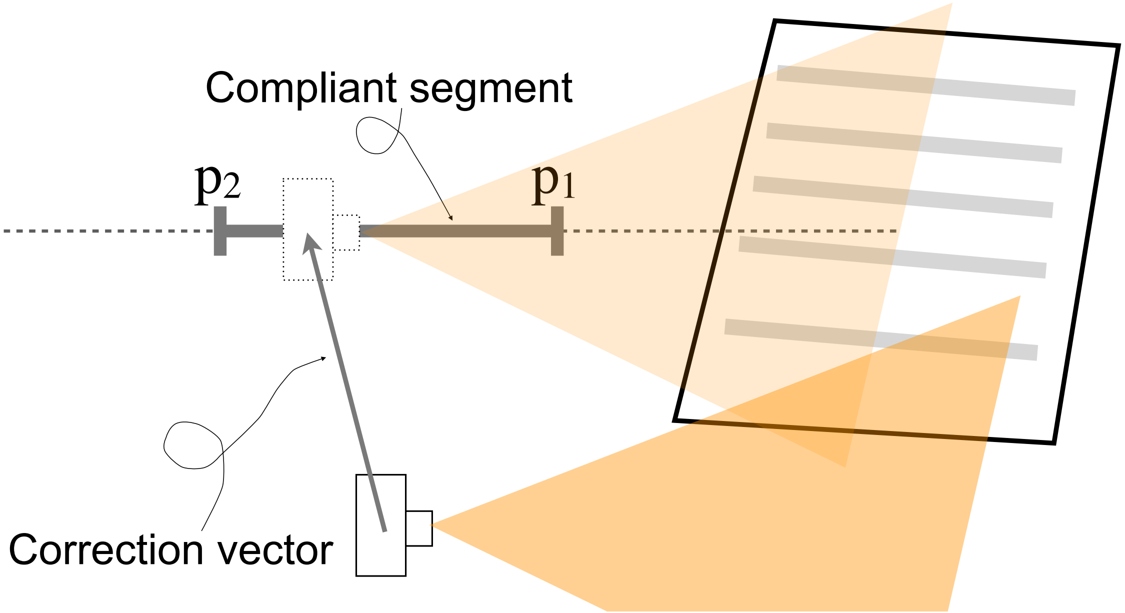


Figure 3.2: A simple guidance example. The current camera pose (shown in solid line) is not compliant, because part of the document is outside of its field of view. If the camera is moved by the correction vector, it will reach a position in the compliant segment. If the orientation is kept constant, any position on the compliant segment is compliant.

and position.

3.2.6 Procedure

After being read and having signed the IRB-approved consent form, each participant were given an introduction to the goals of the experiment and to its procedures. They were informed that, in order to take a "good" (compliant) picture of the document, the camera should be at a height of between approximately one foot and one and a half feet over the document, with the iPhone level (horizontal) and well aligned with the document. Each participant was asked to sit on a chair in front of a small desk, and invited to adjust the height of the chair to ensure that he or she was able to raise his or her iPhone-holding hand comfortably at least 40 cm above the desktop. Participants were informed that they could stand up during the experiment, if they felt that this would increase their comfort, and that they could use either or both hands to hold the phone. Most participants decided to sit for the duration of the experiment, although three participants decided to stand for all or some of the trials. Several of the participants experimented with multiple positions of the phone holding hand throughout the experiment.

After this preliminary phase, each participant performed the experiment, structured as an ordered sequence of sessions: Pre-intervention, Intervention, and Post-intervention. Each session was comprised of 12 identical trials; participants were informed that the first three trials of each session were to be considered practice trials. At the beginning of each trial, the paper sheet was slightly moved and rotated on the desktop, and the iPhone was placed flat (the camera facing downwards) over the document's left corner closest to the participant. In this way the participants frame of reference was reset; each trial simulates a fresh document scanning scenario. Each participant was assigned a Group ID (0 or 1), such that the IDs were evenly distributed across participants.

Pre-Intervention

The goal of each trial was to take a compliant picture of the document using the *snapshot* modality described earlier. The participant was asked to pick up the iPhone, and position it where he or she thought a good picture of the document could be taken. Once they were confident of the position they took a picture by pressing either of the two small volume buttons on the side of the iPhone. Participants were free to re-position the document on the desktop if they wanted to, and could take as much time as they wanted before taking the snapshot.

Several participants found the action of pressing one of the volume buttons difficult to execute, especially when holding the phone with one hand, although others found it very natural. Two participants expressed concern about the possibility that while reaching with a finger for these buttons, the phone may be inadvertently moved, generating blur or resulting in the picture taken from an incorrect location; however, this didn't seem to be the case, and all snapshots taken this way were correctly processed by the system.

Intervention

The goal of these trials was to move the iPhone over the document so as to reach a compliant pose using one of the *hovering* modalities described earlier. Participants in Group 0 used the *hovering: guidance* modality, while participants in Group 1 used the *hovering: just confirmation* modality. The starting procedure at each trial was the same as for the pre-intervention trials. A time-out period T_{to} of 150 seconds was set for each trial: if a compliant pose was not reached within the time-out period, the trial was terminated.

Post-Intervention

This session was identical to the Pre-intervention session. All participants used the *snapshot* modality to try to take compliant pictures of the document. These trials were meant to investigate whether experience with a hovering modality in the Intervention trials could increase the user's awareness of the compliant space, and thus facilitate taking a compliant snapshot of a document without system assistance. At the end of the three sessions, participants were asked to answer a short questionnaire, described in detail in the Results section.

The experiments described in [29] also consider similar interaction modalities to those considered here, albeit under different names. However, the experiment design in [29] and in the study presented here are very different. Participants in the experiments of [29] all underwent the same sequence (Snapshot; Hovering:Just Confirmation; Hovering:Guidance). This design does not allow one to evaluate whether experience with a hovering modality can increase one's skill at taking compliant snapshots without system assistance (which is the reason for the Pre- and Post-Intervention phases of the new design). In addition, the experiment design from [29] did not balance the order of the hovering modalities, resulting in a potentially biased analysis.

3.2.7 Metrics

Accuracy

Each *snapshot* trial can be characterized by a binary variable (*success*) that is equal to 1 if the snapshot resulted in a compliant picture, 0 otherwise. The *success rate* (*SR*) represents the average success value over all trials in a session.

We also derive a "softer" measure of accuracy (*proportion legible*) defined as the number of equivalent 12-point characters in the printable area that are OCR-readable from the image, divided by the total number of characters in the printable area, assuming the the printable area is filled with 12-point characters in an ordered grid. (This grid is designed based on standard inter-character and inter-line spacing.) Note that proportion legible = 1 implies success = 1; the opposite is not true. The proportion legible metric gives an indication of the document area that can be accessed by OCR. Note, however, that this does not translate directly into “readable portion of a document”: if, for example, the right half of a text column is outside of the camera’s field of view, the whole column is not “readable” (even though individual words in the left half can be decoded by OCR). A more useful metric, which we will consider in future work, would take the document structure into account. For each session, we computed the *median proportion legible* over all trials in the session.

Time

For the *hovering* trials, we measure the time from the beginning of the trial until a compliant pose is reached (*time-to-completion,* T_c). If a compliant pose is not reached before the time-out period T_{to}, we simply set $T_c = T_{to}$.

Note that it is not possible to directly compare the results of a *snapshot*–type and of a *hovering*–type trials using success rate or time-to-completion. For snapshot-type trials, the time used to take a snapshot is immaterial; for hovering-type trials, the success rate (that is, the probability of reaching a compliant pose before time-out) is equal to:

$$SR = \mathrm{Prob}(T_c < T_{to}) \tag{3.1}$$

and is thus an increasing function of the chosen time-out period T_{to}. (For $T_{to} \to \infty$, we could assume that a compliant pose will be reached with probability 1 as one simply moves the phone randomly over the document.)

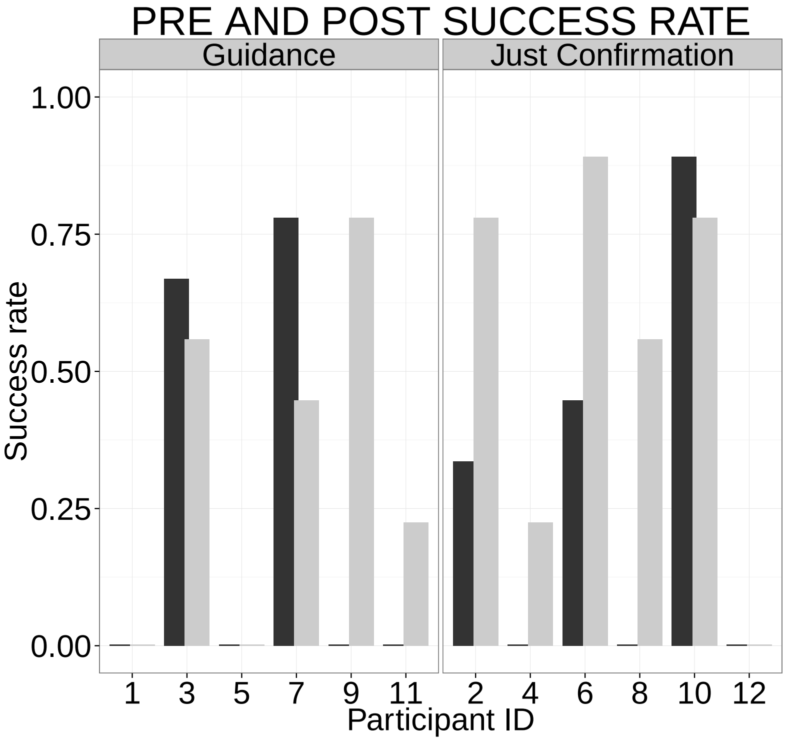


Figure 3.3: Success rate for all participants in the snapshot-type trials. Left: Group 0. Right: Group 1. Black: pre-intervention. Gray: post-intervention.

3.3 Results

3.3.1 Snapshot Modality

General Results

Figs. 3.3 and 3.4 show the results, in terms of success rates and proportion legible, for the pre- and post-intervention trials using the *snapshot* modality. From these plots, it results clear that, while some participants were quite proficient at this task, others had serious difficulties. In particular, seven participants could not take a single compliant picture in the pre-intervention trials; three of them could not take any compliant picture in the post-intervention trials either.

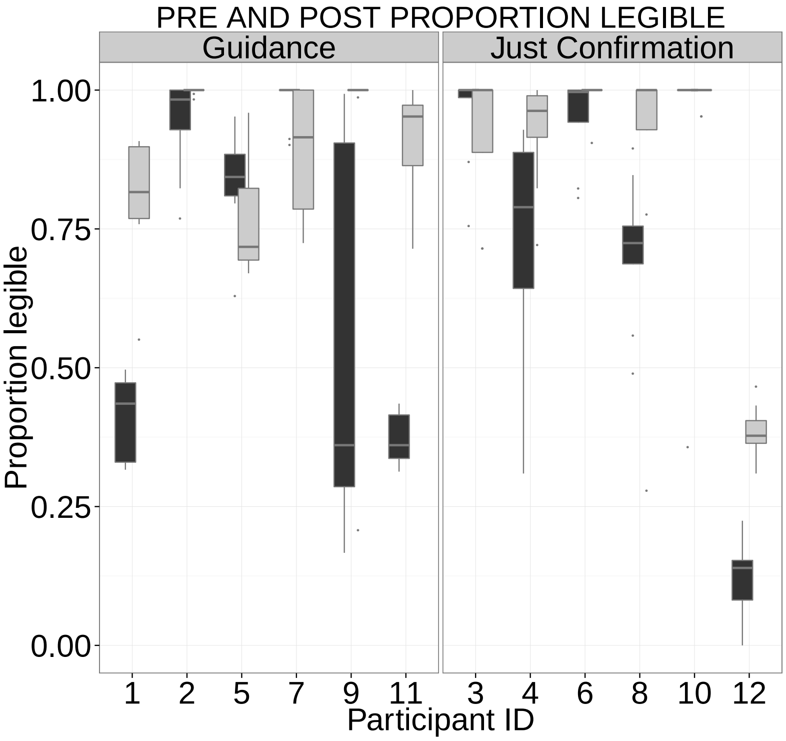


Figure 3.4: Proportion legible for all participants in the snapshot-type trials, shown as box plots. Left: Group 0. Right: Group 1. Black: pre-intervention. Gray: post-intervention.

To investigate the main causes of failure, we need to consider all conditions that can result in a non-compliant pose. The space of poses PS can be divided into four disjoints sets:

PS1: Poses that can be made compliant by simply re-positioning the camera (orientation unchanged) but not by simply re-orienting the camera (position unchanged).

PS2: Poses that can be made compliant by simply re-orienting the camera (position unchanged) but not by simply re-positioning the camera (orientation unchanged).

PS3: Poses that can be made compliant by simply re-orienting the camera or re-positioning the camera.

PS4: Poses that can be made compliant only by re-orienting and re-positioning the camera.

We analyzed the poses of the non-compliant snapshots, in order to obtain proportion of occurrence of the different types of poses above. This is expressed as probabilities (see Tab. 3.1.)

Pr(PS1)	**Pr(PS2)**	**Pr(PS3)**	**Pr(PS4)**
0.35	0.1	0.49	0.06

Table 3.1: The probability distribution of non-compliant poses across the four conditions considered.

This data suggests that in most cases ($\Pr(PS1) + \Pr(PS3) = 0.84$) a simple re-positioning of the camera would have led to a compliant snapshot. In a smaller proportion of cases ($\Pr(PS2) + \Pr(PS3) = 0.59$), a compliant pose would have been reached by simply re-orienting the phone. The more serious situation of a pose requiring both orientation and position adjustment occurs only 6% of the time.

Fig. 3.5 shows the location of the camera at the time of the snapshot for compliant poses (black dots) and non-compliant poses (grey dots). (Remember that locations higher than 42 cm and lower than 28 cm with respect to the document are non-compliant.) The plot suggests that in many cases, non-compliance was due to the participant keeping the phone too close to the document (the difference in height means between compliant and non-compliant poses is significant at $p < 0.001$). Fig. 3.6 shows the histogram of *off-axis angles* (defined as the angle between the camera's optical axis and the normal to the document) at the time of the snapshot. (Note that the off-axis angle, by itself, does not determine compliance: if the camera is located to the side of the document, a moderately

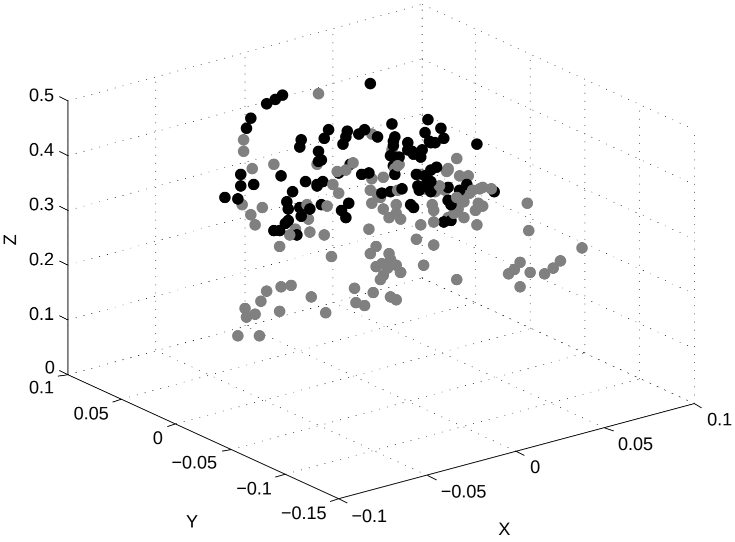


Figure 3.5: 3-D locations of camera pose in the pre- and post-intervention trials, with respect to a reference system centered at the center of the paper sheet (units are in meters). Black: compliant pose. Gray: non-compliant pose.

large off-axis angle may be required for compliance.) This histogram shows that, on average, non-compliant poses were characterized by a larger off-axis angle than compliant poses (the difference in means is significant at $p < 0.001$).

The median time to take a snapshot (over all trials in a pre- or post-intervention session) ranged from 5.6 sec. to 39.3 sec., with a mean of 12.4 sec.

Pre- and Post-Intervention Comparison

We compared the success rate and median proportion legible for pre- and post-intervention sessions using a standard 2 × 2 mixed factorial design model. Note from Fig. 3.3 that among those participants who were able to take compliant pictures in the post-intervention trials, two in Group 0 and four in Group 1 improved their success rate after the intervention session, while two in Group 0 and one in Group 1 worsened their

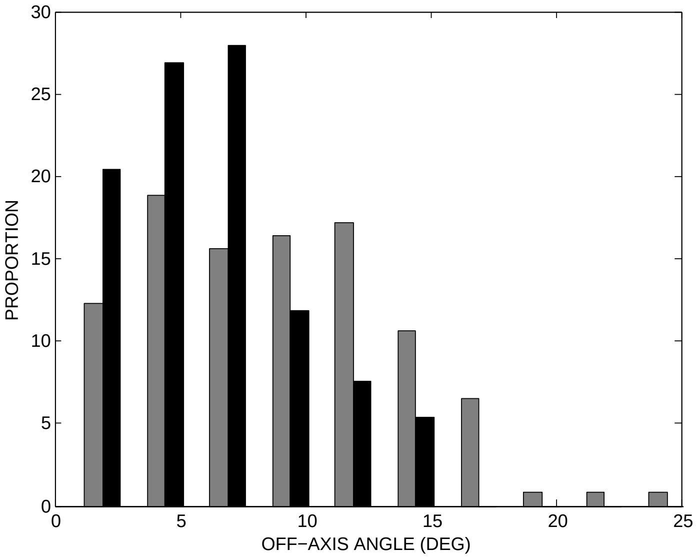


Figure 3.6: Histogram of off-axis angles for compliant (black) and non–compliant (gray) terminal poses in the pre- and post-intervention trials.

performance. The difference in mean success rate between pre- and post-intervention and across groups was not found to be significant at $\alpha = 0.05$. The difference in mean between the pre- and post-treatment median proportion legible is significant at $p = 0.04$ (mean equal to 0.72 for pre-treatment, 0.89 for post-treatment). However, the main effect of intervention type (*guidance* vs. *just confirmation*) was not found to be significant at $\alpha = 0.05$. No significant difference was found between the means of camera height, horizontal offset (distance to the line perpendicular to and centered at the sheet), or off-axis angle at the time snapshots were taken for the pre- and post-intervention trials. However, for the participants that were not able to take a single compliant snapshot in the pre-intervention trials (participants 1,4,5,8,9,11,12; see Fig. 3.3), we noted that the median (across trials) of the horizontal offset decreased from 5.5 cm to 3.7 cm (paired one-sided t-test; $p = 0.03$). This may help explain why all but one of these participants performed better (in terms of

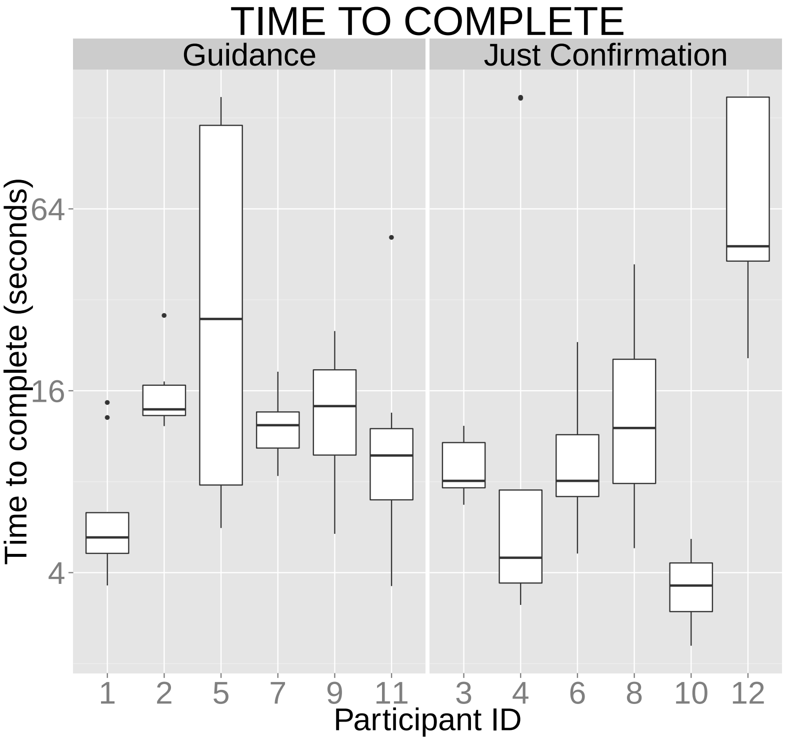


Figure 3.7: Time-to-completion values for all participants shown as a box plot on logarithmic scale. Left: Group 0 (Hovering: Guidance). Right: Group 1 (Hovering: Just Confirmation).

proportion legible) in the post-intervention trials.

3.3.2 Hovering Modalities

Time-to-Completion

Fig. 3.7 shows a box plot of the logarithm of the time-to-completion values for all hovering-type trials (Intervention session). The median (over all trials) time-to-completion ranged from 3.6 sec to 48.1 sec, with an average value of 13.9 sec. We notice that one participant in Group 1 (ID=12) took much longer to complete the hovering trials than the other participants; the mean value of the time-to-completion medians with this participant removed drops to 10.8 sec. Multiple-sample repeated measurements ANOVA analysis did not find a significant difference in the mean time-to-completion between participants in Group 0 (*guidance*) and Group 1 (*just confirmation*).

Equivalent Time-Out Period

In order to compute the equivalent time-out period T_{eto} (defined in the Method section), we used the following algorithm. We first computed the value of the "success" ($S(T_{to})$) binary variable for each hovering-type trial at multiple levels of the time-out period T_{to}. (Note that, although all hovering-type trials were conducted with a fixed time-out period $T_{to_{\max}} = 150$ sec, it is trivial to derive the value of $S(T_{to})$ for any $T_{to} < T_{to_{\max}}$ based on the recorded time-to-completion T_c: $S(T_{to}) = 1$ if $T_c \leq T_{to}$, 0 otherwise.) For each considered value of T_{to}, we tested for difference between the success rate under the *snapshot* modality (pre-intervention) and under the *hovering* modality using a paired one-sided t-test. (Since the difference in mean between the two types of hovering modalities (*guidance* and *just confirmation*) was not found to be significant, we did not differentiate between the two in this analysis.) The equivalent time-out period T_{eto} is equal to the largest value of T_{to} for which the difference between the success rates in the two cases was found to be not statistically significant. Stated differently, for $T_{to} > T_{eto}$, the success rate using the hovering mechanism is significantly higher than the success rate using the snapshot mechanism. Using the standard significance level $\alpha = 0.05$, we obtained T_{eto}=12.1 sec. Note that decreasing the significant level would increase the equivalent time-out-period, but not by a very substantial amount; for example, setting $\alpha = 0.01$ results in a 11% increase of the equivalent time-out period (T_{eto}=13.5 sec).

3.3.3 Participant Surveys

During the experiment, participants were free to try whichever hand positions worked best for them. Several of the participants experimented with multiple positions of the phone holding hand throughout the experiment. Most participants decided to sit for the duration of the experiment, although three participants decided to stand for all or some of the experiment.

As mentioned in Sec. 3.3, during the pre- and post-intervention sessions, the participants were to take snapshots by pressing either volume bottom located at the side of the iPhone. Several participants found this action somewhat difficult to execute, especially if holding the phone with one hand, while others found it very natural. Two participants expressed concern about the possibility that while reaching with a finger for these buttons, the phone may be inadvertently moved, generating blur or resulting in the picture taken from an incorrect location.

Two Group 0 participants lamented the fact that guidance directions were issued in centimeters, a unit they were not accustomed to. Note that we chose centimeters (rather than inches) in order that commands could be issued in whole units with good enough resolution. In future studies, we will offer the user the option to choose the preferred unit.

Group 1 Participant ID 10 strongly disliked the intervention session. This participant was exposed to the confirmation intervention without guidance. This participant median time to complete each intervention trial was 48.13 seconds and was unable to reach a compliant pose within the timeout for three of the nine trials.

Group 0 Participant ID 5 had a substantial decrease between pre- and post-intervention (see Fig. 3.4 and Fig. 3.3. It is interesting to note that this participant demonstrated near perfect ability to capture compliant images in the pre-intervention trials. We should note that, during experiment with this participant, the system crashed in the middle of the intervention session. It took the experimenter six minutes to restore the system, before the experiment could be resumed. Luckily, no log data was lost in the process. This incident may or may not have contributed to the decrease in performance of this participant.

At the end of the experiment, each participant was asked to complete a short survey. Participants were asked to comment on a number of statements using a five-point Likert scale (with 'strongly disagree' represented by '1' and 'strongly agree' represented by '5'). The statements, reported verbatim below along with the median response, differed slightly across the two participant groups.

Questions for Group 0 (*Hovering: Guidance*)	**Median response**
I feel that, after interacting with the system, I am now able to take better pictures of the document by myself.	**4**
It was easy to follow the directions from the system.	**5**
The directions from the system helped me take better pictures of the document.	**4**
If the guidance system were available as an application, I would be interested in using it.	**5**

Questions for Group 1 (*Hovering: Just Confirmation*)	**Median response**
The system helped me take better pictures of the document.	**4**
It was easy to follow the directions from the system.	**5**
If this system were available as an app, I would be interested in using it.	**5**

3.4 Discussion

Participants exhibited a wide diversity of skill taking compliant snapshots without help from the system (Figs. 3.3 and 3.4). By observing the participants during the experiment, it was clear that some were much more "methodical" than others in the way they moved the phone to take a snapshot. Interestingly, as shown by Fig. 3.5, participants tended to take snapshots at a short distance from the document: the maximum recorded height of a snapshot was 44 cm, which is slightly above the maximum compliant height (42 cm). As mentioned earlier, participants were informed that the correct height was approximately between one foot and one and a half feet, but it seems that they preferred to err on the lower end. Of course, since no feedback was provided in the pre-intervention session, participants did not have a means to correct what could be a biased perception of the camera height. However, this tendency did not change even after the Intervention phase, in which participants had a chance to experiment first-hand the range of compliant heights.

Can the proprioception skills that are necessary to correctly position a camera be taught? We note that during the trials performed as part of the pre and post-test, we observed no trend of improvement between the first and the second half of the trials. This makes sense since there is no feedback during the snapshot trials. However, for many participants we observed improvement between the the pre and post-test. In addition, our quantitative results with the experimental system, along with the outcomes from the participant surveys, supports this observation. However, these results do not provide a clear indication of what exactly was learned through the Intervention phase.

As mentioned above, participants in the post-test trials continued to take snapshots from a relatively low height, something that undoubtedly contributed to a fair portion of failures. However, anecdotally a participant in the guidance group said after several trials of the intervention “ahah now i’ve got it”. Similar “aha” moments occurred for other participants during the intervention; at which point the subsequent intervention trials were quickly completed.

We were surprised by the discovery that both the *guidance* and the *just confirmation* intervention modalities produce comparable results. We carefully designed a complex guidance modality, and expected that it would help the user reach a compliant pose faster. This expectation was supported by preliminary results using a similar system with sighted blindfold participants presented in [29]. Although as discussed earlier, the experimental design and the chosen metrics in [29] may have been inappropriate for this type of analysis.

Why is it, then, that the guidance modality, with its rich system feedback, did not prove superior to the just confirmation modality in terms of time-to-completion in the present study? We believe that the reason for this lies in the sub-optimal design of the user interface used in these prior experiments. Upon careful analysis of the videos collected during the trials, we determined two main pitfalls of the current design:

Lack of explicit orientation guidance. As shown in Fig. 3.6, non-OCR-complaint images are often associated with excessive off-axis angles. Our original guidance system gave directions in terms of translation but not of orientation; this was a deliberate choice in order to keep the complexity of directions low. Participants were advised to keep the iPhone horizontal; only upon detection of a large off-axis angle was a synthetic speech warning produced. However, most participants found it difficult to re-orient the phone correctly (horizontally), resulting in the off-axis warning being re-issued several times before the orientation of the iPhone was properly adjusted. When this happened, the whole process was slowed down, which generated frustration among some participants. We now believe that some form of orientation correction guidance would be very beneficial. Indeed, as discussed earlier, in 59% of the non-compliant snapshot cases, a simple camera re-orientation would have been sufficient to make the pose compliant, and in 6% of the cases this correction would in fact have been necessary.

Disruptive guidance modality. The synthetic speech directions produced by the system contained precise metric indication of where to move the phone next. Ideally, the user would move the phone exactly as directed, ending at a compliant pose. In fact, this was rarely the case, due to the difficulty of moving the phone precisely as directed. This resulted in participants following a discrete sequence of movements; after each movement, they would pause and wait for the system to produce the next direction. In contrast, participants in the group that did not use the guidance system moved the phone in continuous motion; this allowed for a larger portion of space to be explored in the same amount of time. The difference in behavior for the two hovering modalities can be noticed in Fig. 3.8. The path marked in blue (*hovering:guidance*) is characterized by non-uniform velocity and several

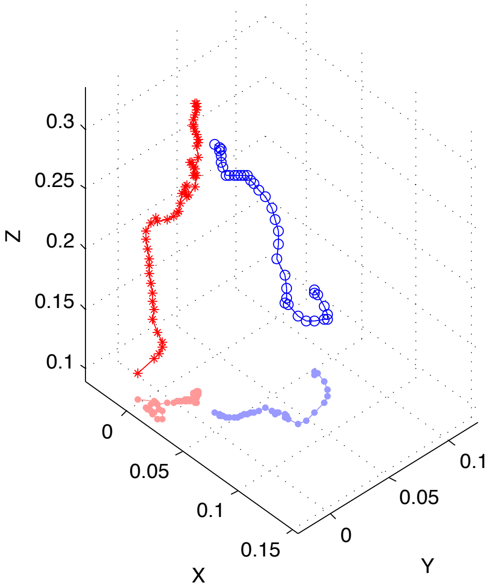


Figure 3.8: The paths represent camera locations during two trials, using the hovering:just confirmation modality (red) and the hovering:guidance modality (blue). Units are in meters. The projection of the paths on the horizontal plane are shown with faded color. Circular blue marks and red asterisks are placed at constant time periods of 0.1 s. Only the portion of the path after a certain time lag is shown as measurements cannot be taken when the camera is too close to the document. This lag was of 6.8 s for the path marked in red and of 3.9 s for the path marked in blue.

abrupt turns in response to a direction, whereas the path marked in red (*hovering:just confirmation*) shows a more uniform motion. In future work we will explore different types of acoustic interface that require less information processing by the user and encourage smooth trajectories.

3.5 Conclusions

We have presented an experimental study that investigated modalities to help a blind person take better pictures of a document faster through the use of image processing software. The overarching goal of this project is to facilitate the use of mobile OCR for printed text access.

The proposed mechanisms have been implemented using special printed fiducials, and could not be used directly with regular printed documents. This investigation explores the "best case scenario" of a perfectly functioning device; similar functionalities on regular printed documents are not out of reach.

Camera orientation can be computed from the device accelerometers and by measuring orientation of detected parallel text lines. By detecting the endpoints of text lines, one can make inferences about whether the text is fully visible (e.g. a line ending at the edge of the image is likely truncated) or, if not, where the camera should be moved for better visibility. Readability of characters can be computed by a fast text spotter (e.g. if characters in a line cannot be spotted, the camera is too far). Localization features could be approximately inferred by computer vision algorithms with heuristics about the visual structure of typical documents. These vision-based algorithms can obtain functionalities similar (albeit less accurate) to using fiducials with real-world documents.

3.6 Acknowledgments

The project described was supported by grant number
****** from NEI/NIH. The authors thank all of the participants for their feedback and effort.

3.7 Transition from user study one to technology and the second user study

The results from our first user study with blind participants left many questions unanswered. Although we showed that the proportion the document legible imaged increased intervention, it is unclear why we found no difference between the two modalities. We made the oversight of not properly balancing our experiment design in the initial user study. In practice this means that from our limited pool of participants we only gave half of them each intervention. We realized that in our next experiment we would expose all the participants to every type of intervention and simply just vary the order. In this way we hope to track down differences between the two modalities. However, we learned a lot about the type of problems that prevent blind people from capturing compliant document images. We carefully studied the orientations and positions blind people typically capture images from. Despite the conflicting results, we found promising data on how blind participants interacted with our system. We decided to move forward with our research and design a new system that works will real documents and conduct a new user study with better experimental control to increase our statistical power.

In order for such a system to work with real documents several problems that were unnecessary to solve with fiducials became necessary to solve immediatly. From viewing a single fiducial pattern with our previous system, we can immediately compute the position and orientation of the phone. However, for a computer vision system to work from partial views of real documents, oriented text line detection is critical. With text lines detected, we can reason about if the text lines have sufficient resolution and are not cropped. In fact we believe we can use the text line information to formulate a guidance heuristic. Further, without fiducial patterns we do not know with certainty if the entire document is within the FOV. Therefore, we must develop a less formal compliance criteria that works from detected text lines over a sequence of frames. Additionally, the video feed is insufficient resolution read small font documents, and therefore we require asynchronous control of the full resolution sensor. However, its imperative that we don't simply capture an image without notifying the user first or it will likely contain severe motion blur. We will discuss in the next chapter how we designed solutions to these technical problems.

Chapter 4

Real-time acquisition guidance on real printed text documents

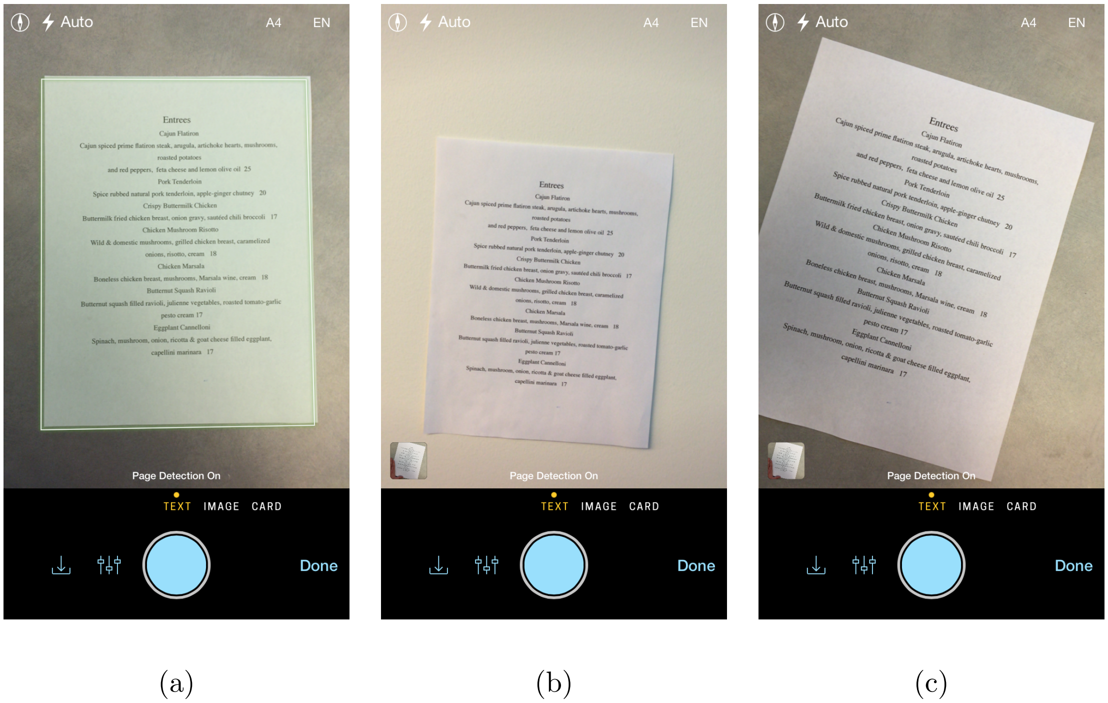
Abstract

Increasing printed document accessibility with guided image acquisition

by

Michael P. Cutter

The advent of mobile OCR (optical character recognition) apps on regular smartphones holds great promise for enabling blind persons to access printed information. Unfortunately, in order for OCR to work, a good, well-framed image of the document needs to be taken, something that is quite difficult to do without sight. We describe a computer vision-based iOS application that allows a blind person to take an OCR-readable high-resolution picture of a document quickly by following real-time spoken directions from the smartphone. Our algorithm analyzes medium-resolution images continuously, producing positioning guidance, and takes a high-resolution snapshot as soon as an OCR-readable image is detected. Experimental results with eight blind participants show that acquiring an OCR-readable image using our system takes about 15 seconds on average. This is three times as fast as with a simpler system that does not produce real-time guidance. Our participants expressed positive comments on the system, and generally felt that their proficiency at taking OCR-readable images had increased by interacting with the system.



Entrees

Cajun Flatiron

Cajun spiced prime flatiron steak, arugula, artichoke hearts, mushrooms,
roasted potatoes
and red peppers, feta cheese and lemon olive oil 25

Pork Tenderloin

Spice rubbed natural pork tenderloin, apple-ginger chutney 20

Crispy Buttermilk Chicken

Buttermilk fried chicken breast, onion gravy, sautéed chili broccoli 17

Chicken Mushroom Risotto

Wild & domestic mushrooms, grilled chicken breast, caramelized
onions, risotto, cream 18

Chicken Marsala

Boneless chicken breast, mushrooms, Marsala wine, cream 18

Butternut Squash Ravioli

Butternut squash filled ravioli, julienne vegetables, roasted tomato-garlic
pesto cream 17

Eggplant Cannelloni

Spinach, mushroom, onion, ricotta & goat cheese filled eggplant,
capellini marinara 17

(a) (b) (c)

Figure 4.1: Requiring all four corners of the document to be visible can be a good strategy in many cases (a), but fails in case the edges are not visible (e.g., white paper on white background; (b)) and can be too restrictive (an image could be OCR readable even if not all corners are visible; (c).) Screenshots from the Prizmo app.

In this chapter, we describe an iPhone app that facilitates acquisition of an OCR-readable image of a document. By "OCR-readable" we mean that all text content in the document is visible in the image, and that the text is imaged at high enough resolution that it can be decoded by a good OCR software. Our system acquires images continuously from the iPhone's camera at medium (VGA) resolution, and processes each image to assess whether the image is *compliant*, that is, if the text image has enough resolution for OCR, and that all text is surrounded by enough white padding. We use a state-of-the-art text spotting algorithm that can quickly analyze an image to detect the presence of text, and an innovative text line aggregation algorithm that allows for identification of individual text lines at any orientation. When the images in a short sequence are all deemed to be compliant, the flash is activated and a high resolution image is snapshot for subsequent OCR processing. In addition, our system produces directions (in the form of synthetic speech) that can help the user move the phone to a position from which a compliant image can be taken.

We tested the system with eight blind volunteers, who operated it under different modalities: (1) taking a picture without any system guidance (except for an acoustic warning when the phone was not kept level); (2) using our guidance system; (3) using a simpler version of the system, which would still take a high resolution picture upon detection of a compliant image, but without giving guidance. The results show that the guidance modality allows for acquisition of an almost perfectly OCR-readable image three times faster (on average) than the equivalent modality without guidance. On average, it took 15 seconds for our participants to take an OCR-readable image with the guidance modality. Our tests also show that if users need to decide by themselves when to take a picture, the chance that the resulting image is OCR-readable is quite low. Interestingly (but not surprisingly), after interacting with out system, users were noticeably more proficients at taking OCR-readable images of the document. Pre- and post-test interviews investigated the previous experience (if any) of our participants with existing mobile OCR apps, and gathered feedback about our experimental system.

4.1 Related Work

4.1.1 Text Detection in Natural Images

The Computer Vision community has witnessed burgeoning interest in *text spotting* techniques. Unlike OCR, text-spotters do not (usually) decode text; rather, they are specialized in the fast detection (and localization) of any text content in the image. A text spotting and localization competition is hosted by the International Conference of Document and Analysis and Recognition (ICDAR); a good collection of state of the art algorithms is described in [48]. The text spotter utilized in this contribution is based on the popular Stroke Width Transform (SWT) algorithm by Epshtein et al. [35].

We should note that the ICDAR robust reading competition (and thus most algorithms developed to succeed in this competition) can only deal with horizontally aligned text. In the mobile OCR application considered in this paper, it is imperative to accurately localized *oriented* text, as a blind person may be unable to correctly align the camera with the text lines. Oriented text localization in natural images has been recently addressed by several authors [74, 51, 71]. In particular, Nassu et. al. [71] proposed a technique to detect text lines on document images using sift [55] keypoints. However, this algorithm is computationally expensive and not amenable to real time implementation on a smartphone.

4.2 Technical Development

We designed an iPhone app that provides two layered modalities. In the first modality (*confirmation*), medium resolution (VGA: 640 by 480 pixels) images are continuously acquired by the iPhone's camera and quickly analyzed to see if they satisfy a specific *compliance* criterion, precisely defined in Sec. 4.2.2. Loosely speaking, an image is compliant if all the text in the document is visible and at good enough resolution to be OCR-readable. As soon as a short sequence of compliant images are detected, the user is prompted to keep the camera still while the system takes a high-resolution image with the flash activated. This image is then passed on to OCR. This modality is similar to the way other apps (Prizmo, TextDetective, kNFB) work – except for the criterion used for compliance definition. The user is expected to aim the camera at the document and slowly move the phone around until the system decides that a good picture can be taken.

The second modality (*guidance*) builds on the confirmation modality, but in addition it provides directions to the user about where to move the camera. The hope is that, by following these directions, the user will be able to quickly reach a position from which a compliant image can be taken.

At the core of both modalities is our algorithm to verify whether an image is compliant. This algorithm comprises a text spotting phase followed by an oriented line grouping algorithm (Sec. 4.2.1). Based on the thus formed text lines, we introduce a compliance criterion that is based the expected size of the characters in the image as well as the white padding around the text area (Sec. 4.2.2). Our procedure to produce meaningful guidance directions is described in Sec. 4.2.5.

4.2.1 Text Spotting and Line Grouping

Connected component segmentation

The first step of the algorithm is the detection of connected components from the Stoke Width Transform (SWT). SWT is an algorithm for the detection of text strokes, based on the observation that text strokes have approximately constant width and tend to form a connected graph within each character. Following the original SWT algorithm [35], we first compute an edge map using Canny [25]. We then cast a ray from each edge pixel in the direction of the local gradient. A ray is accepted if it intersects another edge point with the opposite gradient direction (within a tolerance of $\pm 30°$). Intuitively, accepted rays are those with a good likelihood to section a character stroke. The length of each accepted ray is measured and recorded at each pixel intersected by the ray, resulting in a *stroke width map*. A graph is formed on this map, where two pixels are connected by an edge if they are neighbors in the pixel grid, and if the larger recorded stroke width of the two is less than three times the smaller one.

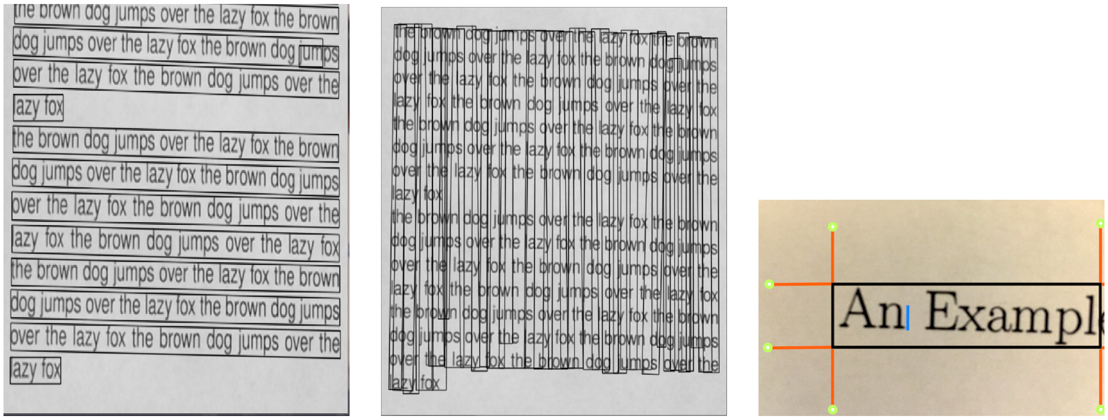
Letter classification

In the original SWT algorithm, connected components are classified as text characters based on certain geometric properties (aspect ratio, height, stroke width variance, and the number of encapsulated connected components). A simple classifier is designed by defining thresholds for these parameters; this classifier is trained on the ICDAR training dataset. In order to increase classification robustness, we considered more features: Euler number (number of holes), perimeter to area ratio, number of horizontal crossings, stroke width variance, and stroke width over height of the connected component (the first three inspired by the work of Nuemann et al. [61]). We use a Random Forest classifier [23] on this features with 100 trees and max depth of five. The training data for this classifier has positive samples selected from the connected components that overlap the ground truth regions, and negative samples mined from the wrong predictions using the original SWT algorithm (in other words: a negative sample is a connected component that was incorrectly classified as "character" by SWT). Similar to [61] we tuned the classifier to favor recall over precision: a connected component is classified as a character if this is the prediction of at least 25 of the trees. In the following, we will use the term "character" to define a connected component that has been classified as such by our algorithm.

Generating Candidate Document Orientations

The next step is to determine text lines. We assume that text lines are mutually parallel in the image, even though this is not strictly true even when the text lines are parallel in the document (due to perspective deformation). Our strategy is to first identify a number of candidate line orientations, which are then validated by associating characters to lines and checking for consistency.

We start from the edge map, which was computed earlier as part of SWT analysis. (In order to ensure real-time processing, the edge map is sub-sampled by 8 in each direction.) We then search for dominant lines using the Hough Transform [34]. Only lines that have length larger than one eight of the the longest side of the input image are kept. We use k-means to cluster the set of line orientations, where for each line we add an orthogonal orientation to the set. This was inspired by the observation that, when applied to latin


the brown dog jumps over the lazy fox the brown
dog jumps over the lazy fox the brown dog jumps
over the lazy fox the brown dog jumps over the
lazy fox
the brown dog jumps over the lazy fox the brown
dog jumps over the lazy fox the brown dog jumps
over the lazy fox the brown dog jumps over the
lazy fox the brown dog jumps over the lazy fox
the brown dog jumps over the lazy fox the brown
dog jumps over the lazy fox the brown dog jumps
over the lazy fox the brown dog jumps over the
lazy fox
An Exampl

(a) (b) (c)

Figure 4.2: (a-b) Two possible text line hypotheses after initial rectifying rotation. Notice that the horizontal hypothesis has no intersecting text lines while the vertical hypothesis has many. (c) The black rectangle represents a text line bounding box. The orange segments (with length equal to three times the estimated x-height, shown by the blue segment) are used to test the white padding condition. Note that the left side of the text line satisfy the white padding condition, but not the right side.

script document images, the Hough Transform predominately hallucinates lines that are either aligned or orthogonal to the actual document text lines (see Fig. 4.2 (a-b)). We consider k=5 clusters; however, if two resulting orientations are within 2 degrees of each other, we only keep one of the two. The resulting set of candidate orientation is denoted by Θ.

Selecting the Best Text Line Orientation

Before assigning characters to text lines, we compute the median x-height across character. (Note: in typography, *x-height* commonly refers to the distance between baseline and mean line of lower-case letters in a typeface.) Specifically, our estimate of the x-height is given by the median value of the set formed by the lengths of the smaller side of the bounding boxes of all characters. For each candidate orientation θ in Θ, we group characters into text lines after first rotating the image around its center by angle θ. Intuitively, if this is the correct text line orientation, we expect all characters in a line to share the same y-coordinate; the distribution of y-coordinates of all characters in the image should have multiple modes, one per line. Based on this observation, we compute the histogram of the y-coordinate of the centroids of all characters and select the largest peak; this should correspond to the most densely populated horizontal line. All characters whose centroid has y-coordinate that differs from the location of the histogram peak by no more than the x-height are associated to this line and removed from the corpus of characters. Then the histogram is computed again on the remaining characters, iterating until no more than two characters can be assigned to the horizontal line defined by the histogram peak. For each line, we compute the minimum bounding box containing all of its characters, with sides pairwise parallel to sides of the (rotated) image.

At this point, we have a set of text lines (denoted by $T(\theta)$) for each orientation, along with the lines' bounding boxes. In order to select the "correct" orientation θ (and associated text lines), we compute a metric that is motivated by two observations: (1) the correct orientation should create lines that contain most of the detected characters; and (2) the bounding boxes of the lines should be well separated (see Fig. 4.2). We translate these observation into an empirical metric that is the linear combination of two terms: (1) the proportion of characters that are associated with a text line, and (2) the inverse of the proportion of text lines that overlap with at least another text line. The orientation that produces the highest value for this metric is selected for further processing.

4.2.2 Compliance Assessment

As mentioned earlier, we define a document image to be *compliant* if it has enough resolution to be OCR readable and if all text in the document is visible. In general, OCR can be assumed to work well when the height of the smallest characters is of 12 pixels [76]; however, based on our experiments, we take a more conservative approach, and require the x-height to be of at least 18 pixels. Note that we run OCR on a high resolution image (3264×2448 pixels), whereas compliance is a computed on a lower resolution images (approximately five time less resolution). Hence, we declare that the image is OCR readable if (in the lower resolution version), the median x-height is of at least 4 pixels.

The second criterion (all text in document visible in the image) is more difficult to quantify. We make the simplifying assumption that the document has substantial white padding around the text. Specifically, we require that there by white padding of width equal to at least three times the median x-height (see Fig. 4.2 (c)). This is a large enough padding to avoid that lines be cropped or that lines in a paragraph are skipped (since the distance between two text lines is usually smaller than the required padding), yet small enough to accommodate for documents with text printed close to the edges. In practice, we check for white padding violation by extending each segment of the rectangular bounding box of each text line by 4 times the median x-height (see Fig. 4.2 (c)), verifying that the new endpoints are contained in the document. If any text line does not have the required padding, the image is not compliant.

Of course, it is possible that consecutive paragraphs in the text are separated by a space wider than our minimum required padding; in this case, our system may trigger an image capture of an individual paragraph, rather than of the whole document. In order to reduce the risk of false detections due to errors in the line grouping phase, we require that at least seven successive frames meet the white padding and x-height constraint before triggering a high resolution image capture process. The end-to-end algorithm runs at about 15 frames per second on an iPhone 6.

4.2.3 High Resolution Capture Process

Good OCR is obtained only if the image is sharp and has little noise. In order to reduce the risk of motion-induced blur, we activate the flash, which reduces exposure time (and thus motion blur) and increases the signal-to-noise ratio of the resulting image. In addition, we alert the user, via acoustic interface, that the image is about to be taken, prompting him or her to keep the camera steady in the process. Specifically, once the capture process is triggered as explained in the previous section, the phone creates a short melody that lasts for one second, after which it utters the word "Wait", activates the flash, and takes the snapshot. While the system is producing the melody, data from the phone's accelerometer is analyzed. If acceleration with magnitude larger than 0.05 g is measured (meaning that the phone is moving, possibly resulting in a blurry picture), the image acquisition process is aborted.

4.2.4 Roll/Pitch Correction Warning

In order to ensure good image quality (and avoid perspective distortion consequent to large slant) our system requires that the phone be kept as horizontal as possible. The roll and pitch angle from the phone's accelerometer are measured at all times; if either of these angles is larger than 7°, the phone produces warning sound. Specifically, the sound has different pitch depending on whether the roll or the pitch threshold is exceeded. By hearing the pitch of the warning sound, the user may figure out in which direction to rotate the phone to make it level. If both angles exceed the threshold, both sounds are produced. If the roll/pitch condition is violated during the high resolution capture process, the process is aborted.

4.2.5 Guidance Generation

If the *guidance* modality is enabled, the phone produces instructions (in the form of synthetic speech) guiding the user to move the phone to a position from which a compliant image of the document can be captured. Remember that compliance is defined in terms of median x-height and on the white padding measured around text lines. Intuitively, if the x-height criterion is violated (median x-height less than 4 pixels), the image has too low resolution, and the user needs to move the camera closer to the document. If the white padding condition is violated on, say, the left side of the image, the camera needs to be moved to the left. If the camera is too close to the document, likely resulting in several sides of the document with no white padding, the camera should be raised. Based on this observations, our guidance algorithm works as follows:

- If the median x-height criterion is violated and the white padding condition is satisfied, the system utters the word “Lower”.

- If padding violation is detected in two non-adjacent sides or in more than two sides of the image, the system utters the word “Raise”.

- If the x-height criterion is satisfied but padding violation is detected in one (e.g., left) or two adjacent (e.g., left and bottom) sides, an horizontal positional command is issued (in our example, “Left” or “Backward Left”).

- If the x-height criterion is violated *and* padding violation is detected in one (e.g., left) or two adjacent (e.g., left and bottom) sides, the system issues one of two commands: either “Lower”, or the proper horizontal positional command issued (in our example, “Left” or “Backward Left”). The command is selected at random between the two, with a larger probability (0.8) assigned to selection of the “Lower” command.

4.3 Experimental Design

4.3.1 Participants

We recruited eight participants for this study (three female and five male). All participants were blind, except for at most some residual light perception. Their ages ranged from 25 to 67, with a median age of 60.5. Four of the eight participants had previous experience with mobile OCR apps. All participants owned a smart phone, although one of the participants only used her iPhone to make phone calls. Two participants (P1 and P7) also had a hearing impairment, but were able to hear instructions from the phone.

4.3.2 Interaction Modalities

We considered three modalities for taking a high resolution image of a document. In the first modality (*snapshot*), participants were asked to move the camera to a position where they thought a good image of the document could be produced, and then to take a

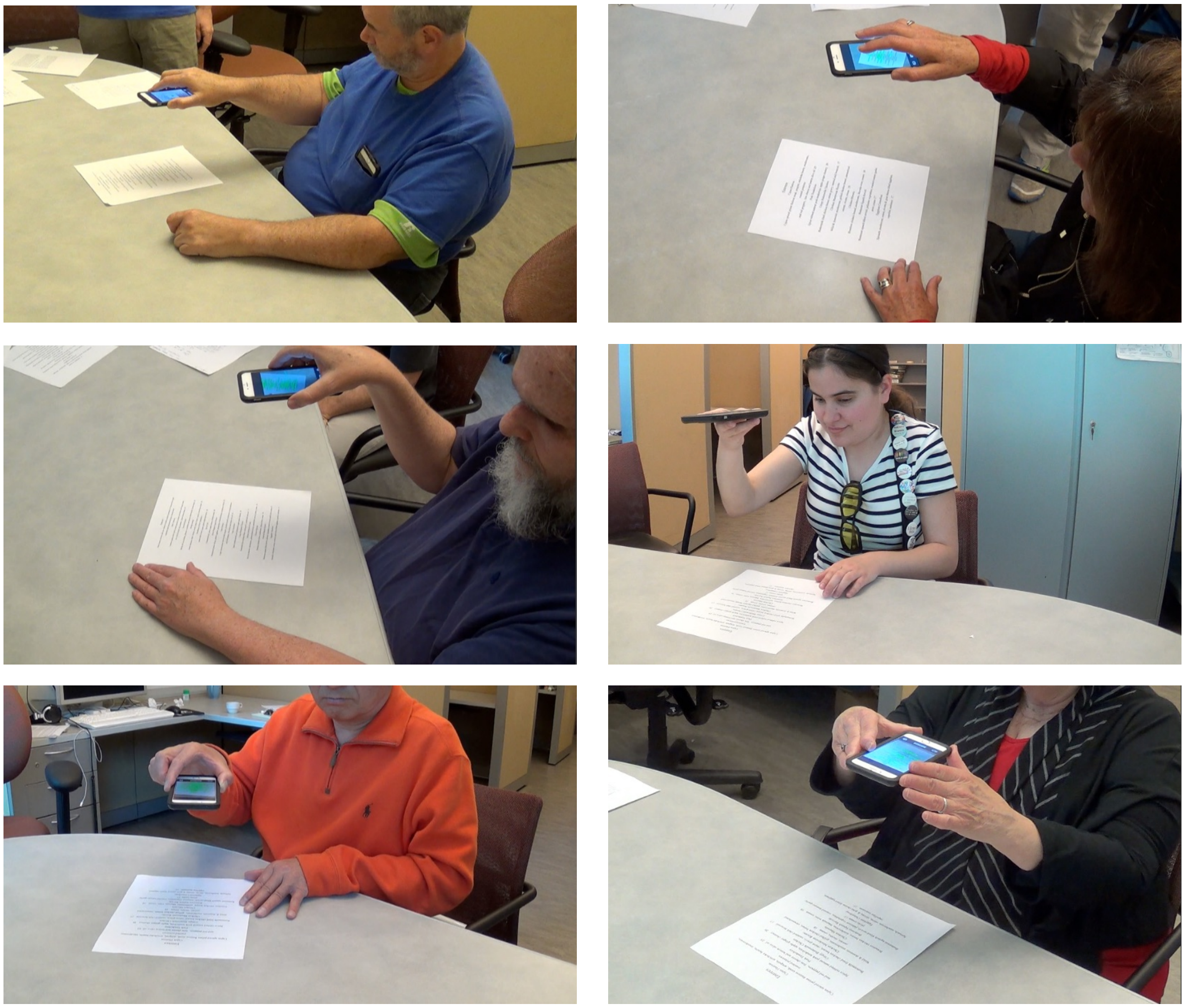


Figure 4.3: Six of the study participants.

snapshot of the document by pressing either of the two volume buttons placed on the side of the iPhone 6. The only feedback produced by the system was the pan/tilt correction warning. Once a volume button was pressed, the high resolution image acquisition process described in Sec. 4.2.3 was started. In the second modality (*confirmation*), the participants moved the phone until a compliant images was detected (Sec. 4.2.2), triggering the high resolution image acquisition process. The third modality (*guidance*) was identical to the confirmation modality, except that guidance instructions were issued by the system as explained in Sec. 4.2.5.

4.3.3 Protocol

Participants were equally divided in two groups (Group 1 and Group 2). Each participant performed a sequence of trials. At the beginning of each trial, a printed document was placed on a desktop, and the iPhone 6 running our app was placed flat on top of the bottom right corner of the document, its camera facing down. The participant was then asked to pick up the phone and take a good snapshot of the document, using one of the three modalities discussed above. More specifically, the participants underwent an ordered sequence of trials organized in four batches as follows: (1) ten trials (*pre-test*) with the snapshot modality; (2) ten trials with either the guidance (Group 1) or the confirmation (Group 2) modality; (3) ten trials with the either the guidance (Group 2) or the confirmation (Group 1) modality; (4) ten trials (*post-test*) with the snapshot modality. After the clearly audible sound generated by the camera when a snapshot was taken, or in case the trial lasted for longer than a time-out period of 180 seconds, the user was asked to place the phone back onto the desktop. Time was measured with a stopwatch, which was started as soon as the participant raised the phone from the desktop, and stopped when the snapshot was taken or after the time-out period.

Two different documents were used: one ("small font") printed in 10 points font with 1.5 inch margins, and one ("large font") printed in 16 points font with 1 inch margins. Both documents, containing a restaurant menu, were printed with black ink on a letter-size ($8\frac{1}{2}$" × 11") paper sheet. In each batch of ten trials, the first five trials were conducted with one document, while the remaining trials were conducted with the other one. The order of documents was randomized for each batch of ten trials.

Participants P1–P3 and P6–P8 tested the system in a lab room at our Engineering building, while tests with participants P4–P5 where conducted in a different building, which was closer to their location and thus easier to reach by public transportation. After signing the IRB-approved consent form, participants were interviewed, with questions ranging over their experience with smartphones, the apps that they use more frequently, their experience (if any) with and the perceived utility of mobile OCR apps. After that, they were read a detailed description of the functionality of the system in its various modalities. Before the start of each batch of trials, users were encouraged to test the system with the modality considered for that batch. Participants were informed that a good image could be taken if the phone was kept at a distance of about 1 foot from the document.

At the end of all trials, participants were asked to respond to the following statements on a 5-point Likert scale from SD (strongly disagree) to SA (strongly agree):

- Q1: I believe the pictures I took without feedback at the start of the experiment were completely readable.

- Q2: I believe the pictures I took without feedback at the end of the experiment were completely readable.

- Q3: The directions from the system helped me take better pictures of the document.

In addition, participants were asked to qualify the ease of pre/post-test trial batches, of trial batches with the confirmation modality, and of trial batches with the guidance batches, on a scale from 1 (very difficult) to 5 (very easy). They were also asked whether they would purchase an OCR app implementing the confirmation modality or the guidance modality if this app costed $10. Finally, we asked them to give us general feedback about the system they tested.

4.3.4 Metrics

For each trial, we took and recorded two measurements: (1) time to complete the trial, as described above; (2) OCR score. The OCR score was obtained by processing the high resolution image acquired in the trial, and feeding it to a professional OCR application ABBYY. Specifically, the OCR score is the ratio of the number of characters that have been correctly recognized, divided by the number of characters in the whole document.

4.4 Results

4.4.1 Quantitative Results

We looked for statistical significance of the modality (confirmation vs guidance) on the time to completion using a 2 by 2 by 5 repeated measures model, with modality and document type (small font, large font) as within-subject factors, and participant group (1 or 2) as between-subject factor. ANOVA resulted in time to completion for the guidance modality ($\bar{x} = 15.1, SE = 3.37$) being significantly smaller ($p = 0.034$, $F = 7.522$) than for the confirmation modality ($\bar{x} = 47.358, SE = 11.96$). The observed power, β, is the probability of correctly accepting the alternative hypothesis. In the case of time to complete the β equals 0.925 so we are confident that guidance does in fact decrease time to complete the task of taking a compliant picture. The whole set of times to completion for these two modalities is showing in Fig. 4.4.1. Note that the average time to completion using the guidance modality is approximately three times smaller than the average time to completion in the confirmation modality.

We also looked at the OCR scores, to verify whether confirmation or guidance resulted in good scores (as expected), and whether our participants were able to achieve similar scores without help from the system. ANOVA on a 2 by 4 by 5 repeated measures model with modality (four levels: pre-test, confirmation, guidance, post-test) and document type as within-subject factors, and participant group as between-subject factor, showed that modality was indeed significant ($p < 0.001$, $F = 11.475$). These are the marginal statistics of OCR scores for the different modalities: pre-test: $\bar{x} = 0.69, SE = 00.08$; confirmation: $\bar{x} = 0.99, SE = 0.00$; guidance: $\bar{x} = 0.97, SE = 0.02$; post-test: $\bar{x} = 0.95, SE = 0.02$. The whole set of OCR scores for the pre- and post-test matches is shown in Fig. 4.4.1. It should be noted all images captured with the confirmation and guidance modality resulted in OCR scores in excess of 0.99, except for one image, that obtained a score of 0.01. This was due to the fact that a thumb of the participant was visible next to the text, and this led to a massive failure of the OCR engine. If this data point is removed, the average OCR score of guidance trails becomes 0.99. Note that the post-test trials resulted in OCR scores that were significantly larger ($p = 0.046$,$F = 5.853$) than pre-test scores (but still substantially worse than guidance scores). For OCR accuracy during the pre and post test we observed β equal to 0.631. This means that a larger sample size is necessary to conclude with certainty that we are accurately rejecting the null hypothesis. However, It would appear that the participants remembered the correct range of positions of the cameras after interacting with our system.

The response (on a Likert scale) to our self-assessment questions described in Sec. 4.3.3 are shown in Tab. 4.4.1 along with their median. All participants but P8 thought that the system helped them take better pictures; two participants (P3 and P7) thought that after using the system, their skill at taking good pictures of the document had increased. In terms of ease of use (Tab. 4.4.1), all participants except for P2 and P8 found the guidance modality easier than the confirmation modality. All participants except for P2 and P3 found the guidance modality to be easier than taking a picture without help from the system. Asked if they would purchase an OCR app that implemented the guidance system at a cost of $10, all participants said that they would; however, if the app only implemented the confirmation system, only seven participants said they would purchase it at that price.

4.4.2 Qualitative Observations

Pre-Trial Interviews

To complement and explain the quantitative results, we ran debriefing interviews to gain more insight into participants' strategies and practices in accessing printed text as

	P1	*P2*	*P3*	*P4*	*P5*	*P6*	*P7*	*P8*	*Med.*
Q1	A	A	SD	N	A	SD	SD	SD	**D**
Q2	A	A	A	N	A	SD	D	SD	**N/A**
Q3	SA	SA	SA	A	SA	D	A	SD	**A/SA**

Table 4.1: Responses on a Likert scale to the three questions listed in Sec. 4.3.3 and their median over all participants. Possible responses are SD (strongly disagree), D (disagree), N(neutral), A (agree), SA (strongly agree).

	P1	*P2*	*P3*	*P4*	*P5*	*P6*	*P7*	*P8*	*Med.*
Pre/Post	3	5	5	2	4	2	3	2	**3**
confirmation	3	5	1	1	4	2	1	4	**2.5**
Guidance	4	4	5	4	5	4	5	3	**4**

Table 4.2: Perceived ease, on a scale from 1 (very difficult) to 5 (very easy) of the three modalities used in the trials, along with their median over all participants.

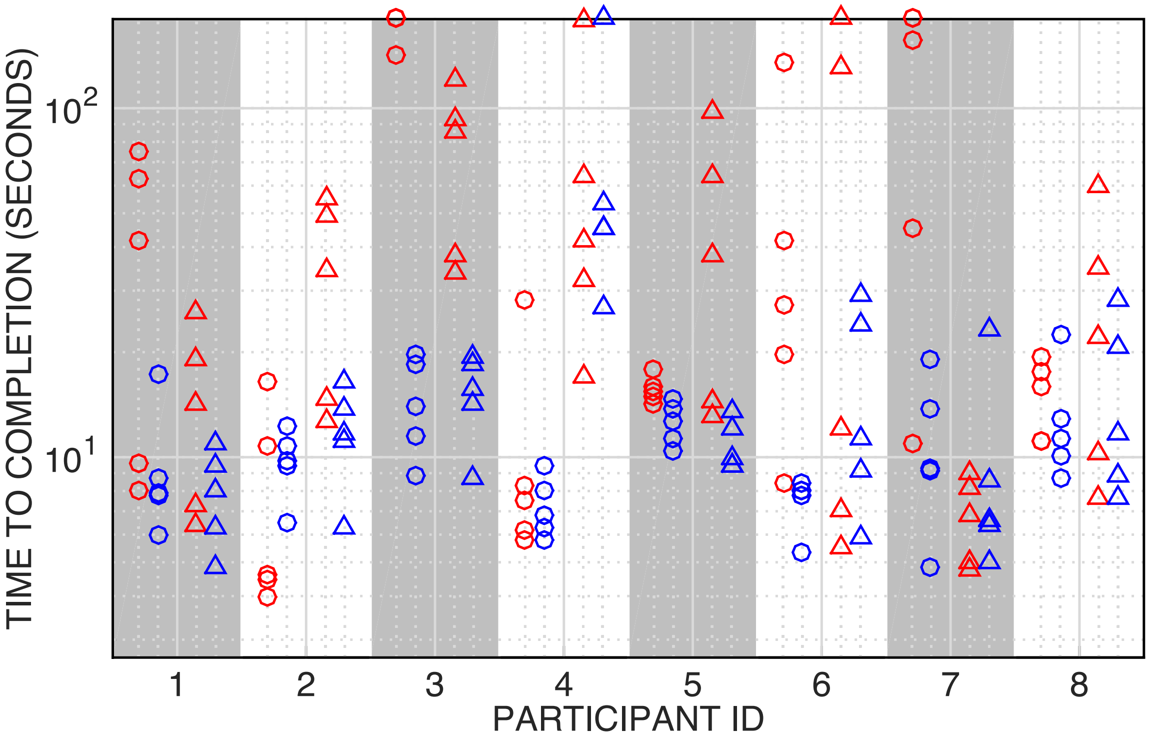


Figure 4.4: (a) Time to completion for all trials on a logarithmic scale. Circles: Large font documents. Triangles: Small font documents. Red: confirmation modality. Blue: Guidance modality.

well as feedback for improvement for our tool.

All of our participants described situations in their daily lives in which they desired to, but could not, access printed text. Examples include handouts distributed in class or at conferences, restaurant menus posted on a wall, and yoga schedules.

Four of the eight participants had previous experience with mobile OCR applications (TextDetective, Prizmo, or kNFBReader). The most commonly cited use case involved physical mail (including determining to whom a letter is addressed). Those who use OCR apps do not seem to use them very often; for example, P5 uses the kNFB app at most four

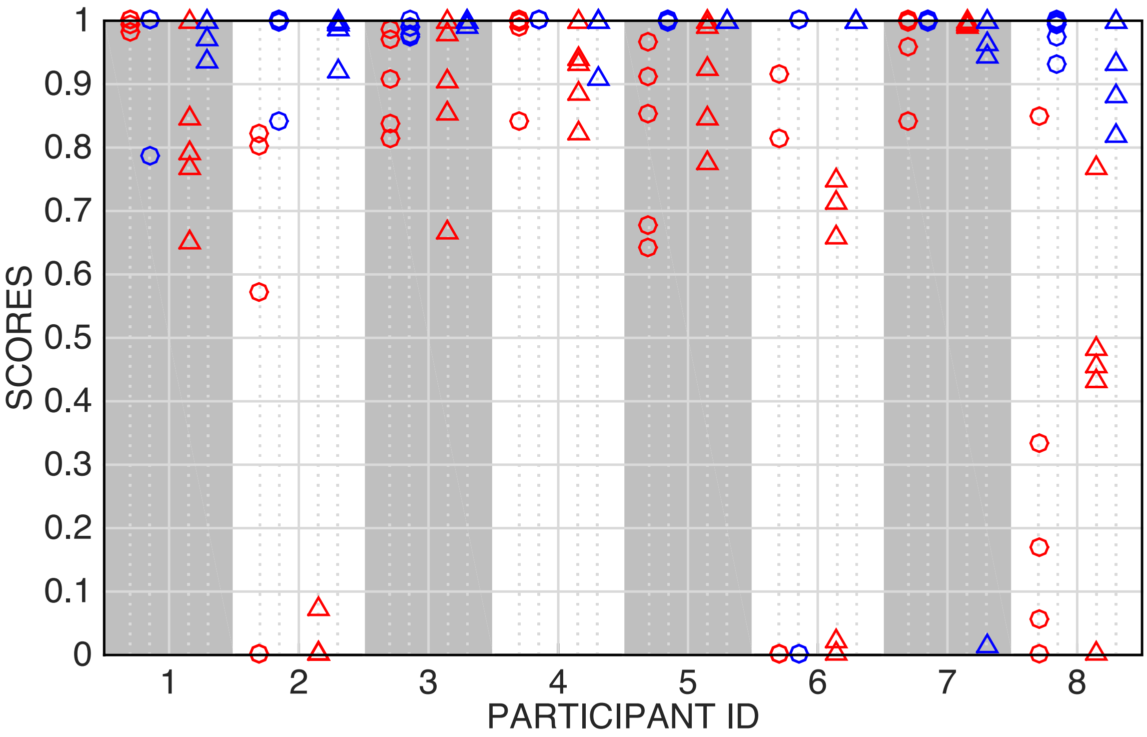


Figure 4.5: (a) OCR scores for all trials. Circles: Large font documents. Triangles: Small font documents. Red: Pre-test trials. Blue: Post-test trials.

to five times a month. Several 4 participants were familiar with the picture summarization app TapTapSee, although only one continues to use it after it became a fee-based program. P8 still uses TapTapSee for reading things that don't fit in a flatbed scanner, such as a pill container. Her strategy to take a well framed picture of pill jar is to brace it between two heavy objects, and take the picture from overhead.

Five of the participants own a flatbed scanner that can be used for OCR. Their opinion is that mobile OCR may be preferable due to increased ease to use, and also because they found that most flatbed scanner OCR software is obsolete.

Observations During Trials

As shown in Sec. 4.4.1, several participants had low OCR scores in the pre-test trials. This stemmed from holding the phone too low and poor document centering. However, most of these problems were resolved after twenty intervention trials when the participant captured their final snapshot images (post-test). To understand what happened between pretest and posttest we observed participants' aiming strategies. Our observation showed that everybody had a different aiming strategy.

Each participant developed his or her own personal strategy for moving the phone and aiming at the document. Six participants held the phone with one hand, while other used two hands (see Fig. 4.3). All participants except for P7 conducted the trials from a seated position. P7 kept standing during the pre-test and post-test trials, and remained seated for the other trials.

P2's camera targeting strategy was to move the phone very quickly to probe various areas. She told us that her yoga instructor mentioned that she moves sharply as well. Several participants would feel the edges of the document to help themselves center and orient the camera correctly. For example, P7 described his strategy as "Feel top edge, then bottom edge, then point at left top". Camera alignment is certainly important; for example, P6 kept the phone misaligned with the text lines by 30° or more, which made compliant acquisition difficult. She appeared to be aware of this problem, and frequently tried to reset the phone orientation by placing it back to the start position halfway through the trial.

However, centering and orienting the camera is not sufficient to obtain a compliant picture: for example, P7 during his pre-test trials, kept the camera too close to the document for successful reading. During the trials in the confirmation modality, participant P5 initially had difficulties achieving a compliant picture (with trials lasting sixty seconds or more), until he found a successful strategy resulting in compliant image acquisition in short time. He described his strategy as gently rocking the phone back and forth while raising it slowly.

Participant Feedback

P1, P4, and P8 suggested that the system should allow the user to "modulate" the amount of guidance given. P1 said: "Sometimes people need a lot of correction at the beginning, but other times I want it to just let me know if its a good picture at the end."

P3 had several issues with the guidance modality. Specifically, he complained about the lag between instructions, and sometime he didn't trust the instructions issued by the system (in different occasions, he remarked: "I don't believe you" and "I am dubious"). Likewise, P4 said he didn't believe the "Raise" or "Lower" instructions given, but found the horizontal positioning instructions sometimes helpful. P6 also complained that sometimes directions in the guidance modality are not produced frequently enough.

P5 remarked that the continuous feedback from the guidance is easier to interact with than the kNFBReader field of view report. The report is a statement such as "top-right edges visible rotated 11 degrees clockwise". As he put it "you have to think about what to do next. Plus you have to hold it in place. Sometimes I rotate it the wrong way". He preferred how the interface didn't require one to request information, but instead provided instructions continuously.

P8 said "Theres a lot of apps out there but there's not a lot that gives instruction. It's a great app. It's so frustrating to scan a document." She described both interventions as “frustration savers ... it might take more time to line up but it will at least save time ultimately because you don’t have to take four to five pictures”.

4.5 Conclusion

We designed and tested an iPhone app that facilitates acquisition of an OCR-readable image by a blind user. Our novel contribution includes a definition of “compliance” of a document image that can be computed quickly using a text spotter and line grouping algorithm, and a simple yet effective procedure for producing guidance instructions. Our experiments have shown that this system allows one to take OCR-readable images in a relatively short time. A simpler system that does not produce spoken directions also allows one to take OCR-readable images, but on average in a time three times as large. In addition, participants felt that use of the guidance made the image acquisition task easier. Our participants seemed to increase their proficiency in taking OCR-readable images of a document by using our system, as shown by comparing their scores in the pre- and post-test. This is also reflected by their responses, in which participants reported increased confidence in quality of their images in the post-test trials compared to the pre-test trials.

Chapter 5

Conclusions

We have shown that is possible and helpful to provide guidance to facilitate mobile OCR We hope that our research provides valuable design heuristics to accessibility researchers and engineers. Through this dissertation we have told the story of the journey of understanding how to help blind people capture OCR-readable document images. We began this dissertation with a formal definition of the range of positions and orientations from where an OCR readable document image can be acquired from. We designed a computer vision algorithm capable of verifying document image compliance and providing instructions for acquisition. We tested this algorithm with participants in a counterbalanced repeated measures experiment. Our experiment revealed that our guidance instructions decreased the time to compliant image capture three times compared to a simpler baseline system which determines compliance but does not vocalize instructions. Future work in this area includes development of guidance techniques capable of helping someone read text posted on the wall and a thorough investigation into an entirely acoustic versus spoken guidance.

Bibliography

[1] Apple developer resources. `developer.apple.com`. Accessed: 2013-11-09.

[2] DocScanner saytext. `http://www.docscannerapp.com/saytext/`. Accessed: 2013-11-09.

[3] Flow, powered by amazon. `http://www.a9.com/whatwedo/mobile-technology/flow-powered-by-amazon/`. Accessed: 2013-11-09.

[4] Look tell money reader. `http://www.looktel.com/products`. Accessed: 2013-11-09.

[5] Prizmo. `http://www.creaceed.com/prizmo`. Accessed: 2013-11-09.

[6] Ray Kurzweil wikipedia. `http://en.wikipedia.org/wiki/Ray_Kurzweil`. Accessed: 2013-11-09.

[7] Knfb reader mobile. knfb Reading Technology, Inc., 2008. http://www.knfbreader.com/.

[8] *10th International Conference on Document Analysis and Recognition, ICDAR 2009, Barcelona, Spain, 26-29 July 2009*. IEEE Computer Society, 2009.

[9] 2011. http://www.speech.cs.cmu.edu/flite/.

[10] 2011. http://www.taptapseeapp.com.

[11] Text detective (blindsight). Blindsight, Inc., 2011. http://blindsight.com/textdetective/.

[12] Aruco: a minimal library for augmented reality applications based on opencv. Universidad D Cordoba, 2012. http://www.uco.es/investiga/grupos/ava/node/26.

[13] 2015. http://www.orcam.com.

[14] Americans with disabilities act of 1990. Pub. L. 101-336. 26, July 1990. 104 Stat. 328.

[15] Shipra Agrawal and Navin Goyal. Analysis of thompson sampling for the multi-armed bandit problem. *arXiv preprint arXiv:1111.1797*, 2011.

[16] Hasan SM Al-Khaffaf, Faisal Shafait, Michael P Cutter, and Thomas M Breuel. On the performance of decapod's digital font reconstruction. In *Pattern Recognition (ICPR), 2012 21st International Conference on*, pages 649–652. IEEE, 2012.

[17] Peter Auer, Nicolò Cesa-Bianchi, Yoav Freund, and Robert E. Schapire. The nonstochastic multiarmed bandit problem. *SIAM J. Comput.*, 32(1):48–77, January 2003.

[18] Kathrin Berkner and Laurence Likforman-Sulem, editors. *Document Recognition and Retrieval XVI, DRR 2009, 16th Document Recognition and Retrieval Conference, part of the IS&T-SPIE Electronic Imaging Symposium, San Jose, CA, USA, January 18-22, 2009. Proceedings*, volume 7247 of *SPIE Proceedings*. SPIE, 2009.

[19] Jeffrey P. Bigham, Chandrika Jayant, Hanjie Ji, Greg Little, Andrew Miller, Robert C. Miller, Robin Miller, Aubrey Tatarowicz, Brandyn White, Samual White, and Tom

Yeh. Vizwiz: nearly real-time answers to visual questions. In *Proceedings of the 23nd annual ACM symposium on User interface software and technology*, UIST '10, pages 333–342, New York, NY, USA, 2010. ACM.

[20] J.P. Bigham, C. Jayant, A. Miller, B. White, and T. Yeh. Vizwiz::locateit - enabling blind people to locate objects in their environment. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on*, pages 65–72, June 2010.

[21] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.

[22] Erin Brady, Meredith Ringel Morris, Yu Zhong, Samuel White, and Jeffrey P. Bigham. Visual challenges in the everyday lives of blind people. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, pages 2117–2126, New York, NY, USA, 2013. ACM.

[23] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[24] Andrés Bruhn, Joachim Weickert, and Christoph Schnörr. Lucas/kanade meets horn/schunck: combining local and global optic flow methods. *Int. J. Comput. Vision*, 61(3):211–231, February 2005.

[25] J Canny. A computational approach to edge detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 8(6):679–698, June 1986.

[26] J. M. Chambers and T. J. Hastie, editors. *Statistical Models in S.* Wadsworth and Brooks/Cole, Pacific Grove, CA, 1992.

[27] Patrick Chiu, Michael Patrick Cutter, KIM Chelhwon, and Surendar Chandra. Generating hi-res dewarped book images, November 30 2012. US Patent App. 13/690,564.

[28] Michael P Cutter and Patrick Chiu. Capture and dewarping of page spreads with a handheld compact 3d camera. In *Document Analysis Systems (DAS), 2012 10th IAPR International Workshop on*, pages 205–209. IEEE, 2012.

[29] Michael P Cutter and Roberto Manduchi. Real time camera phone guidance for compliant document image acquisition without sight. In *Document Analysis and Recognition (ICDAR), 2013 12th International Conference on*, pages 408–412. IEEE, 2013.

[30] Michael P Cutter and Roberto Manduchi. Real time camera phone guidance for compliant document image acquisition without sight. In *Document Analysis and Recognition (ICDAR), 2013 12th International Conference on*, pages 408–412. IEEE, 2013.

[31] Michael P Cutter, Joost Van Beusekom, Faisal Shafait, and Thomas M Breuel. Font group identification using reconstructed fonts. *Proceedings of SPIE*, 7874:78740N, 2011.

[32] Michael Patrick Cutter, Joost van Beusekom, Faisal Shafait, and Thomas Michael Breuel. Unsupervised font reconstruction based on token co-occurrence. In *Proceedings of the 10th ACM symposium on Document engineering*, pages 143–150. ACM, 2010.

[33] Li Deng. The mnist database of handwritten digit images for machine learning research [best of the web]. *Signal Processing Magazine, IEEE*, 29(6):141–142, 2012.

[34] Richard O. Duda and Peter E. Hart. Use of the hough transformation to detect lines and curves in pictures. *Commun. ACM*, 15(1):11–15, January 1972.

[35] B. Epshtein, E. Ofek, and Y. Wexler. Detecting text in natural scenes with stroke width transform. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 2963–2970, June 2010.

[36] B. Epshtein, E. Ofek, and Y. Wexler. Detecting text in natural scenes with stroke width transform. In *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, pages 2963–2970, 2010.

[37] N. Ezaki, K. Kiyota, B.T. Minh, M. Bulacu, and L. Schomaker. Improved text-detection methods for a camera-based text reading system for blind persons. In *Document Analysis and Recognition, 2005. Proceedings. Eighth International Conference on*, pages 257 – 261 Vol. 1, aug.-1 sept. 2005.

[38] L. Gomez and D. Karatzas. Multi-script text extraction from natural scenes. In *Document Analysis and Recognition (ICDAR), 2013 12th International Conference on*, pages 467–471, Aug 2013.

[39] Lluis Gomez and Dimosthenis Karatzas. A fast hierarchical method for multi-script and arbitrary oriented scene text extraction. *arXiv*, 2014.

[40] M. Grundmann, V. Kwatra, and I. Essa. Auto-directed video stabilization with robust l1 optimal camera paths. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

[41] François Guimbretière. Paper augmented digital documents. In *Proceedings of the 16th annual ACM symposium on User interface software and technology*, UIST '03, pages 51–60, New York, NY, USA, 2003. ACM.

[42] R. I. Hartley and A." Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2004.

[43] Kazumasa Iwata, Koichi Kise, Tomohiro Nakai, Masakazu Iwamura, Seiichi Uchida, and Shinichiro Omachi. Capturing digital ink as retrieving fragments of document images. In *ICDAR* [8], pages 1236–1240.

[44] Chandrika Jayant, Hanjie Ji, Samuel White, and Jeffrey P. Bigham. Supporting blind photography. In *The proceedings of the 13th international ACM SIGACCESS conference on Computers and accessibility*, ASSETS '11, pages 203–210, New York, NY, USA, 2011. ACM.

[45] Jayant Kumar, Peng Ye, and D. Doermann. A Dataset for Quality Assessment of Camera Captured Document Images. In *International Workshop on Camera-Based Document Analysis and Recognition (CBDAR)*, pages 39–44, August 2013.

[46] Shaun K. Kane, Brian Frey, and Jacob O. Wobbrock. 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, pages 347–350, New York, NY, USA, 2013. ACM.

[47] Shaun K. Kane, Meredith Ringel Morris, Annuska Z. Perkins, Daniel Wigdor, Richard E. Ladner, and Jacob O. Wobbrock. Access overlays: Improving non-visual access to large touch screens for blind users. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology*, UIST '11, pages 273–282, New York, NY, USA, 2011. ACM.

[48] D. Karatzas, F. Shafait, S. Uchida, M. Iwamura, L. Gomez i Bigorda, S. Robles Mestre, J. Mas, D. Fernandez Mota, J. Almazan Almazan, and L.-P. de las Heras. Icdar 2013 robust reading competition. In *Document Analysis and Recognition (ICDAR), 2013 12th International Conference on*, pages 1484–1493, Aug 2013.

[49] Aliasgar Kutiyanawala, Vladimir Kulyukin, and John Nicholson. Teleassistance in accessible shopping for the blind. *Proc. ICOMP*, 11, 2011.

[50] N. M. Laird and J. H. Ware. Random-effects models for longitudinal data. *Biometrics*, 38(4):963–974, December 1982.

[51] Le Kang, Yi Li, and David Doermann. Orientation Robust Text Line Detection in Natural Images. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, June 2014.

[52] Vincent Lepetit, Francesc Moreno-Noguer, and Pascal Fua. Epnp: An accurate o(n) solution to the pnp problem. *Int. J. Comput. Vision*, 81(2):155–166, February 2009.

[53] Jian Liang, Daniel DeMenthon, and David Doermann. Camera-based document image mosaicing. In *Proceedings of the 18th International Conference on Pattern Recognition - Volume 02*, ICPR '06, pages 476–479, Washington, DC, USA, 2006. IEEE Computer Society.

[54] Jian Liang, David Doermann, and Huiping Li. Camera-based analysis of text and documents: a survey. *International Journal on Document Analysis and Recognition*, 7(2):84–104–104, July 2005.

[55] David G Lowe. Object recognition from local scale-invariant features. In *Computer vision, 1999. The proceedings of the seventh IEEE international conference on*, volume 2, pages 1150–1157. Ieee, 1999.

[56] Roberto Manduchi and James Coughlan. (computer) vision without sight. *Commun. ACM*, 55(1):96–104, January 2012.

[57] Roberto Manduchi and Sri Kurniawan, editors. *Assistive Technology for Blindness and Low Vision*. CRC Press, 2013.

[58] Yannick Morvan. Multi-view-coding-thesis, 2009. http://www.epixea.com/research/multi-view-coding-thesis.html.

[59] Tomohiro Nakai, Koichi Kise, and Masakazu Iwamura. Use of affine invariants in locally likely arrangement hashing for camera-based document image retrieval. In Horst Bunke and A.Lawrence Spitz, editors, *Document Analysis Systems VII*, volume 3872 of *Lecture Notes in Computer Science*, pages 541–552. Springer Berlin Heidelberg, 2006.

[60] Tomohiro Nakai, Koichi Kise, and Masakazu Iwamura. Camera-based document image mosaicing using llah. In Berkner and Likforman-Sulem [18], pages 1–10.

[61] Lukas Neumann. Real-time scene text localization and recognition. In *Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, CVPR '12, pages 3538–3545, Washington, DC, USA, 2012. IEEE Computer Society.

[62] Lukas Neumann and Jiri Matas. Text localization in real-world images using efficiently pruned exhaustive search. In *Proceedings of the 2011 International Conference on*

Document Analysis and Recognition, ICDAR '11, pages 687–691, Washington, DC, USA, 2011. IEEE Computer Society.

[63] Lukas Neumann and Jiri Matas. Scene text localization and recognition with oriented stroke detection. In *The IEEE International Conference on Computer Vision (ICCV)*, December 2013.

[64] Long Quan and Zhong-Dan Lan. Linear N-Point Camera Pose Determination. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(8):774–780, August 1999.

[65] Faisal Shafait, Michael Patrick Cutter, Joost van Beusekom, Syed Saqib Bukhari, and Thomas M Breuel. Decapod: A flexible, low cost digitization solution for small and medium archives.

[66] Huiying Shen and James M. Coughlan. Towards a real-time system for finding and reading signs for visually impaired users. In *Proceedings of the 13th international conference on Computers Helping People with Special Needs - Volume Part II*, ICCHP'12, pages 41–47, Berlin, Heidelberg, 2012. Springer-Verlag.

[67] Roy Shilkrot, Jochen Huber, Connie Liu, Pattie Maes, and Suranga Chandima Nanayakkara. Fingerreader: A wearable device to support text reading on the go. In *CHI '14 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '14, pages 2359–2364, New York, NY, USA, 2014. ACM.

[68] Roy Shilkrot, Jochen Huber, Meng Ee Wong, Pattie Maes, and Suranga Nanayakkara.

FingerReader: A wearable device to explore printed text on the go. In *Proc. ACM CHI*, 2015.

[69] E. Tekin and J.M. Coughlan. An algorithm enabling blind users to find and read barcodes. In *Applications of Computer Vision (WACV), 2009 Workshop on*, pages 1–8, 2009.

[70] Ender Tekin and James M. Coughlan. A mobile phone application enabling visually impaired users to find and read product barcodes. In *Proceedings of the 12th International Conference on Computers Helping People with Special Needs*, ICCHP'10, pages 290–295, Berlin, Heidelberg, 2010. Springer-Verlag.

[71] B. Tomoyuki Nassu, R. Minetto, and L.E. Soares de Oliveira. Text line detection in document images: Towards a support system for the blind. In *Document Analysis and Recognition (ICDAR), 2013 12th International Conference on*, pages 638–642, Aug 2013.

[72] Marynel Vázquez and Aaron Steinfeld. Helping visually impaired users properly aim a camera. In *Proceedings of the 14th international ACM SIGACCESS conference on Computers and accessibility*, ASSETS '12, pages 95–102, New York, NY, USA, 2012. ACM.

[73] C. Wolf and J.-M. Jolion. Object count/area graphs for the evaluation of object detection and segmentation algorithms. *International Journal on Document Analysis and Recognition*, 8(4):280–296, 2006.

[74] Cong Yao, Xiang Bai, Wenyu Liu, Yi Ma, and Zhuowen Tu. Detecting texts of arbitrary

orientations in natural images. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 1083–1090, June 2012.

[75] Chucai Yi and Yingli Tian. Assistive text reading from complex background for blind persons. In *Proceedings of the 4th international conference on Camera-Based Document Analysis and Recognition*, CBDAR'11, pages 15–28, Berlin, Heidelberg, 2012. Springer-Verlag.

[76] Ali Zandifar and Antoine Chahine. A video based interface to textual information for the visually impaired. In *Proceedings of the 4th IEEE International Conference on Multimodal Interfaces*, ICMI '02, pages 325–, Washington, DC, USA, 2002. IEEE Computer Society.

[77] Zhengyou Zhang. A flexible new technique for camera calibration. *IEEE Trans. Pattern Anal. Mach. Intell.*, 22(11):1330–1334, November 2000.

[78] Yu Zhong, Pierre J. Garrigues, and Jeffrey P. Bigham. Real time object scanning using a mobile phone and cloud-based visual search engine. In *Proceedings of the 15th International ACM SIGACCESS Conference on Computers and Accessibility*, ASSETS '13, pages 20:1–20:8, New York, NY, USA, 2013. ACM.

[79] Yu Zhong, Walter S Lasecki, Erin Brady, and Jeffrey P Bigham. RegionSpeak: Quick comprehensive spatial descriptions of complex images for blind users. In *Proc. ACM CHI*, 2015.