Hindawi Publishing Corporation
EURASIP Journal on Image and Video Processing
Volume 2007, Article ID 96357, 13 pages
doi:10.1155/2007/96357

Research Article

Color Targets: Fiducials to Help Visually Impaired People Find Their Way by Camera Phone

James Coughlan[1] and Roberto Manduchi[2]

[1] *Rehabilitation Engineering Research Center, Smith-Kettlewell Eye Research Institute, San Francisco, CA 94115, USA*
[2] *University of California, Santa Cruz, CA 95064, USA*

Received 16 January 2007; Revised 10 May 2007; Accepted 2 August 2007

Recommended by Thierry Pun

A major challenge faced by the blind and visually impaired population is that of wayfinding—the ability of a person to find his or her way to a given destination. We propose a new wayfinding aid based on a camera cell phone, which is held by the user to find and read aloud specially designed machine-readable signs, which we call color targets, in indoor environments (labeling locations such as offices and restrooms). Our main technical innovation is that we have designed the color targets to be detected and located in fractions of a second on the cell phone CPU, even at a distance of several meters. Once the sign has been quickly detected, nearby information in the form of a barcode can be read, an operation that typically requires more computational time. An important contribution of this paper is a principled method for optimizing the design of the color targets and the color target detection algorithm based on training data, instead of relying on heuristic choices as in our previous work. We have implemented the system on Nokia 7610 cell phone, and preliminary experiments with blind subjects demonstrate the feasibility of using the system as a real-time wayfinding aid.

1. INTRODUCTION

There are nearly 1 million legally blind persons in the United States, and up to 10 millions with significant visual impairments. A major challenge faced by this population is that of *wayfinding*—the ability of a person to find his or her way to a given destination. Well-established orientation and mobility techniques using a cane or guide dog are effective for following paths and avoiding obstacles, but are less helpful for finding specific locations or objects.

We propose a new assistive technology system to aid in wayfinding based on a camera cell phone (see Figure 1), which is held by the user to find and read aloud specially designed signs in the environment. These signs consist of barcodes placed adjacent to special landmark symbols. The symbols are designed to be easily detected and located by a computer vision algorithm running on the cell phone; their function is to point to the barcode to make it easy to find without having to segment it from the entire image. Our proposed system, which we have already prototyped, has the advantage of using standard off-the-shelf cell phone technology—which is inexpensive, portable, multipurpose, and becoming nearly ubiquitous—and simple color signs which can be easily produced on a standard color printer. Another advantage of the cell phone is that it is a mainstream consumer product which raises none of the cosmetic concerns that might arise with other assistive technology requiring custom hardware.

Our system is designed to operate efficiently with *current* cell phone technology using machine-readable signs. Our main technological innovation is the design of special landmark symbols (i.e., fiducials), which we call *color targets*, that can be robustly detected and located in fractions of a second on the cell phone CPU, which is considerably slower than a typical desktop CPU. The color targets allow the system to quickly detect and read a linear barcode placed adjacent to the symbol. It is important that these symbols be detectable at distances up to several meters in cluttered environments, since a blind or visually impaired person cannot easily find a barcode in order to get close enough to it to be read. Once the system detects a color target, it guides the user towards the sign by providing appropriate audio feedback.

This paper builds on our previous work [1], in which the color target patterns and detection algorithm were designed

Figure 1: Camera cell phone held by blind user.

heuristically, by describing a principled method for optimizing the design parameters. This method uses training data containing images of different colors rendered by different printers and photographed under multiple lighting conditions, as well as negative examples of typical real-world background images where color targets are not present, to determine which color target pattern is both maximally distinctive and maximally invariant with respect to changing environmental conditions (such as illumination). Once an optimal pattern has been selected, an algorithm that detects the pattern as reliably and quickly as possible can be easily determined.

We have implemented a real-time version of our wayfinding system, which works with any camera cell phone running the Symbian OS (such as the Nokia 7610, which we are currently using). The system is set up to guide the user towards signs using audio beeps, and reads aloud the sign information using prerecorded speech (which will eventually be replaced by text-to-speech). Sign information can either be encoded directly as ASCII text in the barcode, or can encode a link to an information database (which is what our prototype does on a small scale). The signs are affixed to the walls of a corridor in an office building to label such locations as particular office numbers and restrooms. Preliminary experiments with blind subjects demonstrate the feasibility of using the system as a real-time wayfinding aid (see Section 4).

2. RELATED WORK

A number of approaches have been explored to help blind travelers with orientation, navigation, and wayfinding, most using modalities other than computer vision. The most promising modalities include infrared signage that broadcasts information received by a hand-held receiver [2], GPS-based localization, RFID labeling, and indoor Wi-Fi-based localization (based on signal strength) and database access [3]. However, each of these approaches has significant limitations that limit their attractiveness as stand-alone solutions. Infrared signs require costly installation and maintenance; GPS has poor resolution in urban settings and is unavailable indoors; RFIDs can only be read at close range and would therefore be difficult to locate by blind travelers; and Wi-Fi localization requires extensive deployment to ensure complete coverage, as well as a time-consuming calibration process.

Research has been undertaken on computer vision algorithms to aid in wayfinding for such applications as navigation in traffic intersections [4] and sign reading [5]. The obvious advantage of computer vision is that it is designed to work with little or no infrastructure or modification to the environment. However, none of this computer vision research is yet practical for commercial use because of issues such as insufficient reliability and prohibitive computational complexity (which is especially problematic when using the kind of portable hardware that these applications require).

Our approach, image-based labeling, is motivated by the need for computer vision algorithms that can run quickly and reliably on portable camera cell phones, requiring only minor modifications to the environment (i.e., posting special signs). Image-based labeling has been used extensively for product tagging (barcodes) and for robotic positioning and navigation (fiducials) [6–10]. It is important to recognize that a tag reading system must support two complementary functionalities: detection and data embeddings. These two functionalities pose different challenges to the designer. Reliable detection requires unambiguous target appearance, whereas data embedding calls for robust spatial data encoding mechanisms. Distinctive visual features (shapes and textures or, as in this proposal, color combinations) can be used to maximize the likelihood of successful detection. Computational speed is a critical issue for our application. We argue that color targets have a clear advantage in this sense with respect to black and white textured patterns.

Variations on the theme of barcodes have become popular for spatial information encoding. Besides the typical applications of merchandise or postal parcel tagging, these systems have been demonstrated in conjunction with camera phones in a number of focused applications, such as linking a product or a flyer to a URL. Commercial systems of this type include the Semacode, QR code, Shotcode, and Nextcode. An important limitation of these tags is that they need to be seen from a close distance in order to decode their dense spatial patterns. Our approach addresses both requirements mentioned above by combining a highly distinctive fiducial with a barcode.

Direct text reading would be highly desirable, since it requires no additional environment labeling. Standard OCR (optical character recognition) techniques are effective for reading text against a blank background and at a close distance [11], but they fail in the presence of clutter [12]. Recently, developed algorithms address text localization in cluttered scenes [13–16], but they currently require more CPU power than is available in an inexpensive portable unit, our preliminary tests show cell phone processing speed to be 10–20 times slower than that of a portable notebook computer for integer calculations (and slower still if floating point calculations are performed). Barcodes suffer from a similar limitation in that they must be localized, typically by a hand-held scanner, before they can be read. We note that our color target approach solves both the problems of quickly localizing barcodes or text and of designating the specific information that is useful for wayfinding.

We originally introduced the concept of a color target for wayfinding, along with a fast barcode reader, in [1]. However, in [1], the target was designed based on purely heuristic criteria. In this paper, we provide a sound approach to the joint design and testing of the color target and of the detection algorithm.

3. COLOR TARGETS

We have designed the color targets to solve the problem of localizing information on signs. The targets are designed to be distinctive and difficult to confuse with typical background clutter, and are detectable by a robust algorithm that can run very quickly on a cell phone (i.e., up to 2 or more frames/sec. depending on resolution). Once the targets are detected, barcodes or text adjacent to them are easily localized [1]. A variety of work on the design and use of specially designed, easily localized landmarks has been undertaken [6, 7], but to the best of our knowledge, this is the first cell phone-based application of landmark symbols to the problem of environmental labeling.

We use a cascade filter design (such as that used in [17]) to rapidly detect the color target in clutter. The first filter in the cascade is designed to quickly rule out regions of the image that do not contain the target, such as homogeneous regions (e.g., blue sky or white wall without markings). Subsequent filters rule out more and more nontarget locations in the image, so that only the locations containing a target pass all the filter tests in the cascade (with very few false positives).

Rather than relying on generic edge-like patterns, which are numerous in almost every image of a real scene, we select a smaller set of edges: those at the boundaries of particular color combinations, identified by certain color gradients. Some form of color constancy is required if color is to be a defining feature of the target under varied illumination. One solution would be to preprocess the entire image with a generic color constancy algorithm, but such processing generally makes restrictive assumptions about illumination conditions and/or requires significant computational resources. Fortunately, while the appearance of individual colors varies markedly depending on illumination, color *gradients* tend to vary significantly less [18]. We exploit this fact to design a cascade of filters that threshold certain color gradient components. The gradients are estimated by computing differences in RGB channels among three or four pixels in a suitable configuration. The centroid of the three pixels, (x, y), is swept across the entire pixel lattice.

3.1. Target color and test design

A critical task of this project is the selection of a small set of color patches forming our target, along with the design of the visual detection algorithm. The ideal color target should satisfy two main requirements. It should be *distinctive*, meaning that it should be easily recognizable. At the same time, its appearance should be *invariant* with respect to changing environmental conditions (illumination, viewing angle, distance, camera noise). Distinctiveness and invariance are important characteristics of feature detection algorithms for numerous vision tasks (stereo [19], wide-baseline matching [20], object recognition [21, 22], tracking [23]). Compared to typical vision applications, however, we have one degree of freedom more, namely, the choice of the target that we want to recognize. It is clear that target design should be undertaken jointly with algorithm optimization, with the goal of minimizing the likelihood of missing the target (false negative) while maintaining a low rate of false alarms (targets mistakenly detected where there is none).

As mentioned above, our targets display a pattern with a small number of N contiguous color patches. In order to detect the target, the image is scanned by a moving window, which samples N-tuples of pixels (*probes*) in a suitable arrangement. The color patches are shaped as radial sectors, placed so as to form a full circle (see, e.g., Figure 7). Accordingly, the probes are arranged uniformly on a circumference with suitable radius R (see Figure 7). Suppose that the sliding window is placed at the center of the projection of the target in the image. In ideal conditions (i.e., when there is no motion blur, sampling effects can be neglected and the camera is fronto-parallel with the target at the correct orientation), this probing arrangement will sample exactly one pixel per color patch, regardless of the distance to the target (as long as the target projection has radius larger or equal to R). This important feature motivated our choice for the target shape. We will discuss issues related to the optimal choice of R in Section 3.3. It suffices here to observe that sampling artifacts and motion blur are directly related to the distance between probing pixels: the closer the probes, the more significant these effects.

The number of color patches in the target should be chosen carefully. Too many patches make detection challenging, because in this case, the radial sectors containing the color patches become narrow, and therefore the distance between probing pixels becomes small. On the contrary, decreasing the number of patches reduces the distinctiveness of the pattern (other “background” patches may contain the same color configuration). The notion of distinctiveness is clearly related to the false positive rate (FPR), which can be estimated over a representative set of images that do not contain any targets.

Another important design decision is the choice of the detection algorithm. Due to the limited computational power of a cell phone, and the real-time requirement of the system, it is imperative that the algorithm involves as few operations per pixel as possible. For a given algorithm, we can design the target so as to optimize its detection performance. Hence, even a simple algorithm has the potential to work well with the associated optimal target. In comparison, in typical real-world vision applications, the features to be observed may be, and often are, highly ambiguous, requiring more complex detection strategies. Our algorithm performs a cascade of one-dimensional “queries” over individual color channels of pairs of color patches. More specifically, let $c_m = (c_m^1, c_m^2, c_m^3)$ represent the RGB color vector

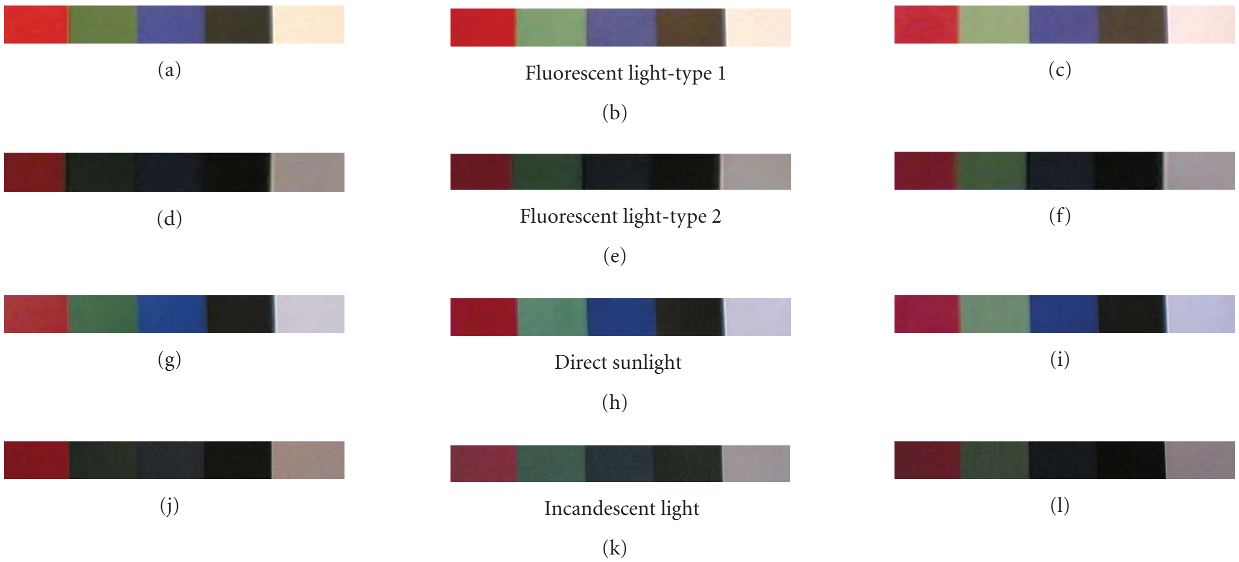


Figure 2: The figure shows a sample of the 24 images taken with the 5 possible color patches under different lighting conditions. Each row contains images of the patches generated by three different printers under the same illumination condition. Empirical statistics from this dataset are used to determine optimal query thresholds.

as measured by the probing pixel for the mth patch. Then, a query involving the mth and nth color patches over the kth color channels ($k = 1, 2, 3$ designates the red, green, and blue color channels, resp.) can be expressed as follows:

$$(c_m^k - c_n^k) \geq T_{m,n}^k, \quad (1)$$

where $T_{m,n}^k$ is a suitable threshold. The quadruplet $Q = (m, n, k, T_{m,n}^k)$ fully characterizes the query. The detection algorithm is thus defined by the sequence of J queries $(Q_1, Q_2, \ldots, Q_J)$. Only if a pixel satisfies the whole cascade of queries, it is considered to be a candidate target location. The advantage of using a cascade structure is that if the first few queries are very selective, then only a few pixels need to be tested in the subsequent queries.

For fixed values N of color patches and J of queries, the joint target-algorithm design becomes one of finding patch colors and queries that give low values of FPR as well as of FNR (false negative rate, or missed target detection). We decided to tackle the problem via exhaustive testing on carefully chosen image sets. In order to make the problem tractable, we confined the choice of color patches to a set containing the following colors: $\{\text{red}, \text{green}, \text{blue}, \text{black}, \text{white}\}$. More precisely, when creating an 8 bit image to be printed out, we select colors from the following RGB representations: $\{(255, 0, 0), (0, 255, 0), (0, 0, 255), (255, 255, 255), (0, 0, 0)\}$. White and black are obvious choices due to their different brightness characteristics and to the fact that they can be easily reproduced by a printer. As for the other three colors, the following argument suggested their choice. First, note that an ideal surface with monochromatic reflectance spectrum has optimal color constancy characteristics: a change in the illuminant spectrum only determines a change in the intensity of the reflected light (and therefore of the measured color). Secondly, if the camera's spectral sensitivities are unimodal, with peaks centered at wavelengths matching the monochromatic light reflected by the color patches, then for each color patch the camera generates a response in only one color channel (the remaining two channels being negligible) [24]. In other words, the use of red, green, and blue is motivated by the fact that we are using an RGB camera.

Unfortunately, the colors produced by a typical printer are far from monochromatic. Furthermore, different printers produce different results for the same input image. The color values read by the camera also depend on the illuminant spectrum, on the white balancing operation in the camera, on the nonlinear transfer function (gamma), and on any brightness adjustment via exposure and gain control. Note that we do not have control over exposure, gain, and white balancing for our camera phone. Finally, specularities may be present as well, but we will neglect them in this work (as they did not seem to be a major problem in our experiments).

Rather than attempting to compensate for the different lighting and exposure conditions via a color constancy algorithm, which may require additional modeling and computation, we decided to use a set of exemplars and to choose target colors and queries that prove robust against varying illumination and background. In addition, we considered three different printer models for our target.

Twenty four images of the five possible color patches, printed on a sheet of paper by three different printers, were taken by our camera phone under very different lighting conditions, both indoor and outdoor. A sample of these images is shown in Figure 2. We will use empirical statistics about this image dataset to determine "optimal" query thresholds, described later in this section.

In order to evaluate the false positive ratio, we also considered the seven "background" images (not containing a target) shown in Figure 3. This image set represents a sample of

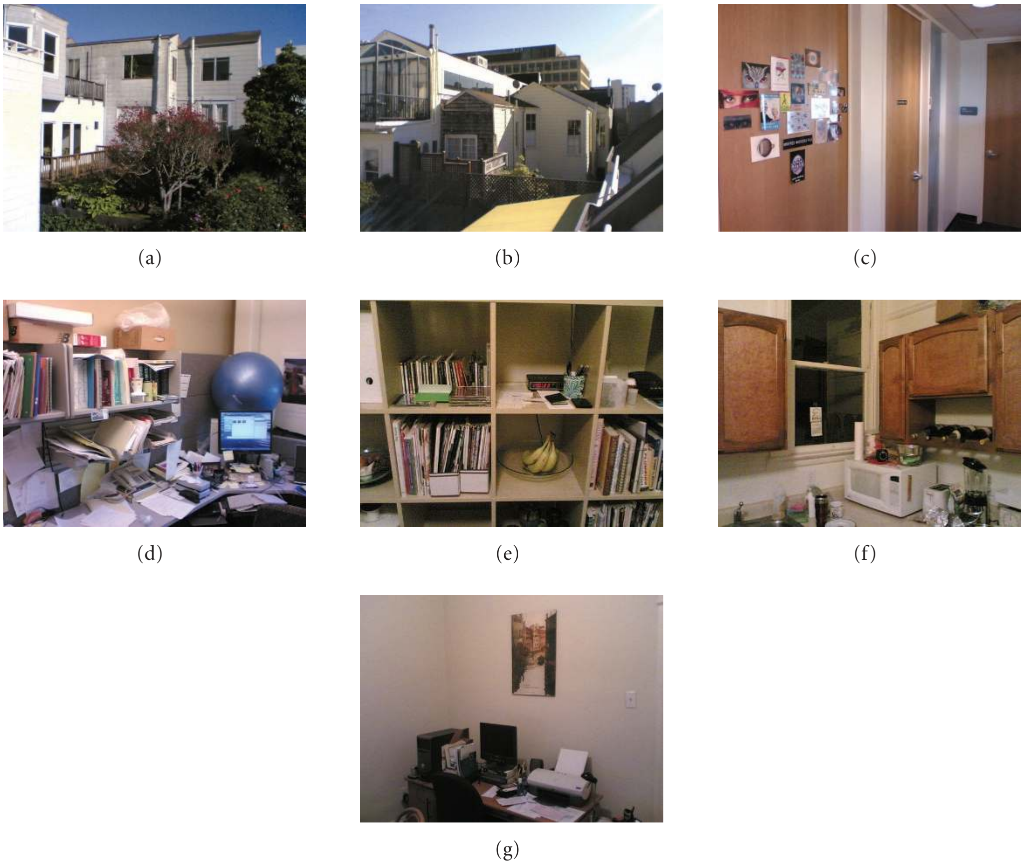

(a) (b) (c) (d) (e) (f) (g)

Figure 3: The “background” images used to evaluate the false positive rate.

different representative situations, including cluttered indoor and outdoor scenes. Ideally, these images would provide adequate information about the statistical characteristics of the background. We reckon that seven images may not represent an adequate sample for representing the environment. Nevertheless, we use this data set as a simple working reference on which to assess our system’s performance, aware that the results may change using a different data set. In a practical implementation of our wayfinding system, it may be possible to collect images of the environment (e.g., the corridors in an office building) where targets are placed.

Given the set $C = \{C_1, C_2, \ldots, C_N\}$ of color patches forming the target and a query $Q_i = (m, n, k, T^k_{m,n})$, we estimate the associated FNR by running the algorithm on the set of target images. For each image, we pick a pixel from color patch m and one from color patch n, and check whether they verify (1). We repeat this test for all pixel pairs from the two patches for the same image, counting the number of pixels that do not pass the test. We then compute the sum of these numbers for all images and divide the result by the overall number of pixel pairs, obtaining the FNR associated with Q_i. We can compute the FNR associated to a query sequence $\mathbf{Q} = (Q_1, Q_2, \ldots, Q_J)$ in a similar fashion. Likewise, we can compute the associated FPR by running the detection algorithm on the background images and counting the number of pixels mistakenly classified as target. Note that the value of the threshold $T^k_{m,n}$ determines the values of FNR and FPR for the query Q_i.

There are several possible criteria to specify an “optimal” threshold for query Q_i. We use a very simple approach: select the largest value of $T^k_{m,n}$ such that the associated FNR is equal to 0. In other words, we choose the most stringent threshold that ensures that all pixel pairs from color patches m and n pass test (1). This is achieved by setting $T^k_{m,n}$ to the minimum value of the color difference $(c^k_m - c^k_n)$ over the dataset of the known color patches (see Figure 2). (The minimum is chosen with respect to all pairs of pixels falling within color patches m and n, over all printers and illumination conditions.) As we will see shortly, this criterion provides us with a straightforward optimization technique. A potential disadvantage of the unbalanced weight placed on the FNR and FPR is that the resulting FPR may be too high for practical use. Our experiments show that this does not seem to be the case. It should also be pointed out that there are subsequent layers of processing to validate whether a candidate pixel belongs to a target image or not. Hence, it is critical that in this first phase, no target pixel is missed by the algorithm.

A convenient feature of our optimization algorithm is that, by forcing FNR = 0, we can separate the computation of thresholds $T^k_{m,n}$ from the choice of color patches and of the query sequence. Indeed, for each pair (m, n) of color patches, $T^k_{m,n}$ is chosen based only on the color patch training images.

Table 1: The best colors for targets with $N = 3$ or 4 patches and the lower bounds for the false positive rate (FPR_{LB}) using thresholds derived from images from each individual printer and from images from all printers.

	$N = 3$ patches		$N = 4$ patches	
	Colors	FPR_{LB}	Colors	FPR_{LB}
Printer 1	(White, red, black)	$9.2 \cdot 10^{-6}$	(White, red, green, blue)	0
Printer 2	(White, red, green)	$5.6 \cdot 10^{-5}$	(White, red, green, blue)	0
Printer 3	(White, red, green)	$5.8 \cdot 10^{-4}$	(White, red, green, black)	$7.5 \cdot 10^{-6}$
All printers	(White, red, black)	$1.6 \cdot 10^{-3}$	(White, red, blue, black)	$5.6 \cdot 10^{-5}$

Once the set of thresholds has been computed, we can proceed to estimate the set of colors C and the set of queries $\mathbf{Q}$. We are considering only targets with a number of colors equal to $N = 3$ or $N = 4$ in this work. Since there are 5 colors to choose from, the number of possible targets[1] is $\binom{5}{N}$. Given a target and the length J of the query sequence, and noting that the order of the queries does not affect the final result, it is easy to see that the number of possible different query sequences is equal to $\binom{3N(N-1)}{J}$, since there are 3 color channels and $N(N-1)$ possible ordered pairs of distinct color patches. For example, for a given 3-color target, there are 3.060 different quadruplets of queries. Although the problem of optimal query sequence selection is NP-hard, it is possible to solve it exhaustively in reasonable time for small values of sequence lengths J. We have considered a maximum value of $J = 5$ in this work for the 3-color target, and $J = 4$ for the 4-color target (since the number of possible queries is much greater in this case). This choice is justified by the experimental observation that the decline in FPR resulting in an increase of J from 4 to 5 (or from 3 to 4 in the 4-color target case) is normally modest, hence larger values of J may not improve performances significantly (details are given later in this section). Thus, for a given color set C, we proceed to test all possible J-plets of queries, and select the one with the lowest associated FPR.

In order to reduce the computational cost of optimization, we select the set of colors C before query optimization, based on the following suboptimal strategy. For each combination of N colors, we consider the FPR associated with the sequence comprising all possible $3N(N-1)$ queries. This sequence being unique (modulo an irrelevant permutation), the associated FPR is computed straightforwardly. The resulting value (FPR_{LB}) has the property that it represents an achievable lower bound for the FPR of any query sequence using those colors. We then select the combination of colors with associated smallest values of FPR_{LB}. By comparing this value with a predetermined threshold, one can immediately check whether the number N of colors is large enough, or if a larger N should be used.

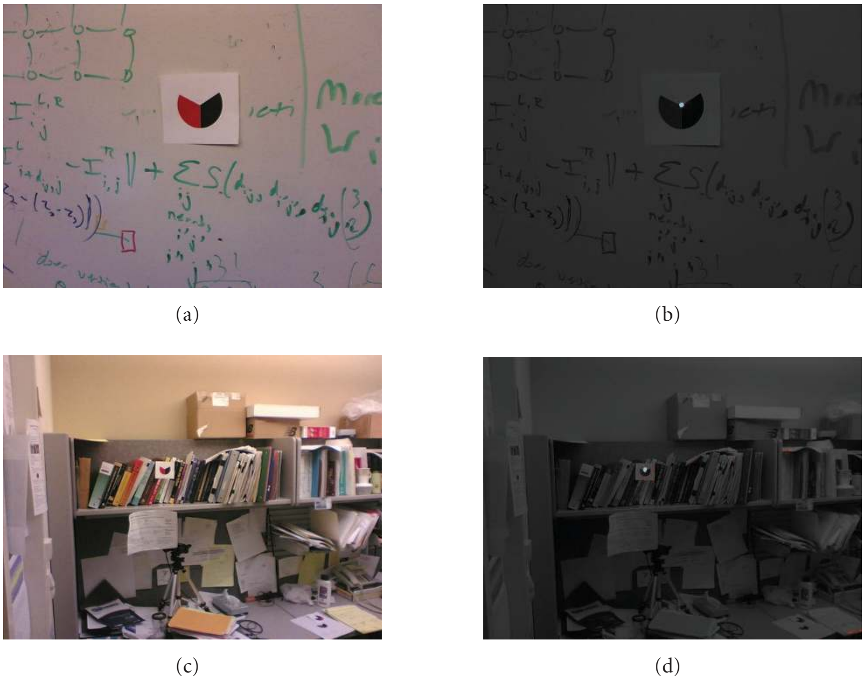
Table 1 shows the best colors and the lower bounds FPR_{LB} using thresholds derived from images from each individual printer, as well as using thresholds derived from the whole collection of images over all printers. The latter case can be seen as an attempt to find an algorithm that is robust against the variability induced by the printer type. It is interesting to note how different printers give rise to different values of FPR_{LB}. In particular, Printer 1 seems to create the most distinctive patterns, while Printer 3 creates the most ambiguous ones. As expected, the algorithm that gives zero false negatives (no misses) from targets created from all printers is also the algorithm that creates the highest rate of false positives. As for the color choice, the white and red patches are always selected by our optimization procedure, while the remaining color(s) depends on the printer type. Note also that the 3-color target has higher FPR_{LB} than the 4-color target for all cases.

As an example of optimal query sequence, we computed the optimal quadruplet of queries associated with the optimal 3-color set for Printer 1 (white, red, black). The queries are (note that the color channels are labeled 1,2,3 to represent red, green, and blue) $Q_1 = (1, 2, 2, 89)$; $Q_2 = (2, 3, 1, 69)$; $Q_3 = (3, 2, 2, -44)$; $Q_4 = (2, 1, 1, -69)$. In simple words, a triplet of probes passes this query sequence if the first patch is quite greener but not much redder than the second patch, and if the second patch is quite redder but not much greener than the third patch. The queries are ordered according to increasing FPR, in order to ensure that most of the pixels are ruled out after the first queries. The average number of queries per pixel before a pixel is discarded (i.e., rejected as a target pixel) is 1.037. The FPR associated with this query sequence is equal to $1.4 \cdot 10^{-3}$. By comparison, note that if five queries are used, the FPR decreases only by a small amount to $1.2 \cdot 10^{-3}$.

Here is another example, involving a quadruplet of queries associated with the optimal 4-color set with thresholds computed over the whole set of printers. In this case, the optimal colors are (white, red, blue, black), and the query sequence is $Q_1 = (1, 2, 2, 58)$; $Q_2 = (2, 4, 1, 33)$; $Q_3 = (2, 3, 1, 22)$; $Q_4 = (2, 1, 1, -90)$. This query sequence ensures that the first patch is significantly greener but not much redder than the second patch, and that the second patch is significantly redder than both the third and the fourth patches. This query sequence has FPR $= 2.2 \cdot 10^{-4}$. (The FPR is only slightly higher, $3.1 \cdot 10^{-4}$, when three queries are used instead of four.) On average, the algorithm computes 1.08 queries per pixel before recognizing that a pixel is not a target. Detection results examples using these two query sequences are shown in Figures 4 and 5.

Additional postprocessing is needed to rule out the few false positives that survive the query cascade. A simple and

[1] Note that we are neglecting the order of color patches in the target, although it may be argued that the order may play a role in the actual FPR.



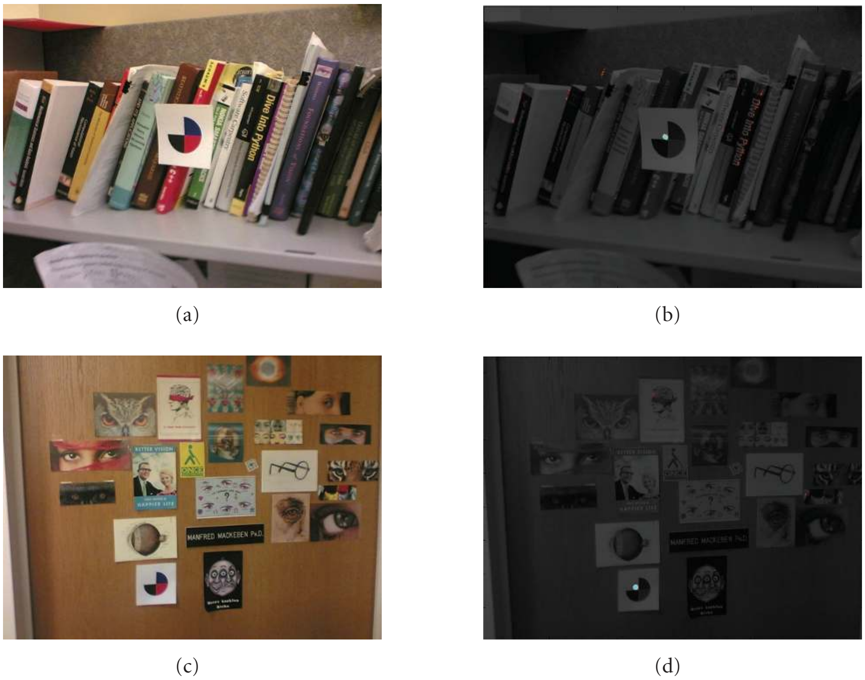
(a)

(b)

(c)

(d)

Figure 4: Some detection results using the sequences with 4 queries described in this section, with 3-color targets. Pixels marked in red were detected by the color-based algorithm but discarded by the subsequent clustering validation test. Pixel marked in blue survived both tests.



(a)

(b)

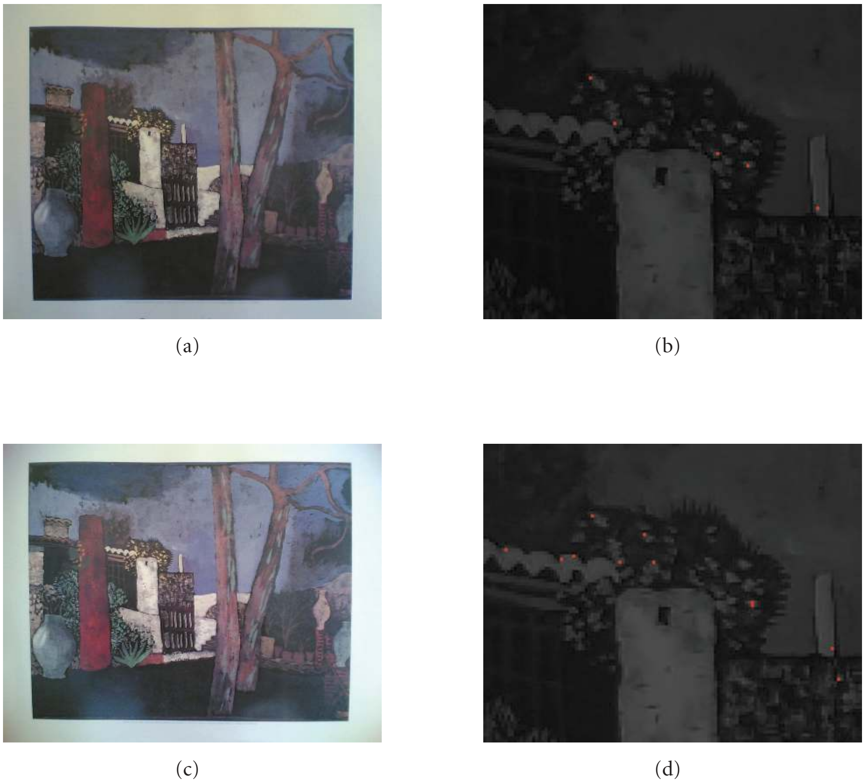
(c)

(d)

Figure 5: Some detection results using the sequences with 4 queries described in this section, with 4-color targets. See caption of Figure 4.

fast clustering algorithm has given excellent results. Basically, for each pixel that passed the query sequence, we compute how many other passing pixels are in a 5 × 5 window around it. If there are less than 13 other passing pixels in the window, this pixel is removed from the list of candidates. Finally, remaining candidate pixels are inspected for the presence of a nearby barcode, as discussed in [1].

We implemented this simple prototype algorithm in C++ on a Nokia 7610 cell phone running the Symbian 7.0 OS. The camera in the phone has a maximum resolution of 1152 by



(a)

(b)

(c)

(d)

Figure 6: Left, sample scene photographed by different cameras (7610 on top, 6681 on bottom). Right, false positives of zoomed-in region of image shown in red. This example illustrates the similarity of the pattern of false positives between cameras.

864 pixels, although we normally operate it at VGA resolution. The algorithm detects multiple targets in a fraction of a second to about half a second (depending on camera resolution). The detection is invariant to a range of scales (from about 0.5 m to as far as 10 m), and accommodates significant rotations (up to about 30 degrees in the camera plane), slant and background clutter.

Note that the color targets need to be well illuminated to be detected, or else image noise will obscure the target colors. One way to overcome this limitation might be to operate a flash with the camera, but this approach would use significant amounts of battery power, would fail at medium to long range, and would be annoying to other people in the environment. Another possibility might be to increase the exposure time of the camera, but this would make the images more susceptible to motion blur; similarly, increasing the camera gain would increase pixel noise as well as the brightness. (Note that the exposure time and gain are set automatically and cannot be specified manually.) In addition, white balancing is set automatically and the background of the color target may affect the color target's appearance in an unpredictable way. Overall, it seems most practical to site the targets at locations that are already well lit; accordingly, we have emphasized the application of our system to indoor environments such as office buildings, which are usually well lit throughout common areas such as corridors and lobbies.

3.2. Comparisons between different cameras

The color target patterns and color target detection algorithms were designed and tested for a variety of illumination conditions and printers. However, all the preceding experiments were conducted using a single camera cell phone model, the Nokia 7610, and it is important to determine whether the results of the experiments generalize to different camera models. In general, we might expect that different cameras will have different imaging characteristics (such as color matching functions), which could necessitate the use of different color difference thresholds. It is impractical to test every possible combination of illumination condition, printer, and camera model, especially given the enormous (and constantly growing) selection of camera models, so in this section, we describe some simple experiments demonstrating that our color target detection algorithm works similarly for three different Nokia models: the 7610, 6681, and N80.

In these experiments, we examined the FPR obtained by the same color target detection algorithm applied to images of background scenes not containing color target patterns, photographed by the three different cameras (see, e.g., Figure 6). Four scenes were used, each under a different illumination condition: an indoor scene illuminated only by outdoor daylight (indirect sunlight through a window), normal indoor (fluorescent) illumination, dark indoor (also

Table 2: Comparing FPRs for images of scenes taken by different cameras (Nokia 6681, 7610, and N80). For each scene, note that the FPR usually varies by no more than about a factor of two from camera to camera.

	6681	7610	N80
Scene 1	$1.02 \cdot 10^{-4}$	$5.99 \cdot 10^{-5}$	$9.88 \cdot 10^{-5}$
Scene 2	$1.63 \cdot 10^{-4}$	$2.03 \cdot 10^{-4}$	$4.37 \cdot 10^{-4}$
Scene 3	$2.78 \cdot 10^{-3}$	$2.78 \cdot 10^{-3}$	$1.57 \cdot 10^{-3}$
Scene 4	$2.58 \cdot 10^{-3}$	$1.90 \cdot 10^{-3}$	$2.69 \cdot 10^{-3}$

fluorescent) illumination, and an outdoor scene (also indirect sunlight). It is difficult to make direct comparisons between images of the same scene photographed by different cameras; not only is it hard to ensure that the photographs of each scene are taken with the camera in the exact same location and at the exact same orientation, but the cameras have different resolutions and slightly different fields of view. To address this problem, we performed a simple procedure to "normalize" the images from the different cameras. In this procedure, we chose scenes of planar surfaces that commonly appear on walls, such as posters and printouts (in our experience, these are a common source of false positives in indoor environments), and held each camera from approximately the same distance to the scenes. We resampled the images from each camera to the same resolution (1152 × 864, the resolution of Nokia 7610). Finally, we placed a featureless rectangular frame against these surfaces to define a region of interest to analyze; after resampling the image to the standard resolution, the image was manually cropped to include everything inside the frame but nothing outside it.

The FPRs, shown numerically in Table 2 and illustrated in Figure 6, were estimated by running the four-color target detector described in the previous section (using the four queries and thresholds obtained for the 7610 camera) across each normalized image. (Results were similar for the three-color target, which we do not include here.) For each scene, note that the FPR usually varies by no more than about a factor of two from camera to camera, despite differences of over two orders of magnitude in the FPR from one scene to the next. These results demonstrate that a color target detector trained on one camera should have similar performance on other cameras. However, in the future, the color target detector can be tailored to each individual camera model or manufacturer if necessary.

3.3. Theoretical and empirical bounds

3.3.1. Maximum detection distance (stationary)

The width of the color target, together with the resolution and the field of view (FOV) of the camera, determine the maximum distance at which the target can be detected. For simplicity's sake, we will only consider 3-color targets in the following. For the Nokia 7610 cell phone, the instantaneous horizontal FOV (IFOV) of a single pixel is approximately 1.5 mrads for the 640×480 resolution, and 0.82 mrads for the 1152×864 resolution. The pixels can be considered square to a good approximation. In order to detect a target at a distance d, it is necessary that all three color patches be correctly resolved. The color at a pixel location, however, is computed by interpolation from the underlying Bayer mosaic, which typically involves looking at color values within a 3 × 3 window centered at the pixel. (We note that our algorithm processes uncompressed image data, without any JPEG artifacts that could complicate this analysis.) This means that, in order to correctly measure the color of a patch, the patch must project onto a square of at least 3 × 3 pixels, so that at least one pixel represents the actual patch color. In fact, we found out that as long as at least half of the pixels within the 3 × 3 window receive light from the same color patch, detection is performed correctly.

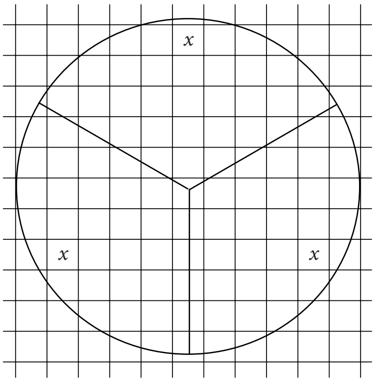


Figure 7: The layout of a 3-patch color target, with the location of the "probing" pixels. The lower two pixels are separated by a buffer of $M = 7$ pixels.

Now, suppose that two measurement pixels are separated by a buffer zone of M pixels as in Figure 7. In our implementation, we chose $M = 7$. The importance of these buffer pixels in the context of motion blur will be discussed in Section 3.3.2. It is clear from Figure 7 that the diameter D of the color target should project onto at least $M + 4$ pixels for color separation. This is obviously an optimistic scenario, with no blurring or other forms of color bleeding and no radial distortion. In formulas, and remembering that the tangent of a small angle is approximately equal to the angle itself:

$$d \le \frac{D}{\mathrm{IFOV} \cdot (M+4)}. \qquad (2)$$

We have considered two target diameters in our experiments, $D = 6$ cm and $D = 12$ cm. Table 3 shows the theoretical bounds, computed using (2), as well as empirical values, obtained via experiments with a color target under two different incident light intensities (175 lux and 10 lux, resp.). A lower detection distance may be expected with low light due to increased image noise. The maximum distances reported in the table include the case when no postprocessing is performed (such as the clustering algorithm of Section 3.1). This provides a fairer comparison with the model of Figure 7, which only requires one-point triplet detection. Of course, postprocessing (which is necessary to reject false positives) reduces the maximum detection distance, since it requires

Table 3: Maximum distances (in meters) for color target detection. Theoretical bounds are reported together with experimental values with and without the postprocessing (PP) module. Values in the case of poor illumination are shown within parentheses.

	$D = 6$ cm			$D = 12$ cm		
	Theor.	Exp.-No PP.	Exp.-PP.	Theor.	Exp.-No PP.	Exp.-PP.
640 × 480	3.6	3.5(3)	2.7(2.4)	7.2	6.7(5.7)	5.6(4.7)
1152 × 864	6.6	5.5(4.5)	4.3(3)	13.2	11.1(8.5)	9.3(6.4)

that a certain number of triplets is found. The experiments were conducted while holding the cell phone still in the user's hand. Note that the experimental values, at least for the case of well-lit target and without postprocessing, do not differ too much from the theoretical bounds, which were obtained using a rather simplistic model.

3.3.2. *Maximum detection distance (panning)*

Searching for a color target is typically performed by pivoting the cell phone around a vertical axis (panning) while in low-resolution (640 × 480) mode. Due to motion, blur can and will arise, especially when the exposure time is large (low-light conditions). Motion blur affects the maximum distance at which the target can be detected. A simple theoretical model is presented below, providing some theoretical bounds.

Motion blur occurs because, during exposure time, a pixel receives light from a larger surface patch than when the camera is stationary. We will assume for simplicity's sake that motion is rotational around an axis through the focal point of the camera (this approximates the effect of a user pivoting the cell phone around his or her wrist). If ω is the angular velocity and T is the exposure time, a pixel effectively receives light from a horizontal angle equal to $\mathrm{IFOV} + \omega T$. This affects color separation in two ways. Firstly, consider the vertical separation between the two lower patches in the color target. For the two lower probing pixels in Figure 7 to receive light from different color patches, it is necessary that the apparent image motion be less than[2] $\lfloor M/2 \rfloor - 1$ (this formula takes the Bayer color pattern interpolation into account). The apparent motion (in pixels) due to panning is equal to $\omega T/\mathrm{IFOV}$, and therefore the largest acceptable angular velocity is $(\lfloor M/2 \rfloor - 1) \cdot \mathrm{IFOV}/T$. For example, for $M = 7$ and $T = 1/125$ s, this corresponds to 21.5°/s. The second way in which motion blur can affect the measured color is by edge effects. This can be avoided by adding a "buffer zone" of $\lceil \omega T/\mathrm{IFOV} \rceil$ pixels to the probing pixels of Figure 7. This means that the diameter of the target should project onto $M + 2 \cdot (2 + \lceil \omega T/\mathrm{IFOV} \rceil)$ pixels. Hence, the maximum distance for detection decreases with respect to the case of Section 3.3.1.

In fact, these theoretical bounds are somewhat pessimistic, since a certain amount of motion blur does not necessarily mean that the target cannot be recognized. In order to get some more realistic figures, we ran a number of experiments, by pivoting the cell phone at different angular velocities in front of a 12 cm target from a distance of 2 meters. Since we could neither control nor measure exposure time, comparison with the theoretical bounds is difficult. When the color target was lit with average light intensity (88 lux), detection was obtained with probability larger than 0.5 at angular speeds of up to 60°/s. With lower incident light (10 lux), this value was reduced to 30°/s, presumably due to larger exposure time.

Table 4: Rates (in frames per minute) attained for different image resolutions with and without target detection module (proc./no proc.) and with and without display in the viewfinder (disp./no displ.).

	No proc./displ.	Proc./displ.	Proc./no displ.
640 × 480	114	110	154
1152 × 864	21	19	20

3.3.3. *Detection rates*

The rate at which target detection is performed depends on two factors: the image acquisition rate, and the processing time to implement the detection algorithm. Table 4 shows the rates attained with and without processing and display (in the viewfinder). Image display is obviously not necessary when the system is used by a blind person, but in our case it was useful for debugging purposes. Note that image display takes 44% of the time in the VGA detection loop. If the images are not displayed, the frame rate in the VGA resolution mode is more than 2.5 frames per second. However, for the high-resolution case, image acquisition represents a serious bottleneck. In this case, even without any processing, the acquisition/display rate is about 21 frames per minute. When processing is implemented (without display), the rate is 20 frames per minute.

Given the extremely low acquisition rate for high-resolution images provided by this cell phone, we use the following duty cycle strategy. The scene is searched using VGA resolution. When a target is detected over a certain number F (e.g., $F = 5$) of consecutive frames, a high-resolution snapshot is taken. Barcode analysis is then implemented over the high-resolution data [1]. The number F of frames should be large enough to allow the user to stop the panning motion, thereby stabilizing the image and reducing the risk of motion blur when reading the barcode.

[2] The symbol $\lfloor \cdot \rfloor$ represents the largest integer smaller than or equal to the argument.

4. EXPERIMENTS WITH BLIND SUBJECTS

We have conducted four proof-of-concept experiments to demonstrate the feasibility of a blind person using a cell phone to obtain location information using a color target. These experiments were performed by three blind volunteers who were informed of the purpose of the experiments. To guide the subjects towards color targets using the system, we devised a simple three-pitch audio feedback strategy: low, medium, or high tones signified the target appearing in the left, center, or right part of the camera's field of view, respectively, and silence signified that no target was visible to the system. We used 640 × 480 camera resolution in the experiments, which allowed the system to process a few frames per second, and a 4-color target.

For the first three experiments, the subjects were informed that the goal of the experiments was to measure their ability to use the color target (and/or associated barcode) to perform simple wayfinding tasks. They were given a brief (approximately, 10-minute) training session in which the experimenter explained the operation of the color target system and had them try it out for themselves (initially with assistance, and then unassisted). During this session, the experimenter advised the subjects that they needed to move the cell phone slowly and smoothly to avoid motion blur, and that they had to hold the cell phone upright and horizontal, taking care not to cover the camera lens with their fingers.

In the first experiment, a blind subject was seated near the wall of a small conference room (approximately, 7 meters by 5 meters), and a color target was placed on another wall in one of four possible locations relative to the subject: left, far left, right, or far right. For each trial, the color target location was chosen at random by the experimenter. The subject was asked to use the cell phone target detector to identify which location the target was in. After a practice session, he identified the location for ten consecutive trials without making any mistakes.

A second experiment featuring a more challenging task was conducted with another blind subject, testing his ability to locate, walk to, and touch a color target in an unknown location. For each trial, the experimenter placed a color target at a random location (at shoulder height) on the walls of the same conference room (which was free of obstacles). Beginning in the center of the room, the second blind subject used the color target detector to find the color target, walk towards it, and touch it. In 20 trials, it took him anywhere from 13 to 33 seconds to touch the target. In 19 of the 20 trials, he touched only the target, while in one trial, he first touched the wall several inches away before reaching the target.

The third experiment was designed to test the ability of a blind subject to find locations of interest in an office corridor using the color target system augmented with barcodes. The goal was to find four locations along the walls of a straight corridor (about 30 meters long) using the cell phone system; he was advised that the labels did not correspond to the actual building layout, so that his familiarity with the building would not affect the outcome of the experiment. A color target with 5-bit barcode was affixed at waist height near each of the four locations in the corridor. The barcodes encoded four separate numbers which were associated with four prerecorded sounds ("elevator," "restroom," "room 417," and "staircase"). The subject was instructed to walk along the corridor to scan for all four labels. When the camera was close enough to the color target, the barcode next to it was read and the appropriate recording was played back. After a training session, the labels were moved to new locations and the subject was able to find all four of them in about two minutes. No false positives or false barcode readings were encountered during the experiment.

Table 5: Results of a color target experiment with blind subjects. The task was to locate Braille signs in the environment (either a corridor or conference room) using the color target detector, and to then read them aloud. Each cell shows the time (in seconds) it took the subject to perform the task in two separate trials.

	Subject 1	Subject 2
Corridor	164, 145	281, 84
Conference room	61, 65	68, 67

Finally, a fourth experiment was recently conducted (see [25] for details) to measure the performance of the color target system as a tool for finding Braille signs, and in addition to investigate the strategies that worked best for blind subjects using this system to explore their environment. The subjects were informed of these experimental goals and given a brief (approximately 10 minute) training session as in the previous experiments. The task in this experiment was to locate and read aloud Braille signs bearing a single letter chosen at random (A through J) located either in the corridor or the conference room (the same locations as in the previous experiments). (For comparison, we repeated the same experiment without the use of the cell phone system, so that the subjects relied exclusively on their sense of touch; details on that experiment are not relevant to this paper and are reported in [25].) The experiment was performed by two blind subjects, each of whom read Braille. Randomization of the placement and content of the Braille signs minimized the effects of familiarity with the layout of the environment.

For each trial in the experiment, two different Braille signs were placed randomly at shoulder height in two different places. In the corridor trials, the Braille signs were placed at seven possible locations (which were described to the subjects in the training session), covering existing Braille signs (designating rooms and office numbers) running the length of the corridor. For the conference room trials, two different Braille letter signs were placed randomly at shoulder height, with the two signs on different walls (also randomly chosen).

We recorded the total time for the subject to find and read aloud both Braille letter signs in each trial; the results are shown in Table 5. An important finding of the experiment is that both subjects adopted similar search strategies for searching with the cell phone system: walking slowly down the corridor with the cell phone camera pointed at one wall (and pointing the camera at the other wall on the way back), and panning the camera in a circle in the middle of the conference room to find the directions to the color targets before advancing towards them.

This finding provides valuable insight into how blind subjects actually use the cell phone system, and suggests the most important areas for improvement. To operate the cell phone system, the subjects had to walk slowly and/or pan the camera slowly to avoid motion blur. Walking was an appropriate strategy for exploring walls at close range, as in corridors (in which the subject was never far from the walls); the walking speed had to be fairly slow, since the apparent motion of the walls would otherwise cause excessive motion blur, which would be exaggerated at close range. Conversely, panning allowed subjects to explore wide expanses of walls at greater distances, as from the vantage point in the center of a room.

Another finding is that, while the subjects were instructed to hold the camera as level as possible, the inability to maintain a level orientation (which is difficult for a blind or visually impaired person to estimate) was the most common cause of problems: if the camera was sufficiently off the horizontal, the subject could walk by a color target without detecting it. (This scenario explains an outlier in our data in Table 5, the inordinately long time that it took Subject 2 to complete the first cell phone trial, 281 seconds.)

While these four experiments are preliminary, they show that blind subjects are able to use a simple cell phone interface to locate signs at a range of distances and orientations and that they are capable of orienting the camera properly and moving it smoothly and slowly enough to prevent interference from motion blur. These results provide direct evidence of the feasibility of our proposed system.

However, the experimental results also underscore the need to address the problems of motion blur and limited rotation invariance in the future. Barring future improvements in cell phone camera technology (i.e., faster exposure times), we could make the system less sensitive to motion blur—and thereby allow the user to walk or pan more quickly—by using larger color targets. This would permit a larger separation between the probe pixels, which would allow greater amounts of motion blur without sacrificing long-distance detection. A more aesthetically acceptable alternative may be to use the same size targets as before but to adopt a multiscale detection approach: a greater probe pixel separation could be used in a first stage of detection for each image, and if nothing is detected in the image, a second stage could be executed with a narrower separation (to detect targets at a greater distance).

One way to improve the system's range of rotation invariance is to use a target with three colors rather than four, since under ideal conditions, the former is invariant to orientation deviations of $\pm 60^\circ$, while the range for the latter is $\pm 45^\circ$. However, we have found that using three colors creates more false positives than using four colors. Another possibility is to use the usual four-color target but to expand the color target search by including multiple orientations of the probe pixels (e.g., over the three orientations 0°, $+20^\circ$, and -20°). It is unclear whether this approach can be implemented without slowing down the detection process too much.

In the future, we will conduct experiments to test the performance of our system under more complicated—but typical—real-world conditions, including unfamiliar buildings and locations, in corridors and rooms of different shapes and sizes, with signs placed at varying heights. These experiments will help us to improve the user interface, which may combine the use of vibratory signals, audio tones, and synthesized speech.

5. CONCLUSION

We have demonstrated a camera cell phone-based wayfinding system that allows a visually impaired user to find signs marked with color targets and barcodes. A key challenge of the system is the limited computational power of the cell phone, which is about 10–20 times slower than a typical notebook computer. Our solution is to place a distinctive color target pattern on the sign, which may be rapidly detected (using integer arithmetic) even in cluttered scenes. This swiftly guides the system to an adjacent barcode, which we read using a novel algorithm that is robust to poor resolution and lighting. An important contribution of this paper is a principled method for optimizing the design of the color targets and the color target detection algorithm based on training data, instead of relying on heuristic choices as in our previous work. Preliminary experiments with blind subjects confirm the feasibility of the system.

A priority for future work is to minimize susceptibility to motion blur and to improve the rotation invariance of the color target detection, for which we have described possible solutions. In addition, we will test the system with more blind subjects (and will need to include low-vision subjects) under more complicated—but typical—real-world conditions to help us to improve the user interface, which may combine the use of vibratory signals, audio tones, and synthesized speech.

Even with these future improvements, some drawbacks of our wayfinding system will remain. For instance, color targets (and associated barcodes) can only be detected when they are in the direct line of sight of the camera, and obviously cannot be detected in the dark; even when minimized, motion blur prevents rapid panning by the cell phone user, and hence limits the speed with which he or she can explore the environment; and oblique viewing angles (which are commonly encountered in corridors) and limitations on the size of the color target signs (imposed by space constraints and cosmetic considerations) restrict the detection range of the system. However, despite these drawbacks, we stress that our proposed system provides wayfinding information that is *otherwise completely inaccessible* to blind and visually impaired persons in a portable, low-cost package based solely on off-the-shelf hardware. Moreover, in the future we envisage integrating our computer vision-based color target system with alternative modalities such as GPS and Wi-Fi (which we note are becoming increasingly common features on cell phones), in order to combine the complementary strengths of each technology and to maximize the power and ease of use of the wayfinding system.

ACKNOWLEDGMENTS

The authors would like to acknowledge support from NIH grant 1 R21 EY017003-01A1, and the first author would also like to acknowledge useful discussions with Dr. Joshua Miele and programming assistance from Dr. Huiying Shen.

REFERENCES

[1] J. Coughlan, R. Manduchi, and H. Shen, "Cell phone-based wayfinding for the visually impaired," in *The 1st International Workshop on Mobile Vision*, Graz, Austria, May 2006.

[2] W. Crandall, B. Bentzen, L. Myers, and J. Brabyn, "New orientation and accessibility option for persons with visual impairment: transportation applications for remote infrared audible signage," *Clinical and Experimental Optometry*, vol. 84, no. 3, pp. 120–131, 2001.

[3] A. M. Ladd, K. E. Bekris, A. P. Rudys, D. S. Wallach, and L. E. Kavraki, "On the feasibility of using wireless ethernet for indoor localization," *IEEE Transactions on Robotics and Automation*, vol. 20, no. 3, pp. 555–559, 2004.

[4] M. S. Uddin and T. Shioyama, "Bipolarity- and projective invariant-based zebra-crossing detection for the visually impaired," in *Proceedings of the 1st IEEE Workshop on Computer Vision Applications for the Visually Impaired (CVAVI '05)*, p. 22, San Diego, Calif, USA, June 2005.

[5] P. Silapachote, J. Weinman, A. Hanson, R. Weiss, and M. A. Mattar, "Automatic sign detection and recognition in natural scenes," in *Proceedings of the 1st IEEE Workshop on Computer Vision Applications for the Visually Impaired (CVAVI '05)*, p. 27, San Diego, Calif, USA, June 2005.

[6] Y. Cho and U. Neumann, "Multi-ring color fiducial systems for scalable fiducial tracking augmented reality," in *Proceedings of the IEEE Virtual Reality Annual International Symposium (VRAIS '98)*, p. 212, Atlanta, Ga, USA, March 1998.

[7] D. Claus and A. W. Fitzgibbon, "Reliable fiducial detection in natural scenes," in *Proceedings of the 8th European Conference on Computer Vision (ECCV '04)*, vol. 4, pp. 469–480, Prague, Czech Republic, May 2004.

[8] L. Naimark and E. Foxlin, "Circular data matrix fiducial system and robust image processing for a wearable vision-inertial self-tracker," in *Proceedings of the International Symposium on Mixed and Augmented Reality (ISMAR '02)*, pp. 27–36, Darmstadt, Germany, September-October 2002.

[9] D. Scharstein and A. J. Briggs, "Real-time recognition of self-similar landmarks," *Image and Vision Computing*, vol. 19, no. 11, pp. 763–772, 2001.

[10] A. State, G. Hirota, D. Chen, W. Garrett, and M. Livingston, "Superior augmented reality registration by integrating landmark tracking and magnetic tracking," in *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '96)*, pp. 429–438, New Orleans, La, USA, August 1996.

[11] A. Zandifar, P. R. Duraiswami, A. Chahine, and L. S. Davis, "A video based interface to textual information for the visually impaired," in *Proceedings of the 4th IEEE International Conference on Multimodal Interfaces (ICMI '02)*, pp. 325–330, Pittsburgh, Pa, USA, October 2002.

[12] J. Liang, D. Doermann, and H. Li, "Camera-based analysis of text and documents: a survey," *International Journal on Document Analysis and Recognition*, vol. 7, no. 2-3, pp. 84–104, 2005.

[13] X. Chen and A. L. Yuille, "Detecting and reading text in natural scenes," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '04)*, vol. 2, pp. 366–373, Washington, DC, USA, June-July 2004.

[14] J. Gao and J. Yang, "An adaptive algorithm for text detection from natural scenes," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '01)*, vol. 2, pp. 84–89, Kauai, Hawaii, USA, December 2001.

[15] A. K. Jain and B. Yu, "Automatic text location in images and video frames," *Pattern Recognition*, vol. 31, no. 12, pp. 2055–2076, 1998.

[16] H. Li, D. Doermann, and O. Kia, "Automatic text detection and tracking in digital video," *IEEE Transactions on Image Processing*, vol. 9, no. 1, pp. 147–156, 2000.

[17] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '01)*, vol. 1, pp. 511–518, Kauai, Hawaii, USA, December 2001.

[18] B. V. Funt and G. D. Finlayson, "Color constant color indexing," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 5, pp. 522–529, 1995.

[19] R. Manduchi and C. Tomasi, "Distinctiveness maps for image matching," in *Proceedings of the 10th International Conference on Image Analysis and Processing*, pp. 26–31, Venice, Italy, September 1999.

[20] A. Baumberg, "Reliable feature matching across widely separated views," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '00)*, vol. 1, pp. 774–781, Hilton Head Island, SC, USA, June 2000.

[21] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[22] C. Schmid, R. Mohr, and C. Bauckhage, "Evaluation of interest point detectors," *International Journal of Computer Vision*, vol. 37, no. 2, pp. 151–172, 2000.

[23] J. Shi and C. Tomasi, "Good features to track," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR '94)*, pp. 593–600, Seattle, Wash, USA, June 1994.

[24] G. Sharma and H. J. Trussell, "Digital color imaging," *IEEE Transactions on Image Processing*, vol. 6, no. 7, pp. 901–932, 1997.

[25] J. Coughlan and R. Manduchi, "Functional assessment of a camera phone-based wayfinding system operated by blind users," in *Conference of IEEE Computer Society and the Biological and Artificial Intelligence Society (IEEE-BAIS), Research on Assistive Technologies Symposium (RAT '07)*, Dayton, Ohio, USA, April 2007.