# Mobile Currency Reader for People with Visual Impairments

Xu Liu liuxu@cs.umd.edu

Department of Computer Science
University of Maryland, College Park

In this paper we present a camera phone-based currency reader that can identify U.S. paper dollars for people with visual impairments. Currently, U.S. paper currency can only be identified visually and this situation will continue for a foreseeable future. Our solution harvests the imaging and computational power on camera phones to read these bills. Considering it is impractical for people with visual impairments to capture high quality images, our currency reader performs real time processing for each captured frame as the camera approaches the bill. We extracted illumination invariant features based on random pixel pairs, trained our currency reader using Ada-boost framework and implemented an efficient recognizer using box-filters for pre-classification on camera phones. Our currency reader processes 10 frames/second and achieves a false positive rate $<10^{-4}$. Our system is trained on front and back at both corners of U.S. paper currencies so that the user does not have to know which side or corner is up for recognition. We performed user study with blind users and received promising recognition time of 21.3 seconds per bill with no error.

## 1. Problem and Motivation

Visual impairments affect a large percentage of population in various ways[1] Current estimates suggest there are approximately 10 million blind or visually impaired individuals in the United States alone. Visual impairments significantly affect quality of life of these populations and limit many daily activities especially using cash or for financial transactions. Currencies are often printed on different sizes of paper or with different tactile for people with visual disabilities to touch and recognize. However, in the U.S. these user-friendly features are not provided for the visually impaired users. The blind community initiated a law suit against the discrimination of the Department of the Treasury and won the case on May 20, 2008[2]. This situation may eventually be resolved from the engraving and printing process, but it may be a prolonged process and will be expensive to replace all currency already in use. Therefore a light weighted currency reader is desired in the near future. The government has been searching for such a solution since 2007 and a recent request for information has been announced by the Bureau of Engraving and Printing, "Denominating US Currency by the Blind and Visually Impaired"[12]. We have designed an efficient objection recognition algorithm and implemented it for currency recognition on camera phones to fulfill the task. Our system is currently being evaluated by the government in a contract with Arinc Inc.

## 2. Background and Related Work

Dedicated devices such as "Kurzweil reader"[3] have been introduced to help reading currency, but they are often bulky and expensive. Novel systems such as iCare[4] have also been developed to help the visually impaired people with pattern recognition. iCare uses a wearable camera for imaging and a PC for computation. We propose an alternative solution to employ the ubiquitous camera phone[10] to identify different denomination in an instant and inexpensive way. The combined imaging and computational power of new devices has inspired us to embed image processing and computer vision algorithms into the devices. Although for this project we target for reading currency, the designed framework can be extended to help the visually impaired users identify other objects as well.

[View video: https://youtu.be/teSDq_XkIE8 ]

Classic pattern recognition algorithms usually include feature extraction and feature classification. Widely used features such as SIFT[5] or SIFT-likes[6,7] have high repeatability. SVM[8] and neural networks[9] can be trained to achieve high accuracy given enough time and space allowance. However, these classic pattern recognition approaches cannot be ported directly to mobile devices. Implementing pattern recognition on mobile devices has three major challenges.1)The limited processing power of the device, 2)the fact that the captured scene could contain complex background resulting in false positive that must be eliminated, and 3) the expectation of the user who typically Fig.1 Mobile Currency Reader expects instant feedback and requires on line (real time) recognition.

These three challenges are related to the speed of the algorithm. The algorithm must be efficient enough to fit in the light-weight device and be able to discard quickly images or pixels that are not of interest, so more time can be allocated to the image that contains objects to be recognized. Ideally, when an algorithm is efficient enough to run in real time, the recognition can be performed on the video stream of the camera, and the user does not have to hit a key to capture an image. We choose to process the image in real time which provides a much smoother user experience and avoids motion blur caused by "click-to-capture," but as noted earlier, it must typically deal with lower quality data.

## 3. Uniqueness of the Approach

Since our system is designed for people with visual impairments, it is impractical for our user to complete sophisticated operations which are already difficult for regular users because of the small keyboard input[14]. The overall design principle we follow is to minimize user operation, especially with respect to keyboard hits. Take the pattern recognizer as an example, a typical mobile pattern recognition system (for example the Kurzweil K1000 device) asks the user to take a snapshot and then the system tries to recognize the result. If the image is imperfect, the recognition may fail and the user will have to repeat the process. However we cannot expect visually impaired user to perform such tasks and it is impractical to ask them to take high quality pictures for recognition. Real time recognition gives smooth user experience. Our pattern recognizer runs on the device and processes approximately 10 frames per second so that the user gets instant response as the camera moves. This introduces the challenge that we must process the video stream from the camera at a very high speed. We address this problem using a boosted object detector with both high efficiency and accuracy.

In our approach, we tackle these challenges in two steps and achieve high accuracy, real time recognition on mobile devices. First, we use a very fast pre-classifier to filter the images and areas of an image with low probability of containing the target. This step is inspired by the Viola-Jones face detector[9] and the Speed-Up Robust Feature[7], both of which use box filters to detect objects and features rapidly. Second, we use a set of local pixel pairs to form weak classifiers and use Ada-boost[13] to train strong but fast classifiers from these weak classifiers. The idea of using local pixel pairs is inspired by Ojala, et al.'s work on Local Binary Patterns [11] and more recent work by Pascal, et al.[6]. The advantage of local pixel pairs lies in their robustness to noise, blur and global lighting changes which are significant challenges for mobile recognition. The details are presented in the next two sub-sections.

### 3.1 Fast Classification with Random Pixel Pairs

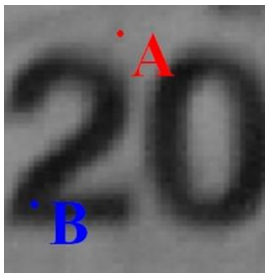

Fig. 2 An example of random pixel pair

In this section we introduce a fast classifier based on random pixel pairs. One challenge of pattern recognition on the mobile device is usability. Traditional "capture and recognize" approaches may not be friendly to mobile users. When the recognition fails (because of motion blur, de-focusing, shadows, or any other reason) the user will have to click and try again. Instead, real time recognition is preferred. At the same time, recognition may benefit from the fact that images are taken by a cooperative user. Unlike a mounted camera with the objects moving in the scene, the camera (phone) is moving itself and the user is usually approaching the object to be recognized. We assume that the object rests approximately to the same relative position of the camera when being recognized. Under this assumption, we can subtract the background, extract and normalize the object, then perform the recognition under a relative stable setup. Our primary concern, the features, is the key subject of this section. The features we seek must be distinct (not to raise any false alarms), robust (to tolerate weak perspective and lighting variation) and fast to compute on the phone. When considering feature, the first option may be SIFT[5] key points which outperform most of other features in terms of accuracy. However, the speed of SIFT is a significant challenge for mobile devices. The Gaussian convolutions are too computationally intensive for most of today's mobile devices. Meanwhile some recent results of efficient and robust feature (point) extraction are built from simple elements such as box filters and random pixel pairs [6,7].

A random pair of pixels have a relatively stable relationship in that one pixel is brighter than the other. An example of a random pair is shown in Figure 2 where pixel A is brighter than pixel B. The advantage of using pixel pairs is that their relative brightness is not affected by environmental lighting variations. Since the same relationship may also occur in general scenes, we select the pairs that appear more frequently in the inliers (currency images) and less frequently in the outliers (non-currency images).

Our feature set consists of a large set of binary values of pairwise intensity comparisons. For a w*h image, there are as many as $\binom{w \times h}{2}$ different pairs of pixels to compare. Our goal aims to find those pairs that uniquely define the object of interest. We achieve this goal with a learning approach. Attempting to recognize a twenty dollar bill, we first perform background subtraction and normalization, and collect a set of positive samples and an even larger set of negative samples, Figure 3. We train our recognizer by selecting discriminating intensity pairs. For each pair of pixels (i,j), we define $P^+(i,j)$ to be the number of positive samples with greater intensity at pixel i than at pixel j, similarly, define $P^-(i,j)$ to be the number of negative samples with greater intensity at pixel i than at pixel j. Our goal will then be to choose pairs (i,j) to maximize $P^+(i,j)$ and minimize $P^-(i,j)$. One naive way to achieve this goal is to maximize $\frac{P^+(i,j)}{P^-(i,j)}$, but this will not work because the large collection of negative samples make $P^-(i,j)$ random. Nevertheless, numerous pairs satisfy all the positive samples, among which we would like to choose the most distinct ones. Although the number of hits of pair (i,j) in the negative samples cannot help us judge whether the choice of (i,j) is good, it helps to measure the distance from the closest negative sample to the positive sample. As shown in Figure 4, we will maximize the margin between positive samples and the closest negative samples by scoring positive and negative samples. A higher score indicates a higher probability of an inlier and lower score for outliers. After training, we can use a threshold on scores to classify the pattern. Using this criteria, we develop the following algorithm. Assign an initial score of zero for all negative samples and keep a pointer q that always points to the highest score among all negative samples. This can be done efficiently using a heap structure. (1) Generate a random pair (i,j).

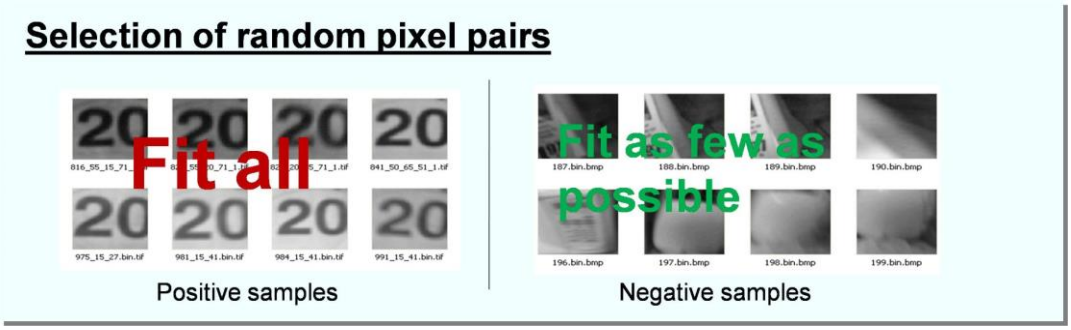

Fig. 3 Positive and negative samples

(2) If (i,j) satisfies negative sampleq, then go back to step (1).
(3) If (i,j) does not satisfy the all positive samples, go back to step (1).
(4) For all negative samples, increase its score by 1 if it satisfies pair (i,j) and modify pointer q to point to the negative sample with highest score, hence, the score of q is not increased.
(5) Go back to (1) until we have n pairs.

Using this algorithm, we collect n discriminating pairs that represent the object and the gap G between positive and negative samples will be:

$$G = \min\{P^+(i,j)\} - \max\{P^-(i,j)\} = n - P^-(i,j)$$

$P^+(i,j)$ and $P^-(i,j)$ stand for the highest score of positive and negative samples respectively. During each round, all positive samples get increased by 1, while the closet negative sample does not. The gap G between positive and negative samples is therefore enlarged. In the recognition phase, we will use this score to judge whether an object is recognized or rejected, and our threshold will be placed in the middle of the gap. Ideally we would like to see all positive samples with high scores and all negative samples with low scores. However, there might be a negative sample that is similar to the inliers, and our algorithm can spot the most confusing outliers and establish a boundary between the inliers and those outliers. The accuracy of this detection algorithm is further enhanced using the Ada-boost[13] algorithm.

### 3.2. Initial Design

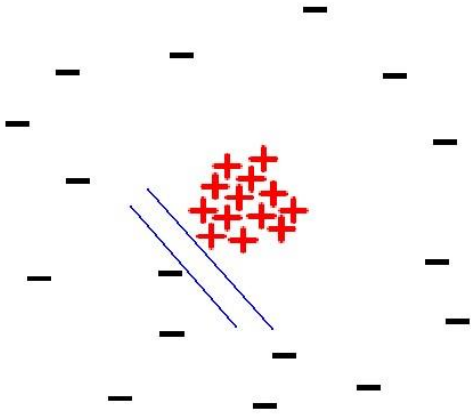

Fig. 4 Maximize margin between positive and negative samples

In order to detect and recognize a bill, we first binarize the image and remove irrelevant background. Black pixels touching the boundary of the image are regarded as backgrounds since the bill always has a white boundary along the edge. After removing the background some noise might still exists. We further refine the location of a bill by running a breadth-first-search (BFS) from the image's center to remove the remaining noise. The complexity of this step is linear in the number of pixels in the image and after processing we know the exact position of the feature area. We then normalize extracted area to a rectangle with an aspect ratio of 4:1 for recognition.

We collected 1000 samples of captured currencies of each side of the most common U.S. bills. Each has four sides, two front and two back. We also collected 10000 samples of general scenes which are not currency. For each side of a given bill, we use Ada-boost[13] to train a strong classifier from a set of weak classifiers. The weak classifiers must be computationally efficient because hundreds of them must be computed in less than 0.1 second.

We define a weak classifier using 32 random pairs of pixels in the image.A weak classifier will provide a positive result if more than 20 pairs are satisfied and negative otherwise. 10 weak classifiers selected using Ada-boost form a strong classifier that identifies a bill as long as it appears in the image. To recognize a bill we only need 32x10=320 pair-wise comparisons of pixels. Our system is trained to read $1,$5,$10,$20,$50 and $100 bills and can process 10 frames/second on a Windows Mobile (iMate Jamin) phone at a false positive rate $<10^{-4}$. It should be pointed out that this framework is general so that new notes (e.g. $2) can be easily added to the system.

### 3.3. Revised Design

Although the initial design of the currency reader satisfies our primary requirements of real time recognition and has a high accuracy, it could be further improved after an experimental study of its practical use. Users with visual disabilities identified two major disadvantages of the initial design. First, it required the coverage of the entire right hand side of the bill, i.e. the upper right and bottom right side of the bill must be captured at the same time. However, it may be difficult to accomplish such coverage without a great deal of practice. Second, users with visual disabilities like to fold the bills in different ways to distinguish among denominations, but folding can change the shape of the right hand side of a bill and may disturb the recognition.

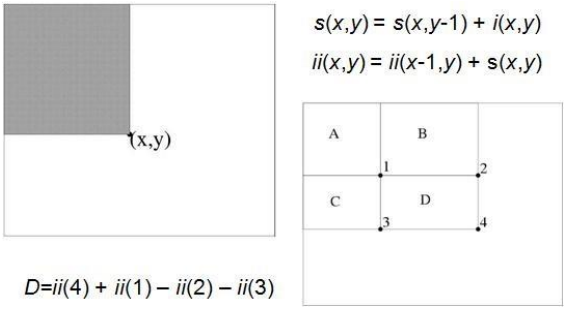


Fig. 5 Integral Image

This suggests the use of a smaller feature area for recognition because it is easier to capture and less likely to be disturbed by folding. We have refined our currency reader to identify a feature area with the number denomination as shown in Figure6. Feature areas are first detected using a fast pre-classifier and then identified using a strong classifier based on random local pixel pairs, as described in Section 3.1.

### 3.4. Pre-classification and Filtering

To detect an object in an image, an exhaustive search is usually inefficient because most of the areas in the image do not contain the object in which we are interested. A pre-classification which filters these areas is therefore important and can speed the detection by ten times or more. In our research we found that a box filter computed using an integral image is very efficient and can be applied to mobile devices. In an integral image, at each pixel the value is the sum of all pixels above and to the left of the current position. The sum of the pixels within any rectangle can be computed in four table lookup operations on the integral image in constant time as shown in Figure 5. If we replace the original image with an image with the squared gray scale value at each pixel, we can then compute the standard deviation (second order moment) within any rectangle in O(1) time. Any order of moment can be computed in O(1) time using an integral image. Both the Viola-Jones face detector[9] and SURF[7] benefit from the speedup of the box filter and integral image.

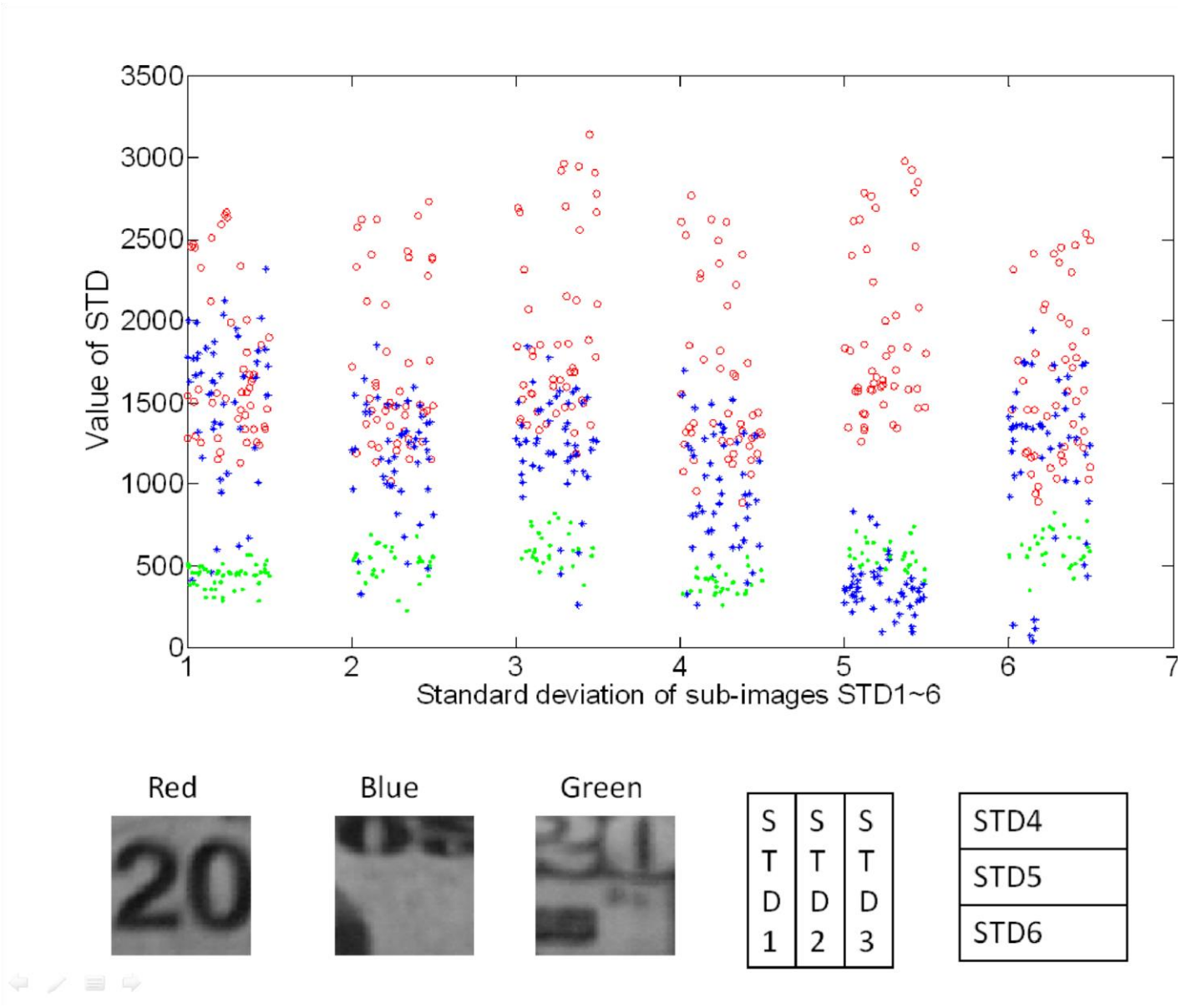


Fig. 6 Standard deviation of 6 sub-images at 3 corners of a 20 dollar bill

We found that the standard deviation (STD) in the sub-image of an object is relatively stable and combination of STDs can be used as a cue to filter non-interest image areas. In Figure 6, we divide the feature area of a twenty dollar bill into 6 boxes (3 vertical and 3 horizontal). The STD in each sub-window falls in a relatively stable range and we search only within these ranges for the potential corner patterns to recognize. In each sub-window an STD range may span at most 50% (red) or even less

(blue) of possible STD. Assuming the STD in each sub-window is independent and is equally distributed in an arbitrary scene, the box filter can eliminate $1-(1/2)^6$=98.4% of the computation by discarding low probability regions. In our experiment we found the pre-classification can speed the algorithm by 20 times on a camera phone.

## 4. Results and Contributions

### 4.1 User Evaluation

To meet the requirements of users with visual disabilities, we pay special attention to the details of the user interface. Every operation of the software is guided by a voice message. It requires two key presses to activate the camera to prevent accidental activation. The software automatically exits after being idle for two minutes to save battery power. The user has the option of "force" recognition of a bill by pressing the center button. The software will search for additional scales and positions for the feature area in "forced" recognition.

We have performed user evaluation with the system of the refined design. Ten blind users were asked to identify four bills using the camera phone currency reader. Each user was given a brief two minute introduction on how to use the device and the software, they were asked to continue until all four bills are recognized. The total time (including entering and exiting the program) was recorded to measure the usability of the software. On average, users recognized a bill in 21.3 seconds, as shown in Figure 7. Our currency reader raised no false positive during the experiment.

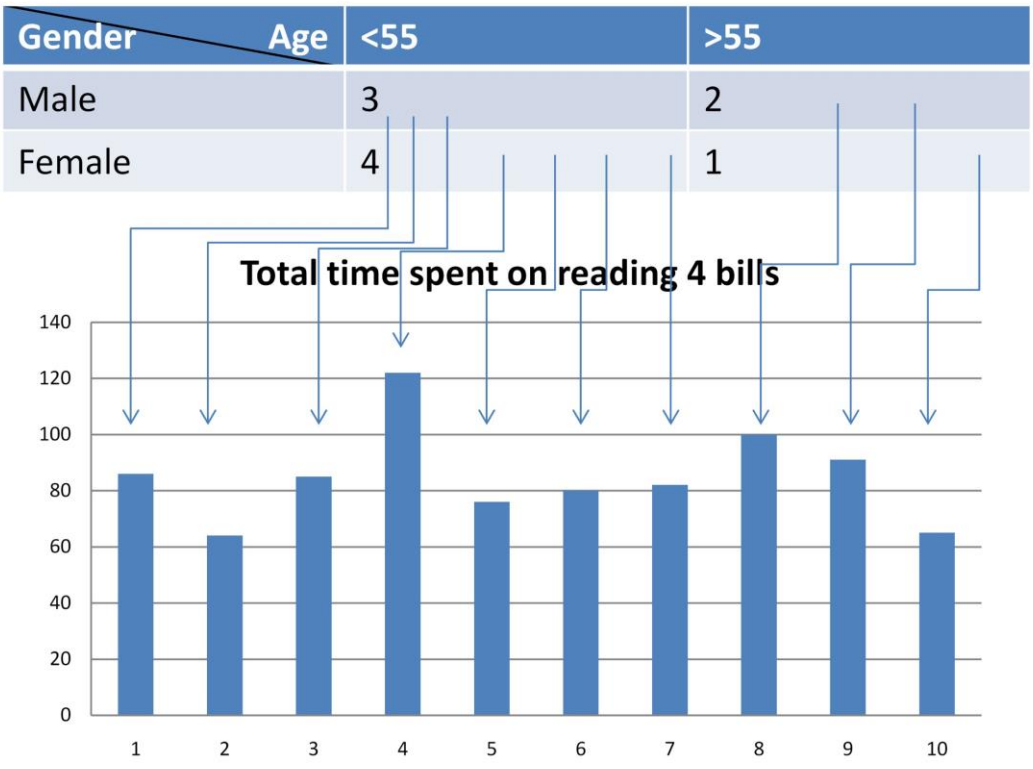


Fig. 7 User evaluation of mobile currency read

4.2 Contribution

In this paper we have presented an efficient pattern recognition algorithm that performs real time recognition on camera-enabled mobile devices. Using this algorithm we developed a currency reader for the blind users to recognize U.S. paper dollars using camera phones. We extracted illumination invariant features based on random pixel pairs, trained our currency reader using Ada-boost framework and implemented an efficient recognizer using box-filters for pre-classification on camera phones. Our currency reader processes 10 frames/second and achieves a false positive rate $<10^{-4}$. More importantly we have performed user study with blind users and received promising result. It should be pointed out that our framework is general so that new notes (e.g. the new $5 printed in 2007) can be easily added to the system.

## 5.References


1. C.T. Baker Massof, R.W. Hsu and F.H. Barnett. Visual Disability Variables. I,II: The Importance and Difficulty of Activity Goals for a Sample of Low-Vision Patients.Archives of Physical Medicine and Rehabilitation, 86(5):946-953, 2005.
2. http://www.acb.org/press-releases/final-edit-paper-currency-ruling-080520.html
3. http://www.knfbreader.com
4. S. Krishna, G. Little, J. Black, and S. Panchanathan. A wearable face recognition system for individuals with visual impairments. Proceedings of the 7th internationalACM SIGACCESS conference on Computers and accessibility, pages 106-113, 2005.
5. D.G. Lowe. Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision, 60(2):91-110, 2004.
6. 2005.6. M. zuysal, P. Fua, V. Lepetit. Fast keypoint recognition in ten lines of code. Computer Vision and Pattern Recognition, 2007. CVPR07. IEEE Conference on, pages 1-8, 2007.h;8, 2007.
7. H. Bay, T. Tuytelaars, and L. Van Gool. SURF: Speeded up robust features. Proceedings of Ninth European Conference on Computer Vision, Graz, May 7, 13:404-417,2006.
8. C. Cortes and V. Vapnik. Support-vector networks. Machine Learning, 20(3):273-297, 1995.
9. P. Viola and M. Jones. Robust real-time face detection. International Jounrnal on Computer Vision, 57(2):137-154, 2004.
10. Tim Kindberg, Mirjana Spasojevic, Rowanne Fleck, and Abigail Sellen. The ubiquitous camera: An in-depth study of camera phone use. IEEE Pervasive Computing,4(2):42-50, 2005
11. T. Ojala, M. Pietikainen, and T. Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Transactions onPattern Analysis and Machine Intelligence, 24(7):971-987, 2002.
12. https://www.fbo.gov/index?s=opportunity&mode=form&id=2c92455a3397558700aed0d3161dceb6&tab=core&_cview=0&cck=1&au=&ck=
13. Y. Freund and R.E. Schapire. A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. Journal of Computer and System Sciences,55(1):119-139, 1997.
14. A.K. Karlson, B.B. Bederson, and J. Contreras-Vidal. Understanding One-Handed Use of Mobile Devices. Handbook of Research on User Interface Design andEvaluation for Mobile Technology, 2008.