Hindawi Publishing Corporation
EURASIP Journal on Image and Video Processing
Volume 2007, Article ID 64295, 11 pages
doi:10.1155/2007/64295

Research Article

A Multifunctional Reading Assistant for the Visually Impaired

Céline Mancas-Thillou,[1] Silvio Ferreira,[1] Jonathan Demeyer,[1] Christophe Minetti,[2] and Bernard Gosselin[1]

[1] *Circuit Theory and Signal Processing Laboratory, Faculty of Engineering of Mons, 7000 Mons, Belgium*
[2] *Microgravity Research Center, The Free University of Brussels, 1050 Brussels, Belgium*

Received 15 January 2007; Revised 2 May 2007; Accepted 3 September 2007

Recommended by Dimitrios Tzovaras

In the growing market of camera phones, new applications for the visually impaired are nowadays being developed thanks to the increasing capabilities of these equipments. The need to access to text is of primary importance for those people in a society driven by information. To meet this need, our project objective was to develop a multifunctional reading assistant for blind community. The main functionality is the recognition of text in mobile situations but the system can also deal with several specific recognition requests such as banknotes or objects through labels. In this paper, the major challenge is to fully meet user requirements taking into account their disability and some limitations of hardware such as poor resolution, blur, and uneven lighting. For these applications, it is necessary to take a satisfactory picture, which may be challenging for some users. Hence, this point has also been considered by proposing a training tutorial based on image processing methods as well. Developed in a user-centered design, text reading applications are described along with detailed results performed on databases mostly acquired by visually impaired users.

1. INTRODUCTION

A broad range of new applications and opportunities are emerging as wireless communication, mobile devices, and camera technologies are becoming widely available and acceptable. One of these new research areas in the field of artificial intelligence is camera-based text recognition. This image processing domain and its related applications may directly concern the community of visually impaired people. Textual information is everywhere in our daily life and having access to it is essential for the blind to improve their autonomy. Some technical solutions combining a scanner and a computer already exist: these systems scan documents, recognize each textual part of the image, and vocally synthesize the result of the recognition step. They have proven their efficiency with paper documents but present the drawbacks of being limited to home use and exclusively designed for flat and mostly black and white documents.

In this paper, we aim at describing the development of an innovative device, which extends this key functionality to mobile situations. Our system uses common camera phone hardware to take textual information, perform optical character recognition (OCR), and provide audio feedback. The market of PDAs, smartphones, and more recently PDA phones has grown considerably during the last few years. The main benefit to use this hardware is to combine small-size, lightweight, computational resources and low cost. However, we have to allow for numerous constraints to produce an efficient system. A PDA-based reading system does not only share common challenges that traditional OCR systems meet, but also particular issues. Commercial OCRs perform well on "clean" documents, but they fail under unconstrained conditions, or need the user to select the type of documents, for example forms or letters. In addition, camera-based text recognition encompasses several challenging degradations:

(i) *image deterioration*: solutions to the poor resolution and without-auto-focus sensors, image stabilization, blur or variable lighting conditions need to be found;
(ii) *low computational resources*: the use of a mobile device such as a PDA limits the processing time and the memory resources. This adds optimization issues in order to achieve an acceptable runtime.

Moreover, these issues are even more highlighted when the main objective is to fulfill the visually impaired' requirements: they may take out of field or with strong perspective images, sometimes blurry or in night conditions. A user-centered design in close relationship with blind people [1] has been done to develop algorithms *with in situ* images.

Around the central application, which is natural scene (NS) text recognition, several applications have been developed such as Euro banknotes recognition, object recognition using visual tags, and color recognition. To help the visually impaired acquire satisfying pictures, a tutorial using a test pattern has also been added.

This paper will focus more on image processing integrated into our prototype and is organized as follows. Section 2 will deal with state-of-the-art of camera-based text understanding and commercial products related to our system. In Section 3, the core of the paper, an automatic text reading system, will be explained. Further, in Section 4, the prototype and the other image-driven functionalities will be described. We will present in Section 5 detailed results in terms of recognition rates and comparisons with commercial OCR. Finally, we will conclude this paper and give perspectives in Section 6.

2. STATE-OF-THE-ART

Up to now and as far as we know, no commercial product shares exactly the same specifications of our prototype, which may be explained by the challenging issues. Nevertheless, several devices share common objectives. First, these products are described and then, applications with analogous algorithms are discussed. We compare the different algorithmic approaches and we highlight the novelty of our method.

2.1. Text reader for the blind

The K-NFB Reader [2] is the most comparable device in terms of functions and technical approach. Combining a digital camera with a personal data assistant, this technical aid puts character recognition software with text-to-speech technology in an embedded environment. The system is designed to the unique task of portable reading machine. Its main drawback is the association of two digital components (a PDA and a separate camera, linked together in an electronic way) which increases price but offers high resolution images (up to 5 megapixels). By using an embedded camera in a PDA phone, our system processes only 1.3 megapixels images. Moreover, this product is also not multifunctional as it does not integrate any other specific tools for blind or visually impaired users. In terms of performance, the K-NFB Reader has a high level of accuracy with basic types of document. It performs well with papers having mixed sizes and fonts. On the other hand, this reader has a great deal of difficulty in the area of documents with colors and images and results are mitigated when trying to recognize product packages or signs. The AdvantEdge Reader [3] is the second portable device able to scan and read documents. It also consists of a merging of two components, a handheld micro computer (SmallTalk using Windows XP) enhanced with a screen reading software and a portable scanner (Visionner). The aim of mobility is partially reached and only flat documents may be considered. Their related problems are thus completely different from ours. Figure 1 shows the portability of the similar products compared to our prototype.

(a) (b) (c)

Figure 1: (a) AdvantEdge reader, (b) K-NFB reader, (c) our prototype.

This comparison shows that our concept is novel as all other current solutions use two or more linked machines to recognize text in mobile conditions. Our choice of hardware leads to the most ambitious and complex challenge due to the poor quality and the wide diversity of the images to process in comparison with the images taken by the existing portable solutions.

2.2. Natural scene text reading algorithms

Automatic sign translation for foreigners is one of the closest topics in terms of algorithms. Zhang et al. [4] used an approach which takes advantage of the users by selecting an area of interest in the image. The selected part of the image is then recognized and translated, with the translation displayed on a wearable screen or synthesized in an audio message. Their algorithmic approach efficiently embeds multiresolution, adaptive search in a hierarchical framework with different emphases at each layer. They also introduced an intensity-based OCR method by using local Gabor features and linear discriminant analysis for selection and classification of features. Nevertheless, a user intervention is needed which is not possible for blind people.

Another technology using related algorithms is license plate recognition, as shown in Figure 2. This field encompasses various security and traffic applications, such as access-control system or traffic counting. Various methods were published based on color objects [5] or edges assuming that characters embossed on license plates contrast with their background [6]. In this case, textual areas are known a priori and more information is available to reach higher results such as approximate location on a car, well-contrasted and separated characters, constrained acquisition, and so on.

In terms of algorithms, text understanding systems include three main topics: text detection, text extraction, and text recognition. About automatic text detection, the existing methods can broadly be classified as edge [7, 8], color [9, 10], or texture-based [11, 12]. Edge-based techniques use edge information in order to characterize text areas. Edges of text symbols are typically stronger than those of noise or background areas. The use of color information enables to segment the image into connected components of uniform color. The main drawbacks of this approach consist of the high color processing time and the high sensibility to uneven lighting and sensor noise. Texture-based techniques attempt to capture some textural aspects of text. This approach is frequently used in applications in which no a priori information is provided about the document layout or the text

to recognize. That is why our method is based on this latest while characterizing the texture of text by using edge information. We aim at realizing an optimal compromise between two global approaches.

A text extraction system usually assumes that text is the major input contributor, but also has to be robust against variations in detected text areas. Text extraction is a critical and essential step as it sets up the quality of the final recognition result. It aims at segmenting text from background. A very efficient text extraction method could enable the use of commercial OCR without any other modifications. Due to the recent launch of the NS text understanding field, initial works focused on text detection and localization and the first NS text extraction algorithms were computed on clean backgrounds in the gray-scale domain. In this case, all thresholding-based methods have been experienced and are detailed in the excellent survey of Sezgin and Sankur [13]. Following that, more complex backgrounds were handled using color information for usual natural scenes. Identical binarization methods were at first used on each color channel of a predefined color space without real efficiency for complex backgrounds, and then more sophisticated approaches using 3D color information, such as clustering, were considered. Several papers deal with color segmentation by using particular or hybrid color spaces as Abadpour and Kasaei [14] who used a PCA-based fast segmentation method for color spotting. Garcia and Apostolidis [15] exploited a character enhancement based on several frames of video and a k-means clustering. They obtained best nonquantified results with hue-saturation-value color space. Chen [16] merged text pixels together using a model-based clustering solved thanks to the expectation-maximization algorithm. In order to add spatial information, he used Markov random field, which is really computationally demanding. In next the sections, we propose two methods for binarization: a straightforward one based on luminance value and a color-based one using unsupervised clustering, detailed in fair depth in [17].

The main originalities of this paper are related to the prototype we designed and several points need to be highlighted.

(i) We develop a fully automatic detection system without any human intervention (due to the use by blind users) but also which work with a large diversity of textual occurrences (document papers, brochures, signs, etc.). Indeed most of the previous text detection algorithms are fitted to operate in a particular context (only for a form or only for natural scenes) and fail in other situations.

(ii) We use dedicated algorithms for each single step to reach a good compromise in terms of quality (recognition rates and so on) and time and memory efficiency. Algorithms based on human visual system are exploited at several positions in the main chain for their efficiency and versatility faced to the large diversity of images to handle.

(iii) Moreover, as the whole chain has to work without any user intervention, a compromise is done between text detection and recognition, in order to validate textual candidates at several occasions.

(a) (b)

Figure 2: (a) A license plate recognition system and (b) a tourist assistant interface (from Zhang et al. [4]).

3. AUTOMATIC TEXT READING

3.1. Text detection

The first step of the automatic text recognition algorithm is the detection and the localization of the text regions present in the image. The mainstream of text regions is characterized by the following features [18]:

(i) characters contrast with their background as they are designed to be read easily;

(ii) characters appear in clusters at a limited distance around a virtual line. Usually, the orientation of these virtual lines is horizontal since that is the natural writing direction for Latin languages.

In our approach, the image consists of several different types of textured regions, one of which results from the textual content in the image. Thus, we pose our problem locating text in images as a texture discrimination issue. Text region must be firstly characterized and clustered. After these steps, a validation module is applied during the identification of paragraphs and columns into the text regions. The document layout can then be estimated and we can finally define a reading order to the validated text bounding boxes as described in Figure 3.

Our method for texture characterization is based on edges density measures. Two features are designed to identify text paragraphs. The image is firstly processed through two Sobel filters. This configuration of filters is a compromise in order to detect nonhorizontal text at different fonts. A multiscale local averaging is then applied to take into account various character scales (local neighborhood of 12 and 36 pixels). Finally to simulate human texture perception, some form of nonlinearity is desirable [19]. Nonlinearity is introduced in each filtered image by applying the following transformation Y on each pixel value x [20]:

$$Y(x) = \tanh(a \cdot x) = \frac{1 - \exp^{-2ax}}{1 + \exp^{-2ax}}. \quad (1)$$

For $a = 0.25$, this function is similar to a thresholding function like a sigmoid.

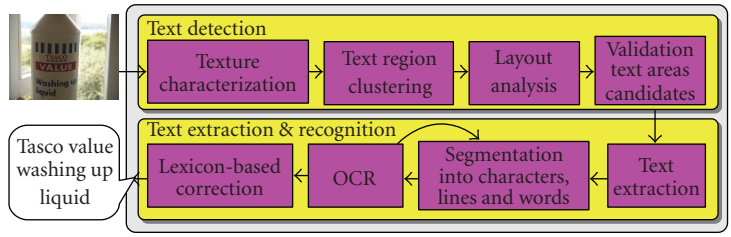


Figure 3: Description scheme of our automatic text reading.

The two outputs of the texture characterization are used as features for the clustering step. In order to reduce computation time, we apply the standard k-means clustering to a reduced number of pixels and a minimum distance classification is used to categorize all surrounding nonclustered pixels. Empirically, the number of clusters was set to three, value that works well with all test images taken by blind users. The cluster whose center is closest to the origin of feature vector space is labeled as background while the furthest one is labeled as text.

After this step, the document layout analysis may begin. An iterative cut and merge process is applied to separate and distinguish columns and paragraphs by using geometrical rules about the contour and the position of each text bounding box. We try to detect text regions which share common vertical or horizontal alignments. At the same time, several kinds of false detected text are removed using adapted validation rules:

(i) fill ratio of pixels classified as text in the bounding box larger than 0.25,
(ii) X/Y dimension ratio of the bounding box between 0.2 and 15 (for small bounding boxes) and between 0.25 and 10 (for larger ones),
(iii) area size of the text bounding box larger than 1000 pixels (the minimal area size to recognize a small word).

When columns and paragraphs are detected, the reading order may be finally estimated.

3.2. Text segmentation and recognition

Once text is detected in one or several areas I^D, characters need to be extracted. Depending on image types to handle, we developed two different text extraction techniques, based either on luminance or color images. For the first one, a contrast enhancement is applied to circumvent lighting effects of natural scenes. The contrast enhancement [21] is issued from visual system properties and more particularly on retina features and leads to I^D_{enhanced}:

$$I^D_{\text{enhanced}} = I^D * H_{\text{gangON}} - \left(I^D * H_{\text{gangOFF}}\right) * H_{\text{amac}} \tag{2}$$

with

$$H_{\text{gangON}} = \begin{pmatrix} -1 & -1 & -1 & -1 & -1 \\ -1 & 2 & 2 & 2 & -1 \\ -1 & 2 & 3 & 2 & -1 \\ -1 & 2 & 2 & 2 & -1 \\ -1 & -1 & -1 & -1 & -1 \end{pmatrix},$$

$$H_{\text{gangOFF}} = \begin{pmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & -1 & -2 & -1 & 1 \\ 1 & -2 & -4 & -2 & 1 \\ 1 & -1 & -2 & -1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{pmatrix},$$

$$H_{\text{amac}} = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 2 & 2 & 2 \\ 1 & 2 & 3 & 3 \\ 1 & 2 & 2 & 2 \\ 1 & 1 & 1 & 1 \end{pmatrix}. \tag{3}$$

These three previous filters assess eye retina behavior and correspond to the action of ON and OFF ganglion cells ($H_{\text{gangON}}, H_{\text{gangOFF}}$) and of the retina amacrine cells (H_{amac}). The output is a band-pass contrast enhancement filter which is more robust to noise than most of the simple enhancement filters. Meaningful structures within the images are better enhanced than by using classical high-pass filtering which provides more flexibility to this method. Based on this robust contrast enhancement, a global thresholding is then applied, leading to $I_{\text{binarized}}$:

$$I_{\text{binarized}} = \left(I^D_{\text{enhanced}} > \text{Otsu}_{\text{threshold}}\right) \tag{4}$$

with $\text{Otsu}_{\text{threshold}}$ determined by the popular Otsu algorithm [22].

For the second case, we exploit color information to handle more complex backgrounds and varying colors inside textual areas. First, a color reduction is applied. Considering properties of human vision, there is a large amount of redundancy in the 24-bit RGB representation of color images. We decided to represent each of the RGB channels with only 4 bits, which introduce very few perceptible visual degradation. Hence the dimensionality of the color space $\mathcal{C}$ is $16 \times 16 \times 16$ and it represents the maximum number of colors. Following this initial step, we use the k-means clustering with a fixed number of clusters equal to 3 to segment $\mathcal{C}$ into three colored regions. The three dominant colors (C_1, C_2, C_3) are extracted based on the centroid value of each cluster. Finally, each pixel in the image receives the value of one of these colors depending on the cluster it has been assigned to. Three clusters are sufficient as experienced on the complex and public ICDAR 2003 database [23], which is large enough to be applicable on other camera-based images, when text areas are already detected. Among the three clusters, one represents obviously background. Only

two pictures left which correspond depending on the initial image to either two foreground pictures or one foreground picture and one noise picture. We may consider combining them depending on location and color distance between the two representative colors as described in [17]. More complex but heavier text extraction algorithms have been developed but we do not use them as we wish to keep a good compromise between computation time and final results. This barrier will disappear soon as hardware advances in leaps and bounds in terms of sensors, memory, and so on.

In order to use straightforward segmentation and recognition, a fast alignment step is performed at this point. Based on the closest bounding box of the binarized textual area and successive rotations in a given direction (depending on initial slope), the text is aligned by considering the least high bounding box. As the alignment is performed, the bounding box is now more accurate. Based on these considerations and properties of connected components, the appropriate number of lines N_l is computed. In order to handle small variations and to be more versatile, an N_l-means algorithm is performed by using y-coordinate of each connected component as detailed in [1]. Word and character segmentation are iteratively performed in a feedback-based mechanism as shown in Figure 3. First, character segmentation is done by processing individual connected components and followed by the word segmentation, which is performed on intercharacter distance. An additional iteration is performed if recognition rates are too low and a Caliper distance is applied to possibly segment joined characters and to recognize them better afterwards. The Caliper algorithm computes distances between topmost and bottommost pixels of each column of a component and enables to easily identify junctions between characters.

About character recognition, we use our in-house OCR, tuned in this context to recognize 36 alphanumeric classes without considering accent, punctuation and capital letters. To detail more, we use a multilayer perceptron fed with a 63-feature vector where features are mainly geometrical and composed of characters contours (exterior and interior ones) and Tchebychev moments [17]. The neural network has 1 hidden layer of 120 neurons, and trained on more than 40 000 characters. They have been extracted on a separate training set, but acquired by blind users as well in realistic conditions. Even a robust OCR is error-prone in a lower percentage and a post-processing correction solution is necessary. Main ways of correcting pattern recognition errors are either combination of classifiers to statistically decrease errors by adding information from different computations or by exploiting linguistic information in the special case of character recognition. For this purpose, we use a dictionary-based correction by exploiting finite state machines to encode easily and efficiently a given dictionary, a static confusion list dependent of OCR and a dynamic confusion list dependent of the image itself. As this extension may be considered out of scope, more details may be found in [24].

Our whole automatic text reading has been integrated in our prototype and also used for other applications, as described in Section 4.

Figure 4: User interface for blind people.

4. MULTIFUNCTIONAL ASSISTANT

4.1. System overview

The device is a standard personal digital assistant with phone capabilities (PDA phone). Hardware has not been modified; only the user interface is tuned for the blind. Adapting a product dedicated to general audience rather than developing a specific electronic machine allows us to profit from the fast progress in embedded device technologies while keeping a low cost. The menu is composed of the multidirectional pad and a simulated numerical pad on the touch screen (from 0 to 9 with $*$ and #). For the blind, those simulated buttons are quite small in order to limit wrongly pressed keys while taking their marks. A layer has been put on the screen to change the touch while pressing a button, as shown in Figure 4.

The output comes only from a synthetic voice[1] which helps the user to navigate through the menu or provide the results of a task. An important point to mention is the automatic audio feedback for each user action, in order to navigate and guide properly.

One of the key features of the device is that it embeds many applications and fills needs which normally require several devices. The program has also been designed to easily integrate new functionalities (Figure 5). This flexibility enables us to offer a modular version of our product which fits the needs of everyone. Hence, users can choose applications according to their level of vision but also to their wills.

Additionally to the image processing applications described in this section, the system also integrates dedicated applications like the ability to listen to DAISY[2] books, talked newspapers or telephony services.

4.2. Object recognition

In the framework of object recognition (Figure 6), we chose to stick a dedicated label onto similar-by-touch objects.

Blind people may fail to identify tactically identical objects such as milk/orange bricks, bottles, medicine boxes. In

[1] We have used the Acapela Mobility HQ TTS which produces natural and pleasant-sounding voice.

[2] A standard format for talking books designed for blind users [25].

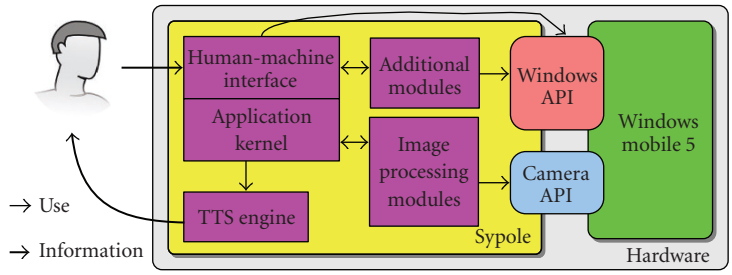


Figure 5: A block diagram of the architecture and design of our system.

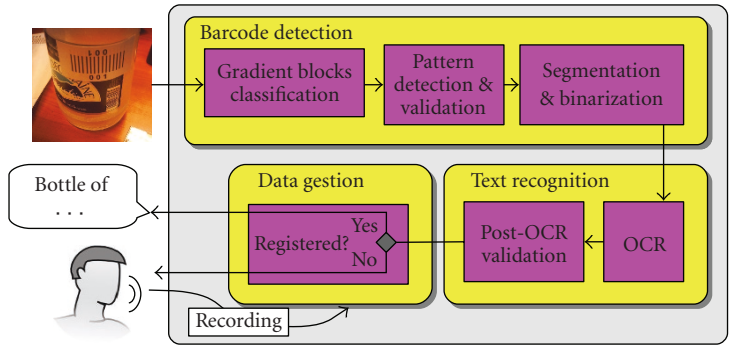


Figure 6: Description scheme of our object recognition system.

order to remedy this need we chose a solution based on specific labels to put onto the object. This is the best solution for several reasons. Text recognition of product packages may lead to erroneous results due to artistic display and very complex backgrounds. A solution using Braille stickers is useful and efficient only for people knowing this language and is limited in size for the description.

Based on these considerations, the solution of a dedicated label, superimposed on objects to be tactically found, was chosen. Once the barcode is stuck, the user takes a picture of it. The system recognizes the barcode as a new code and asks the user to record a message describing it (such as "orange juice bought Friday, 10th," e.g.). During the further use, the user will take a snapshot of the object and if the system recognizes the tag, it plays the audio message previously recorded. This application has been duplicated by blind users as a memo. They stuck the label onto a fridge and recorded audio messages every night as a reminder for the following morning!

Contrarily to the generic text recognition system detailed in Section 3, we can use here a priori information about the tag and recognize it easier. Figure 7 illustrates the pattern of the tag similar to a classical barcode (designed with a bigger size to take into account the bad quality of the image sensors). Two numbered areas have been symmetrically added in order to increase final results in case of out of field images. Moreover, as only these areas are processed, it enables not only to circumvent image processing failures but also to provide free-rotation pictures. The global idea to localize the tag in the image is that this region of interest (ROI) is characterized by gradient vectors strong in magnitude and sharing the same direction. First off, the energy gradient image is computed in magnitude and direction. We then use a technique of classification by blocks. The whole image is divided into small blocks of 8×8 pixels. Gradient magnitudes of pixels are summed to estimate if the block contains enough gradient energy and if the pixels share a common gradient direction. We categorize these directional blocks into four main directions ($0°, 45°, 90°$ and $135°$). An example of this classification result is shown in Figure 7(b). The detection of the tag can now be operated by analyzing each main direction. Blocks of the same direction are clustered and candidate ROIs are selected. A validation module is then applied to verify the presence of lines into the candidate region. When the presence of minimum four lines is validated, the candidate ROI is selected. This procedure is illustrated in Figure 7(c). Limits of the barcode are redefined more precisely using the ends of these lines previously isolated. We can simultaneously estimate accurately the skew of the barcode. If required, a rotation is applied and finally we isolate both regions (if any) representing the code to be recognized by OCR.

Once the barcode numbers have been detected (once or twice depending on image quality and framing), the numbered area is analyzed. First, it is binarized by our gray-level-based thresholding described in Section 3, meaning a contrast enhancement inspired by visual properties and followed by a global thresholding. Then, connected components are computed and fed into our in-house OCR. For this application, the recognizer has been trained on a particular data set based on several pictures taken by end users and for 11 classes only, 10 digits completed by a noise class to remove spurious parts around digits. In the case of low recognition quality for

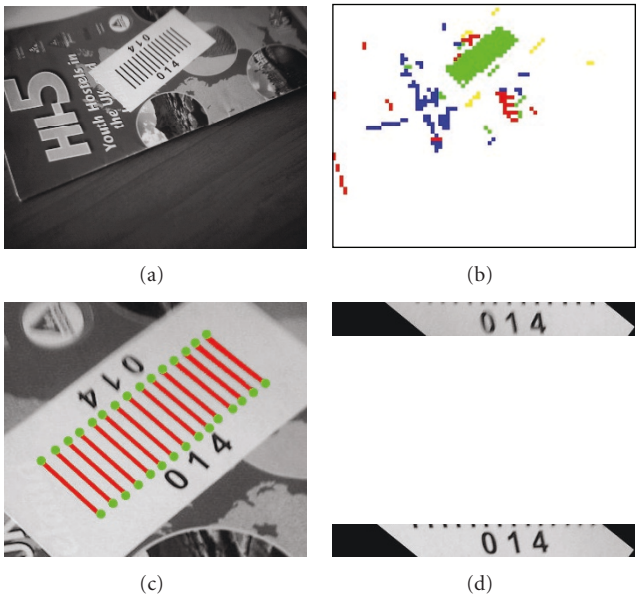


Figure 7: (a) Original image, (b) results of classification by gradient blocks, (c) validation process by detection of "lines," (d) final regions of interest.

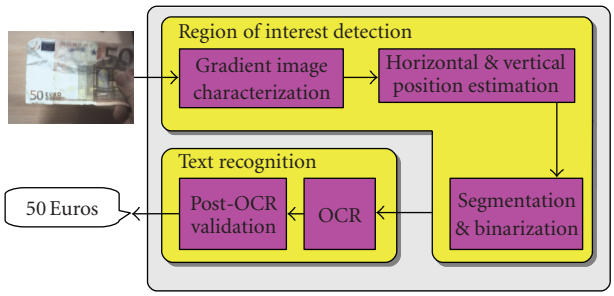


Figure 8: Description scheme of our banknote recognition system.

the first numbered area, the second one, if any, is analyzed afterwards to increase recognition rates.

4.3. Banknote recognition

This application provides a mean to blind people to verify the value of their banknotes. The user takes a picture of a banknote and after analysis and correction, the system provides an audio answer, with the value of the banknote. We pay attention here to drastically reduce false recognitions for obvious reasons. The main framework, displayed in Figure 8, is explained in this subsection. Similarly as the previous application, we use a priori information about the pattern to recognize. Indeed we have information about the position and the size of the ROI (always in the same zone for all banknotes, as displayed in Figure 9) but also about the text we have to recognize (only numbers of 5, 10, 20, 50, 100, 200, and 500 Euros). Banknote recognition could have been processed by color information or template matching for banknote images but we chose text recognition mainly for two reasons:

Figure 9: Examples of banknotes to recognize. The banknote value which is analyzed by image processing is highlighted by a red square.

(i) Sensors of embedded cameras are still poor and combined with uneven lighting effects, they lead to non-smooth colors. Moreover perturbing colors in the picture background may be present and text detection is hence more reliable.

(ii) in addition, for computation cost and memory, we chose to specialize one main chain into different applications instead of using totally different algorithms for each application.

By using one-dimensional signals (gradient image profiles) the detection algorithm scans the image firstly vertically using sliding windows and then horizontally to find the candidate regions. As the detection is turned into a one dimensional problem, this process is very fast.

Afterwards, the binarization method takes advantage of previously computed information: the gradient image. Indeed, the pattern of the text region of interest is known in this application: dark characters on bright background. The idea is to firstly estimate pixels representing the background and those representing the characters. This can be done by using the previously computed gradient pixels, which are the transition between these two states and are tagged as unknown pixels. When this first estimation is operated, we can compute a global binarization threshold T by using in the calculation only contributions from pixels classified as character and as background. We use the following formula:

$$T = \frac{m^b * nb^b + m^c * nb^c}{nb^b + nb^c} \quad (5)$$

with m^b the mean value of pixels classified as background, nb^b number of background pixels, m^c the mean value of pixels classified as character, and nb^c number of characters pixels. This method was selected for two reasons: its efficiency when the system is designed to recognize a text area having a priori information about the background and the characters colors like in this application, and its computational time, which remains very low thanks to information already computed during the previous steps.

Once the value of the banknote is binarized, a compromise between computation time and high-quality results is done until the end. Hence, the first preliminary test is to count the number N_{cc} of connected components. If N_{cc} is larger than 10, we reject this textual area. One of the main advantages is to quickly discard erroneously detected areas

by keeping a reasonable computation time. Actually, based on the low quality and the image resolution, text detection is a challenging part and assuming several areas enables to consider properly detected areas without missing them.

Following this segmentation into connected components, our home-made OCR is applied and tuned to recognize only the five classes 0, 1, 2, 5 and noise needed for this application. The noise class is useful to remove erroneous detected areas, such as the part with the word "EURO."

A simple correction rule is then applied to always provide best possible answers to end users. The application of banknote recognition has to be very efficient as the consequence may be damageable for blind people. Hence if recognition results are not values of traditional Euro banknotes, they are rejected. A second loop is then processed to handle joined characters, which may happen in extreme cases.

Based on image quality and degradations to handle, banknotes may have been acquired with perspective, blur, or uneven lighting which connects numbers of the banknote value. Hence, a Caliper distance is performed as described in Section 3 to optimally separate those characters and the same recognition and correction are then performed.

The methods previously described to recognize banknote values have been tuned to Euro banknotes (especially for the text detection part). Nevertheless, the extension to another currency is quite straightforward and may be handled easily. An all-currency recognizer has not been chosen for efficiency purposes but the code has been developed to be easily adapted.

4.4. Color recognition

This software module can be used to determine the main color of an object by taking a picture of it. Firstly, the algorithm analyzes only the central half part of the picture. Indeed, empirical tests have shown that the main color of an object is over-represented in the center of the picture as the background noise is rather present next to the edges. A first reduction of colors of the original RGB image is applied to decrease the number of colors to 512. This operation is very fast as we keep the 3 most significant bits of each color byte. The second step is a color reduction based on the color histogram. The 10 most important colors of the histogram are preserved. A merging is then applied to fuse similar colors using the Euclidian distance in the Luv color space and a fixed threshold. Finally, the most representative color of the remaining histogram is compared to a color lookup table and the system provides an audio answer with two levels of luminance (bright/dark) for each color.

4.5. Acquisition training for the blind

Taking pictures in the best conditions is the very starting point of a successful image processing chain. Indeed, most of the preprocessing chain can generally be eliminated by choosing the appropriate field of view, orientation, illumination, zoom factor, and so forth. However, this fact that seems so obvious for most of people is not natural and easy for blind people. For them, taking a picture requires training and is self-dependent of each person. In order for blind people to autonomously train themselves and develop their own marks, we have developed an imaging system for acquisition training.

(a) (b) (c) (d)

Figure 10: (a) Acquisition, (b) binarization, (c) first segmentation, (d) second segmentation.

(a) (b) (c) (d)

Figure 11: Output messages: (a) the assistant and the target are strongly nonparallel, (b) the field of view is incomplete. Moving the assistant back, (c) the picture has been taken correctly, (d) slightly rotate the assistant counterclockwise.

The underlying algorithm analyzes the structure of the target composed of nine black dots, as shown in Figure 10. After segmentation of the black dots, the relative position of each of them is analyzed and different types of defaults can be derived, such as the target position in the field of view, the global rotation of the target, perspective effects (horizontal or vertical) or illumination conditions (insufficient or saturated illuminations).

The processing chain includes four steps, as described in Figure 10. First off, a binarization of the gray-level image is performed with a global thresholding, depending on histogram distribution. Then, a first segmentation is applied. All the connected components of the binarized picture are labeled S_i. Only the square surfaces are kept in the image. Hence, surfaces S_i with a ratio $\text{Width}(S_i)/\text{Height}(S_i)$ in the range $[0.75; 1.5]$ are removed. Following that, if the number of remaining surfaces is larger than 9, we analyze the distance between the center of mass of all different surfaces. This allows to easily determining the surfaces of the target; the others are removed. Finally, we compute the angle between the lines connecting the different surfaces. On this basis, parameters like global orientation, field of view, perspective effects are derived.

The self-learning imaging system allows blind people to train themselves to take pictures. In order to progressively adapt the user to take pictures, the embedded software enables to process only one type of effects (e.g., rotation). When the user feels sufficiently confident, he may ask the software to give the dominant effect. Examples of images taken by blind people and the generated feedback are shown in Figure 11.

(a) (b)

Figure 12: Examples of NS text, difficult to recognize, either with blur or too tiny characters.

5. RESULTS

5.1. Material and databases constitution

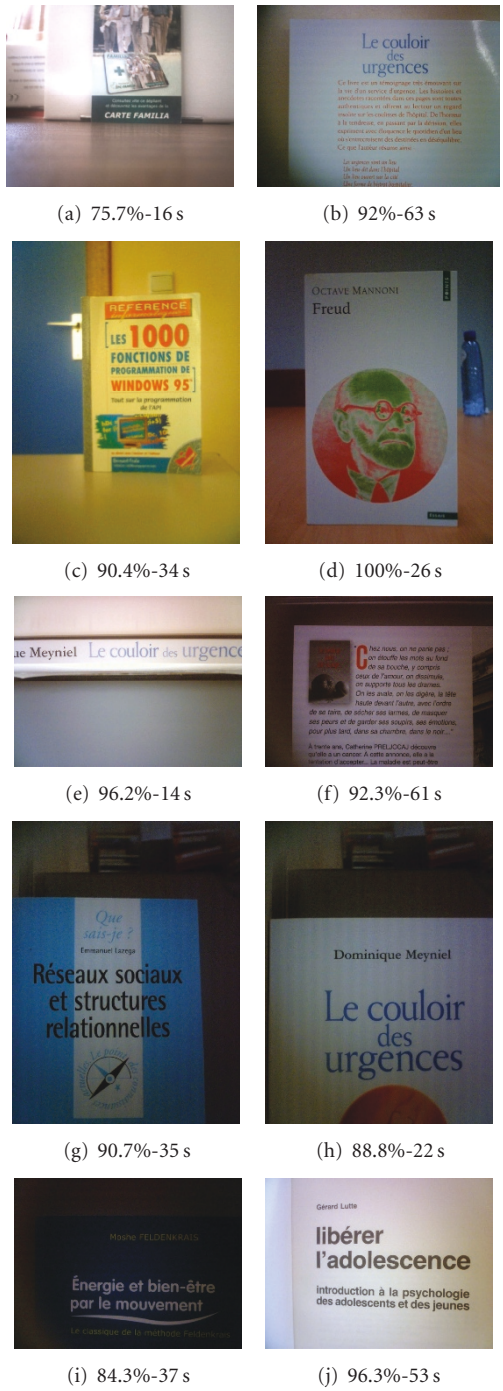
All tests have been made on a Pocket PC, with a 520 MHz Intel XScale processor. The embedded camera has a resolution of 1.3 megapixels. Images have been mainly taken by end users, meaning blind people. Distance between objects with text, tags onto objects or banknotes is from 10 to 30 cm in order to get possible readability. For comparison of some applications, a commercial OCR has been used on a PC using the same database and refers to ABBYY FineReader 8.0 Professional Edition Try&Buy.[3]

5.2. Automatic text reading results

One important point to note in this application is the difficulty to meet sensor requirements for satisfying images and blind acquisition. Due to the sensor, and numerous inherent degradations, blur, tiny characters for OCR, uneven lighting and so on, a large number of images taken during test sessions by blind users leads to no recognition at all, as the ones shown in Figure 12.

Results are detailed in Figure 13 to simultaneously show the diversity of images and corresponding recognition rates and processing time, which is dependent on text density to analyze. Runtime corresponds to detection of textual areas, alignment, binarization, segmentation into lines, words and characters, recognition and linguistic-based correction. Minimum time is 14 seconds and maximum one is 63 seconds. The code still needs to be optimized. We compare results with a commercial OCR described in Section 5.1 with no limitation in hardware and for images of Figure 13, 79.8% characters have been recognized in average against 90.7% for our system. The false positive rate (when nontext is considered as text) is lower than 2%. This result is satisfactory and very low due to a two-step validation procedure. First, the text detection system uses rejection rules based on global measures about text region candidates (bounding box, fill ratio, etc.). Moreover, the following steps of OCR and correction reject most of the false text areas by considering two additional constraints: characters must be recognized with a significant probability and words must belong to a given lexicon or be included in a line with several meaningful words.

Main failures are due to too tiny characters (less than 30 dpi), blur during acquisition, and low resolution. Much effort has to be provided in terms of versatility to handle a



(a) 75.7%-16 s (b) 92%-63 s

(c) 90.4%-34 s (d) 100%-26 s

(e) 96.2%-14 s (f) 92.3%-61 s

(g) 90.7%-35 s (h) 88.8%-22 s

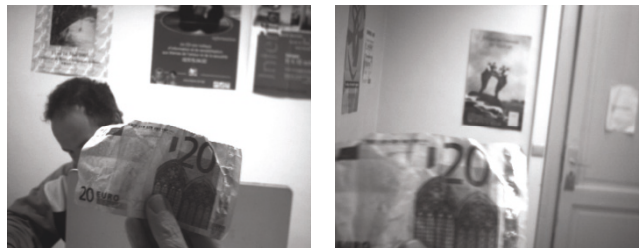
(i) 84.3%-37 s (j) 96.3%-53 s

Figure 13: Different images with their corresponding recognition rate and processing time.

[3] http://france.abbyy.com/download/?param=46440.

(a) (b)

Figure 14: Examples of dedicated barcodes onto a CD or a medicine box.



(a) (b)

Figure 15: Examples of banknotes, hard to handle and acquire properly and hence to recognize.

larger diversity of images and new ways to ensure satisfying acquisition by the visually impaired. Very soon, hardware and software will meet for commercial exploitations. Until now, word recognition rates (which lead to comprehensive word after text-to-speech algorithm) are too low to be used by blind people.

5.3. Object recognition results

About object recognition, the database includes 246 images with barcodes inside, as those displayed in Figure 14.

One of our concerns is to provide very high-quality results with very low false recognition rates, meaning that if the result has a low confidence rate, the prototype asks the user to take another snapshot. Hence, we have a recognition rate of 82.8% on the first snapshot. 17.2% of nonrecognition is divided into 15.2% of no results where a second snapshot is required and around 2% of wrong recognition. False recognition rates may be decreased even more by knowing the range of values of barcodes used by a single user, at home for example. We may choose to add this a priori information if necessary.

In the permanent concern of computation time to deliver satisfying results, fusion of both numbered areas is not considered. Actually, around 86% of recognized barcodes are reached by using only the first detected numbered area. Hence, by considering only the first numbered area, the computation time is drastically reduced in main situations. If no recognition is done, the second one, if any, may be analyzed. From database described above, a fusion process to reinforce confidence rates would create confusion in 1.2% of the cases as the first and second numbered areaS may lead to different results. It is important to note that in the 1.2% confusion, the right answer was provided by the first numbered area, which adds no errors in our method.

For results comparison, we use the commercial OCR, which completely fails without preliminary text detection. In order to fine results, we use our text detection and provide numbered areas to OCR. Error rate is 12.2% in average against our low error rate of around 2%.

The average computation time is 3.1 seconds. It corresponds to image acquisition, detection of the barcode, possible rotation, cropping of two possible numbered areas, binarization and recognition.

5.4. Banknote recognition results

For banknote recognition evaluation, the database includes 326 images as the ones shown in Figure 9. This application has to provide highly efficient results and we have only around 1% of false banknotes values after our process. This leads to a good recognition of around 84% and a second snapshot to take is necessary in around 15% cases. At this point, it is interesting to mention the difficult way for blind people to acquire satisfying images. For barcodes onto objects, a snapshot of the object has to be taken but without worrying of object orientation and position. In the case of banknotes, several ways have been experienced: put the banknote on a table (if any), hold the banknote, as properly as possible, with one hand and take the snapshot with another one, and so on. Hence, blur is a very frequent degradation leading to difficult images to handle such as the ones shown in Figure 15.

Similarly as object recognition evaluation, we compare results with the commercial OCR, which fails for all images without text detection. After providing already detected text areas, error rate drops to 13.9%. Hence, our error rate of 1% is very satisfying even if for some images, a second snapshot is required.

For this application, the average computation time is 1.2 seconds, which includes detection of the banknote value, binarization, possible segmentation into individual characters, recognition, and validation.

5.5. Color recognition results

Results are very sensitive to the quality of the image sensor and the lighting conditions. When the color is preserved into the original image, the algorithm presents a correct answer in more than 80% of cases. In situations of poor illumination or artificial lights, true colors can be altered in the original image.

6. CONCLUSION

We have presented an innovative mobile reading assistant specially designed for visually impaired people. The main application of our technical aid is text recognition in mobile situations. No assumption is done about the kind of documents or natural scene text to describe; hence this approach offers the opportunity to process a large variety of text occurrences. One limitation consists in the low quality of the images to process by using an existing camera phone that is commonly available. Nevertheless, we can already achieve acceptable results and the progress of these mobile devices in which our software may be installed is promising. As opposed to generic text recognition, we described other image processing functions like object or banknote recognition which have a priori information about the pattern to detect in the image and to identify. By adapting our algorithms in those cases, we can currently reach high recognition rates while keeping a low error rate. A key idea of our system is to be modular in the way that it can continuously integrate new image processing technologies, but also third-party technologies, such as GPS positioning or other input/output modalities. Our aim is to build the most complete and adapted talking assistant for blind users.

ACKNOWLEDGMENT

This project is called *Sypole* and is funded by Ministère de la Région wallonne in Belgium.

REFERENCES

[1] J.-P. Peters, C. Mancas-Thillou, and S. Ferreira, "Embedded reading device for blind people: a user-centred design," in *Proceedings of the 33rd Applied Imagery Pattern Recognition Workshop (AIPR '04)*, pp. 217–222, Washington, DC, USA, October 2004.

[2] "K-NFB Reader website," http://www.knfbreader.com/, May 2007.

[3] "AdvantEdge Reader website," http://www.atechcenter.net/, May 2007.

[4] J. Zhang, X. Chen, J. Yang, and A. Waibel, "A PDA-based sign translator," in *Proceedings of the 4th IEEE International Conference on Multimodal Interfaces (ICMI '02)*, pp. 217–222, Pittsburgh, Pa, USA, October 2002.

[5] E. R. Lee, P. K. Kim, and H. J. Kim, "Automatic recognition of a car license plate using color image processing," in *Proceedings of the IEEE International Conference on Image Processing (ICIP '94)*, vol. 2, pp. 301–305, Austin, Tex, USA, November 1994.

[6] S. Draghici, "A neural network based artificial vision system for licence plate recognition," *International Journal of Neural Systems*, vol. 8, no. 1, pp. 113–126, 1997.

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

[8] M. Pietikäinen and O. Okun, "Text extraction from grey scale page images by simple edge detectors," in *Proceedings of the 12th Scandinavian Conference on Image Analysis*, pp. 628–635, Bergen, Norway, June 2001.

[9] W.-Y. Chen and S.-Y. Chen, "Adaptive page segmentation for color technical journals' cover images," *Image and Vision Computing*, vol. 16, no. 12-13, pp. 855–877, 1998.

[10] Y. Zhong, K. Karu, and A. K. Jain, "Locating text in complex color images," *Pattern Recognition*, vol. 28, no. 10, pp. 1523–1535, 1995.

[11] V. Wu, R. Manmatha, and E. Riseman, "Textfinder: an automatic system to detect and recognize text inimages," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 11, pp. 1224–1229, 1999.

[12] A. K. Jain and S. Bhattacharjee, "Text segmentation using Gabor filters for automatic document processing," *Machine Vision and Applications*, vol. 5, no. 3, pp. 169–184, 1992.

[13] M. Sezgin and B. Sankur, "Survey over image thresholding techniques and quantitative performance evaluation," *Journal of Electronic Imaging*, vol. 13, no. 1, pp. 146–168, 2004.

[14] A. Abadpour and S. Kasaei, "A new parametric linear adaptive color space and its PCA-based implementation," in *Proceedings of the 9th Annual Computer Society of Iran Computer Conference (CSICC '04)*, vol. 2, pp. 125–132, Tehran, Iran, February 2004.

[15] C. Garcia and X. Apostolidis, "Text detection and segmentation in complex color images," in *Proceedings of IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '00)*, vol. 4, pp. 2326–2329, Istanbul, Turkey, June 2000.

[16] D. Chen, *Text detection and recognition in images and video sequences*, Ph.D. thesis, École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland, August 2003.

[17] C. Mancas-Thillou, *Natural scene text understanding*, Ph.D. thesis, Faculté Polytechnique de Mons, Mons, Belgium, 2007.

[18] R. Lienhart and A. Wernicke, "Localizing and segmenting text in images and videos," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 12, no. 4, pp. 256–268, 2002.

[19] D. Dunn, W. E. Higgins, and J. Wakeley, "Texture segmentation using 2-D Gabor elementary functions," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 2, pp. 130–149, 1994.

[20] A. K. Jain and F. Farrokhnia, "Unsupervised texture segmentation using Gabor filters," *Pattern Recognition*, vol. 24, no. 12, pp. 1167–1186, 1991.

[21] M. Mancas, C. Mancas-Thillou, B. Gosselin, and B. Macq, "A rarity-based visual attention map—application to texture description," in *Proceedings of IEEE International Conference on Image Processing*, pp. 445–448, Atlanta, Ga, USA, October 2006.

[22] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.

[23] "Robust Reading Competition," http://algoval.essex.ac.uk/icdar/RobustWord.html, May 2007.

[24] R. Beaufort and C. Mancas-Thillou, "A weighted finite-state framework for correcting errors in natural scene OCR," in *Proceedings of the 9th International Conference on Document Analysis and Recognition (ICDAR '07)*, Curitiba, Brazil, September 2007.

[25] "Daisy website," http://www.daisy.org/, May 2007.