Crosslingual Access of Textual Information using Camera Phones

L. Jagannathan and C. V. Jawahar
Center for Visual Information Technology
International Institute of Information Technology
Gachibowli, Hyderabad, India

Abstract

In this paper we describe a prototype system that recognizes text in images, captured by camera phones and provide access to the content in a different language. Two Indian languages – Hindi and Tamil are used to demonstrate such a system. The prototype system is built using off-the-shelf components, and in house developed algorithms. The acquired image is first transferred to a server, which corrects the perspective distortions, detects recognizes and then translates the text (word). This translated text is sent back to the camera-phone in a suitable form. We have also described here, the Hindi and Tamil OCRs which we use for the character recognition. We also propose methods to make the recognizer efficient in storage and computation. The translated text, along with any additional information, is transmitted back to the user.

1 Introduction

We are often confronted with the task of reading or understanding text present on road signs, billboards and buses to carry out our day-to-day activities. When we are unfamiliar with the language, automatic recognition of such text can be of immense help. A cross-lingual access device that can read text and convert it from one language to another would prove to be a boon in such situations. To facilitate widespread use, such a device needs to be built from off-the shelf components. We demonstrate that a cross lingual access device could be built using camera phones and the available infrastructure. Camera phone is used to capture images with textual content. This image is then transfered to a central server, where it is processed and recognized. The recognized text is then converted to the language of users choice and delivered appropriately.

The implementation of such a system poses many new technical challenges. Traditionally, scanners were used for digitization in document image analysis systems. The captured images were then converted to the textual representation using Optical Character Recognizers(OCR). The scanners, besides being bulky and costly, are not suitable for non-contact imaging. They also fail to image text written on non-planar surfaces [1]. In such situations, CCD cameras prove to be beneficial. Cameras offer greater mobility and are easier to use. Camera phones being ubiquitous can play an important role in serving as a cross-lingual access device.

Textual images acquired through cameras introduce new challenges to the recognition process in the form of perspective distortions [2]. When imaged with a perspective camera, parallel and perpendicular lines do not retain these properties in the image. This necessitates additional processing modules for correcting the distortions. For example, a camera-based business card reading system needs images to be rectified before being OCRed and understood.

Imaging with the help of a camera phone introduces many additional challenges. Images acquired through the camera phones are usually noisy and have complex, cluttered background [3]. Conventional scanners usually provide clean high resolution images with simple background structure. They employ orthographic (parallel) projection for the imaging. Compared to the scanners, camera-phones are of low resolution and and follow a perspective projection model. Novel image understanding algorithms are hence needed to address the situations created by camera based digitization. It is expected that many of these limitations will be overcome soon by the advances in hardware technology and algorithmic innovations.

Reading text using a camera phone or any perspective camera is quite challenging. Images from such cameras are optically distorted due to lens effects. With perspective distortion, character recognition becomes much more difficult compared to scanned documents. Doerman [1] presents an overview of the general challenges associated with the analysis of images acquired through cameras. Images from camera phones suffer from perspective distortion and low resolution. Typical imaging environments have uneven background and non-uniform lighting that makes the job of camera based text reading even more challenging.

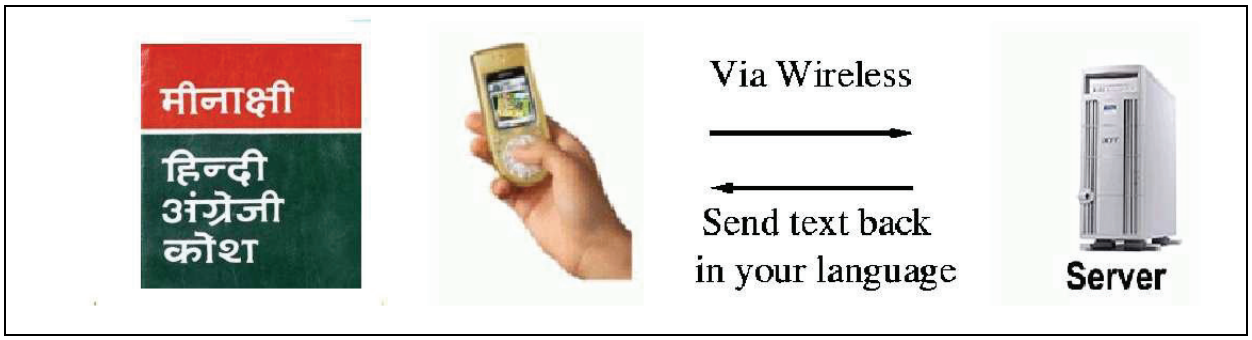

Figure 1: Overview of the process. Camera phone captures the document image. Sends it to the server and gets back the required result based on the settings.

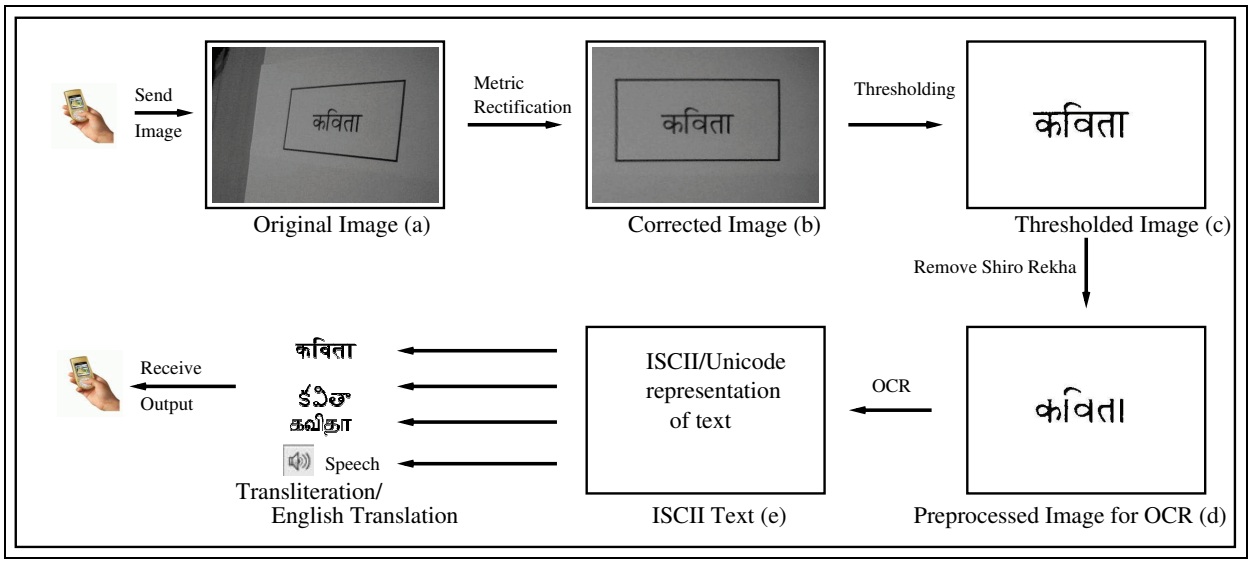

Figure 2: Process of Text Reading

Even with all these limitations, recognition of text present in images acquired through cameras, is shown to be possible. These recognition algorithms follow a different approach, differing from the conventional document analysis systems. Clark and Mirmehdi [4] demonstrate a method to rectify the text prior to the recognition. Plenty of research has been carried out on recognition of objects (often 3D) under perspective imaging [5]. A spectrum of perspective correction techniques based on recent results from Multiple View Geometry [5], is presented in [2]. Detection and recognition of text can also be attempted without perspective correction. Camera based text reading systems are popular for reading license plates in video analysis applications [6]. This paper describes the prototype system that has been built from off-the shelf components for the crosslingual access. The procedure employed for the perspective correction and brief details of the OCRs employed are explained in the next sections. Results are then described in Section 6.

2 Description of the Prototype System

The prototype system consists of off-the shelf available components. We employ a camera phone (Nokia 3660) with a built in digital VGA camera. This camera captures images at 640×480 resolution for the text access. Figure 1 demonstrates the overview of the process to access text in an unknown language.

There are two possibilities in processing and accessing text using camera phones. (1) One can develop applications to under-

stand the text on the camera platform. Such applications require small footprints and high efficiency to be usable on the mobile device. (2) For most Indian languages, we do not have robust and efficient recognizers. Hence it would be better to develop the application on a server. Communication between the server and the camera-phone could happen through technologies like Multimedia Messenging Service(MMS), Infrared or Bluetooth. This is the approach that we follow (see figure 1).

For our experiments we used Bluetooth for transferring images. Bluetooth wireless technology is a low cost short range wireless specification for connecting mobile devices. The system currently recognizes text written in Hindi and Tamil and provides cross-lingual access. Text is written inside a bounding box to aid projective correction. We employ a server running Linux to receive images from the camera phone. The received image is preprocessed and perspective distortions are removed [2]. The textual region is isolated and then fed to the OCR. The recognized text is then sent back to the camera-phone in the language of the users choice.

3 Preprocessing and Perspective Correction

In this section, we describe the image processing modules and perspective correction process. We assume that the images of the words has a bounding rectangle. This could be the border of the sign board. For more advanced perspective correction techniques, refer to [2].

Textual image received from the cell-phone is first preprocessed for detection and isolation of boundaries for perspective correction. The received image has a resolution of 640×480. This low resolution image is first binarised using an adaptive thresholding algorithm. Adaptive thresholding ensures that the system is robust to changes in lighting conditions. Illumination variation is a serious problem in any camera-based imaging systems. We assume that the image has a large white background that contains the black text. This assumption is reasonable as in many real situations this is available. For example, signboards and number-plates have distinct white background. In the next step, we look for the presence of the bounding rectangle, as a boundary of the object or known apriori. The boundary is visible as a regular quadrilateral, which gets rectified as a rectangle after the perspective rectification.

Planar images in multiple views are related by a homography [5]. Hence images from one view could be transferred to another view with the help of just a homography matrix. The homography matrix H is a 3×3 matrix defined only up to a scaling factor. Hence it has 8 unknowns. Recovery of fronto-parallel view involves the estimation of the homography to the frontal view. This could be done by using various methods using prior knowledge of the structure of the textual image [2]. A projective homography can be understood to be the product of three components – similarity(H_s), affine (H_a) and projective (H_p), i.e., $H = H_s H_a H_p$. We are interested in removing the projective and affine components to obtain a similarity transformed (i.e., translated, rotated and/or scaled) version of the original image. When the text is surrounded by a well defined boundary, the boundary can be extracted to correct for projective deformations using the following procedure [2].

1. Identify a pair of parallel lines in the text that intersect in the perspective image.
2. Compute the point of intersection of the two transformed parallel lines.
3. Find the equation of the line ($\mathbf{l} = [l_1, l_2, l_3]$) passing through these points of intersection and rectify the image by removing projective component. ($\mathbf{l}_\infty = H_p^{-1}\mathbf{l}$)
4. Identify a pair of perpendicular lines and remove the affine components.
5. Resultant image is a frontal view (similarity transformed) version of the original image.

4 Script Recognition

Once the image is corrected for projective distortions, it is necessary to recognize the script in the image so that an appropriate OCR could be used. Separation and recognition of scripts is a well studied problem. Since our cross-lingual access system is restricted to the recognition of Tamil and Hindi, we use features specific to these scripts to distinguish between them. Hindi and some other languages like Bangla have a *Sirorekha*, a horizontal bar joining the characters. Also vowel modifiers in Hindi appear above or below the actual character. Textual character in Hindi not only have word-level connected components but have their lower and upper *matras* distinguished by projections. Tamil and other south Indian languages, on the other hand, have each

	அ	ஆ	இ	ஈ	உ	ஊ	எ	ஏ	ஐ	ஒ	ஓ	ஔ
க	க	கா	கி	கீ	கு	கூ	கெ	கே	கை	கொ	கோ	கௌ
ச	ச	சா	சி	சீ	சு	சூ	செ	சே	சை	சொ	சோ	சௌ
ஞ	ஞ	ஞா	ஞி	ஞீ	ஞு	ஞூ	ஞெ	ஞே	ஞை	ஞொ	ஞோ	ஞௌ
ப	ப	பா	பி	பீ	பு	பூ	பெ	பே	பை	பொ	போ	பௌ
ம	ம	மா	மி	மீ	மு	மூ	மெ	மே	மை	மொ	மோ	மௌ
ல	ல	லா	லி	லீ	லு	லூ	லெ	லே	லை	லொ	லோ	லௌ
வ	வ	வா	வி	வீ	வு	வூ	வெ	வே	வை	வொ	வோ	வௌ

Figure 3: Formation of characters in Tamil. Columns represent vowels and rows represent consonants.

(a) (b)

Figure 4: (a) Positional information can be effectively used to distinguish between characters (b) Figure showing the motion of the camera around the textual image

character as a connected component. Further, vowel modifiers apart from being connected to the character (can be connected to the top or bottom) can also appear to the left or right of a character as separate connected components. Hence much difference is not observed in the horizontal projections of such languages. Script is recognized using the above information. The text is assumed to be distinct from the background and hence it is easily extracted.

5 Tamil and Hindi OCRs

A pattern classification system which recognizes a character image is at the heart of any OCR. We employ a Support Vector Machine based classifier for the character recognition [7]. Feature extraction [8] is done using PCA (Principal Component Analysis) for building a compact representation. When samples are projected along principal components, they can be represented in a lower dimension with minimal loss of information.

Recognition using a SVM classifier involves the identification of an optimal hyperplane with maximal margin in the feature space [8]. The training process results in the identification of a set of important labeled samples called support vectors. Discriminant boundary is then represented as a function of the support vectors. The support vectors are the samples near the decision boundary and are most difficult to classify. We build a Directed Acyclic Graph of ${}^{N}C_2$ binary classifiers where N is the number of classes. Every component passes through N classifiers before being classified. More information regarding the Directed Acyclic Graph approach for character recognition can be found in [7].

In [7], we analyzed the performance of the SVM-based OCR with different Kernel functions for the classification. It was observed that the accuracy of the OCR did not vary significantly with the increase in complexity of the Kernels. We exploit this advantage in building an efficient SVM-OCR which needs much smaller representation for the storage.

With a direct SVM-OCR, the model files (support vectors, and Lagrangians [8]) occupy large amount of space; usually of the order of several hundreds of megabytes. Since higher order kernel functions did not improve the accuracy, we restricted our classifier to linear kernels. A single decision boundary is stored in the model file for each node in place of the support vectors. This resulted in a reduction of the size of the model file by a factor of 30. This reduced model file could be ported to a PDA or to devices where storage space is precious. We are presently exploring the option of porting the OCR into a camera-phone.

After classification of the sequence of components, class labels are converted to the UNICODE output specific to a language.

Hindi OCR Hindi is written in Devanagari script and is spoken by the largest number of people in India. Though various dialects of Hindi exists the underlying script do not vary. The complex script employed by Hindi presents the OCR research community with novel challenges. Words appear together as a single component and characters are often slightly modified by the *matras*. In [7], we discuss the details of the OCR built for two languages: Hindi and Telugu. During preprocessing, the *Sirorekha* is identified and used to separate the upper *matras* from the remaining word. Horizontal projection profiles are then used to separate the lower *matras*. The *Sirorekha* itself is removed to obtain three different zones of the word –upper, middle and lower zones. During training of the classifier system, the upper, lower and the middle components are trained separately. Since the vowel modifiers are separated from the main characters, the total number of classes to be recognized is reduced. During testing, a component is sent to the appropriate classifier depending on its position. Since the upper, lower and middle zones of the word have separate classifiers, a component from one part will not be confused with a component from a different part increasing the accuracy.

Tamil OCR Tamil is one of the Dravidian languages and is spoken in Tamil Nadu, a southern state of India. We extend our earlier work [7] for the cross lingual access of Hindi and Tamil text. For this, we have developed a Tamil OCR whose details are given here. Tamil shares the same structure of the script as the other south Indian languages with minor differences. The Tamil script is different from Hindi as characters are not connected with vowel modifiers and the vowel modifiers themselves may appear before or after the actual character. The characters in Tamil are formed from two sets of base characters. The first set consists of 12 characters (called *uyirezhuthu*). These form the vowels of the language. The other set (called *meiyezhuthu*) consists of 18 characters which form the consonants. The combinations of these produce 216 (18×12) different characters. Some of the characters that are formed are shown in Figure 3. The columns represent vowels and the rows different characters and their modified forms. These modified characters may be composed of different connected components. These different components may also appear before or after the character. In some cases, the vowel modifiers appear as a part of the character itself. However the number of different components produced are limited. The reduction in the number of characters in the language, is due to the absence of stronger forms of some consonants. . For example the pronunciations *'ka'* and *'ga'* have the same underlying script in Tamil whereas they are represented by different characters in Hindi. The basic language consists of about 120 different classes to recognize. Hence Tamil is simpler to analyze owing to the lesser number of classes present. For the purpose of recognition, characters are treated in their entirety. Features from images are extracted using PCA and classified using SVM's under the DAG architecture similar to Hindi. Post-processing was done to differentiate between characters that were frequently misclassified. In some cases, positional information of connected components was also used to resolve ties. Such an example is given in figure 4(a). The order of the connected components alone does not suffice in this case. The positions of the two connected components are also used to resolve the ambiguity. While the connected components of figure 4(a) i) does not overlap in their x-coordinate, the connected components in figure 4(a) ii) overlap.

Cross-Lingual Access India is a multilingual country with various scripts and languages. For the effective access of relevant information, cross-lingual techniques are needed. The basic approaches in cross lingual information access can be categorized into three broad categories: (a) Corpus-base Approaches (b) Machine Translation based Methods (c) Dictionary based Techniques.

In corpus based methods, the source language query is translated using a parallel text corpus to a target language query. This approach requires a parallel collection in the domain of the queries and the target collection. In Machine translation based approaches, a machine translation system is employed for query translation. Such systems aim at correct language translation of source language texts. For the situations, in which camera phone is used for information access, dictionary-based techniques are most appropriate. In fact, usually one will be interested in reading and understanding a specific word rather than a complete sentence.

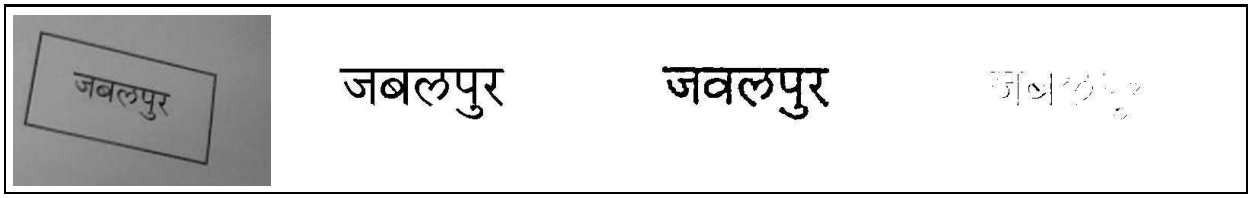

Figure 5: (a) Shows the image captured by the camera. The original frontal view image is showed in (b). The rectified image of (a) is shown in (c). The difference between (b) and (c) is shown in (d).

The dictionary based approach is often viable when linguistic resources are scarce. These approaches use bilingual dictionaries for query translation. The basic necessity for query translation is at word-level. If the source language words appear in inflected forms, they cannot be readily translated, because they do not match with dictionary headwords(which are in base forms). If there is a morphological analyser(parser) available, words can be normalized to the lemma, i.e. the normal dictionary entry form. Approximate string matching techniques are used for dictionary lookup in these cases.

For Indian languages, which share a common set of alphabet, a still simpler solution is to transliterate the words across languages. This allows the user to read words written in other languages. In applications like reading city names, such transliteration schemes are sufficient. We provide mechanism for transliteration and dictionary-based translation in the present prototype system.

6 Results

The system was tested with 30 images of city names prepared in both Hindi and Tamil. The city names were printed in fonts *Naidunia* for Hindi and *TM-TTValluvar* for Tamil. The city names were printed with a font size of 120. When images were taken from various angles close to the frontal view, the recognition accuracy of the system was close to 100%. The accuracy of the system was limited only by the performance of OCR for small variations in angle from the frontal view. To estimate the robustness of the system towards arbitrary angles, we moved the camera around the image with increasing angle from the frontal view as shown in figure 4(b). The distance of the camera from the text image was kept constant. For angles approximately upto about 60°, the recognition accuracy was close to 100%. Errors occured in cases where the character was very distorted for the OCR to recognize it accurately. For angles more than 60° from the frontal view, the results were not consistent. This was because the quality of the rectified images was not suitable for recognition. The images contained too much noise to be recognized by the OCR and hence the accuracy of the system deteriorated in such cases. Figure 5 shows an example image captured by the camera, the frontal view image, the rectified image and the error image. Note that the difference between the rectified image and the frontal view image is not significant.

7 Conclusion

A prototype system for cross-lingual access of text from document images is presented. Future work involves automatic recognition and isolation of text from cluttered background. The system could also be extended for various other applications like bar code reader, signboard reader, etc. Currently work is being done to automatically rectify the image without the knowledge of the bounding box.

References

[1] D. Doerman, J. Liang, and H. Li, "Progress in camera-based document image analysis," *Proceeding International Conference of Document Analysis and Recognition*, pp. 606–616, 2003.

[2] L. Jagannathan and C. V. Jawahar, "Perspective correction methods for camera based document analysis," *Proc. First Int. Workshop on Camera-based Document Analysis and Recognition (CBDAR), Seoul*, pp. 148–154, 2005.

[3] M. Mirmehdi, P. Clark, and J. Lam, "Extracting low resolution text with an active camera for ocr," in *Proc. IX Spanish Symposium on Pattern Recognition and Im age Processing*, pp. 43–48, May 2001.

[4] P. Clark and M. Mirmehdi, "Estimating the orientation and recovery of text planes in a single image," *Proc. 12th British Machine Vision Conf.*, pp. 421–430, 2001.

[5] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2000.

[6] H. Li, D. Doerman, and O. Kia, "Automatic text detection and tracking in digital videos," *IEEE Transaction on Image Processing*, vol. 9, no. 1, pp. 147–167, 2000.

[7] C. V. Jawahar, MNSSK Pavan Kumar, and S. S. Ravikiran, "A bilingual OCR system and its applications," *Proc. Int. Conference of Document Analysis and Recognition*, pp. 408–413, 2003.

[8] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*. John Wiley and Sons, 2002.