Character Recognition System for Cellular Phone with Camera

K. S. Bae, K. K. Kim[0], Y. G. Chung, W. P. Yu
{Intelligent Task Control Research Team, Human Robot Interaction Research Team[0]},
Electronics and Telecommunications Research Institute, Korea
{pavin, kyekyung[0], ykchung, ywp}@etri.re.kr

Abstract

This paper describes camera based character recognition system, which is implemented for mobile devices such as PDA and cellular phones with color cameras. First, we have developed camera based character recognition system for PC that includes techniques such as image enhancement, local adaptive binarization and blob coloring to effectively extract character region and remove the noise of camera captured images. Then, we converted the PC based OCR system to the embedded OCR system for cellular phones. Several functions were specially developed since most of mobile telecommunication devices have no real number computing functions and have limitation of memory space. In this paper, we are addressing the related problems and our system mechanisms.

Keywords: character recognition system, local adaptive binarization, WIPI

1. Introduction

In the past, a major purpose of telephone usage was voice conversation. As the mobile telecommunication devices are popularized, the purpose of telephone is changing to manage personal information. Considering the fast popularization of cellular phone with digital camera, necessity of camera based character recognition system will increase, because it has many application fields such as navigation system, the smart tour guide, data compression of document images, robot automatic traveling, *etc*. However, camera captured images have many noises due to low brightness contrast and influence of illumination variance. It makes hard to extract character region and to recognize characters. Many methods have been proposed to tackle these problems [1-5]. Besides there is constraint that real number operations must be converted to an integral number operations to embed the character recognition system into mobile devices because most of mobile devices have no real number operator functions.

In this paper, to recognize characters from a camera captured document image, a locally adaptive binarization method is proposed. And also image enhancement algorithm before binarization is applied to increase robustness from illumination variation. Local adaptive binarization algorithm provides better binarization results compared with other proposed binarization algorithms. Experiments are carried out with camera-captured images.

2. Character recognition system for a Camera captured image

As shown in Fig 1, our recognition system consists of three steps: preprocessing, segmentation and recognition.

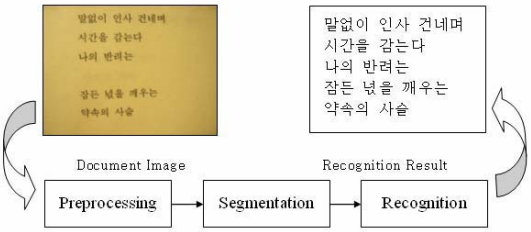


Fig 1. Document Image Recognition System

2.1 Preprocessing

Traditional research of character recognition was mostly based on scanned images. The extraction of character region on scanned images is not difficult, because they have white background and black character foreground, due to uniformed optimal illumination. But camera-captured images have many noises due to low brightness contrast and various illuminated environment. Therefore, the performance of a camera based character recognition system depends on

Proceedings of the 29th Annual International Computer Software and Applications Conference (COMPSAC'05)
0730-3157/05 $20.00 © 2005 IEEE
Authorized licensed use limited to: Reprints Desk. Downloaded on July 23,2025 at 20:15:03 UTC from IEEE Xplore. Restrictions apply.

how effectively it removes noises in the preprocessing step.

In the preprocessing step, a camera-captured image is first converted into a gray level image and then image enhancement algorithm is applied to the image. The enhanced image is binarized and then the noises are removed by blob coloring.

2.1.1. Image enhancement. A gray level image has been binarized for classifying foreground pixels from background ones. Partly very low contrast of intensity on image exists because of illumination variation and photographing angle such as figure 2 (a). It causes misclassification of foreground characters from background. To overcome this problem, image enhancement by gray level normalization is processed before binarization. The formula for enhancement of an image is described as follows:

$$f_1(x, y) = (L-1)\frac{f(x, y) - \min}{\max - \min}$$

$$\max = \max[f(x, y)]$$

$$\min = \min[f(x, y)]$$

$$for \quad 1 \le x \le M \text{ and } 1 \le y \le M \qquad (1)$$

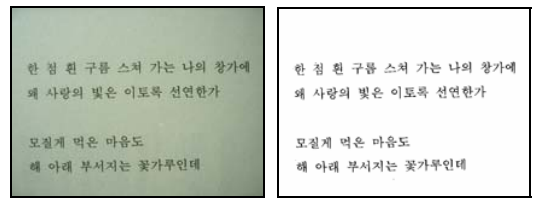
Where $f(x, y)$, $f_1(x, y)$ denotes the gray level value at pixel (x, y) and the pixel level after image enhancement, respectively. L denotes gray level range of image to be converted and M denotes the height and width of image. Max and min are maximum value and minimum value among pixels on image, respectively.

2.1.2. Binarization. We proposed locally adaptive thresholding method that is robust to variation of illumination. This method binarizes an image using the difference of maximum and minimum intensity value of pixels in sub-window.

Our proposed binarizing method yields better results compared with other methods. The thresholding value of T_{rxr} is selected automatically in rxr sub-window. The formula for binarizing camera-captured image is described as follows:

$$T_{rxr} = t \times (g_{high} - g_{low}) \qquad 0.4 \le t \le 0.7 \qquad (2)$$

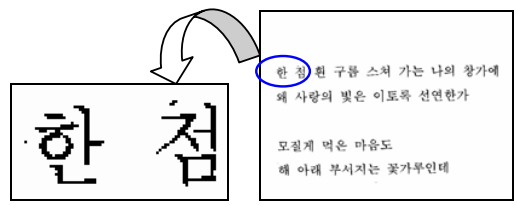
Where g_{high} and g_{low} are maximum and minimum intensity value of pixels in rxr sub-window, respectively. The range of t has been obtained through the many binarization experiments over various kinds of document images.



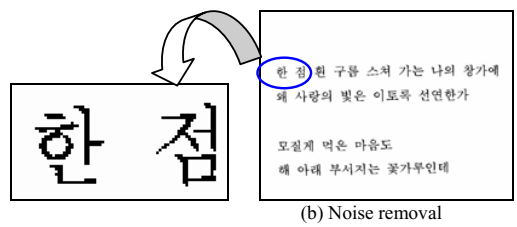
(a) Original image (b) Binarized image

Fig 2. Locally Adaptive Threshold method

2.1.3. Noise Reduction. The blob coloring process is applied to the binarized image. And then noises on the image are removed through analyzing location and size of blobs. Fig 3 is the result of noise reduction by blob coloring.



(a) Binarized image



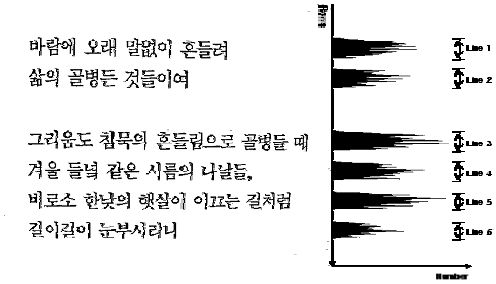
(b) Noise removal

Fig 3. Noise reduction by noise reduction

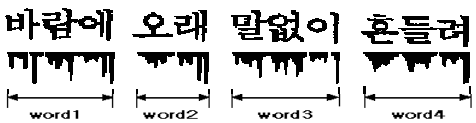
2.2 Segmentation

In segmentation phase, extracting the correct character regions is very important for the recognition of characters. In regular sequence, the binary image is segmented into lines, words and characters. There are many segmentation methods well known such as projection, region growing and tracing contour *etc*. In our system, we use the segmentation method by projection. The binary image is first horizontally projected to extract character line regions as shown in Fig 4 (a). And then regions of words and characters are extracted from a character line region by vertical projection as in Fig 4 (b) and (c).

Proceedings of the 29th Annual International Computer Software and Applications Conference (COMPSAC'05)
0730-3157/05 $20.00 © 2005 IEEE
Authorized licensed use limited to: Reprints Desk. Downloaded on July 23,2025 at 20:15:03 UTC from IEEE Xplore. Restrictions apply.



(a) Line segmentation



(b) Word segmentation

(c) Character segmentation

Fig 4. Horizontal and Vertical Projection

Korean character named "Hangul" consists of combined consonants and vowels as in Fig 5. We can classify Hangul into six types using structural feature of Hangul. Six types of Hangul are shown in Fig 5.

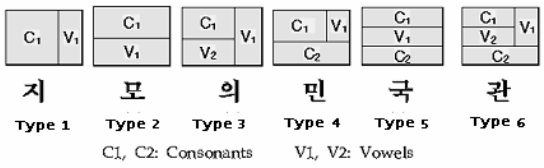


Fig 5. 6 types of Korean Characters

The characters in type 2, 4, 5 and 6 are fully segmented by only vertical projection. But the characters in type 1 and 3 are not, because they have a discrete vertical direction vowel in their right. Therefore, we have to make a rule that could combine this discrete vowel with the other element (consonant and vowel). Fig 6 shows result of vertical projection. We can see that space of each character is larger than interval between vowels and consonant. We first calculate average distance of each character. Then, if interval is less than average, we regard them as discrete vowel and combine them. Fig 7 shows result of segmentation.

Fig 6. Example of discrete vowels (g2, g6)

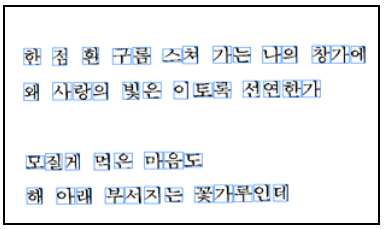


Fig 7. Result of segmentation

2.3 Recognition

In recognition phase, segmented characters are normalized, extracted features and classified by Neural Network. Finally, characters are recognized.

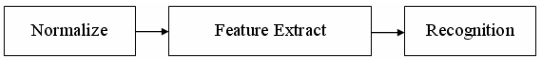


Fig 8. Process of character recognizer

2.3.1. Normalization. To recognize characters that have various font and size, input characters should be normalized as a size of training samples.

Fig 9. Result of normalization

2.3.2. Feature Extraction. To classify character type into six types of Hangul, features are extracted from binary image. Structural features are extracted, including mesh, chain code, gradient, transition and distance of segmented individual character image [7].

Mesh features by summing foreground pixels of segmented character and chain code features of eight directional slopes in sub-windows are used. The transitions are the number of foreground pixel to background pixel transition points. The transition features are extracted in both horizontal and vertical directions. The distance features are distances from the image boundary to the first foreground pixel.

2.3.3. Character Recognition. After the preprocessing and segmentation phase, the individually segmented characters are normalized. And 256 features for type

Proceedings of the 29th Annual International Computer Software and Applications Conference (COMPSAC'05)
0730-3157/05 $20.00 © 2005 IEEE
Authorized licensed use limited to: Reprints Desk. Downloaded on July 23,2025 at 20:15:03 UTC from IEEE Xplore. Restrictions apply.

classification and 328 features for individual character recognition are extracted and then transmitted to Neural Network. The Neural Network calculates output value using transmitted input features and weight vectors.

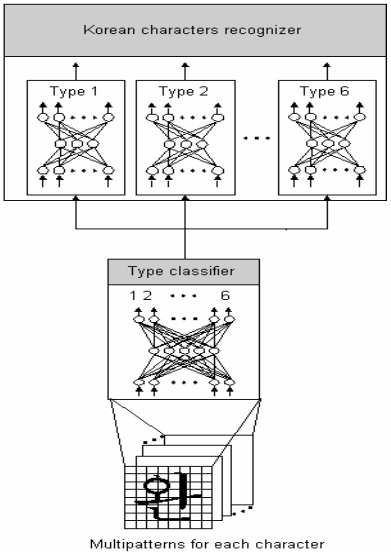


Fig 10. Character recognizer using NN

3. Embedding in Cellular phone

Our character recognition system for PC could not be directly embedded into a mobile telecommunication device because most of mobile telecommunication devices have no real number operator functions and they have only integer based computing operators. Also, even if real number operations are performed on an emulator, response of character recognition in real time is not easy to get, because about 400,000 decimal weight values should be processed via additions and multiplications to classify types and recognize individual characters. Also sigmoid function could not be used because sigmoid function is exponential function and mobile device could not calculate exponential function. Thus, we have to convert real number operation to integer operation and perform polynomial approximation for sigmoid function.

3.1. Conversion of real number operation into integer operation

The most important matter that we should consider is a decrease of recognition error rate. As operations are repeated, accumulated errors are enlarged because fractal values below the decimal point have not been counted. To minimize these errors, we have to change integer type into decimal type that has a virtual decimal point. In our research, we change 64 bits decimal data into 32 bits integral data.

3.2. Approximation of Sigmoid function

There are many activation functions in neural network methods. We used a sigmoid function among them.

$$f(net) = \frac{1}{1+\exp(-net)} \qquad 0 \le f(net) \le 1$$

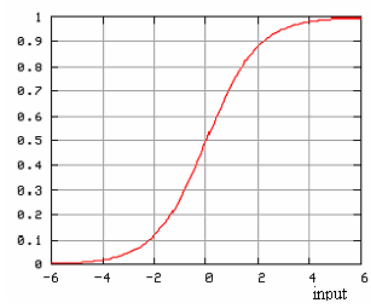


Fig 11. Sigmoid function

As mentioned earlier, we need an approximation method to substitute for sigmoid function because mobile telecommunication devices have no exponential function. As shown in Figure 11, a sigmoid function is symmetrical function. We split positive portion (0~40) of input axis into 4~128 partitions and approximate them. Figure 12 shows approximation algorithm. After splitting, a polynomial approximation is applied to each partition and then, we can find simple approximation equation that is most similar to original sigmoid function. At this time, we can get max-error point of original sigmoid function through approximation equation and then, repeat splitting based on this point. Process like this is repeated until positive portion (0-40) is divided into 128 partitions. We have made the approximated sigmoid function for the mobile cellular phones. Fig 13 shows the process results of polynomial approximation by a sigmoid function.

Proceedings of the 29th Annual International Computer Software and Applications Conference (COMPSAC'05)
0730-3157/05 $20.00 © 2005 IEEE
Authorized licensed use limited to: Reprints Desk. Downloaded on July 23,2025 at 20:15:03 UTC from IEEE Xplore. Restrictions apply.

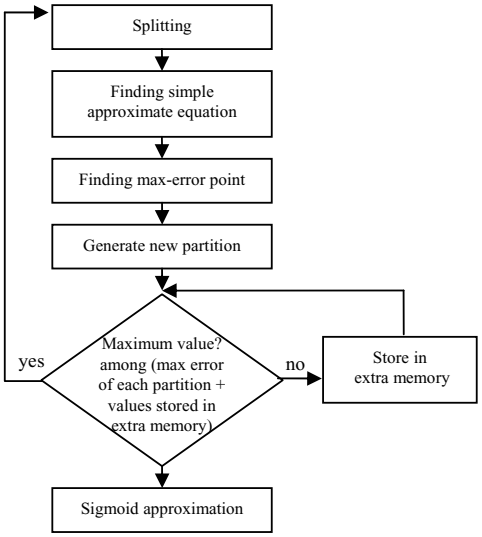


Fig 12. Approximation algorithm of sigmoid function

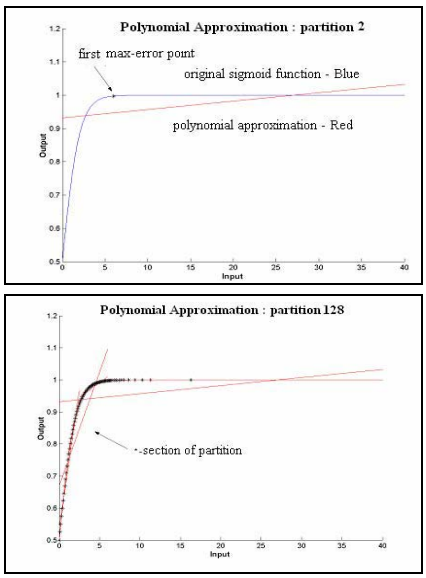


Fig 13. The process of polynomial approximation

3.3. WIPI (Wireless Internet Platform for Interoperability)

As wireless Internet service market grow up in earnest, people feel keenly the necessity of wireless Internet platforms. To solve several problems resulted from various platforms of various mobile telecommunication companies, the plan of mobile standard platform development is started since July, 2001 in Korea. Since then, WIPI (Wireless Internet Platform for Interoperability) come into the world due to the effort of Expert Group including Mobile Telecommunication Companies, Electronics and Telecommunications Research Institute (ETRI) and Telecommunications Technology Association (TTA).

ETRI, a member of 3GPP, made a formal introduction of WIPI at the 3GPP conference of Vancouver in Canada (May, 2002). More detailed information about WIPI can be referred to [10].

In our research, we applied our character recognition system to WIPI in software develop environment that are offered by SK Telecom, Korean Company.

Fig 14. Embedded recognizer in WIPI Emulator

4. Experiments

To evaluate our system, we have carried out experiments with camera document images of the ETRI database. A multi-layer perceptron has been implemented with 256 input neurons, 100 hidden neurons and 6 output neurons to classify six kinds of character types. And the other MLPs have been implemented with 314 input neurons, and 120, 76, 54, 520, 301, and 55 output neurons to recognize characters according to each character type. MLPs for recognition are trained with 11260(10/character class) and tested with 6756 characters extracted form ETRI database. Training and testing database is constructed with documents inputted by digital camera with 1280x1024 resolutions.

Classification performance of character type is shown in Table 1. Six types of Hangul are classified and 99.19% of classification rate is obtained. The characters included in TYPE 1 to 5 have good classification performance. But Type 6 has poor classification result even though is is the smallest database set. In confusion matrix, some characters in character Type 6 are misclassified frequently into Type 4 because character structure of Type 4 and 6 is similar to each other.

Table 1: Classification performance of character type.

Proceedings of the 29th Annual International Computer Software and Applications Conference (COMPSAC'05)
0730-3157/05 $20.00 © 2005 IEEE
Authorized licensed use limited to: Reprints Desk. Downloaded on July 23,2025 at 20:15:03 UTC from IEEE Xplore. Restrictions apply.

Character type	Total chs.	Classified chs.	Classification rates (%)
Type 1	720	719	99.86
Type 2	456	456	100.00
Type 3	324	319	98.46
Type 4	3120	3107	99.58
Type 5	1806	1793	99.28
Type 6	330	307	93.03

The recognition rates according to six character types are shown in Table 2. Recognition result of 97.52% is obtained. Some characters of Type 6 "뒷", "쉼" and "뛴" are misrecognition into "윗", "쉽" and "쥔". Another character recognition error is caused mainly by individual character segmentation. In the future, revised segmentation algorithm and training with more document database are needed to improve overall recognition performance.

Table 2: Recognition performance according to character type

Character Type	Recognition rates (%)
Type 1	97.83
Type 2	99.76
Type 3	97.67
Type 4	98.52
Type 5	98.11
Type 6	89.00

5. Conclusions

This paper addressed an embedded optical recognition character system of a camera captured document image which can be implemented in mobile devices such as PDAs and cellular phones.

Local adaptive binarization and image enhancement algorithm using gray level normalization is proposed to process camera captured images. The proposed binarization algorithm shows excellent performance compared with other well known binarization algorithms. Binary image is projected horizontally and vertically. Then, individual characters are extracted. Features are extracted from individual characters and entered into input nodes of our neural net based recognizer. The neural net recognizer classifies character types and correctly recognizes the characters. We especially made the real number operation module for converting decimal operations into integer operations and then it was applied to the WIPI emulator. Hereby, we confirmed that our system could be embedded in cellular phone practically.

Experiments have been carried out using the camera-captured images database in ETRI. Experimental result shows 99.19% of classification rate of and 96.82% of recognition rate as in table 1 and 2. Our adaptive binarization process played a key role for extracting the enhanced character features and special functions such as conversion modules of real number operations provided a good role for practical usage. Through these development activities, we could get encouraging results.

6. References

[1] H. Fujisawa, H. Sako, Y. Okada, and S. W. Lee: Information capturing camera and developmental issues, Proc.of the 6th ICDAR, pp. 205-208, 2001.

[2] M. Seeger and C. Dance: Binarizing camera images for OCR, Proc. of the 6th ICDAR, pp. 54-58, 2001.

[3] K. Wang, J. A. Kangas, and W. Li: Character segmentation of color images from digital camera, Proc. of the 6th ICDAR, pp. 210-214, 2001.

[4] L. Fan, L. Fan and C.L.Tan: Binarizing document image using coplanar prefilter, Proc. of the 6th ICDAR pp. 34-38, 2001.

[5] A. Dawoud and M. Kamel: Binarization of document image using image dependent model, Proc. of the 6th ICDAR, pp. 49-53, 2001.

[6] S. Madhvanath, G. Kim, and V. Govindaraju: Chaincode contour processing for handwritten word recognition, IEEE Trans., on PAMI 21, pp. 928-932, 1999.

[7] K.K. Kim, J.H. Kim, and C.Y. Suen: Segmentation-based recognition of handwritten touching pairs of digits using structural features, Pattern Recognition Letters, pp.13-24, 2002.

[8] B. Kosko: Neural networks and fuzzy system, Prentice-Hall, Englewood Cliffs, NJ, 1992.

[9] H.R Byun, M.C Roh, K.C Kim, Y.W Choi, S.W Lee: Text Extraction in Complex Images, Document Analysis Systems V, pp.329-340, 2002

[10] http://www.wipi.or.kr/

Proceedings of the 29th Annual International Computer Software and Applications Conference (COMPSAC'05)
0730-3157/05 $20.00 © 2005 IEEE
Authorized licensed use limited to: Reprints Desk. Downloaded on July 23,2025 at 20:15:03 UTC from IEEE Xplore. Restrictions apply.