체인 정합과 확장된 그룹핑 방법을 사용한 곡선형 텍스트 라인 추출

Nguyen Noi Bai† · 윤 진 선†† · 송 영 준††† · 김 남†††† · 김 용 기†††††

요 약

본 논문은 정형화되지 않은 텍스트 라인들을 추출하기 위한 방법을 보여주고 있다. 텍스트 라인들은 각기 다른 각도로 구성되고, 심하게 굴곡이 있는 모양, 그리고 텍스트 라인내의 약간의 단어 사이의 공간이 생기게 된다. 그러한 텍스트 라인들은 포스터, 주소, 그리고 예술 문서 등에서 발견된다. 제안하는 방법은 기존의 직관적인 그룹핑 방법에 기반을 두고 있지만, 하나의 라인에서 발생하는 불충분한 특징점들과 모호한 회전 등을 극복하기 위한 방법을 개발하였다. 본 논문에서 텍스트 라인들은 몇 개의 연결된 성분들로 구성되고, 이 성분들은 하나의 문자 또는 연결된 문자들의 검은색 화소들의 집합이라고 가정하였다. 제안하는 방법은 반복적으로 증가되는 임계값과 가까운 성분들은 하나의 체인으로 병합하게 되고 확장되어 길어진 체인들은 라인의 원시 체인으로서 인지된다. 그때 원시 체인들은 텍스트 라인의 부분적 회전에 따라 좌우로 확장되어 진다. 텍스트 라인의 부분적인 회전은 원시 체인이 확장될 때, 체인들의 각 면에서 재구성될 것이다. 이러한 과정을 통해서 모든 텍스트 라인들이 구성되어 진다. 제안 방법은 로고와 슬로건에서 사용된 곡면으로 쓰여진 텍스트 라인들에 대해서 실험한 결과 직선 텍스트 라인은 98%, 곡선 텍스트 라인은 94%로서 높은 추출율을 보여주고 있다.

키워드 : 문서영상 분석, 문서영상 분할, 텍스트라인 추출, 곡선 텍스트라인 추출

Extracting curved text lines using the chain composition and the expanded grouping method

Nguyen Noi Bai† · Jin-Seon Yoon†† · Young-Jun Song††† · Nam Kim†††† · Yonggi Kim†††††

ABSTRACT

In this paper, we present a method to extract the text lines in poorly structured documents. The text lines may have different orientations, considerably curved shapes, and there are possibly a few wide inter-word gaps in a text line. Those text lines can be found in posters, blocks of addresses, artistic documents. Our method based on the traditional perceptual grouping but we develop novel solutions to overcome the problems of insufficient seed points and varied orientations in a single line. In this paper, we assume that text lines contained some connected components, in which each connected components is a set of black pixels within a letter, or some touched letters. In our scheme, the connected components closer than an iteratively incremented threshold will make together a chain. Elongate chains are identified as the seed chains of lines. Then the seed chains are extended to the left and the right regarding the local orientations. The local orientations will be reevaluated at each side of the chains when it is extended. By this process, all text lines are finally constructed. The proposed method is good for extraction of the considerably curved text lines from logos and slogans in our experiment; 98% and 94% for the straight-line extraction and the curved line extraction, respectively.

Key Words : Document Image Analysis, Document Image Segmentation, Text Lines Extraction, Curved Text Lines Extraction

1. Introduction

Extraction of text lines is a part of off-line text recognition to arrange characters in the correct order as they were written. There are many published works in this field during the last decade. In well structured documents, text lines are straight and parallel, simple techniques like projection profile [1], Hough transform [2] can be used successfully. But in poorly structured documents, some text lines have remarkable curvature and sometimes there are wide gaps between words in a single line. Simple approaches mentioned above will not

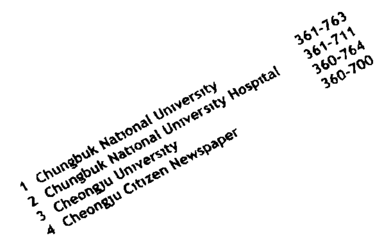
※ 이 논문은 2003년도 학술진흥재단의 지원에 의하여 연구되었음(과제번호: KRF-2003-070-C00025)
† 준 회 원 : 충북대학교 전기전자컴퓨터공학부 석사과정
†† 준 회 원 : 충북대학교 BK21충북정보기술사업단 박사후연구원
††† 종신회원 : 충북대학교 충북BIT연구중심대학육성사업단 초빙조교수
†††† 정 회 원 : 충북대학교 전기전자컴퓨터공학부 교수
††††† 정 회 원 : 충북대학교 천문우주학과 교수
논문접수 : 2007년 5월 28일, 심사완료 : 2007년 9월 16일

work in these documents.

There are a few pieces of published work on extraction of multi-oriented or curved text lines. In Zahour et al. [3], the documents are split into vertical strips and projection profiles are sought inside each strip. These piecewise projections are used to adapt to local orientations within a more global scheme. Projection based methods only deal with text lines of moderate curvature. Wong et al. proposed the Run-Length Smoothing Algorithm [4]. Black pixels along horizontal direction will be smeared if their distance is within a predefined threshold. Some techniques can be added to separate wrongly smeared points, such as in Douglas et al. [5]. This method requires an adequate threshold and does not work on considerably curved lines. In the work of Felbach and Tonnies[6], baseline units are detected as the minima points of the writing. These units are grouped gradually to form text lines. In this approach, several parameters must be determined manually depending on the script size.

To deal with an arbitrarily curved text line, Yan [7] proposed a fuzzy curve-tracing algorithm. There, character pixels are grouped based on the fuzzy c-means algorithm. Then the cluster centers are connected to form the initial curve of the text path. Finally, another clustering process is to make the path smooth. This method needs to choose the appropriate smoothness parameter and the author only focused on building a single text line but not consider the case that two text lines can be clustered into one. U.Pal et al. [8] exploited some specific properties of Indian scripts to extract text lines from Indian artistic documents. This approach is only applied for Indian documents. Goto and Asu [9] proposed a local linearity method, in which they firstly split the document image into some small sub-regions. Next, local orientation is estimated within each sub-region. Finally, they extend local orientations of the sub-regions to extract the text lines. But the main limitation of this method is that it is unable to handle variable size characters.

In the class of grouping methods, Likoforman-Sulem and Faure have developed an iterative method based on perceptual grouping [10]. Anchor points are detected by selecting connected components elongated in specific directions (0°, 45°, 90°, 125°). Each anchor, then each alignment, is extended to the left and the right. By this technique, the authors assumed a fixed orientation within every single line, so it was not suited for remarkably curved lines. It was also difficult to apply this method in



(a) Address Block (zip codes) with some wide inter-word gaps

(b) Slogan with remarkably curved text lines

(Fig. 1) Examples of poorly structured documents.

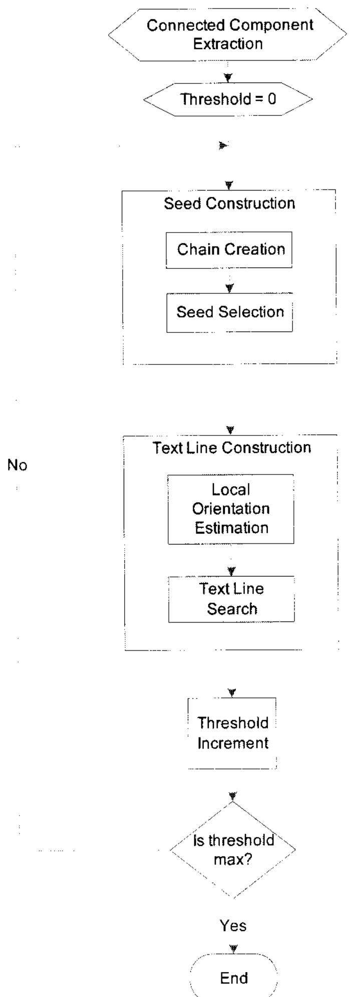
the case of insufficient anchor points for every text line.

It can be seen that abovementioned works are not appropriate for handling remarkably curved text lines or text lines containing some big word-gaps. In this paper, we introduce a new approach to extract text lines from poorly structured documents which contains irregular inter-word distances and curved text lines (Fig. 1). We use the grouping strategy suggested in [10] but some significant changes are made to suit for the mentioned kinds of documents. Besides anchor points, seeds of text lines are built from potential sets of connected components. Moreover, the text line orientations are adjusted to adapt the remarkable curvature. In the next sections, the described method is used for processing the left-to-right written documents, but it can be easily expanded to include the up-to-down written documents by a simple transposition.

2. Definitions and overview of the algorithm

In this paper, to describe the algorithm conveniently we frequently used the following definitions.

Connected components: a connected component is a set of black pixels in which every pair of the black pixels can be connected by one or some continuous paths (sequences of black pixels). In a document, a connected component may be a character, a part of character or a group of touched characters. For short, we also use the single word “component” to indicate connected component from here.



(Fig. 2) Flowchart of the algorithm

Chains: we define a chain as a spatially ordered set of connected components. For two consecutive connected components X_i and X_{i+1} in a chain, there is a relationship: X_i is the left neighbor of X_{i+1} or X_{i-1} is the right neighbor of X_i, vice versa.

Seeds of text lines: a seed of a text line is a connected component or a chain which is used to search the remaining components in the same text line. Each seed has a realizable orientation so that the local orientations can be estimated from these seeds. If a seed is a connected component, it is called an anchor point. If a seed is a chain, it is called a potential chain.

Our algorithm is schemed in the flowchart in the Fig.2.

Firstly, the connected components are extracted by the classical image processing routine. Each connected component is given a unique number in the connected component labeling stage.

The algorithm includes an iterative loop. At the first iteration, all connected components are free components, which are still not grouped to any other component. A distance threshold is initiated to equal to zero and it is incremented every iteration. At each iteration, chains of components are built from free connected components under the constrain: the distance between two successive components must be within the threshold. Some of these chains will be chosen as the seeds of text lines by a shape criterion (see Section 3). The local orientations are calculated at the left end and the right end of each seed. These seeds are extended to the left and the right to construct text lines. The local orientations are reevaluated whenever a new component is added to the seed. After one iteration is completed, some new text lines are detected. Connected components lying in these text lines become caught components. Opposite to free components, the caught components are grouped with other components in a searched text line. At the next iterations, the threshold is incremented and the process is repeated for the remaining free components until the threshold reaches the predefined maximum value.

3. Seed Construction

At the beginning of each iteration, free connected components are linked together to create some chains. The linking process obeys the following conditions:

A free component is linked with two other free components at most: one in its left side and another in its right side. These components must be its two nearest free components according to each side.

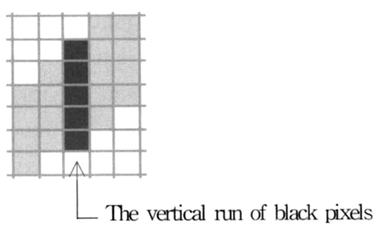
Components are linked if the distance between them is within a distance threshold D. The threshold D is incremented every iteration.

Each sequence of linked components forms a chain. In most documents, although a few chains possibly contain some components belonging to different text lines (this situation happens if the threshold exceeds the inter-line gaps), a majority of chains are correctly text-line segments, especially with thresholds smaller than inter-line gaps. It is resulted from the observation that the inter-character or inter-word distances are often shorter than inter-line distances. This fact can be exploited to select seeds of text lines.

Among the created chains, some chains will be chosen as the seeds of text lines. This selection starts with the estimation of the height and the length for each chain. To estimate the heights and the lengths, each chain is smeared (Fig. 3). Firstly, we do horizontally smearing. Every horizontal run of white pixels between two black pixels of the same connected component will be filled

(a) Chain (b) Smeared Components (c) Smeared Chain

(Fig. 3) Smearing



(Fig. 4) Black vertical run

with black pixels. Then we also shade the horizontal run of white pixels between two successive components in the chain. Next, the similar strategy is applied for every vertical run of white pixels. After smeared, chains appear as compact black blocks (Fig. 3). In each compact block, we take the average value for vertical runs of black pixels (the used unit is the number of black pixels within every vertical run, see Fig.4). This average value is the block's height. The length of each block is simply its horizontal length

The seeds of text lines can be recognized by their elongate shapes. It means that their lengths are large in comparison to their heights. Actually in the human vision, we may feel an object having a certain orientation if its length is about twice its height. But to estimate the orientation exactly (see Section 4), we select the chains whose length are fourth times more than their height to be the seeds of text lines.

At the first iteration, the threshold D is set to zero, every chain is identical with a single connected component. Therefore, in this case, seeds of text lines coincide with anchor points in the literature [10], which are connected components elongated in specific orientations. At the next iterations, the threshold D is gradually increased, the seeds of text line will be the potential chains composed of several components. The potential chain can be considered as an expansion for the concept of anchor point. When the threshold D is smaller than the inter-line gaps, the components in a potential chain exactly belong to the same text line, hence the selected seeds are proper. These seeds are used to search the remaining components lying in their lines before the threshold is incremented at the next iteration.

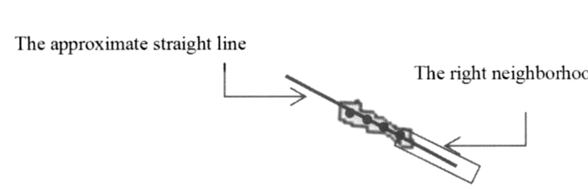
The use of the iteration and the incremented threshold takes the advantage that it can be adaptive to irregular gap distances in the documents. In the next section, we will describe the prediction whether a connected component belongs to the same text line with the seed and the complete text line construction.

4. Text Line Construction

After seeds of text lines are selected at a specific iteration, the text lines are grown by extending the seeds to the left and the right. Firstly, we truncate one section in the left end and another section in the right end of the seed (Fig. 5). The lengths of those sections are chosen by four times of the seed's height. In turn, each section is divided into four parts and the centers of gravity are sought for every part. The straight line which approximately crosses the four centers of gravity is found. The orientation at each end of the seed is the incline angle of the corresponding straight line. For each seed end, we define a directional neighborhood region as the rectangular area along the local orientation (Fig. 6). A neighbor component is predicted to belong to the same text line with the seed by the shortest distance criterion. The predicted component is integrated with the seed to form an extended seed, called an alignment. The local orientations are reevaluated for the new ends of the formed alignment. Another component is continuously sought from each new end until no component is detected anymore.

Under the extension of alignments, the conflict may appear when several right neighborhoods meet one component (similar situation to left neighborhoods). The solution for the conflict was suggested in some earlier works, such as by Likforman-Sulem and Faure in [10], where a set of rules is applied taking into account the quality of each alignment and neighboring components of the higher order. Actually, there is no the best accepted

(Fig. 5) Truncation



(Fig. 6) Orientation and neighborhood

solution for this kind of conflict. In this paper, for simplification, we join the conflicting component to the nearest alignment. The solution can be expanded by adding other quality measures of alignments used in the literature.

As stated earlier, at each iteration, new potential chains (also new seeds of text lines) are constructed from free components. These new seeds are extended accompanying with the adjusted local orientation. The adjustment of the local orientation during the extension helps fit the curvature. So the method is suitable for the considerably curved text lines.

5. Simulation results

In our experiment, the method was applied for various document images of address blocks, magazines, slogans and logos. Documents which contain multi-oriented, considerably curved text, as well as some big inter-word gaps were considered for the experiment. Document images were digitized at 100dpi and their sizes are from 200x300 to 500x700 pixels. There were totally 610 text lines.

Document Image Analysis is a complex system and it involves various problems: separation of text from graphics, segmentation of tables and forms, segmentation of math zones, partition of columns, extraction of text lines, words and characters, as well as character recognition. There has been many published works mentioning to different problems of Document Image Analysis. In this paper, we concentrated on extraction of text lines only. Therefore, in our experiments, we assumed that document images were single column pages and included no figures and tables. If any figures and tables appeared in a document image, they were removed manually before our algorithm was applied. The performance of the proposed method as well as some factors which affected the performance will be discussed in the following sections.

5.1 Effect of noise and small scripts

Because of imperfection in the image threshold process, some non-text components (or noise) appear and deteriorate the scheme performance. Thus, these components need to be filtered away before the algorithm is applied. Moreover, subscripts, superscripts, accents or commas should be also eliminated due to their dislocation from the text line and because they may be confusedly chosen as anchor points. These kinds of scripts as well as noise usually have small sizes. Therefore, small sized components are filtered away during the connected component extraction stage by the size thresholds. The size thresholds are chosen according to the average size of scripts. In our experiments, the size thresholds are one forth of the average script sizes, in which the sizes are measured by the height of components after smeared. Eliminated components will be put to the nearest extracted text lines when the process of the text lines extraction is completed.

5.2 Accuracy of the local orientation estimation

The accuracy of the local orientation estimation affects the performance of the extraction scheme. The accuracy of the local orientation estimation depends on the length of the left or the right truncated section used for the estimation. If the truncated section to be used is too short, the orientation computation will be not precise. Otherwise if the truncated section is too long, the resultant orientation will be likely a global orientation rather than a local orientation. To compromise on these effects, we choose the length of each truncated section is four times of its average height. Depending on the shapes of text lines, the accuracy of the local orientation estimation varies from several degrees to over ten degrees. We evaluate errors by applying transformations of rotation for sample sections and then comparing the estimated orientation with the rotation angles. If the sizes of components in the truncated section are regular and the section is relatively straight, the error of estimation can reduce to $\pm 2^0$. For the complex shapes of truncated sections, the errors are sometimes over $\pm 10^0$. It is proper to our vision because it is difficult to give an orientation for a complex shape.

5.3 Statistical results

Among the 610 text lines, 307 lines were straight lines on some magazines, which possibly include big inter-word gaps, and 303 text lines were taken from 20 logos and slogans, which contain considerably curved lines. The resultant text lines extracted by the algorithm are described as compact black clusters. See (Fig. 7) and

(a) The result corresponding to the Fig.1 (a) (b) The result corresponding to the Fig.1 (b)

(Fig. 7) Results of the proposed method
(Text lines are represented by compact clusters)

(a) Original Page (b) Page after removing the figure (c) Extraction Result

(Fig. 8) One another result illustrating the experiment on a magazine page

(a) An example for the broken error

(b) An example for the merged error

(Fig. 9) Errors (The left figure is the source document and the right figure is the result)

<Table 1> Statistical Results

Document Type	Extraction Cases		Number of Text Lines	Rate (%)	
Straight (Data Size 307)	Correct		301	98	
	Errors	Broken	2	1.3	2
		Merged	4	0.7	
Curved (Data Size 303)	Correct		285	94	
	Errors	Broken	11	3.6	6
		Merged	7	2.4	

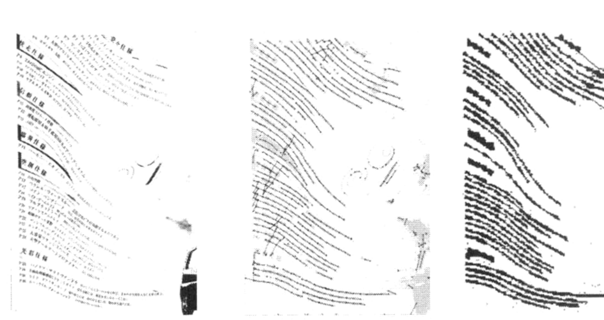
(Fig. 8) where results of extracted text line in our scheme are shown. We compared these resultant lines with the actual text lines to examine the accuracy of our method. This comparison is performed by manual inspection. To bring out a quantitative evaluation of the extraction method, we show the number of correct resultant lines and incorrect resultant lines together with corresponding errors. We divide the resultant text lines into three categories according to the kinds of errors: free error lines, broken lines, and merged lines. The free error lines are the correctly extracted lines. If a text line is broken into separate parts under the algorithm, it is called a broken line, see (Fig. 9) (a). If a text line is merged into another text line, it is a merged line, see the line "need" in (Fig. 9) (b). If a text line is both broken and merged, depending on the number of breaks and merged components, it will be considered as a broken error or a merged error. If the number of breaks is more than the number of merged components in a text line, the text line is a broken type. Otherwise, it belongs to the merged error type. The broken line error in (Fig. 9) (a) is resulted from the character "r" stands outside the neighborhood of the word "The". The merged line errors in (Fig. 9) (b) show up because the connected component "Peacemaker" is so long that it touches the component "eed" after it is smeared.

5.4 Comparisons with other methods

It is difficult to provide a complete comparison between the proposed method and other existing methods because extraction of curved text lines is a new area and there has not been a standard database for the comparison. However, we try to carry out comparisons with typical methods in some aspects.

In comparison with the work of H.Goto and H.Aso [9], although their extraction rate of straight text lines was provided, the extraction of curved text lines was only shown qualitatively in their work. Therefore the comparison was performed quantitatively on the straight lines but qualitatively on curved lines. The straight-line extraction rate of our method showed an equivalently high performance, our rate is 98% while theirs is 98.1%. For the extraction of curved text lines, the proposed method was qualitatively compared with H.Goto's method. The result is given on the following images:

The result of extraction in the work of Goto and Aso was presented by the lines connecting black points. It can be seen from the (Fig. 10) that there many breaks and crosses of text lines in Goto's result. In contrast, our proposed method showed a better result in which only one break occurred.



(a) Original Page (b) Goto's method (c) Our method

(Fig. 10) Qualitative Comparison with Goto's Method

〈Table 2〉 Comparison with Pal and Roy's method

Document Type	Extraction Rate (%)	
	Pal and Roy's method	Our proposed method
Straight Text Lines	91%	98%
Curved Text Lines	89%	94%

In the work of Pal and Roy [8], the authors described the results in the table of extraction distributions. For each text line, the authors defined an accuracy index, and the number of text lines according to certain accuracy was shown in a table. If we consider that text lines of accuracy over 96% is correctly extracted text lines, and text lines of accuracy below 96% are errors, the rate of correctly extracted text lines in their work will be 91% for the straight-line extraction (including single oriented and multi-oriented documents) and 89% for the curved-line extraction. These rates in our method are 98% and 94% for the straight-line extraction and the curved-line extraction, respectively. Despite of different databases, the extraction rates somewhat reflected the higher performance of the proposed method because we tried to use a general and random set of document images. Another advantage of the proposed method over the work of Pal and Roy is the applicability of our proposed method for different languages, while Pal and Roy's method can be used only for Indian characters. In our experiments, the method is applied for many English documents, several Indian and Japanese documents. The results showed that the method can work well with these languages.

6. Conclusions

The proposed method is rather effective for extracting text lines from many poorly structured documents. Our method has some advantages over other methods. Our proposed method is cable of handling variable character sizes. Moreover, this method is able to adapt with irregular gaps as well as to accept some big gaps within a text line. Because of this advantage, our method can be applied for address blocks, in which we may encounter the mentioned kinds of gaps. The method can also deal with considerably curved text lines. It was used successfully to extract many considerably curved text lines from logos and slogans in our experiment.

This method can be more improved and expanded to deal with overlapping/touching lines in the future. The traditional strategy to resolve overlapping lines in the previous grouping methods can be integrated to our method without much difficulty. In addition, we will enhance the method to process documents which comprise both left-to-right text lines and up-to-down text lines. This can be done by a parallel algorithm for both arrangements of text lines. Finally, we will try to add new criteria in the construction of text lines to resist noise caused by the low quality of documents.

References

[1] Shapiro V., Gluhchev G., Sgurev V., "Handwritten document image segmentation and analysis", Pattern Recognition Letters, Vol.14, No.1, pp. 71-78, 1993.

[2] Pu Y., Shi Z., "A natural learning algorithm based on Hough transform for text lines extraction in handwritten documents", In Proceedings of the 6 Intl. Workshop on Frontiers in Handwriting Recognition, Taejon, Korea, pp. 637- 646, 1998.

[3] Zahour, A., Taconet, B., Mercy, P., Ramdane, S., "Arabic hand-written text-line extraction", Proceedings of the 6th ICDAR, Seattle, pp. 281 - 285, 2001.

[4] Wong K., R. Casey, F. Wahl, "Document analysis systems", IBM Journal of Research and Development, Vol. 26, No. 6, 1982.

[5] Douglas J. Kennard, William A. Barrett, "Separating lines of text in free-form handwritten historical documents", Second International Conference on Document Image Analysis for Libraries (DIAL'06), pp. 12-23, 2006.

[6] Feldbach M., Tönnies K.D., "Line detection and segmentation in historical church registers", Proc. of ICDAR'01, Seattle, pp. 743-747, 2001.

[7] H. Yan, "Detection of curved text path based on the Fuzzy Curve-Tracing (FCT) algorithm", in Proc. 6th Int. Conf. Document Analysis Recognition, pp. 266 - 269, 2001.

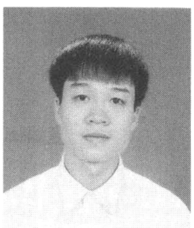
[8] U. Pal and Partha Pratim Roy, "Multioriented and curved text lines extraction from Indian documents", IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics, Vol. 34, No. 4, August 2004.

[9] H.Goto and H. Aso, "Extracting curved lines using local linearity of the text line", Int. J. Doc. Anal. Recognit., Vol. 2, pp. 111 - 118, 1999.

[10] Likforman-Sulem L., Faure C., "Extracting lines on handwritten documents by perceptual grouping", Advances in Handwiting and drawing: a Multidisciplinary Approach, C. Faure, P. Keuss, G. Lorette, A. Winter (Eds), pp. 21-38, Europia, Paris, 1994.

[11] Laurence Likforman-Sulem, Abderrazak Zahour, Bruno Taconet, "Text line segmentation of historical documents:

a survey", Special Issue on Analysis of Historical Documents, International Journal of Document Analysis and Recognition, Springer, Vol. 9, No. 2-4, pp. 123-138, 2007.



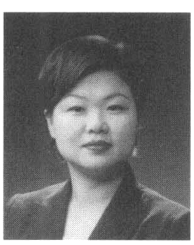
Noi Bai Nguyen

e-mail : noibai@osp.chungbuk.ac.kr

2005년 Vietnam National University, College of Technology (공학사)

2005년~현재 충북대학교 전기전자컴퓨터공학부 석사 과정

관심분야 : 영상 처리, 영상 인식, 3차원 입체영상 디스플레이



윤 진 선

e-mail : jsyoon7889@lycos.co.kr

1997년 충북대학교 정보통신공학과(공학석사)

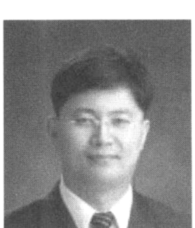
2002년 충북대학교 정보통신공학과(공학박사)

1993년~1995년 옥천공동직업훈련원 정보통신 교사

2006년 충북대학교 전기전자컴퓨터공학부 초빙전임강사

2006년~2007년 충북대학교 BK21충북정보기술사업단 박사후연구원

관심분야 : 광암호화, 정보보안, 광패턴인식, 영상처리



송 영 준

e-mail : songyjorg@dreamwiz.com

1996년 충북대학교 정보통신공학과(공학석사)

2004년 충북대학교 정보통신공학과(공학박사)

1996년~1998년 LG전자 주임연구원

1998년~2000년 하이닉스반도체 주임연구원

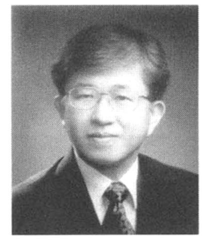
2000년~2003년 한국전자통신연구원 홈네트워크팀 선임연구원

2004년~2005년 프리즘테크 선임연구원

2005년~2006년 충북대학교 전기전자컴퓨터공학부 박사후연구원

2006년~현재 충북대학교 충북BIT연구중심대학육성사업단 초빙조교수

관심분야 : 영상처리, 컴퓨터비전, 얼굴 인식, USN



김 남

e-mail : namkim@chungbuk.ac.kr

1983년 연세대학교 전자공학과(공학석사)

1988년 연세대학교 전자공학과(공학박사)

1992년~1993년 미국 Stanford 대학교 방문 교수

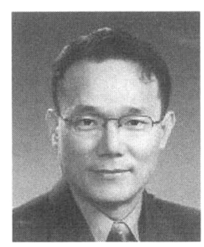
2001년~2002년 미국 California Technology Institute 방문교수

1989년~현재 충북대학교 전기전자컴퓨터공학부 교수

1999년~현재 컴퓨터 정보통신연구소 연구원

2005년~현재 충북BIT연구중심대학육성사업단 단장

관심분야 : 광정보처리, 광통신, 3차원 입체영상 디스플레이, 이동통신 및 전파



김 용 기

e-mail : ykkim153@chungbuk.ac.kr

1988년 독일 베르린 자유대 물리학부(물리학 디플롬)

1992년 독일 베르린 공대 물리학부(이학박사)

1992년~1995년 연세대학교 박사후연구원

1995년~현재 충북대학교 천문우주학과 교수

2002년~2006년 충북대학교 천문대장

관심분야 : 천문정보처리, 영상처리, 전파환경, 이동통신 및 전파