Journal of The Korea Society of Computer and Information
Vol. 18, No. 12, December 2013
www.ksci.re.kr
http://dx.doi.org/10.9708/jksci.2013.18.12.065

스캔된 문서에서의 도장 검출

유 견 아 *, 김 경 혜*

Seal Detection in Scanned Documents

Yu Kyeonah *, Kim Kyung-Hye*

요 약

디지털 시대의 도래에 따라 문서들이 기록 보관되기 위해서 혹은 네트워크를 통해 전송되기 위해서 스캔되는 경우가 많아졌다. 스캔된 문서에서 가장 큰 비중을 차지하는 것은 텍스트이며 텍스트 이외에는 문서 작성자를 나타내는데 사용되는 도장이 가장 많이 포함되어 있다. 스캔된 문서의 중요성이 부각되면서 스캔된 문서로부터 텍스트를 인식하는 연구는 많이 진행되어 상용화된 제품도 개발된 것에 비해 문서가 포함하고 있는 도장에 대한 정보는 버려지고 있는 실정이다. 본 논문에서는 도장이 포함된 컬러 혹은 흑백 문서 영상에서 도장 영역을 검출하여 도장 이미지를 저장하는 방법을 연구한다. 스캔된 문서에서 도장의 외곽선 후보만을 남기고 다른 기타 구성 요소를 제거하는 전처리 과정과 도장의 특징을 이용하여 남은 요소 중에 최종 관심 영역을 선정하는 방법을 제안한다. 또한 검출된 관심 영역의 도장 정보가 텍스트와 겹친 이미지인 경우에는 템플릿 매칭을 통해 데이터베이스로부터 가장 유사한 도장을 찾아 대신 저장할 수 있게 한다. 구현된 시스템은 학교에서 일반적으로 생성되는 여러 유형의 문서들을 대상으로 검증하고 그 결과를 분석한다.

▸ Keywords : 도장 영역 검출, 도장 정합, 기울기 보정

Abstract

As the advent of the digital age, documents are often scanned to be archived or to be transmitted over the network. The largest proportion of documents is texts and the next is seal images indicating the author of the documents. While a lot of research has been conducted to recognize texts in scanned documents and commercialized text recognizing products are developed as highlighted the importance of the scanned document, information about seal images is discarded. In this paper, we study how to extract the seal image area from the color or black and white document containing the seal image and how to save the seal image. We propose a

•제1저자 : 유견아 •교신저자 :유견아
•투고일 : 2013. 9. 27, 심사일 : 2013. 10. 17, 게재확정일 : 2013. 11. 20.
* 덕성여자대학교 컴퓨터학과(Dept. of Computer Science, Duksung Women's University)
※ 본 연구는 2012년도 덕성여자대학교 교내연구비 지원에 의해 수행되었다.

preprocessing step to remove other components except for the candidate outlines of the seal imprint from scanned documents and a method to select the final region of interest from these candidates by using the feature of seal images. Also in case of a seal imprint overlapped with texts, the most similar image among those stored in the database is selected through the template matching process. We verify the implemented system for a various type of documents produced in schools and analyze the results.

▸ Keywords : Seal image detection, Seal matching, Incline compensation

I. 서 론

디지털화되는 문서의 양이 늘어남에 따라 해당 디지털 영상으로부터 텍스트 및 그림, 도표 등의 정보를 올바르게 추출 및 인식하는 알고리즘의 꾸준히 연구되어 왔으며[1][2] 텍스트를 인식하는 기능은 이미 제품화되어 사용되고 있다[3]. 도장은 우리나라 및 동양권에서 문서작성자 인증의 방법으로 많이 사용해 왔기 때문에 대부분의 문서에는 도장 이미지를 포함하고 있는데 도장 정보에 대해서는 별도의 그래픽 도구를 이용하여 수동으로 추출하고 저장해야 하는 번거로움이 있다. 본 논문에서는 이를 해결하기 위해 도장이 텍스트나 표와 같은 다른 개체와 구분되는 특징을 이용하여 스캔된 문서로부터 도장 이미지 영역을 검출하고 도장 이미지를 저장하는 방법을 연구하고 구현한다.

시스템의 입력은 컬러 혹은 흑백으로 스캔된 문서이다. 컬러 문서인 경우에는 컬러 정보를 이용하여 붉은 색 영역이 아닌 부분들을 제거하여 흑백 문서에 비해 간결해진 형태로 처리 과정이 시작되며 이후의 과정은 흑백 문서의 경우와 동일하다. 문서에서 도장을 찾는 일의 가장 기본은 도장의 외곽선을 찾는 것이다. 문서로부터 선분을 추출하기 위해 우선 모폴로지와 스무딩을 통하여 텍스트와 같은 요소들을 블러링하는 전처리 과정을 거친다. 선분들이 검출되면 데이터베이스에 저장되어 있는 도장 크기의 최대치와 최소치를 이용하여 해당하지 않는 선분들을 제거하고 남아 있는 선분들로부터 근사화된 다각형 가운데 가장 도장의 특성에 가까운 다각형을 관심 영역으로 정한다. 이와 같이 검출된 도장 이미지가 텍스트를 포함하거나 질이 심하게 떨어지는 영상 등, 저장 후 정보로서의 가치가 떨어지는 경우에는 템플릿 매칭을 통해 데이터베이스로부터 가장 유사한 이미지를 선택해 저장하는 정합 알고리즘을 개발한다. 해당 시스템은 OpenCV 라이브러리를 사용해 개발되고 테스트되었다. 기존의 도장 영역 추출 방식들이 도장 인식을 위해 정합 알고리즘과 연계하여 정확도를 높이고자 했던 것과는 달리 본 논문에서 제안하는 방법은 신속하게 도장 후보 영역을 추출하고 적합하지 않은 결과에 대해서만 복잡한 정합 단계를 수행하도록 한 것이 차별화되는 점이다.

본 논문의 구성은 2장에서는 도장 검출 및 인식에 관한 관련 연구를 살펴보고 3장에서는 전체 시스템의 개요를 설명한 후 4장과 5장에서 각각 도장 영역 추출과 도장 정합을 위해 제안하는 방법을 설명한다. 6장에서는 해당 시스템을 시뮬레이션한 결과와 그 분석에 대해 기술하고 마지막으로 7장에서 향후 연구 방향 및 결론으로 마무리한다.

II. 관련 연구

도장이나 인장은 우리나라 뿐 아니라 동양의 여러 나라에서 본인 인증의 수단으로 오랫동안 사용되어 왔기 때문에 도장 인식 및 정합에 관한 연구가 로고 및 표지판 등에 대한 연구와 더불어 패턴 인식의 한 분야로써 꾸준히 진행되어 왔다.

문자와 도장이 다른 색으로 되어 있다는 특성이 있기 때문에 이를 이용한 연구가 다수 있다. [4]에서는 컴퓨터 비전에서 퍼지 적분을 이용하여 특정 컬러 클러스터를 추출하는 방법을 응용하였고 [5]에서는 컬러 클러스터를 이용하여 후보를 분류하고 기하학적 특성과 컬러 특성을 이용하여 색이 다른 로고와 텍스트로부터 스탬프 이미지를 분류하였다. [6]에서는 학습 단계를 통하여 비공식적인 모양의 스탬프 이미지를 인식하는 알고리즘을 제안하였는데 이와 같이 컬러 분석을 이용하는 방법은 디지털화된 문서 전체에 적용할 수 없다는 한

계가 있다.

〔7〕에서는 컬러로 도장이 구분되지 않는 저품질의 문서를 입력으로 받아 연결된 모서리(edge) 정보를 이용하여 문서 내 타원형의 도장을 찾아내는 연구를 수행하였다. 〔8〕에서는 전경의 공간 밀도를 계산하여 회전, 잡음, 스케일에 관계없이 로고를 찾아내는 간단한 방법을 제안했으나 도장의 경우와 같이 내용이 일반 텍스트인 경우에는 적용하기 힘들다.

도장의 형태적 정보를 추출한 뒤 내부의 문자 인식을 통해 최종적으로 일치하는 도장 영상을 뽑아내는 연구 〔9〕와 같이 최근에는 도장 인식을 위해 문자 인식 알고리즘을 사용하는 추세를 보이고 있다. 〔10〕에서는 텍스트와 그림, 도표 등의 여러 구성 요소가 포함된 문서 내에서 도장 영역만을 분리해 내기 위하여 GHT와 도장 안에 있는 알파벳 문자 정보를 사용하였으며 Chao Ren 등 〔11〕〔12〕은 중국어로 구성된 도장에서 글자 간격별로 구획화를 수행 후 문자의 기울기를 바로잡아 도장 내 문자를 인식하는 방법을 제안하였는데 이와 같이 방식들은 도장의 진위여부를 판별하는데 초점을 맞춘 것으로 본 연구의 목표에는 과도한 분석 방법이라고 할 수 있다.

〔13〕에서는 모양과 레이아웃 정보를 기반으로 도장을 분류하고 등록하는 방법을 제시하였다. 일반적으로 도장의 모양이 사각형, 원형, 타원형이라는 사실에 착안하여 분류하고 그 안에 글씨가 이산적으로 분포하는 성질을 이용하였는데 본 연구에서 도장의 정합을 하기 위해 이와 유사한 접근 방식을 사용하는데 〔13〕에서는 문서에 있는 도장 영역을 추출하는 과정은 없다.

국내에서는 도장 영상의 히스토그램을 이용해 잡음을 최소화한 뒤 이진화를 거쳐 원형 비교 방법을 통해 도장 영상 간 정합도를 구한 연구가 진행된 바 있으며〔14〕, 영상에 대해 이진화와 평활화를 거친 뒤 ART1 알고리즘을 사용하여 도장 영역을 인식한 연구 〔15〕가 있다. 두 연구 모두 스캔된 문서에서의 도장 영역 검출에 대한 연구가 아니며 정합 방법에 있어서도 본 연구와 차이가 있다.

III. 도장 검출 시스템

본 논문이 제안하는 시스템의 입력은 디지털 형태로 변환된 문서 영상이며 문서에는 한 개의 도장이 미리 정해지지 않은 임의의 위치에 있다는 것을 가정한다. 도장의 윤곽선을 찾아내기 위해 허프 직선 변환 알고리즘을 이용하는데 효과적인 결과를 얻기 위해 텍스트 등의 기타 요소를 제거하는 전처리 과정을 거치게 된다. 전처리 과정에서는 이진화, 모폴로지 연산 및 스무딩 등의 전처리 과정을 통해 문서 영상내에서 도장의 외곽선과 같은 선분이외의 기타 영상을 블러링한다. 단, 컬러 문서의 경우에는 임계값을 이용하여 붉은 색이 아닌 구성요소들을 제거하고 시작한다. 허프 직선 변환 알고리즘을 이용한 결과에 임계값을 적용하여 도장의 후보 가능 영역을 표나 텍스트와 같은 기타 구성요소로부터 구분한다. 남은 선분들로부터 특징점 추출과 다각형 근사화를 통해 윤곽선을 추출하고 이렇게 생성된 도장 후보 영역으로부터 도장 내부 픽셀 정보를 이용하여 최종 도장 이미지 영역을 선정하고 전체 영상에서 관심 영역(ROI, Region of Interest)만을 잘라내는 작업을 수행하여 도장 영역 검출을 완료한다. 전체 시스템 개요를 나타내는 그림 1에서 실선 상자 부분이다.

이와 같이 추출된 ROI는 도장 이미지로 저장되게 되는데 도장 이외에도 텍스트를 포함하거나 질이 심하게 떨어지는 영상 등, 저장 후 정보로서의 가치가 떨어지는 경우에 데이터베이스에서 매치되는 도장을 찾아 저장하는 후처리 과정을 선택적으로 수행할 수 있게 한다. 허프 원 변환 알고리즘을 통해 도장이 원인지 사각형인지 판단해 내고 가로, 세로 비율을 통해 형태를 알아낸다. 제안하는 방법으로 기울기를 보정한 후, 템플릿 매칭을 통해 정합을 수행한다. 이 때 형태와 비율 정보는 데이터베이스에서 후보 도장을 필터링하는데 이용된다. 이 과정은 그림 1의 점선 상자 부분에 해당한다.

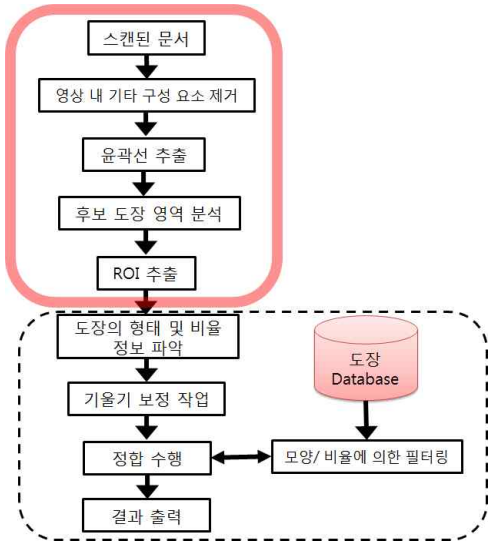


그림 1. 시스템 흐름도
Fig. 1. System Flowchart

3.1 도장 이미지 영역 추출

도장과 구분되어야하는 문서의 구성 요소로는 여러 크기의

텍스트, 표, 다른 이미지들이 있는데 도장이 다른 구성 요소들과 갖는 차별성을 이용하여 도장이 포함되어 있는 영역을 추출해나가도록 한다. 이 과정은 그림 2와 같이 크게 3단계로 구성되어 있다.

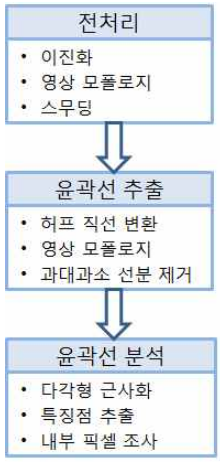


그림 2. 도장 이미지 영역 추출 단계
Fig. 2. Seal Image Region Extraction Steps

① 전처리 과정

도장 이미지 영역을 추출하기 위해서는 도장의 윤곽선을 찾는 것이 가장 중요하다. 그런데 도장 찍힘의 특성상 끊김이 존재하므로 허프 직선 변환 함수를 사용하여 윤곽선을 추출할 때 하나의 선분으로 인정하는 직선 성분의 최소 길이를 가장 작은 도장의 크기의 10%로 정하였다. 이와 같이 하는 경우의 문제점은 텍스트들로부터도 선분이 많이 추출된다는 것인데 이를 방지하기 위해 허프 직선 변환 함수를 적용하기 전, 텍스트 부분을 블러링하기 위한 전처리를 실행한다. 전처리의 첫 번째는 이미지 처리 분야에서 배경을 구분하기 위해 일반적으로 사용되는 이진화 과정으로 각각의 픽셀값을 임계값과 비교해서 임계값 이하이면 0, 임계값보다 크면 1로만 표현하는 것을 말한다. 컬러와 흑백 문서에 따라 임계값을 달리하여 이진화 작업을 수행한다. 이진화 결과 문서에서 짧은 윤곽선들을 없애주고자 침식 연산을 사용한다. 침식 연산은 커널 아래에서 국지적 최소값을 계산하는 것과 유사하여 커널 아래에서 최소값을 취하므로 결과적으로 영상 내부에서 어두운 영역이 확장되게 되고 문서 내 작은 텍스트들이 확장되어 두꺼워지게 된다. 이 과정은 도장의 내부 영역도 블러링 하여 나중에 기타 이미지와 구분되는 성질로 이용할 수 있다. 보다 정확한 윤곽선 검출을 위한 스무딩 처리로 전처리 과정을 마무리한다.

② 윤곽선 추출

전처리 결과로부터 선을 추출하기 위해 허프 직선 변환 함수를 사용한다. OpenCV에서 제공하는 cvHoughLines2() 함수는 주어진 파라미터 값에 따라 임의 선택된 선분을 리턴해 주는데 본 논문에서는 파라미터 중에 방법은 확률적 허프 변환으로 하여 계산상 효율적으로 선분을 찾아내도록 하였으며 직선의 최소 길이를 나타내는 파라미터와 일직선상에 있는 두 직선 성분을 하나의 직선 성분으로 간주하기 위한 최대 거리를 나타내는 파라미터는 저장된 도장의 최대, 최소 지름의 크기를 기준으로 정하였다. 즉, 도장 윤곽선에 일반적으로 나타나는 끊김을 고려하여 직선성분의 최소 길이는 가장 작은 도장 크기의 10%로 하여 짧은 선분도 허용하게 하였으며 일직선상에 있는 두 직선 성분을 하나의 직선으로 간주하기 위한 최대 거리는 가장 작은 도장 크기의 15%로 하여 상당 부분의 끊김도 허용하게 한다. 허프 직선 변환 결과에 작은 영역들을 지워 단순화시켜 특징점 추출이 용이하도록 침식 연산을 수행하고 윤곽선 분석에 사용될 윤곽선만 남기고자 도장의 최대, 최소 크기의 범위에서 벗어나는 선분을 제거하는 과정을 거친다.

③ 윤곽선 분석

윤곽선 분석은 추출한 윤곽선들을 다각형 근사화 하고 근사화 된 윤곽선에서 특징점을 추출함으로서 시작된다. 다각형에 속하는 특징점 사이의 거리를 계산하여 두 특징점 사이의 거리가 최대 도장의 길이보다 긴 경우에는 값을 0으로 저장하여 앞으로의 과정에서 고려되지 않도록 하고 그 이외의 특징점 사이의 최대 거리를 지름으로 한 원을 찾는다.

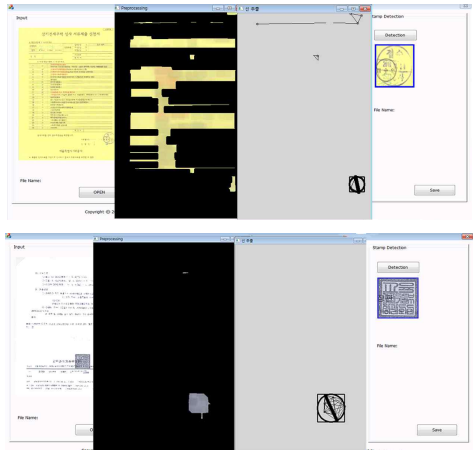


그림 3. ROI 추출 단계
Fig. 3. Step of Extracting the ROI

허프라인의 특성상, 도장이 아닐 경우, 안에 채워진 부분

이 검게 처리가 되어있다. 그러므로 최대 특징점 사이의 거리를 찾아서 원을 그리고 그 내부의 픽셀들을 조사하여 검은 영역이 없을수록 도장 영역일 확률이 높으므로 검정 영역의 비율이 가장 낮은 원을 선택한다. 이와 같은 방식으로 구해진 원을 감싸는 사각형을 그려 4방향으로 스캔해가며 최상단의 좌표(topx, topy)와 최하단의 좌표(bottomx, bottomy), 최우측(rightx, righty)과 최좌측(leftx, lefty)의 좌표를 각각 구하여 이로 얻어진 사각형을 ROI로 저장한다.

3.2 도장 정합

그림 3과 같이 검출된 ROI 안에 도장 이외 텍스트 등을 포함하고 있는 경우가 있다. 도장 정보가 활용되기 위해서는 이를 제거한 정확한 도장이 저장되어야 할 필요가 있으므로 템플릿 매칭에 의한 정합을 실행하여 ROI에 있는 도장과 가장 가까운 도장 원본을 도장 데이터베이스로부터 찾는 것을 시도한다. 이 과정은 크게 도장의 형태 정보 파악, 기울기 보정, 정합의 3단계로 이루어져 있다.

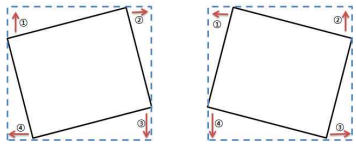
① 형태 정보 파악

윤곽선 검출 결과에서 도장의 형태를 파악하기 위하여 허프 원 변환 함수를 사용한다. OpenCV에서 제공하는 cvHoughCircles() 함수는 주어진 파라미터 조건을 만족하는 모든 원을 검출하여 해당 원의 위치와 검출된 원의 총 개수를 반환한다. 따라서 외곽선의 형태가 사각형일 경우 반환되는 원의 개수는 0이므로 이 성질을 이용하여 도장의 형태 정보를 파악한다. 즉 반환되는 원의 개수(이하 k)가 0이면 사각형, k가 1이면 원형의 도장으로 정의한다는 것이다. 이와 같이 획득한 정보에 ROI를 추출할 때 얻은 4모서리의 좌표를 이용하여 높이와 너비의 비율을 구해 해당 도장의 모양을 결정한다.

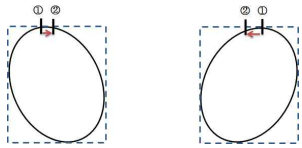
② 기울기 보정

도장은 찍힐 때 마다 각도의 차이가 발생할 수밖에 없으므로 정합 단계 전에 기울기를 보정한다. 도장은 15도 이상 기울여 찍히지는 않았다고 가정한다. 도장 이미지 영역을 추출할 때 얻은 4모서리의 좌표를 사용하여 도장의 형태에 따라 다른 기울기 보정 알고리즘을 적용하였다.

사각형 도장에서 (lefty 〈 righty) and (bottomx 〈 topx)의 조건이 만족하면 시계방향으로 보정하고 (lefty 〉 righty) and (bottomx 〉 topx) 조건이 만족하면 시계 반대방향으로 보정한다. 본 논문에서 가정한 바와 같이 도장은 15도 이하로 기울어져 있지 않다는 점에 착안하여 4개의 화살표 중 가장 짧은 거리를 기준으로 하여 기울기 보정을 수행한다. 그림 4 (a)의 좌측 예 경우에 도장의 각 모서리를 ②만큼 시계 방향으로 회전을 수행하고, 우측 예 경우에는 ①의 길이만큼 반시계 방향으로 회전을 수행한다.



(a) 사각형 도장이 기울어진 2가지 경우



(b) 타원형 도장이 기울어진 경우

그림 4. 기울기 보정

Fig. 4. Compensating the tilt

타원형의 경우에는 가장 최상위점인 (topx, topy)를 ROI의 너비 중앙에 오게끔 영상 변환을 함으로써 기울기 조정을 수행한다. 그림 4 (b)에서 ①은 도장의 최상위점인 (topx, topy)를, ②는 ROI 너비의 중앙점을 각각 나타낸다. 타원형의 도장이 왼쪽으로 기울어진 경우에는 ①이 ②보다 왼쪽에 위치하고, 도장이 오른쪽으로 기울어진 경우에는 ①이 ②보다 오른쪽에 위치하므로 이 정보를 이용하여 해당 도장이 어느 쪽으로 기울어져 있는지를 파악하여 기울기 조정을 수행한다. 원형 도장의 경우에는 도장의 상하좌우에 관한 정보가 전혀 없기 때문에 ±15도 범위 내에서 지그재그 형태로 1도 씩 회전하면서 정합률이 가장 높은 DB 데이터를 선택하는 방식을 취한다.

③ 도장 정합

도장의 검출 및 인식에 관한 최근 연구 동향을 보면 연구의 목적이 대부분 도장의 진위여부를 가리는데 있기 때문에 도장안의 문자 인식을 통해 정합률을 극대화하려는데 초점이 맞추어져 있어 과도하게 정합 알고리즘이 복잡하게 되고 연산 시간이 길어지게 된다. 본 연구에서는 정합의 목적이 DB에서 가장 유사한 도장을 찾아내는 데에 있기 때문에 상대적인 비교가 중요하다는 것에 착안하여 단순 템플릿 매칭으로 정합을 시도한다. 그러나 비트별 비교를 기반으로 하는 템플릿 매칭은 여전히 시간 소모적인 과정이기 때문에 정합 연산을 최소화할 수 있도록 하는 두 가지 면에서 차별화된 알고리즘을 사용하였다. 첫째, 도장 영상에 대해 위에서 설명한 단계를 거치게 되면 시스템은 입력 영상 내 도장의 크기와 모양 정보,

가로와 세로의 비율 등 도장에 대한 여러 가지 정보를 알아내게 된다. 이러한 정보를 바탕으로 DB 데이터들에 대해 가지치기를 수행하여 실제 정합을 수행하는 경우의 수를 대폭 줄인다. 둘째, 도장이 찍힐 당시 압력 차이 정도나 번짐 현상, 기울기의 오차, 잡음 등에 의한 영향을 받기 쉽기 때문에 단순히 입력 도장 영상과 DB 데이터들에 대해 픽셀 대 픽셀 별로 값이 일치하는가를 확인하는 대신에 도장 영상을 구획화하여 다음과 같이 정합률을 구한다.

step 1: 각 셀의 흰색 픽셀에 대한 검정색 픽셀의 비율을 구한다.
step 2: DB 데이터의 각 구획 내 검정색 픽셀 비율과, 해당 구획의 위치에 대응되는 입력 영상 내 구획 내 검정색 픽셀 밀도 간의 차이를 구한다.
step 3: 해당 차이가 지정한 임계값 이내에 있다면 trueCnt를, 임계값을 벗어나면 falseCnt를 증가시킨다.
step 4: falseCnt에 대한 trueCnt의 비율을 100단위로 정규화하여 입력 영상과 해당 DB 영상 간의 정합률을 구한다.

그림 5는 문서에서 추출한 도장(좌측)과 DB 데이터(우측)을 18x18로 구획화하여 비교하고 falseCnt와 trueCnt를 조정하는 예를 보여준다.

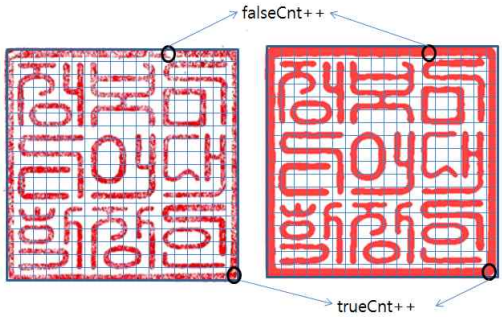


그림 5. 구획화에 의한 정합
Fig. 5. Matchinig by Division

IV. 시뮬레이션 결과 및 분석

입력 영상은 유사한 크기를 가지는 24 bit 트루 컬러로 스캔된 공문서들이고, Windows 7 32bits 운영체제 상에서 MFC를 이용한 C++언어와 OpenCV 라이브러리를 사용해 개발한 시스템으로 실험을 수행하였다. 본 실험에 사용된 데이터베이스는 총 100개의 순수 도장 영상 데이터를 가지고 있으며 모양, 크기, 구획화한 자료와 함께 저장되어 있다. 테스트 문서 집합은 컬러 문서의 비율은 59%, 표가 포함되어 있는 비율은 약 30%, 표나 선이 포함되어 있는 비율은 약 50% 정도로 학교에서 생성되는 공문서의 다양한 형태를 비율에 맞추어 구성하였다.

그림 6은 도장 영역을 추출하는 시뮬레이션 결과 화면이다. 그림 6의 맨 좌측 윈도우는 입력받은 문서를 디스플레이하고 중간은 전처리 과정 결과를 보여주며 우측 윈도우에 추출된 도장 이미지 영역이 디스플레이된다. 이렇게 디스플레이된 결과에 저장하기를 선택하도록 하여 저장하기가 선택되면 추출이 성공한 것이고 저장하기가 선택되지 않은 경우에 정합을 수행하게 된다.

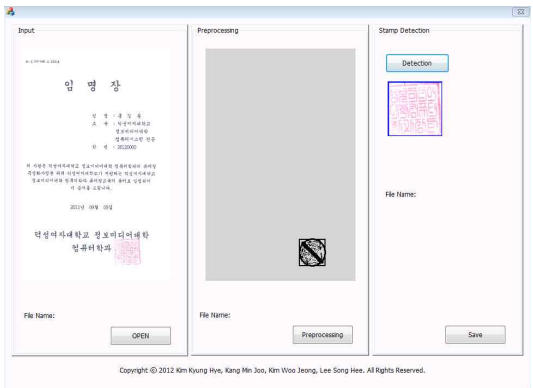


그림 6. 시뮬레이션 결과 화면
Fig. 6. Simulation Results Display

테스트 집합의 문서에 대해 이와 같은 시뮬레이션을 반복한 결과, 102건 중에 11건에 대해 잘못된 ROI를 리턴하여 11%의 오인식률을 보였다. 테스트 집합의 구성에 따라 오인식률은 차이가 나는 것이 당연하기 때문에 오인식 결과를 초래하는 문서들에 대한 분석이 없이는 오인식률 자체의 의미가 없다. 오인식 결과의 예를 분석하면 다음과 같다. 우선 컬러 문서의 경우에는 모든 경우에 ROI 추출에 성공하였는데 이는 컬러 정보를 이용하여 허용 임계치 이외의 구성 요소들을 제거하기 때문에 흑백문서에 비해 매우 간단한 형태의 문서에 대해 분석을 시작하게 되기 때문이다. 흑백 문서의 경우에는 크게 다음과 같은 오인식 분류가 있는데 1. 구성 요소에 회사 로고와 같이 도장과 유사한 이미지가 있는 경우, 2. 도

장의 경계가 너무 희미하게 찍힌 경우, 3. 텍스트가 밀집한 부분의 일부가 도장의 외곽선을 형성하는 경우이다. 본 연구의 방법이 우선 도장의 외곽선을 추출하고 그 내부는 글씨로 채워 있다는데에 기반한 방법이기 때문에 초래되는 오인식 경우들이다.

도장 이미지 추출 단계의 결과가 저장하기에 적합하지 않은 경우에 데이터베이스에 저장된 도장 이미지와 정합을 시도한다. 이 부분에 대한 실험은 일단 데이터베이스에 저장되어 있는 도장 이미지 중에 실제 존재하는 도장들을 각도와 찍히는 압력을 다양하게 찍은 이미지들에 대해 실행되었다. 충분한 실험 후에 도장 이미지 영역 추출 결과와 연동하여 도장 이미지가 다른 구성 요소와 오버랩되어 바로 저장하기가 되지 않은 도장 이미지에 대해 테스트되었다.

본 연구의 정합 과정의 특징은 도장 정보를 바탕으로 DB 데이터들에 대해 가지치기를 수행하여 실제 정합을 수행하는 경우의 수를 대폭 줄이는 것과 비트별이 아닌 일정 구획별로 템플릿 매칭을 하는 것이다. 실험 결과는 가지치기에 의해 평균 전체 데이터베이스 도장의 8.3%에 대해서만 템플릿 매칭을 하는 것으로 나왔다. 실제로 본 시스템이 정합을 한번 실행하는데 소요되는 평균 시간은 28ms인데 반해 도장 정보 비교에 의해 가지치기를 수행하는 시간은 평균 1ms에 불과해 많은 시간이 절약됨을 확인할 수 있었다. 또한 표 1은 픽셀별로 정합하는 것보다 도장 세선의 크기에 따라 구획화 하여 정합하는 것의 결과가 더 우수한 것을 보여준다. 40x40의 원본을 그대로 비교한 경우와 18x18으로 구획화한 경우를 비교하였을 때, 같은 도장끼리의 매칭률은 구획화한 경우 증가하였고 다른 도장과의 매칭률은 작지만 감소하였다. 같은 도장에 대한 매칭률이 비교적 낮은 이유는 대부분의 비교 도장의 상태가 좋지 않은 경우에만 정합을 시도하기 때문인데 다른 도장끼리 비교한 결과에 비해 상대적인 수치가 중요하기 때문에 실제 저장될 도장을 찾는 문제에서의 오탐률은 이에 비해 훨씬 좋다. 표 2는 정합 단계별로 잘못 판단되는 비율이다.

오판단률을 보상하기 위해 시스템은 상위 몇 개의 데이터를 최종 결과로 출력하게 되기 때문에 그림 7과 같이 좌측 문서의 도장 영상에 대해 상위 3개의 데이터를 출력한 결과를 보여주는 것이다.

표 1. 매칭률 비교
Table 1. Comparison of Matching Ratio

	같은 도장	다른 도장
픽셀별 비교	76.6 %	59.0 %
구획별 비교	87.6 %	57.4 %

표 2. 정합 단계별 오판단률
Table 2. Ratio of wrong answers for each step of the matching process

비율 정보 오판단율	모양 정보 오판단율	정합 오판단률
3.37%	1.12 %	5.8%

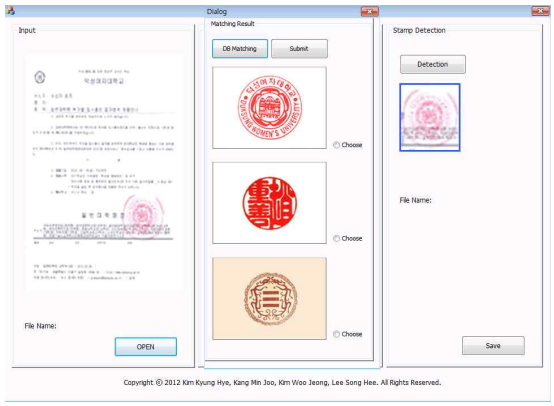


그림 7. 매칭되는 데이터베이스 자료
Fig. 7. Database data matched

V. 결론

본 연구에서는 스캔된 문서에서 도장 이미지 영역을 찾아 저장하는 시스템을 개발하였다. 기존의 방법 대부분은 영역 검출을 도장 정합에 기반하는데 비해 본 연구에서는 간단한 도장 이미지 영역 검출 알고리즘을 이용하고 시간이 많이 걸리는 정합 과정은 필요한 경우에만 사용하도록 하여 효율을 꾀하였다. 102개의 문서를 테스트한 결과 89%의 성공률을 보였으며 도장 정합 기능을 제공하여 보다 정확한 도장 영상을 저장할 수 있도록 하였다.

본 시스템은 도장 이미지 영역 검출 부분과 도장 정합 부분은 독립적으로 개발되어 전자의 출력이 후자의 입력으로 사용되도록 되어 있는데 도장 이미지 영역 검출의 3단계에서 후보 다각형 모두에 대해 도장 정합을 시도하여 가장 높은 정합률을 보이는 영역을 선택하도록 하면 오인식률을 감소시킬 수 있을 것이다. 또한 시스템의 목적에 따라 다른 도장 정합 알고리즘으로 대치되어 사용될 수 있어 정교하고 복잡한 정합 방식을 사용하는 경우에도 도장 영상 추출 단계에서 제공하는 후보 영역에 대해서만 정합이 실행되므로 효율적 수행이 가능하다. 이 방법을 사용하면 여러 개의 도장을 포함하는 문서에 대한 도장 인식도 가능해 질 것이다.

참고문헌

〔1〕 L.A. Fletcher and R. Kasturi, "A robust algorithm for text string separation from mixed text/graphics images", IEEE Transactions on Pattern Analysis and Machine Vol 10(6), pp 910 - 918, 1988.

〔2〕 V. Wu, R. Manmatha, and E.M. Riseman, "Textfinder: an automatic system to detect and recognize text in images", IEEE Transactions on Pattern Analysis and Machine Intelligence Vol 21(11), pp 1224 - 1229, 1999.

〔3〕 J. Fitzpatrick, "Five Best Text Recognition Tools", http://lifehacker.com/5624781/five-best-text-recognition-tools, 2010.

〔4〕 A. Soria-Frisch, "The fuzzy integral for color seal segmentation on document images", International Conference on Image Processing, vol. 1, pp. 157-160, 2003.

〔5〕 B. Micenkova and J. van Beusekom, "Stamp Detection in Color Document Images", Proceedings of the International Conference on Document Analysis and Recognition, pp 1125-1129, 2011.

〔6〕 P. Forczmanski, "Stamp detection in scanned documents", Annales UMCS, Informatica, pp 61-68, 2010.

〔7〕 G. Zhu, S. Jaeger, and D. Doermann, "A Robust Stamp Detection Framework On Degraded Documents", Proceedings of the SPIE Conference on Document Recognition and Retrieval, pp 1-9, 2006.

〔8〕 T. D. Pham, "Unconstrained logo detection in document images", Pattern Recognition 36 (12), pp. 3023-3025, 2003.

〔9〕 H. Liu, Y. Lu, and Q. Wu, "Automatic Seal Image Retrieval Method by Using Shape Features of Chinese Character", Systems, Man and Cybernetics, pp 2871-2876, 2007.

〔10〕 P. Roy, U. Pal, and J. Llados, "Document Seal Detection Using GHT and Character Proximity Graphs", Pattern Recognition, pp. 1282-1295, Volume 44, issue 6, 2011.

〔11〕 C. Ren, D. Liu, and Y. Chen, "A New Method on the Segmentation and Recognition of Chinese Characters for Automatic Chinese Seal Imprint Retrieval", Proceedings of the International Conference on Document Analysis and Recognition, pp 972-976, 2011.

〔12〕 C. Ren and Y. Chen, "Chinese Payee Name Recognition Based on Seal Information of Chinese Bank Checks", International Conference on Frontiers in Handwriting Recognition, pp 538-541, 2012.

〔13〕 X. Wang and Y. Chen, Seal Image Registration Based on Shape and Layout Characteristics, The 2nd International Congress on Image and Signal Processing, pp 1-5, 2009.

〔14〕 M. Song and K. Han, "Development of a System for Recognizing Stamp Images", Journal of Korea Intelligent Information System, 9(1), pp 125-137, 2003.

〔15〕 Y. Lim, I. Bak, J. Lee, K. Park, J. Kim, K. Kim, "Recognition of a Seal Image by Using Smoothing Method and ART1 Algorithm", Proceedings on Korea Multimedia Society, pp 17-22, 2002.

〔16〕 G. Bradski, A. Kaehler, Learning OpenCV: Computer Vision with the OpenCV Library, 2nd Ed., Hanbit Media, 2010.

저 자 소 개

유 견 아

1986: 서울대학교
제어계측공학과 공학사.
1988: 서울대학교
제어계측공학과 공학 석사.
1995: Univ. of Southern California
컴퓨터학과 공학박사
현 재: 덕성여자대학교
컴퓨터학과 교수
관심분야: 인공지능, 경로계획 알고리즘
Email : kyeonah@duksung.ac.kr

김 경 혜

2011: 덕성여자대학교
컴퓨터 공학과 공학사.
2013: 덕성여자대학교
전산정보통신학과 석사.
관심분야: 패턴인식, 게임 인공지능
Email : kkroong39@duksung.ac.kr