

---

# 효과적인 문자 인식을 위한 저 품질 문자 영상의 이진화 및 획 재구성 방법

김도현\* · 차의영\*

Binarization and Stroke Reconstruction of Low Quality Character Image  
for Effective Character Recognition

Do-Hyeon Kim\* · Eui-Young Cha\*

---

이 논문은 2007년도 부산대학교 컴퓨터 및 정보통신연구소의 연구비를 지원받았음

---

## 요 약

영상의 이진화는 영상을 물체와 배경으로 구분하는 전처리 과정으로써 처리해야 할 대상이 되는 물체를 규정짓기 위한 매우 중요한 처리 과정이다. 본 논문에서는 저 품질 문자 영상에서 효과적인 문자 인식을 위한 효율적인 이진화 방법 및 획 영상 재구성 방법을 제안한다. 먼저 전역적 이진화 방법의 장점과 지역적 이진화 방법의 장점을 반영하여 문자 영역에 대한 이진화를 수행한 다음 이진화된 획 영상에 대한 분석을 통해 획에 붙어 있는 획 잡영 제거와 획 부분에 파여진 공백 잡영에 대한 채움 과정을 수행하여 고 품질의 획 영상으로 재구성하였다. 제안하는 문자 영상을 위한 이진화 알고리즘은 적응적인 임계값 선택 방법에 의해 속도와 성능의 효율성을 추구할 수 있도록 하였으며 이진화 결과로 인한 획 표면 잡영에 대해 단계적인 제거 과정을 수행하여 획 영상을 재구성함으로써 고 품질의 이진 영상을 획득할 수 있었다.

## ABSTRACT

Image binarization is an important preprocessing to identify the objects of interest by dividing pixels into the background and object. We proposes an efficient binarization method and a stroke reconstruction method of the low quality character image for an effective character recognition. First, the character image is binarized by using the both advantages of local and global thresholding method and then the noise elimination around the character stroke and the hole filling on the stroke by the analysis of the binarized stroke image are performed to enhance the quality of the character stroke.

Proposed binarization algorithm for character image achieved an efficiency of both processing speed and performance by the adaptive threshold selection. Moreover, We could get a high quality binary image by a stroke reconstruction of the step-by-step denoising process.

## 키워드

문자 이진화, 지역 이진화, 획 영상 개선, 잡영 제거

## I. 서론

문서 영상 분석 및 문자 인식을 통한 산업 자동화는 컴퓨터의 발전과 함께 급속도로 발전하고 있으며 많은 연구와 관심이 집중되고 있다. 문자 인식 과정에서 스캐너나 카메라로부터 입력되는 영상의 해상도가 낮거나 품질이 나쁠 경우에는 전처리 과정에서 소실되는 정보가 많아져 정확한 문자 획의 특징을 추출하기 어려운 경우가 많다. 특히 가로 및 세로 방향으로 획의 출현 회수를 인식 특징으로 사용할 경우 문자 획에 붙어있는 잡영으로 인해 변별력 있는 특징이 추출되지 못하거나 인식 성능에 오히려 악영향을 줄 우려가 있다.

영상의 이진화는 영상을 물체와 배경으로 구분하는 처리 과정으로 처리해야 할 대상이 되는 물체를 규정짓기 위한 매우 중요한 전처리 과정이며 일반적으로 임계값(threshold)을 통해 그 값보다 큰 픽셀들과 작은 픽셀들로 이진화하는 접근 방법을 사용한다. 영상에서 물체와 배경이 서로 잘 구분되는 명도값을 가지고 있는 경우에는 명도 레벨 히스토그램(gray level histogram)상에서 물체 영역과 배경 영역으로 2개의 분포 형태를 보이므로 그 사이 골짜기 부분에서 임계값을 찾는 방법을 사용할 수 있다. 하지만 히스토그램 자체의 특성상 2차원 영상 정보를 1차원 통계적인 정보로 변환한 것이므로 히스토그램 자체가 영상에서 물체와 배경의 명도값 분포 상태를 제대로 반영하지 못하는 경우가 매우 많다. 다시 말해서 영상 자체로 보았을 때는 물체와 배경이 구분되는 밝기 분포를 가지고 있다고 하더라도 2개의 봉우리 형태로 히스토그램이 나타나지 않고 하나의 봉우리 형태로 나타나는 경우도 많으며 이런 경우에는 히스토그램의 분포 분석만으로는 적절한 임계값을 찾기 곤란하다. 더욱이 물체와 배경의 경계 명확하지 않은 지점은 임계값이 어떻게 설정되는가에 따라 그 결과 영상이 매우 다르게 되며 많은 획들로 구성되어 있는 문자 영상의 경우에는 특히 이진화 결과에 따라 인접 획이 서로 붙거나 흐린 획이 소실되는 문제가 발생하기 쉽다.

## II. 관련 연구

이진화 방법은 크게 전역적 이진화 방법(global thresholding)과 지역적 이진화 방법(local thresholding)으

로 구분할 수 있다.

전역적 이진화 방법은 영상 전체를 분석하여 결정된 하나의 임계값을 모든 픽셀에 적용하여 이진화하는 방법으로써 영상 전체 명도값의 평균을 임계값으로 결정하는 평균법, 물체 또는 배경의 분포 비율에 대한 정보를 기반으로 임계값을 결정하는 타일법, 물체와 배경 픽셀들로 구분되는 히스토그램 분포 분석을 통해 히스토그램 상의 골짜기(valley)를 찾는 방법[1], 임의의 임계값을 기준으로 물체와 배경 픽셀을 구분하고 구분된 각각의 물체 픽셀들의 평균값과 배경 픽셀들의 평균값을 다시 평균한 값으로 임계값을 결정하는 방식을 반복적으로 수행하여 임계값을 결정하는 반복법[2], 물체 픽셀 클래스와 배경 픽셀 클래스간의 분산을 최대로 하는 임계값을 찾는 방법[3], 원영상을 비트 평면으로 분해하여 최상위 비트평면에서 나누어진 배경과 문자 부분에서 각각의 평균 명도값을 구한 다음 두 평균값의 평균값으로 임계값을 결정하는 방법[4], 엔트로피를 이용하는 방법[5] 등이 있으며 그 중에서 Otsu의 방법[3]이 가장 널리 사용되고 있다.

지역적 이진화 방법은 하나의 픽셀을 이진화하기 위해서 그 픽셀 주위의 일정한 윈도우 영역을 참조하여 임계값을 결정하는 방법으로 윈도우 영역 내에서 전역적 이진화 방법을 적용하여 임계치를 구하고 이를 이용해서 해당 윈도우 영역을 이진화하는 임계치 방법과 윈도우 영역 내의 주위 픽셀과의 명도값을 비교하여 중심 픽셀을 이진화하는 비교 방법으로 다시 세분화할 수 있다. 이러한 지역적 이진화 방법은 각각의 픽셀 또는 윈도우마다 다른 임계값을 적용하는 방법으로써 지역 윈도우 영역 내 명도값의 최대값과 최소값의 차이를 이용해 임계값을 검출하는 DMM(difference of maximum and minimum intensity value) 방법[6], 지역 윈도우 영역에서 이웃하는 픽셀들과의 평균 및 표준편차 등의 통계적 수치 기준으로 임계값을 결정하는 방법[7-9], 문서 영상의 윈도우 영역에서 각 윈도우 영역의 저주파 영역에서의 DCT 계수를 기준으로 영상 신호의 활동도를 구하여 문자 블록과 배경 블록으로 나눈 다음 문자 블록에 대해 Otsu의 방법[3]을 사용하여 이진화하는 방법[10,11] 등이 있으며 그 이외에 실제 이진화를 수행하기 전에 여러 가지 영상 개선 기법을 사용한 후 이진화를 수행하는 방법[12], SOM (Self Organizing Map) 신경망을 이용한 방법[13] 등 응용 목적에 따라 다양한 방법이 연구되어 왔다.

전역적 이진화 방법은 알고리즘 자체가 간단한 경우가 많으며 구현이 쉽고 처리 속도가 매우 빠르기 때문에 단순한 영상인 경우에는 효율적으로 사용할 수 있으나 영상에서 물체와 배경의 구분이 명확하지 않거나 주변 잡영이나 조명 환경 등에 의해서 영상의 품질이 나쁜 경우에는 좋지 않은 결과를 얻게 되는 문제점이 있다. 반면 지역적 이진화 방법은 중심 픽셀 주변의 지역적인 밝기 특성을 이용할 수 있어서 배경 영상의 밝기 변화에 적응적으로 적절한 임계값을 설정할 수 있으며 배경과 물체의 구분이 애매 모호한 영역에서 효과적으로 물체 영역을 구분할 수 있는 장점이 있지만 지역 윈도우 영역에 대한 계산 비용으로 전역적 이진화 방법에 비하여 실행 속도가 상당히 느리고 순수 배경 영역을 물체 영역으로 이진화하여 잡영을 생성하거나 순수한 물체 영역 내에 홀(hole)이나 획 표면에 잡영이 발생하는 등의 문제점이 있다. 이러한 점을 감안하여 본 논문에서는 문자 인식을 위한 전처리 단계로써 먼저 전역적 이진화 방법의 장점과 지역적 이진화 방법의 장점을 반영하여 문자 영역에 대한 효율적이고 적응적인 이진화 방법을 제안하고 이진화된 획 영상을 개선시키는 방법으로 획에 붙어 있는 획 잡영 제거와 획 부분에 파여진 공백 잡영에 대한 채움 과정을 통해 고 품질의 획 영상을 재구성하는 방법을 제안한다.

### III. 문자 영상의 적응적 이진화 방법

8bit 256 level의 문자 명도 영상은 일반적으로 어두운 문자 획 부분과 밝은 배경 부분으로 이루어져 있으며 획과 배경과의 경계 부분은 그 명도값이 점진적으로 변하는 부분이므로 임계값이 어떻게 결정되는가에 따라 그 결과 영상의 품질이 결정된다. 특히 복잡한 획으로 구성된 문자 영상은 이진화 결과에 따라 이 경계 부분에서 이웃하는 획들이 서로 접하게 되거나 소실되어버리는 치명적인 결과를 얻게 될 가능성이 많다. 이러한 문자 영상의 특성상 전역적 이진화 방법을 사용하면 양호한 문자 영상이 아닌 경우 오히려 이진화 결과를 얻을 수 없게 된다. 따라서 문자 영상의 이진화를 위해서 특정 픽셀의 주변 픽셀 분포를 고려하여 임계값을 결정하는 지역적 이진화 방법을 사용하는 접근 방법이 보다 효과적이다. 하지만 기존의 대부분의 방법들에서 관찰되는 지역적 이

진화 방법의 단점은 서론에서 언급한 바와 같이 배경 처리나 처리 속도면에서 매우 비효율적이다. 이것은 문자 획과 배경의 경계 부분은 지역적 이진화 방법에 의해 임계값을 검출하여 이진화하고 그 이외의 문자와 배경이 확실한 부분은 전역적 이진화 방법으로 결정된 임계값을 사용하는 접근 방법을 사용한다면 수행 속도와 획의 품질 측면에서 매우 만족할만한 결과를 얻을 수 있다. 본 논문에서는 이러한 접근 방법으로 먼저 지역적 이진화가 필요한 픽셀인지 아닌지를 판단하는 과정을 통해 적응적으로 임계값을 선택하도록 하여 전역적인 임계값을 적용해야 하는 픽셀인 경우에는 빠르게 이진화를 수행하도록 하고 지역적인 임계값을 적용해야 하는 픽셀인 경우에는 정확하게 이진화할 수 있도록 하였다.

#### 3.1. 픽셀 클래스 분류

문자 영상에서의 픽셀들은 명도값에 따라 크게 매우 어두운 명도값을 가지는 문자 클래스, 매우 밝은 명도값을 가지는 배경 클래스, 그리고 문자 또는 배경이 될 수 있는 혼합 클래스의 3가지 유형으로 분류할 수 있다. 문자 클래스나 배경 클래스인 픽셀은 이진화 결과가 임계값에 크게 민감하지 않기 때문에 전역적인 임계값으로도 충분히 좋은 결과 영상을 획득할 수 있다.

이렇게 3가지 유형으로 분류하기 위해서는 문자 클래스와 혼합 클래스, 혼합 클래스와 배경 클래스를 구분할 수 있는 2개의 경계 임계값이 필요하다. 이 임계값은 전역적으로 검출된 임계값과 문자 영상 전체의 표준편차값을 이용하여 식 (1), (2)와 같이 설정한다.

$$T_{low} = T_G - p * S_G \tag{1}$$

$$T_{high} = T_G + p * S_G \tag{2}$$

여기서,  $T_G$ 는 전역 이진화 임계값으로 반복 임계값 적용 기법[2]에 의해 결정된 임계값이며  $S_G$ 는 문자 영상 명도 값들의 표준 편차(standard deviation)이다.  $p$ 는 픽셀의 클래스를 분류하기 위한 경계 임계값을 설정할 때 영상 전체의 표준편차를 반영하는 정도를 나타내는 값이다. 이 값이 크면 표준 편차량을 많이 반영하게 되므로 혼합 클래스로 분류되는 픽셀들이 많아지게 된다. 반복 임계값 적용 기법은 구현이 간단하며 Otsu 기법과 거의 유사한 속도와 성능을 나타낸다.

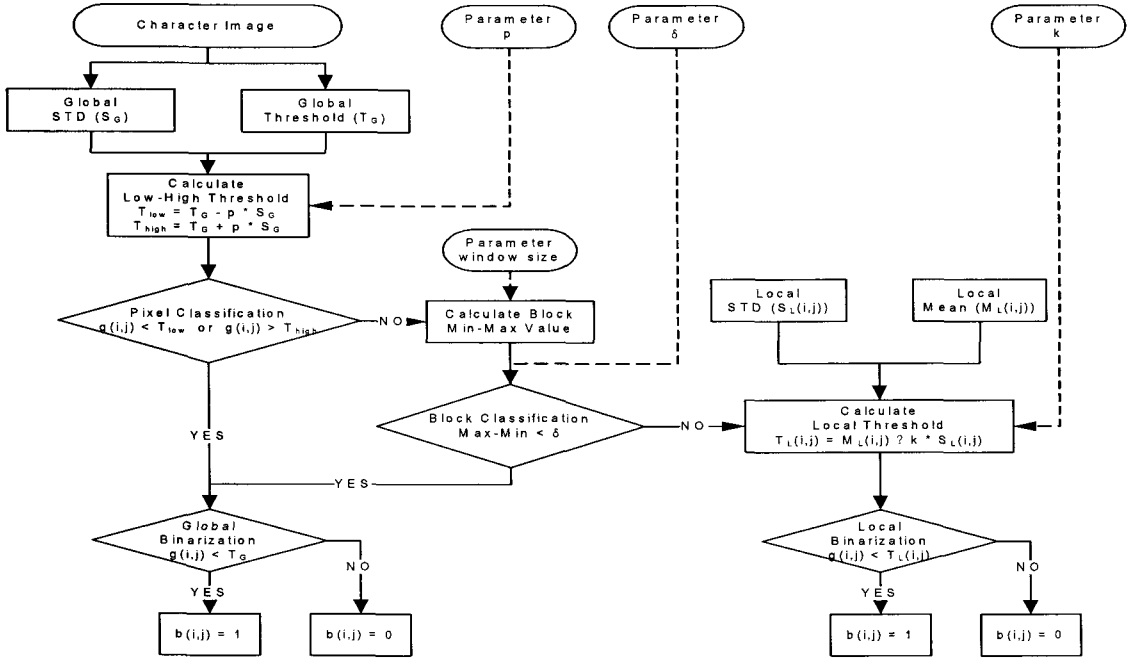


그림 1. 문자 영상의 적응적 이진화 과정  
(실선 : 처리과정, 점선 : 파라메타 입력)

Fig. 1 Adaptive binarization process for character image

임의의 픽셀  $g(i, j)$ 은 그 명도 값이  $T_{low}$ 보다 작은 경우에는 문자 클래스로,  $T_{high}$ 보다 큰 경우에는 배경 클래스로, 두 임계값 사이인 경우 혼합 클래스로 분류한다. 분류된 픽셀의 클래스가 문자 클래스이거나 배경 클래스인 경우에는 사전에 계산된 전역 임계값  $T_G$ 로 픽셀을 이진화한다. 그렇지 않은 경우에는 지역 윈도우 블록 분류 과정을 거친다.

### 3.2. 지역 윈도우 블록 분류

픽셀 분류 단계에서 혼합 클래스로 분류된 픽셀은 문자 획 또는 배경이 될 수 있는 애매모호한 명도 값을 가지는 픽셀임을 의미한다. 이러한 픽셀들은 주변의 일정한 윈도우 영역 내에 있는 픽셀들의 명도값 분포를 참조하여 결정된 지역 임계값으로 이진화 한다. 이때 이 윈도우 영역 내에 있는 주변 픽셀들의 분포가 문자 획에 해당하는 픽셀들과 배경에 해당하는 픽셀들로 구성되어 있는 경우에는 적절한 지역 임계값을 찾을 수 있지만, 그렇지 않은 경우 부적절한 지역 임계값이 얻어질 수 있으므로 영상이 심하게 훼손될 수 있다. 이와 같은 현상은 지

역 이진화 임계값을 구할 때의 윈도우 크기가 문자의 크기에 맞게 적절하게 설정되지 않을 경우 흔히 발생될 수 있는 심각한 문제이다. 따라서 이러한 지역 임계값 결정 방법의 단점을 개선하기 위해서 윈도우 블록 내 존재하는 픽셀의 최대값과 최소값을 검출하여 이 두 값이 일정한 수치( $\delta$ ) 이상의 차이가 있을 때에만 지역 임계값을 사용하여 이진화하고 그렇지 않은 경우는 전역 임계값을 사용하여 이진화하는 방법을 사용한다.

### 3.3. 지역 이진화

윈도우 블록 분류 단계에서 지역 임계값을 사용해야 하는 경우로 판단된 경우에는 픽셀  $g(i, j)$ 을 기준으로 하는 윈도우 블록 내 픽셀들의 평균  $M_L(i, j)$ 과 표준편차  $S_L(i, j)$ 를 이용하여 식 (3)과 같이 지역 이진화 임계값  $T_L(i, j)$ 을 계산하며,  $g(i, j)$ 의 값이  $T_L(i, j)$ 보다 작은 경우에는 문자로, 큰 경우에는 배경으로 이진화 한다.

$$T_L(i, j) = M_L(i, j) - k \cdot S_L(i, j) \quad (3)$$

여기서,  $k$ 는 윈도우 블록 내 픽셀들의 표준 편차를 이진화 임계값에 반영하는 정도를 나타내는 상수 값으로 그 값이 크면 문자 획이 가늘어지고 작으면 굵어진다.

그림 1은 제안하는 이진화 시스템의 전체적인 처리 과정을 나타낸다.

#### IV. 문자 영상의 획 재구성 방법

영상을 지역 임계치 설정에 의해 이진화한 경우 전역적 이진화 방법에 비해 대체적으로 획 구분 능력이 뛰어난 반면 획의 표면 처리가 매끄럽게 되지 못하는 현상이 발생되며 이와 같은 현상은 지역적 임계치 설정 방법에서 공통적으로 나타나는 현상이다. 이와 같은 획 표면 잡영을 제거하기 위해서 본 논문에서는 주변 이웃 픽셀들의 정보를 참조하여 획 표면 잡영은 제거하고 공백 잡영은 획 픽셀로 채우는 방법으로 획의 형태를 재구성하는 방법을 제안한다.

##### 4.1. 획 잡영 및 공백 잡영

획 잡영은 그림 2의 (a)와 같이 획의 길이보다 상대적으로 짧은 획의 바깥 부분에 붙어 있는 획 픽셀을 의미하며, 공백 잡영은 그림 2의 (b)와 같이 획의 길이보다 상대적으로 짧은 획 픽셀 사이에 있는 배경픽셀을 의미한다. 이러한 획 잡영 및 공백 잡영 제거를 위해 median 필터나 morphology 필터를 사용할 경우 획 자체가 소실되거나 획과 획이 서로 붙어버리는 결과를 초래하므로 문자 인

식을 위한 특징 추출 시 치명적인 문제점이 될 수 있다. 따라서 본 논문에서는 주변 이웃 픽셀들의 정보를 참조하여 이러한 잡영을 판단하고 획 잡영은 제거하고 공백 잡영은 획 픽셀로 채우는 방법을 도입하여 전체적인 획의 품질을 개선시킬 수 있도록 하였다. 획 품질을 개선하는 방법은 먼저 수평 획에 대해 획 잡영 라인을 검출하고 이를 제거하며 수직 획에 대해서도 획 잡영 라인을 검출하여 제거한다. 일반적으로 획 잡영은 한 픽셀이 아닌 여러 픽셀로 구성되므로 이를 획 잡영 라인이라 칭한다. 다음으로 획에 존재하는 공백 잡영 라인을 검출하고 이를 제거한다. 공백 잡영 라인은 획의 표면에 존재하는 한 픽셀 이상의 공백 부분을 의미하며 이 부분을 획 픽셀로 채움으로써 잡영을 제거한다.

##### 4.2. 획 잡영 라인 제거

먼저 수평 획 잡영 라인을 제거하는 알고리즘은 다음과 같다.

###### Step 1. 변수 초기화

유효 공백 라인 길이와 잡영 후보 라인 길이의 값을 0으로 초기화한다. 잡영 후보 라인 길이는 획에 붙은 잡영의 길이이며 유효 공백 라인 길이는 잡영 후보 라인이 시작되기 전에 계산된 공백 라인의 길이로써 일반적인 공백의 길이가 아닌 이웃 획에 근접하여 나타난 공백의 연속된 길이이다. 이 유효 공백 라인의 길이가 획 잡영을 판단하는 중요한 기준이 된다. 참고적으로 공백 잡영 제거 알고리즘에서는 유효 공백 라인 대신 유효 획 라인이 사용되며 잡영 후보 라인은 획 부분에 파인 공백의 연속된 길이가 된다.

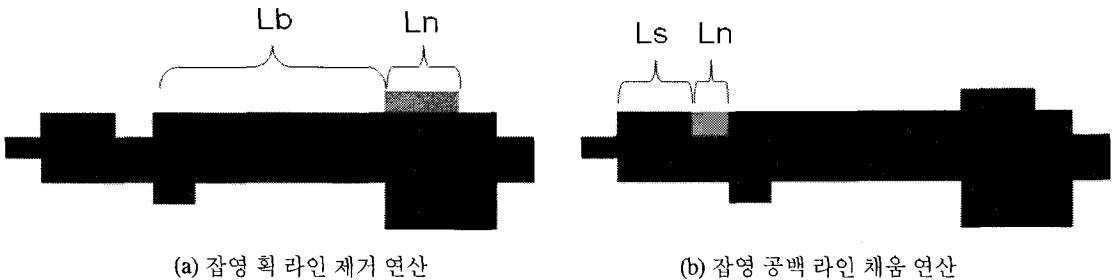


그림 5. 유효 공백 라인, 유효 획 라인, 잡영 후보 라인 길이 예  
 Fig. 2 Illustration of the valid blank line, valid stroke line, candidate noise line length  
 Lb : 유효 공백 라인 길이, Ls : 유효 획 라인 길이, Ln : 잡영 후보 라인 길이  
 Lb : Valid blank line length, Ls : Valide stroke line length, Ln : Candidate noise line length

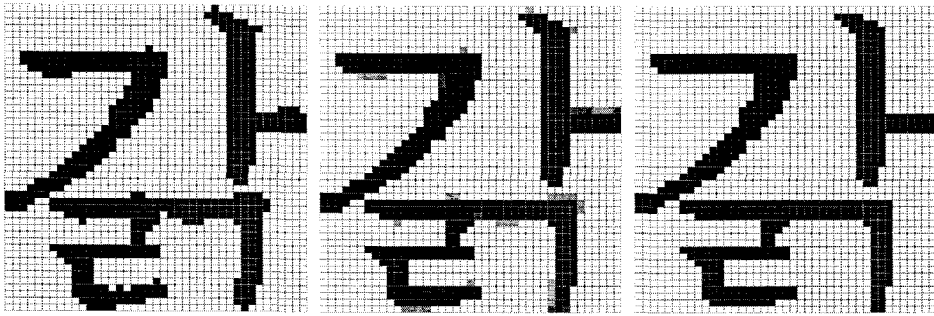


그림 3. 문자 획 개선  
 (a) 원본 영상 (b) 잡영 라인 (c) 최종 결과  
 Fig 3. Stroke Enhancement  
 (a) Original Image (b) Noise line (c) Final result

**Step 2. 영상 열/행 단위 정방향 탐색**

영상의 위쪽부터 아래쪽까지 각각의 행 단위로 영상의 왼쪽에서 오른쪽으로 한 픽셀씩 검사하면서 현재 픽셀  $b(i, j)$ 가 획 픽셀인 경우 Step 3, 배경 픽셀인 경우 Step 4를 수행한다.

**Step 3. [획 픽셀 처리] 잡영 후보 라인 계산**

**Step 3.1** 현재 획 픽셀인  $b(i, j)$ 의 상하 이웃 픽셀  $b(i, j-1)$ ,  $b(i, j+1)$ 의 값이 서로 동일한 경우에는  $b(i, j)$ 가 제거될 대상이 아니므로 유효 공백 라인의 길이와 잡영 후보 라인의 길이의 값을 0으로 초기화한다.

**Step 3.2**  $b(i, j)$ 의 상하 이웃 픽셀  $b(i, j-1)$ ,  $b(i, j+1)$ 의 값이 서로 동일하지 않은 경우에는 획에 붙어있는 잡영일 가능성이 있으므로 Step 3.2.1을 수행한다.

**Step 3.2.1**  $b(i-1, j-1)$ 과  $b(i, j-1)$ 이 동일하고,  $b(i-1, j+1)$ 과  $b(i, j+1)$ 이 동일하면 잡영 후보 라인이 계속됨을 의미하므로 잡영 후보 라인의 길이를 1 증가시킨다. 그렇지 않은 경우는 새로운 형태의 획 잡영이 발생한 것으로 잡영 후보 라인의 길이를 1로 설정하고 이전에 계산된 유효 공백 라인의 길이를 0으로 초기화한다.

**Step 4. [배경 픽셀 처리] 잡영 획 라인 제거 및 유효 공백 라인 계산**

**Step 4.1** 먼저 이전에 잡영 후보 라인이 존재한 경우에는 현재 배경 픽셀이 나타나기 전 잡영 라인이 존재한 경우이므로 이 잡영 라인을 제거하는 과정을 수행한다. Step 4.1.1에 의해 잡영 라인을 제거하거나 그대로 두

고, 그렇지 않은 경우 Step 4.1.2에 의해 유효 공백 라인의 길이를 계산한다. 마지막으로 현재 배경 픽셀이 나타났으므로 잡영 후보 라인 길이를 0으로 설정한다.

**Step 4.1.1** 계산된 잡영 후보 라인이 유효 공백 라인보다 짧은 경우는 잡영 후보 라인이 잡영이라고 판단한다. 따라서 잡영 후보 라인 길이만큼 잡영 라인을 제거하고 그 길이만큼 유효 공백 라인 길이를 더 증가시킨다.

**Step 4.1.2** 계산된 잡영 후보 라인이 유효 공백 라인보다 긴 경우는 잡영 후보 라인이 잡영이 아닌 획의 일부라고 판단하고 그대로 유지하며 유효 공백 라인 길이는 0으로 초기화한다.

**Step 4.2** 현재 배경 픽셀인  $b(i, j)$ 의 주변 상하 픽셀  $b(i, j-1)$ ,  $b(i, j+1)$ 의 값이 서로 동일하지 않은 경우나 모두 획 픽셀인 경우에는 유효 공백 라인인 경우이며 Step 4.2.1을 수행한다. 그 이외의 경우는 의미없는 배경 픽셀이므로 유효 공백 라인의 길이를 0으로 초기화한다.

**Step 4.2.1** 이전에 유효 공백 라인이 존재하고  $b(i-1, j-1)$ 과  $b(i, j-1)$ ,  $b(i-1, j+1)$ 과  $b(i, j+1)$ 이 동일하면 유효 공백 라인이 계속됨을 의미하므로 유효 공백 라인의 길이를 1증가시킨다. 그렇지 않은 경우 즉,  $b(i-1, j-1)$ 과  $b(i, j-1)$ 의 값이 다르거나,  $b(i-1, j+1)$ 과  $b(i, j+1)$ 의 값이 다르면 유효 공백 라인이 시작되는 상태이므로 유효 공백 라인 길이를 1로 설정한다.

**Step 5. 영상 열/행 단위 역방향 탐색**

이상의 과정을 영상의 위쪽부터 아래쪽까지 각각의

행 단위로 영상의 왼쪽부터 오른쪽까지 한 픽셀씩 검사하면서 정방향으로 잡영 라인을 탐색한다. 결과적으로 잡영 라인이 나타나기 전 충분한 길이의 유효 공백 라인이 나타나지 않으면 그 잡영 라인은 제거되지 않으므로 역방향으로 탐색할 필요가 있다. 따라서 각각의 행 단위로 영상의 오른쪽부터 왼쪽 방향으로 역방향으로 탐색하면서 Step1~Step4의 과정을 반복하여 미처 제거되지 못했던 잡영 라인을 제거한다.

이와 같은 과정을 수직 획에 대해서도 동일하게 수행하면 획에 붙어있는 수직 잡영 획 라인을 제거할 수 있다. 다시 말해서, 영상의 왼쪽부터 오른쪽까지 각각의 열 단위로 위쪽에서 아래쪽으로 한 픽셀씩 검사하면서 수직 획 잡영 라인을 탐색한다. 주변 이웃 픽셀은 상하 픽셀  $b(i, j-1)$ ,  $b(i, j+1)$  대신 좌우 픽셀  $b(i-1, j)$ ,  $b(i+1, j)$ 을 참조한다.

### 4.3. 공백 잡영 제거

획 잡영 라인이 제거되었으면 다음 과정으로 공백 잡영 라인을 검출하고 이를 획 픽셀로 채워줌으로써 잡영 라인을 제거한다. 공백 잡영 라인을 검출하고 제거하는 방법은 대상이 되는 획 픽셀 색깔과 공백 픽셀 색깔이 획 잡영 라인을 검출하고 제거하는 방법에서의 색깔과 서로 반대이다. 예컨대, 획 픽셀의 값을 0으로, 공백 픽셀의 값을 255로 설정하여 획 잡영 라인을 제거하였다면, 이 두 색을 서로 반대로 하면 공백 잡영 라인을 제거하는 방법이 된다.

## V. 실험 및 결과

### 5.1. 실험 문자 영상

제안하는 이진화 알고리즘의 성능을 검증하기 위해 200~400 DPI로 스캔된 다양한 크기의 8bit 256 gray level의 개별 한글 문자 영상을 대상으로 실험하였으며 굴림, 고딕, 궁서, 돋움, 바탕의 5가지 글꼴 문자를 각각 2개씩 조합한 한글 완성형 2350자 영상 23500개(2350x5x2)를 사용하여 이진화 성능과 수행 속도를 비교 분석하였다.

### 5.2. 이진화 성능 평가

성능 평가를 위해 사용된 이진화 알고리즘은 전역적 이진화 방법으로 가장 많이 사용하는 Otsu의 방법[3]과

지역적 이진화 방법으로 널리 사용되는 Niblack, Sauvola, Bernsen 방법을 포함한 여러 가지 방법을 사용하였다. 지역적 이진화 방법들은 윈도우 블록 크기에 대한 영향을 고려하여 8x8, 15x15의 두 가지 경우로 그 성능을 비교하였다. 제안하는 이진화 방법에서 사용된 인자는  $p = 0.5$ ,  $\delta = 16$ ,  $k = 0.1$ 로 설정하였으며 다른 이진화 방법도 관련 변수들을 제안한 방법과 동일한 값으로 설정하여 객관적으로 비교 평가할 수 있도록 하였다.

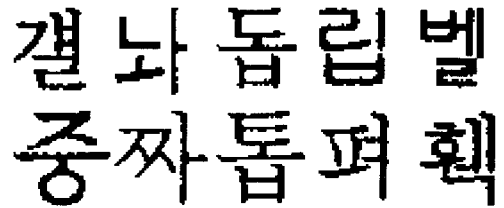


그림 4. 원본 이진 영상  
Fig. 4 Original binary images

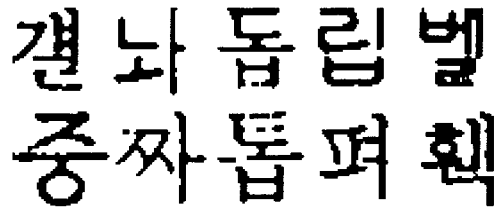


그림 5. 중간값 필터링  
Fig. 5 Median filtering

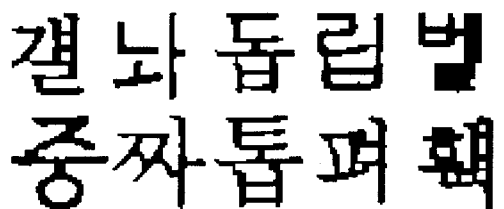


그림 6. 형태학적 닫힘 연산  
Fig. 6 Morphology closing

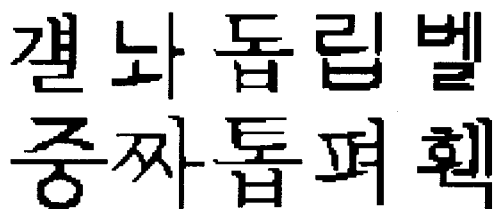


그림 7. 제안된 획 재구성 방법  
Fig. 7 Proposed reconstruction method

표 1. 이진화 성능 비교

Table 1. Comparison of Binarization methods

PR : Proposed Binarization only, PBR : Proposed Binarization + Reconstruction

Sample	Local Block Size	1	2	3	4	5	6	7	8	9	10
Mean	-	가	뱅	개	현	툼	툼	툼	툼	툼	툼
Bimodal [1]	-	가	뱅	개	현	툼	툼	툼	툼	툼	툼
Iterative [2]	-	가	뱅	개	현	툼	툼	툼	툼	툼	툼
Otsu [3]	-	가	뱅	개	현	툼	툼	툼	툼	툼	툼
Niblack [7]	8	가	뱅	개	현	툼	툼	툼	툼	툼	툼
	15	가	뱅	개	현	툼	툼	툼	툼	툼	툼
Sauvola [9]	8	가	뱅	개	현	툼	툼	툼	툼	툼	툼
	15	가	뱅	개	현	툼	툼	툼	툼	툼	툼
Bernsen [11]	8	가	뱅	개	현	툼	툼	툼	툼	툼	툼
	15	가	뱅	개	현	툼	툼	툼	툼	툼	툼
PB	8	가	뱅	개	현	툼	툼	툼	툼	툼	툼
	15	가	뱅	개	현	툼	툼	툼	툼	툼	툼
PBR	8	가	뱅	개	현	툼	툼	툼	툼	툼	툼
	15	가	뱅	개	현	툼	툼	툼	툼	툼	툼

표 1은 여러 가지 이진화 방법에 의한 결과 영상을 비교하고 있다. Sample 1, 2와 같은 영상은 양호한 문자 영상이며 전역적 이진화 방법이나 지역적 이진화 방법 모두 대체로 양호한 결과 영상을 획득할 수 있었다. Sample 3~6은 문자 획이 두껍거나 복잡한 형태의 획을 포함하는

문자 영상이며 전역적 이진화 방법은 문자가 두껍게 이진화되면서 획이 접합되어 획 구분이 되지 않는 매우 좋지 않은 결과를 나타내는 경우가 많다. 반면 지역적 이진화 방법은 획의 변별력은 좋으나 잡영이 발생하거나 획 내부의 홀이 발생하는 문제점이 가끔씩 발생하였으며



제안한 방법은 이러한 지역적 이진화 방법의 단점이 보완되었음을 나타내고 있다. Sample 7-9는 문자의 크기가 작고 획을 이루는 픽셀들의 명도 값이 불균일하여 대비가 낮은 경우의 예로서, 임계값 설정이 어렵기 때문에 전역적 이진화 방법을 사용할 경우 획이 접합되거나 소실되는 극단적인 현상을 나타내지만 지역적 이진화 방법을 사용할 경우 대체적으로 양호한 결과를 나타냄을 알 수 있다. Sample 10은 특정한 획을 이루는 픽셀의 굵기와 명도 값이 다른 획 픽셀과 다르게 나타난 경우의 예로써 윈도우 크기가 작으면 획의 일부가 소실되는 현상을 관찰할 수 있었다.

이진화 성능의 종합적인 평가를 위해 표 2과 같이 이진화 결과에 따른 문자 획의 결합 발생 정도, 획 훼손 발생 정도, 잡영 발생 정도, 획 품질 등을 기준으로 전체적인 성능 평가를 수행하였다. 제안하는 이진화 방법을 적용할 경우 문자 획의 명도값이 균일하지 않은 경우나 영상 자체의 해상도가 낮은 경우의 문자 영상에서는 가끔 획 훼손 현상이 발생하였지만 복잡한 형태의 획으로 이루어진 문자 영상에서도 획 결합 없이 획의 품질이 양호한 이진화 문자 영상을 얻을 수 있었다.

5.3. 획 품질 성능 개선 평가

획의 품질은 median filtering, morphology closing을 수행한 결과와 제안한 알고리즘을 비교 평가하였다. 그림 4는 몇 개의 이진 문자 샘플 영상의 예이며 그림 5은 각각의 영상에 3x3 median filtering을 수행한 결과를 나타낸다. Median filter의 특성상 blurring이 발생하지 않는다는 장점이 있지만 획의 위치 및 관계가 고려되지 않기 때문에 한 픽셀 두께의 획이 소실되는 현상과 근접한 획이 서로 붙어버리는 문제점을 관찰할 수 있으며 획 자체의 품질도 개선되지 않음을 알 수 있다. 그림 6는 3x1 마스크와 1x3 마스크를 사용하여 morphology closing (dilation+erosion) 연산을 수행한 결과이다. 획이 소실되는 현상은 없으며 획의 거친 정도는 다소 완화 되었지만 획 결합 현상이 치명적으로 많이 발생하여 문자 획의 두께 및 획 사이의 거리와 마스크의 크기에 민감하게 반응함을 알 수 있다. 그림 7은 제안된 획 잡영 및 공백 잡영 라인을 제거하는 알고리즘을 사용한 결과이며 획의 거친 표면과 획에 파인 부분이 효과적으로 제거되었으며 연결성도 유지하고 있으며 획 결합 현상도 발생하지 않음을 알 수 있다.

표 2. 처리 속도  
Table 2. Processing Time

Thresholding method	Total (ms)	Average (ms)
Mean	14927	0.64
Bimodal	17291	0.74
Iterative	15927	0.68
Otsu	16057	0.68
Niblack	112344	4.78
Sauvola	111296	4.74
Bernsen	59474	2.53
PB	42317	1.80
PBR	46599	1.98

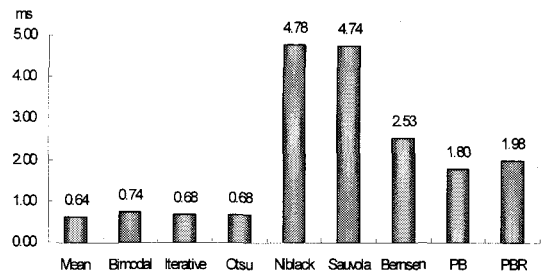


그림 8. 글자당 평균 처리 속도  
Fig. 8 Average processing time per character

5.4. 이진화 및 획 재구성 처리 속도

제안하는 문자 이진화 및 획 재구성 방법은 문자 인식의 전처리 과정으로 처리 시간에 대한 비용을 고려하지 않을 수 없다. 제안하는 이진화 방법은 전역 임계값과 지역 임계값을 모두 사용하는 방법임에도 불구하고 처리 속도에서도 지역 이진화 방법에 비해 처리 속도가 빠름을 알 수 있으며 획 재구성 방법 또한 표 2와 같이 문자당 평균 2ms 정도의 처리 속도로 대용량 문서 처리 및 문자 인식에 효율적으로 활용할 수 있음을 알 수 있다.

VI. 결론

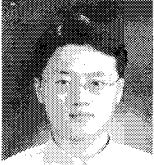
본 논문에서는 영상 분석 및 문자 인식을 통한 대용량 문서 처리의 전처리 단계로서 저 품질 문자 영상에서 효

울적인 이진화 알고리즘과 획 재구성 방법을 제시하였다. 제안하는 이진화 알고리즘은 전역적 이진화 임계값과 지역적 이진화 임계값을 적절히 조합하여 획과 획 사이의 간격이 좁거나 구분이 불분명한 저 품질 영상에 대해서도 획 결합 없이 양호한 품질의 이진화 문자 영상을 얻을 수 있었으며 속도에서도 글자당 평균 2ms의 속도로 기존의 지역적 이진화 알고리즘에 비해 빠른 처리 능력을 보여주었다. 또한 이진화 결과 발생하는 획 주변의 잡영을 제거하기 위해서 주변 이웃 픽셀들의 정보를 참조하여 획 표면 잡영은 제거하고 공백 잡영은 획 픽셀로 채우는 방법을 도입하여 전체적인 획의 품질을 개선함으로써 효과적인 문자 인식 및 문서 처리가 가능할 것으로 기대된다.

#### 참고문헌

- [1] A. Rosenfeld, A.C. Kak, *Digital Picture Processing. 2nd Edition*, Academic Press, 1982.
- [2] Milan Sonica, Vaclay Hlavac, Roger Boyle, *Image Processing, Analysis, and Machine Vision. 2nd Edition*, PWS Publishing, pp.129-130
- [3] N. Otsu, "A threshold selection method from gray-level histogram," *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. SMC-9, No. 1, pp. 62-66, 1979.
- [4] 김하식, 김강, 조경식, 전중식, "비트평면 패턴을 이용한 최적 이진화 방법," *한국컴퓨터정보학회 논문지*, 제6권, 제4호, pp.1-5, 2001년 12월
- [5] J.N.Kapur, P.K.Sahoo, A.K.Wong, "A New Method for Gray Level Picture using Entropy of the Histogram," *Computer Vision, Graphics, and Image Processing*, Vol.29, pp.273-285, 1985.
- [6] 장대근, 전병태, "카메라 문서 영상의 이진화 및 기울어짐 보정 방법," *한국컴퓨터정보학회 논문지*, 제10권, 제3호, pp.143-150, 2005년 7월
- [7] W. Niblack, *An Introduction to Image Processing*, Prentice-Hall, pp.115-116, 1986.
- [8] Zheng Zhang, Chew Lim Tan, "Recovery of Distorted Document Images from Bound Volumns," *Document Analysis and Recognition*, pp.429-433, 10-13 Sept. 2001.
- [9] J. Sauvola, M. Pietikainen, "Adaptive document image binarization," *Pattern Recognition* 33, pp.225-236, 2000.
- [10] B.Gatos, I.Pratikakis, S.J.Perantonis, "Adaptive degraded document image binarization," *Pattern Recognition*, Vol.39, No.3, pp.317-327, March 2006.
- [11] John Bernsen, "Dynamic thresholding of grey-level images," in *Proceedings of the 8th International Conference on Pattern Recognition*, Paris, France, pp.1251-1255, Oct. 1986.
- [12] 박준효, 장익훈, 김남철, "PDA 응용을 위한 명함 영상의 회전 보정," *한국통신학회 논문지*, 제28권, 제12C호, pp.1225-1238, 2003년 12월
- [13] 신기택, 장익훈, 김남철, "변형된 Quadratic 필터를 이용한 PDA로 획득한 명함 영상의 블록 적응 이진화," *한국통신학회 논문지*, 제29권, 제6C호, pp.801-814, 2004년 6월
- [14] 최경주, 변혜란, 이일병, "효과적인 이진화를 위한 영상개선기법의 정의 및 구현," *정보과학회논문지*, 제26권, 제2호, pp.284-296, 1999년 2월
- [15] Hamza, H.; Smigiel, E.; Belaid, E., "Neural based binarization techniques," *Document Analysis and Recognition*, Vol. 1, pp.317-321, Aug. 2005.

## 저자소개



**김도현(Do-Hyeon Kim)**

2001년 부산대학교 전자계산학과 졸업.

2003년 부산대학교 전자계산학과 석사 졸업.

2007년 부산대학교 컴퓨터공학과 박사 재학 중.  
※관심분야: 패턴인식, 영상처리 및 컴퓨터비전, 퍼지 및 신경망, 제어자동화시스템.



**차의영(Eui-Young Cha)**

1979년 경북대학교 전자공학과 졸업.

1982년 서울대학교 전자계산학과 석사 졸업.

1998년 서울대학교 컴퓨터공학과 박사 졸업.  
1981년~1985년 한국전자기술연구소 연구원.  
1995년~1996년 University of London 방문교수.  
1985년~현재 부산대학교 컴퓨터공학과 교수  
※관심분야: 컴퓨터비전, 신경망 이론, 웨이블릿