

Character Matching Using a Hausdorff Distance

Kyeongtaek Kim[†] · Ji Hun Kyung

Department of Industrial and Management Engineering, Hannam University

Hausdorff 거리를 이용한 문자 매칭

김경택[†] · 경지훈

한남대학교 산업경영공학과

The Hausdorff distance is commonly used as a similarity measure between two-dimensional binary images. Since the document images may be contaminated by a variety of noise sources during transmission, scanning or conversion to digital form, the measure should be robust to the noise. Original Hausdorff distance has been known to be sensitive to outliers. Transforming the given image to grayscale image is one of methods to deal with the noises. In this paper, we propose a Hausdorff distance applied to grayscale images. The proposed method is tested with synthetic images with various levels of noises and compared with other methods to show its robustness.

Keywords : Grayscale Hausdorff Distance, Character Matching, Binary Image

1. 서 론

디지털화가 일반화됨에 따라 서류를 디지털 형태로 보관하는 경우가 증가하고 있다. 그 방법 중의 하나가, 인쇄된 서류를 스캐닝하여 보관하는 방법이다. 서류를 스캐닝할 때 부딪히는 가장 일반적인 문제는 여러 가지 이유로 스캐닝된 이미지에 노이즈가 포함된다는 사실이다. 예를 들면, 저 해상도의 기기로 기록하거나, 스캐닝되는 문서가 고정되어 있지 않을 때, 이미지에 노이즈가 포함된다. 이미지에 포함되는 노이즈 중의 하나가, 마치 소금과 후추를 뿌려놓은 것처럼, 검은 점과 흰 점이 이미지 전체에 뿌려져 있는 흑백 점잡음(salt and pepper noise)이다[4]. 특히 디지털화된 문서 이미지를 자세히 살펴보면, 컬러, 회색톤, 흑백 문서를 불문하고, 이러한 흑백 점잡음이 포함되어 있는 경우를 발견할 수 있다[4, 9]. 흑백 점잡음이 포함된 이미지에서는 노이즈의 영향을 줄이기 위하여, 노이즈

제거 필터[9]를 사용하거나, 그레이스케일(grayscale)로 변환한 후[12] 잡음의 영향을 축소하는 방법을 사용해왔다.

문자인식시스템은 전처리(pre-processing), 문자 이미지 분리, 문자 인식, 이렇게 크게 3단계로 구분된다[2]. 전처리 단계에서는 배경잡음 제거, 크기조정(scaling), 문자 비틀림의 조정 등을 통하여, 읽어 들인 문서에 내재된 여러 변동성을 줄여 후속 단계의 성능 개선을 도모한다. 문자 이미지 분리 과정은 문자의 경계를 올바르게 인식하여, 그 결과를 다음 단계로 넘긴다. 문자 인식 단계에서는 신경망 모형, Hausdorff 거리 모형, 서포트 벡터 머신(Support Vector Machine) 모형, 템플릿 매칭(Template Matching) 모형 등을 사용하여 문자를 인식한다[2]. 일반적으로, 전처리 단계에서 노이즈를 제거 하여도, 문자 인식 단계의 입력 데이터에 노이즈가 포함되어 있다고 가정하고, 문자 인식을 하는 것이 타당하다.

컬러 이미지의 경우, 명도나 컬러에 관한 데이터를 이용할 수 있다. 그러나, 흑백 이미지의 경우, 오직 배경 픽셀과 전경 픽셀만으로 이루어져 있어 명도가 매우 조악하다. 따라서, 흑백 이미지의 비교는 전경 픽셀 간의 비교로 이루어진다[1].

문자 인식 단계에서 널리 쓰이는 Hausdorff 거리 모형에서는 입력된 이미지와 원본 문자 이미지들과의 Hausdorff 거리를 계산하여, 문자를 인식하는 방법이다. 본 논문은 흑백 점잡음이 포함된 흑백 이미지를 입력으로 받아 그레이스케일로 변환한 다음, 저장된 원본 문자 이미지와의 Hausdorff 거리를 계산하여, 가장 유사한 문자를 찾는 문제를 다룬다.

흑백 이미지 파일을 그레이스케일 이미지로 변형하면, 노이즈의 강도를 약화시켜, 두 이미지의 매칭을 보다 정확하게 하여준다. Zhao 등[12]은 흑백 이미지를 그레이스케일 이미지로 변형한 후 Hausdorff 거리 계산하여, 문자 매칭 문제를 해결하는 방법을 제시하였다. 이 연구에서 노이즈가 추가된 영어 알파벳을 만들어 노이즈가 없는 문자와 매치 시킬 때, Grayscale Hausdorff 거리가 원래의 Hausdorff 거리보다 우수한 척도임을 보였다. 그러나, 몇 퍼센트의 노이즈가 사용되었는지를 밝히지 않았으며, 오직 한 수준의 노이즈만을 사용하여, Grayscale Hausdorff 거리가 어느 정도의 노이즈 범위에서 Hausdorff 거리보다 우수한 척도인지도 알 수 없다.

본 논문에서는 새로운 Grayscale Hausdorff 거리를 제시하고, 흑백 점잡음의 수준에 따라 기존의 Grayscale Hausdorff 거리와 본 논문에서 제시한 Grayscale Hausdorff 거리를 계산하여, 문자를 정확히 매치하는 비율의 변화를 살펴본다.

2. Hausdorff 거리의 개요

2.1 Hausdorff 거리

Hausdorff 거리는 두 집합의 비일치도(mismatch)를 측정하는 비선형 척도로 유클리디언 거리, 맨하탄 거리 등의 일반적인 거리 함수를 이용하는 최대최소(maxmin) 기법이다[5]. 두 점 a 와 b 사이의 거리 $d(a, b)$ 를 다음과 같이 정의하자.

$$d(a, b) = \|a - b\| \quad (1)$$

두 개의 유한집합 $A = \{a_1, a_2, \dots, a_m\}$ 와 $B = \{b_1, b_2, \dots, b_n\}$ 가 주어졌을 때, A 의 한 점 a 에서 B 까지의 Hausdorff 거리 $d(a, B)$ 는 다음과 같이 정의된다.

$$d(a, B) = \min_{b \in B} d(a, b) = \min_{b \in B} \|a - b\| \quad (2)$$

집합 A 로부터 집합 B 까지의 Hausdorff 거리 $h(A, B)$ 는 다음과 같이 정의된다.

$$\begin{aligned} h(A, B) &= \max_{a \in A} d(a, B) \\ &= \max_{a \in A} \min_{b \in B} d(a, b) \\ &= \max_{a \in A} \min_{b \in B} \|a - b\| \end{aligned} \quad (3)$$

유사한 방법으로, 집합 B 로부터 집합 A 까지의 Hausdorff 거리 $h(B, A)$ 는 다음과 같이 정의된다.

$$\begin{aligned} h(B, A) &= \max_{b \in B} d(b, A) \\ &= \max_{b \in B} \min_{a \in A} d(b, a) \\ &= \max_{b \in B} \min_{a \in A} \|b - a\| \end{aligned} \quad (4)$$

집합 A 와 집합 B 사이의 Hausdorff 거리 $H(A, B)$ 는 앞에서 구한 $h(A, B)$ 와 $h(B, A)$ 를 이용하여 다음과 같이 정의된다.

$$H(A, B) = \max(h(A, B), h(B, A)) \quad (5)$$

2.2 Hausdorff 거리의 변형

그러나, Hausdorff 거리는 이상치에 매우 민감하다[4, 12]. 만일 두 유한집합 A 와 B 의 모든 점들이 완전히 일치하고, 오직 A 의 한 점이 B 의 어떤 점에서도 멀리 떨어져 있다면, Hausdorff 거리는 이 점에 의하여 결정되므로 큰 값을 가지게 되어, 두 유한집합이 아주 다르다는 결론으로 이끌 수 있다. 이상치에 대한 이러한 민감도를 갖는 Hausdorff 거리는 두 집합 사이의 유사성을 측정하는 척도로 사용되기에는 치명적인 결함을 가지고 있으므로, 이상치에 대한 민감도를 낮추기 위하여 Hausdorff 거리를 변형한 여러 방법들이 제안되었다.

Huttenlocher 등[4]이 제시한, 집합 A 로부터 집합 B 까지의 Ranked Hausdorff 거리 $h_R^k(A, B)$ 는, 집합 A 의 각 점으로부터 집합 B 까지의 Hausdorff 거리 중에서 k 번째($1 \leq k \leq |A|$)의 최대값으로 다음과 같이 정의되며, k 값의 선택에 따라 이상치의 영향을 배제할 수 있다.

$$\begin{aligned} h_R^k(A, B) &= k^{th} \max_{a \in A} d(a, B) \\ &= k^{th} \max_{a \in A} \min_{b \in B} d(a, b) \\ &= k^{th} \max_{a \in A} \min_{b \in B} \|a - b\| \end{aligned} \quad (6)$$

Dubuisson 등[3]이 제시한, 집합 A 로부터 집합 B 까지의 Modified Hausdorff 거리 $h_M(A, B)$ 는, 집합 A 의 각 점으로

부터 집합 B까지의 Hausdorff 거리의 평균값으로 다음과 같이 정의되며, 이상치의 영향을 감소시키는 효과를 갖는다.

$$\begin{aligned} h_M(A, B) &= \frac{1}{|A|} \sum_{a \in A} d(a, B) \\ &= \frac{1}{|A|} \sum_{a \in A} \min_{b \in B} d(a, b) \\ &= \frac{1}{|A|} \sum_{a \in A} \min_{b \in B} \|a - b\| \end{aligned} \quad (7)$$

Modified Hausdorff 거리는 Ranked Hausdorff 거리와는 달리, 파라미터 값을 요구하지 않는 장점이 있다.

Lu 등[7]이 제시한, 집합 A로부터 집합 B까지의 Weighted Hausdorff 거리 $h_w(A, B)$ 는, 집합 A의 각 점으로부터 다음과 같이 집합 B까지의 Hausdorff 거리의 가중평균을 사용하여 정의되며, 이상치의 영향을 감소시키는 효과를 갖는다.

$$\begin{aligned} h_w(A, B) &= \frac{1}{N_A} \sum_{a \in A} w(a) \cdot d(a, B) \\ &= \frac{1}{N_A} \sum_{a \in A} w(a) \cdot \min_{b \in B} d(a, b) \\ &= \frac{1}{N_A} \sum_{a \in A} w(a) \cdot \min_{b \in B} \|a - b\| \end{aligned} \quad (8)$$

여기에서 $N_A = \sum_{a \in A} w(a)$ 이다.

Paumard[8]가 제시한, 집합 A로부터 집합 B까지의 Censored Hausdorff 거리 $h_c(A, B)$ 는, 집합 A의 각 점으로부터 집합 B까지의 거리의 Q번째 랭크된 값들 중에서 P번째 랭크된 값으로 다음과 같이 정의된다.

$$\begin{aligned} h_c^{P,Q}(A, B) &= P^{th} \max_{a \in A} Q^{th} d(a, B) \\ &= P^{th} \max_{a \in A} Q^{th} \min_{b \in B} d(a, b) \\ &= P^{th} \max_{a \in A} Q^{th} \min_{b \in B} \|a - b\| \end{aligned} \quad (9)$$

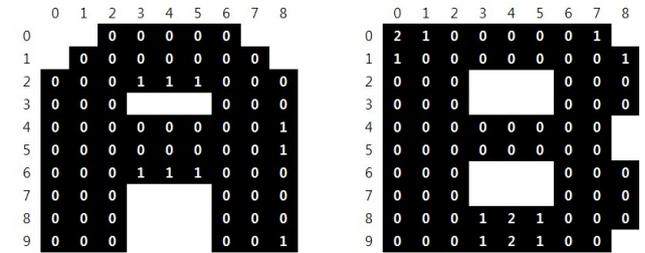
이 외에 M-estimation Hausdorff 거리[6, 10], Doubly Modified Hausdorff 거리[11], Least Trimmed Squared Hausdorff 거리[6, 10] 등이 제시되어 있다.

2.3 Grayscale Hausdorff 거리

흑백 이미지 문자 매칭 문제에서 Hausdorff 거리를 계산할 때 사용되는 두 개의 집합은 이진수로 표현되는 두 개의 이미지 파일을 의미한다. 앞 절에서 언급한 Hausdorff

거리들은 기본적으로 이상치가 포함된 이미지 파일을 변형없이 그대로 두고, Hausdorff 거리의 정의를 변형하여, 문자 매칭 문제를 다루었다.

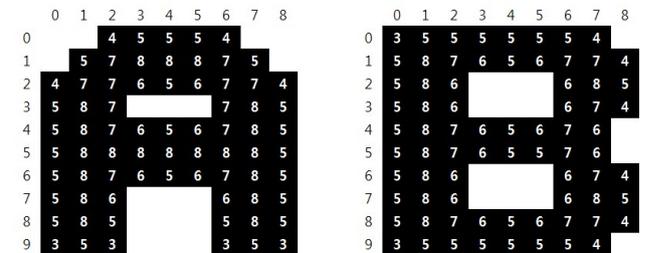
본 논문에서는 흑백이미지를 그레이스케일 이미지로 변형한 후 Hausdorff 거리 계산하여, 문자 매칭 문제를 해결하는 방법을 다룬다. 흑백 이미지 파일을 그레이스케일 이미지로 바꾸려면, 먼저 window의 크기를 결정해야 한다. 본 논문에서는 3×3 window를 사용한다. 어떤 픽셀이 검정색일 때, 주변 8개의 픽셀 중에서 검정색의 픽셀 수가 그 픽셀의 그레이스케일 값이 된다. 예를 들어 주변 8개의 픽셀이 모두 검정색이면, 그레이스케일 값은 8이 된다. 만일 주변 8개의 픽셀이 모두 흰색이면 그레이스케일 값은 0이 된다.



(a) Distance from image A to B (b) Distance from image B to A

<Figure 1> Hausdorff Distance between Image A and B

<Figure 1>은 흰 바탕에 검은 픽셀로 나타낸 알파벳 대문자 A와 B의 이진 이미지를 보여준다. <Figure 1>의 (a)에 있는 검은 바탕의 흰색 숫자는 이미지 A의 한 점에서 이미지 B까지의 Hausdorff 거리 $d(a, B)$ 를 나타낸다. 유사한 방법으로 <Figure 1>의 (b)에 있는 검은 바탕의 흰색 숫자는 이미지 B의 한 점에서 이미지 A까지의 Hausdorff 거리 $d(a, B)$ 를 나타낸다. <Figure 2>는 3×3 window를 사용할 때, <Figure 1>의 이미지 파일에 대한 그레이스케일 값을 검정색 픽셀 위의 흰색 숫자로 보여주고 있다.



(a) Grayscale of image A

(b) Grayscale of image B

<Figure 2> Grayscale of Image A and Image B

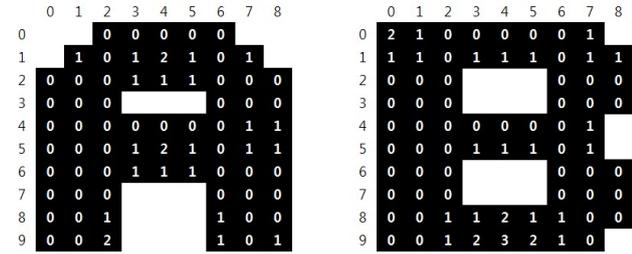
그레이스케일 이미지를 사용하여, Zhao 등[12]은 이미지 A로부터 이미지 B까지의 Grayscale Hausdorff 거리 $h(A,$

B)와 A의 한 픽셀 a_t 에서 B까지의 Grayscale Hausdorff 거리 $d(a_t, B)$ 를 다음과 같이 정의하였다.

$$h(A, B) = \max_{a_t \in A} d(a_t, B) \quad (10)$$

$$d(a_t, B) = \min \left(\min_{b_{t-1} \in B} \|a_t - b_{t-1}\|, \min_{b_t \in B} \|a_t - b_t\|, \min_{b_{t+1} \in A} \|a_t - b_{t+1}\| \right) \quad (11)$$

여기에서 t는 그 픽셀의 그레이스케일 값을 나타낸다. 이 모형은 A의 한 픽셀 a_t 에서 B까지의 Grayscale Hausdorff 거리 $d(a_t, B)$ 를 계산할 때, 동일 그레이스케일 값을 가진 픽셀까지의 최소값뿐만 아니라 ± 1 만큼 차이가 나는 그레이스케일 값을 가진 픽셀까지의 최소값을 반영한다. 이미지 A로부터 이미지 B까지의 Grayscale Hausdorff 거리 $h(A, B)$ 는 이미지 A의 각 점으로부터 이미지 A까지의 Hausdorff 거리의 최대치로 정의한다.



(a) Distance from image A to B (b) Distance from image B to A
 <Figure 3> Grayscale Hausdorff Distance From/To Image A To/From B

<Figure 3>은 이미지 A의 각 픽셀에서 이미지 B까지의 Grayscale Hausdorff 거리 및 이미지 B의 각 픽셀에서 이미지 A까지의 Grayscale Hausdorff 거리를 나타낸다. <Figure 3>을 이용하여 이미지 A로부터 이미지 B까지의 Grayscale Hausdorff 거리 $h(A, B)$ 를 구하면 2가 되고, $h(B, A)$ 는 3이 되어, 이미지 A로부터 이미지 B까지의 Grayscale Hausdorff 거리 $H(A, B)$ 는 3이 된다.

3. 제시 방법론 및 실험 결과

3.1 제시 방법론

본 논문에서는 식 (10) 및 식 (11)로 표현되는 Grayscale Hausdorff 거리를 변형하여, 이미지 A로부터 이미지 B까지의 Grayscale Hausdorff 거리 $h(A, B)$ 와 A의 한 픽셀 a_t

에서 B까지의 Grayscale Hausdorff 거리 $d(a_t, B)$ 를 다음과 같이 정의한다.

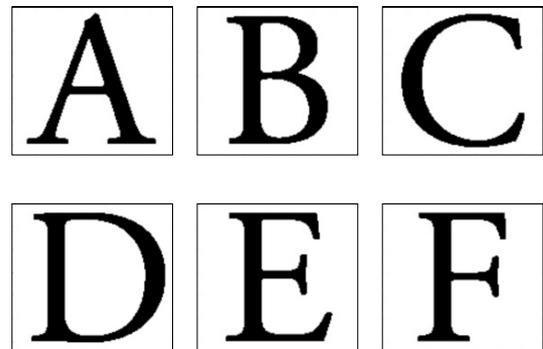
$$h(A, B) = \frac{1}{|A|} \sum_{a_t \in A} d(a_t, B) \quad (12)$$

$$d(a_t, B) = \min \left(\min_{b_{t-1} \in B} \|a_t - b_{t-1}\|, \min_{b_t \in B} \|a_t - b_t\|, \min_{b_{t+1} \in A} \|a_t - b_{t+1}\| \right) \quad (13)$$

이 모형을 Modified Grayscale Hausdorff 거리 모형이라 부르자. 이 모형은 A의 한 픽셀 a_t 에서 B까지의 Grayscale Hausdorff 거리 $d(a_t, B)$ 를 계산할 때, 동일 그레이스케일 값을 가진 픽셀까지의 최소값뿐만 아니라 ± 1 만큼 차이가 나는 그레이스케일 값을 가진 픽셀까지의 최소값을 반영한다는 점에서는 Grayscale Hausdorff 거리와 동일하다. 다만, 이미지 A로부터 이미지 B까지의 Grayscale Hausdorff 거리 $h(A, B)$ 는 이미지 A의 각 점으로부터 이미지 B까지의 Hausdorff 거리의 평균값으로 정의한다는 점에서, 최대치로 정의하는 Grayscale Hausdorff 모형과 다르다. 이와 같은 정의를 통하여 이상치의 영향을 감소시킴으로써, 노이즈의 증가에 따라 문자 매칭의 하락비율이 완만해 질 것으로 기대한다.

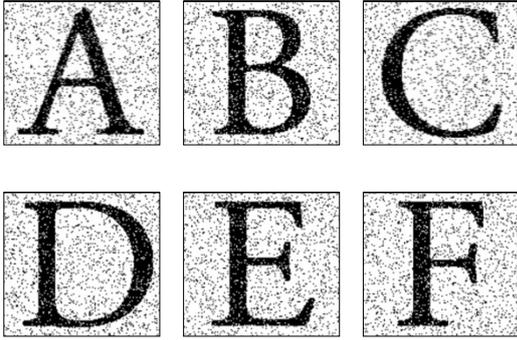
3.2 실험 결과

본 논문에서는 Adobe Garamond Pro 폰트의 알파벳 스물여섯 글자의 이미지를 원본 문자의 이미지로 사용하였다. <Figure 4>는 실험에 사용한 노이즈가 없는 원본 문자의 이미지들의 예를 보여준다.



<Figure 4> Noise-free Letters

주어진 이미지에 0%에서 10%까지 매 2%씩 픽셀 사이즈가 1인 흑백 점잡음을 이미지 전체에 랜덤하게 증가시킨 6세트의 테스트 이미지를 만들었다.



<Figure 5> Letters with 10% Salt-and-Pepper Noise

<Figure 5>는 실험에서 사용한 흑백 점잡음이 10%인 이미지들의 예를 보여준다. 이 이미지들에 대하여 다음 4개의 모형을 사용하여 원본 이미지들과의 거리를 계산한 다음, 거리가 가장 짧은 원본 이미지의 문자를 그 테스트 이미지의 문자로 매치시킨다.

1) 모형 1

기본적인 Hausdorff 거리의 그레이스케일 버전으로 다음 식으로 표현된다.

$$h(A, B) = \max_{a_t \in A} d(a_t, B) \quad (14)$$

$$d(a_t, B) = \min_{b_t \in B} \|a_t - b_t\| \quad (15)$$

2) 모형 2

본 논문에서 사용되는 다른 척도와 구별하기 위하여 사용하는 Grayscale Hausdorff 거리의 다른 이름이다. 식 (10)과 식 (11)로 표현된다.

3) 모형 3

Modified Hausdorff 거리의 그레이스케일 버전으로 다음 식으로 표현된다.

$$h(A, B) = \frac{1}{|A|} \sum_{a_t \in A} d(a_t, B) \quad (16)$$

$$d(a_t, B) = \min_{b_t \in B} \|a_t - b_t\| \quad (17)$$

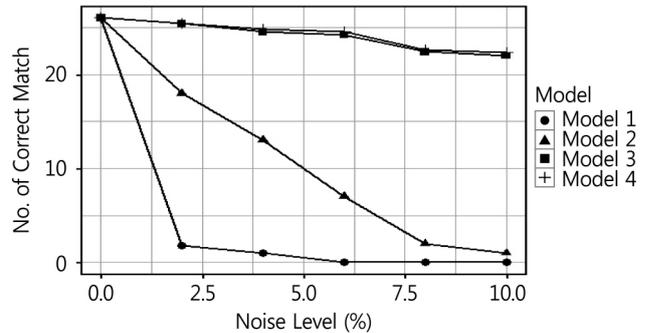
4) 모형 4

본 논문에서 제시한 Modified Grayscale Hausdorff 거리 모형으로 식 (12)와 식 (13)으로 표현된다.

본 실험에서는 두 픽셀 사이의 거리 $\|a-b\|$ 는 다음과 같은 city-block 거리를 사용하였다.

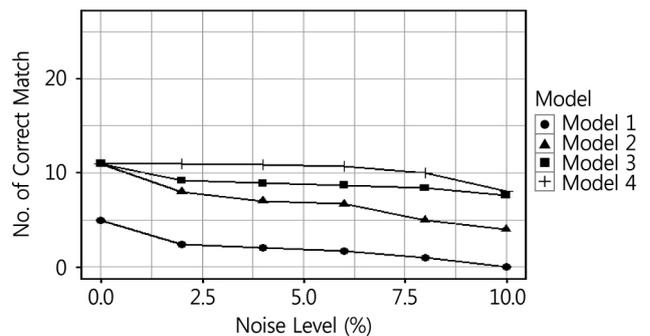
$$\|a-b\| = |a_x - b_x| + |a_y - b_y| \quad (18)$$

여기에서 a_x 는 픽셀 a 의 x 축 좌표 값이며 a_y 는 픽셀 a 의 y 축 좌표 값이다. 주어진 4개의 Grayscale Hausdorff 거리를 사용하여 노이즈가 포함된 테스트 이미지(0%, 2%, 4%, 6%, 8%, 10%)와 노이즈가 없는 원본 문자의 이미지 사이의 거리를 계산하였다. 노이즈가 0%인 테스트 이미지의 경우, 4개의 모형 모두 매칭되는 문자를 완벽하게 찾아냈으나, 노이즈가 있는 경우, 모두 거리가 무한대로 계산되었다. 이는, 테스트 이미지를 그레이스케일 이미지로 변환한 후, Grayscale Hausdorff 거리를 계산하는 과정에서, 픽셀의 그레이스케일 값 t 가 1이거나 2인 경우, 한 쪽 이미지에 그러한 값이 존재하나, 다른 쪽 이미지에 그러한 값이 존재하지 않는 경우가 발생하였기 때문이었다. 이런 경우, 계산 값은 무한대가 되어, 사용된 4개의 모형 모두에서 거리가 무한대가 된다. 이를 방지하기 위하여, 본 실험에서는 이러한 경우가 발생하면 계산하지 않고 무시하였다.



<Figure 6> Number of Correct Match for Original Fonts

<Figure 6>는 4개의 모형을 사용한 실험 결과를 보여준다. 노이즈가 없는 경우, 4개의 모형 모두 테스트 이미지에 해당하는 문자를 정확하게 찾아내었다. 그러나, 모형 1을 사용하면, 낮은 수준의 노이즈에서도 정확도가 급격히 하락하며, 모형 2를 사용하면, 노이즈의 증가에 따라 정확도가 꾸준히 하락 하였다. 모형 3과 모형 4를 사용하면, 10%까지의 노이즈에서 해당하는 문자를 비교적 정확하게 찾아내었다.



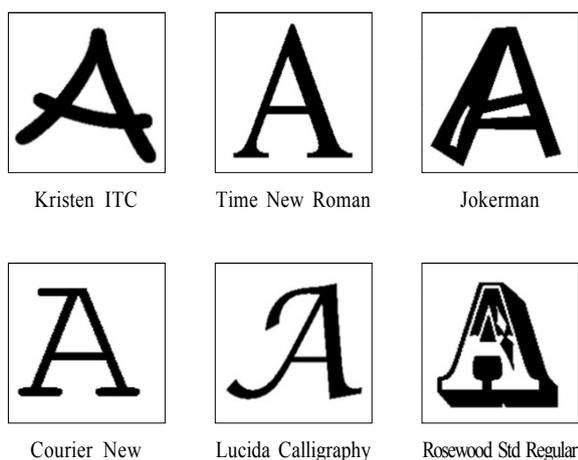
<Figure 7> Number of Correct Match for Italic Fonts

그러나, 동일 폰트의 이탤릭체를 테스트 이미지로 하여 실험한 결과 <Figure 7>에서 보듯이 4개의 모형 전부에서 매칭의 정확도가 크게 떨어졌으며, 전체적으로는 모형 4가 모형 3보다 정확도가 약간 높았다.

<Table 1> Number of Correct Match for Right-Moved-Image by 5%

Model	1	2	3	4
No. of Correct Match	22	26	26	26

원본 문자의 이미지에서 문자의 위치를 5% 우측으로 이동하여 노이즈가 없는 테스트 이미지를 만들어, 문자의 위치에 따른 민감도를 살펴보았다. <Table 1>은 모형 2, 3, 4를 사용하면 문자의 위치가 변하였음에도 불구하고 노이즈가 없는 상태에서는 정확하게 해당 문자를 인식하였음을 보여준다.



<Figure 8> Fonts used for Test Images

<Table 2> Number of Correct Match for Test Fonts

Font	Model			
	1	2	3	4
Kristen ITC	2	9	5	11
Times New Roman	7	19	8	10
Jokerman	0	7	9	9
Courier New	2	8	8	8
Lucida Calligraphy	2	4	4	5
Rosewood Std Regular	4	6	6	4

<Figure 8>에 나타난 바와 같은 여러 가지 폰트들을 테스트 이미지로 사용하여 노이즈가 없는 상태에서, 원본 폰트(Adobe Garamond Pro)를 사용하여 문자 매칭을 실험

한 결과를 <Table 2>에 요약하였다. 서로 다른 폰트에 대한 문자 매칭의 정확도는 폰트에 따라 편차를 보였으며, 다른 폰트들을 인식하는데 사용하기에는 문자 매칭의 정확도가 떨어짐을 알 수 있다.

4. 결론

본 연구에서는 흑백 점잡음을 가진 이미지와 매치하는 문자를 찾아내기 위하여 Hausdorff 거리를 사용하였다. 노이즈 효과를 감소시키기 위하여, 그레이스케일로 변환한 후, 4개의 Grayscale Hausdorff 거리에 대하여 여러 수준의 노이즈를 가진 테스트 이미지와 원본 문자를 매치시키는 실험을 하였다. 그 결과, 모형 3과 본 논문에서 제시한 Modified Grayscale Hausdorff 거리 모형(모형 4)을 사용하는 것이 나머지 두 모형을 사용하는 것보다 문자 매칭의 정확도가 훨씬 더 높은 결과를 가져왔다. 추가적으로, 전체 픽셀에 대하여 오른쪽으로 5% 치우친 문자 이미지를 만들어 실험을 한 결과, 문자 매칭의 정확도는 약간 낮아졌다. 원본 폰트의 이탤릭체를 테스트 이미지로 사용하여 실험을 한 결과, 문자 매칭의 정확도는 본 논문에서 제시한 모형이 가장 높았으나, 모든 모형에서 정확도가 현저히 떨어졌다. 이러한 결과들을 종합하여 볼 때, 본 논문에서 제시한 모형은, 문자의 위치변화나 저 수준의 흑백 점잡음에 거의 영향을 받지 않으므로, 폰트의 종류가 제한된 문자의 인식에 사용이 가능할 것으로 보인다. 이러한 예로서 문서인식, 자동차 번호판 인식 등이 있다. 그러나, 실험결과에서도 알 수 있듯이, 서로 다른 폰트를 사용하는 경우, 문자 매칭의 정확도가 현저히 떨어져, 이 모델을 적용하는 것은 불가능하다.

본 연구의 한계로는, 노이즈의 크기를 1 픽셀로 한정 한 바, 크기가 2 픽셀 이상인 경우에 대한 실험에서도 유사한 결과를 얻는지 여부를 알 수 없다는 점에 있다. 아울러, 숫자 및 특수문자의 매칭에 관한 실험과 다른 언어의 문자에 대한 실험도 진행되지 못했다.

본 모형의 적용범위를 확장하기 위한 추후 연구과제로, 여러 종류의 폰트를 몇 개의 유사 그룹으로 나누어, 각 그룹의 대표 폰트를 선택하여, 이들을 모두 원본 문자 이미지로 사용하여 문자 매칭의 정확성을 측정하는 실험이 있을 수 있다.

Acknowledgement

This study has been partially supported by a 2013 Research Fund of Hannam University, Korea.

References

- [1] Baudrier, E., Nicolier, F., Millon, G., and Ruan, S., Binary-Image Comparison with Local-Dissimilarity Quantification. *Pattern Recognition*, 2008, Vol. 42, pp. 1461-1478.
- [2] Choudhary, A., Rishi, R., and Ahlawat, S., A New Approach to Detect and Extract Characters from Off-Line Printed Images and Text. *Procedia Computer Science*, 2013, Vol. 17, pp. 434-440.
- [3] Dubuission, M.P. and Jain, A.K., A Modified Hausdorff Distance for Object Matching. *Proc. of the 12th International Conference on Pattern Matching*, 1994, Vol. 1, pp. 566-568.
- [4] Farahmand, A., Sarrafzadeh, A., and Shanbehzadeh, J., Document Image Noises and Removal Methods. *Proceedings of the International Multi Conference of Engineering and Computer Science*, 2013, Vol. 1, pp. 436-440.
- [5] Huttenlocher, D.P., Klanderman, G.A., and Rucklidge, W.J., "Comparing Images Using the Hausdorff Distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1993, Vol. 15, No. 9, pp. 850-863.
- [6] Kwon, O.-K., Sim, D.-G., and Park, R.-H., Robust Hausdorff Distance Matching Algorithms using Pyramidal Structures. *Pattern Recognition*, 2001, Vol. 34, No. 10, pp. 2005-2013.
- [7] Lu, Y., Tan, C.L., Huang, W., and Fan, L., An Approach to Word Image Matching based on Weighted Hausdorff Distance. *Proc. of the 6th International Conference on Document Analysis and Recognition*, 2001, pp. 921-925.
- [8] Paumard, J., Robust Comparison of Binary Images. *Pattern Recognition Letters*, 1997, Vol. 18, No. 10, pp. 1057-1063.
- [9] Premchaiswadi, N., Yimngam, S., and Premchaiswadi, W., A Scheme for Salt and Pepper Noise Reduction on Graylevel and Color Image. *Proceedings of the 9th WSEAS International Conference on Signal Processing, Computational Geometry and Artificial Vision*, 2009, pp. 57-61.
- [10] Sim, D.-G., Kwon, O.-K., and Park, R.-H., Object Matching Algorithms using Robust Hausdorff Distance Measures. *IEEE Trans. on Image Processing*, 1999, Vol. 8, No. 3, pp. 425-429.
- [11] Takacs, B., Comparing Faces using the Modified Hausdorff Distance. *Pattern Recognition*, 1998, Vol. 31, No. 12, pp. 1873-1881.
- [12] Zhao, C., Shi, W., and Deng, Y., A New Hausdorff Distance for Image Matching. *Pattern Recognition Letters*, 2005, Vol. 26, pp. 581-586.

ORCIDKyeongtaek Kim | <http://orcid.org/0000-0002-5636-2487>Ji Hun Kyung | <http://orcid.org/0000-0002-0359-5594>