Journal of Korea Multimedia Society Vol. 23, No. 11, November 2020(pp. 1361-1371)
https://doi.org/10.9717/kmms.2020.23.11.1361

한글 획요소 추출 학습에서 적용 글자의 확장에 따른 추출 성능 분석

전자연†, 임순범††

Analysis of Extraction Performance according to the Expanding of Applied Character in Hangul Stroke Element Extraction

Ja-Yeon Jeon†, Soon-Bum Lim††

ABSTRACT

Fonts have developed as a visual element, and their influence has rapidly increased around the world. Research on font automation is actively being conducted mainly in English because Hangul is a combination character and the structure is complicated. In the previous study to solve this problem, the stroke element of the character was automatically extracted by applying the object detection by component. However, the previous research was only for similarity, so it was tested on various print style fonts, but it has not been tested on other characters. In order to extract the stroke elements of all characters and fonts, we performed a performance analysis experiment according to the expansion character in the Hangul stroke element extraction training. The results were all high overall. In particular, in the font expansion type, the extraction success rate was high regardless of having done the training or not. In the character expansion type, the extraction success rate of trained characters was slightly higher than that of untrained characters. In conclusion, for the perfect Hangul stroke element extraction model, we will introduce Semi-Supervised Learning to increase the number of data and strengthen it.

Key words: Hangul Stroke Element, Expansion of Object Detection, Automatic Extraction, Components

1. 서 론

현재 정보통신의 발달로 영상 등의 미디어 매체를 중심으로 활발한 커뮤니케이션이 진행되고 있어 글꼴은 다양한 정보 전달의 수단뿐만 아니라 다양한 효과의 시각적인 요소로 발전하고 있다[1,2]. 이에 맞게 영문을 중심으로 최근 딥러닝 기술을 이용하여 글꼴을 자동으로 분류하거나 유사 글꼴을 판단하는 연구가 활발히 진행되고 있다[3-6]. 하지만 한글은 조합형 글자로 영문과 달리 구조가 복잡하여 글자 이미지 기반으로 분석하기에 비교적 어려움이 많아 글꼴을 자동으로 분류하거나 판단하는 시스템이 상대적으로 미흡하다. 그렇기에 육 천종이 넘는 한글 글꼴[7]의 특징을 수작업으로 축적하고 분석해야 하는 문제점을 해결하고자 선행 연구로 기계학습과 딥러닝을 이용하여 한글 글꼴의 획요소를 자동으로 검출하고 추출하였다[8-10].

선행 연구에서는 한글 글꼴을 시각적인 요소로 분

※ Corresponding Author : Soon-Bum Lim, Address: (04310) Cheongpa-ro 47-gil 100, Yongsan-gu, Seoul, Korea, TEL : +82-2-710-9424, FAX : +82-2-710-9704, E-mail : sblim@sm.ac.kr
Receipt date : Sep. 16, 2020, Revision date : Oct. 7, 2020, Approval date : Oct. 22, 2020
† Dept. of IT Engineering, Graduate School, Sookmyung Women's University
(E-mail : jyjeon@sookmyung.ac.kr)
†† Dept. of IT Engineering and Research Institute ICT Convergence, Sookmyung Women's University
※ This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (NRF-2018R1A4A1025559) and This research was supported by stage 4 BK21 project in Sookmyung Women's Univ of the National Research Foundation of Korea Grant.

석(Visual Font Recognition)하기 위해 컴포넌트 단위로 객체를 검출하는 기법을 적용하였다. 이미지에서 컴포넌트 단위로 객체을 검출하는 대표적인 사례로는 신체를 머리, 팔, 다리 등의 부위별로 검출하여 사람의 행동을 감지하는 연구가 있으며, 동물의 얼굴과 팔, 다리를 검출하여 FGC(Fine-Grained Classification) 단위의 동물의 품종을 분류하는 연구 등이 있다[11-13]. 따라서 이미지 검출에 사용되었던 기법과 마찬가지로 한글 글자의 구조 정보를 이용하여 구성 요소 중 하나인 획요소를 SVM(Support Vector Machine)을 이용하여 시도해보고 또 다른 방법으로 Faster R-CNN 기법을 사용하여 글자에서 자동으로 추출해서 100%의 정확도의 검증 결과를 얻었다.

그러나 선행 연구는 한글 글꼴의 유사도 판단을 목적으로 하여 학습한 네 개의 글자에서만 획요소를 추출하는 방식으로 검증을 진행하였다. 학습 모델의 성능은 다른 글자에서는 아직 검증되지 않았다. 한글은 총 11,172자로 구성되어 있어 일일이 모든 글꼴과 글자에서 획요소 검출 학습을 진행하기는 현실적으로 불가능하여[14-16] 본 연구에서는 획요소 추출의 적용 대상을 학습한 글자 외의 다른 글자로 확장했을 때에도 제대로 추출이 가능한 지 확인해보고자 한다. 모든 글자와 글꼴을 대상으로 획요소가 자동으로 추출되어 이를 바탕으로 글꼴의 시각적 분석(Visual Font Recognition)을 진행한다면 한글 글꼴의 유사도 판단 외에도 글꼴 추천, 분류, 검색 등 한글에서도 모든 분야에서 자동화가 가능해질 것이다.

본 논문의 구성은 다음과 같다. 2장에서 본 논문의 기초가 되는 컴포넌트 단위의 객체 검출 연구와 선행 연구로 진행한 한글 획요소 추출 연구를 소개한 다음, 3장에서 한글 획요소 추출 적용 글자 확장을 위해 설계한 실험 모델을 설명한다. 여기에서는 실험을 진행할 학습 및 적용 모델과 실험을 진행할 태스크, 마지막으로 실험 및 평가 방법의 순서로 서술한다. 그리고 4장에서는 제안한 방법으로 실험한 결과를 통계적인 방법으로 분석하여 확장 유형에 따른 추출 성능을 평가하고 5장에서 결론을 맺는다.

2. 관련 연구

2.1 SVM-HMM 기반의 신체의 구성 요소 검출을 통한 행동 인식 확장 연구

SVM-HMM(Support Vector Machine-Hidden Markov Mode) 모델을 활용하여 각도와 상관없이 사람을 검출하고 행동을 인식하는 모델을 제안하였다[11]. 검출 과정에서 사람의 머리, 팔, 다리 등의 컴포넌트로 개별적으로 분류하는 학습을 진행하고 검출된 신체 부위의 포즈 변화를 분석하는 알고리즘을 적용하여 걷기, 조깅, 손 흔들기, 권투 등과 같은 행동을 인식한다. 따라서 감지된 손과 팔을 기반으로 권투, 손 흔들기 등의 행동을 판단할 수 있고, 다리 등을 이용하여 걷기, 달리기 등과 같은 행동을 인식한다.

해당 연구에서는 야외, 실외, 실내 등 다양한 조건에서 촬영된 남녀 25명의 이미지 데이터베이스를 기반으로 모델을 검증하였다. 그 결과, 제안된 방법이 전신 기반의 모델보다 더 나은 탐지율을 보였으며, 특히 걷기, 달리기, 권투, 박수 등의 동작에서 높은 정확도를 보였다.

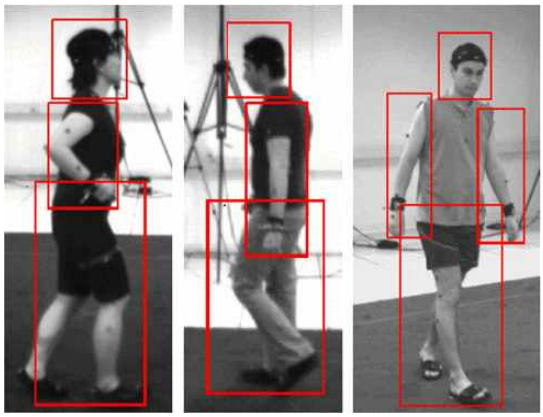


Fig. 1. Human-Body Component Detector proposed in Section 2.1[11].

2.2 MP-CNN(Multi-Part Convolutional Neural Network)를 사용한 동물 품종 분류 연구

야생 동물을 효과적으로 모니터링하고 감지하는 시스템을 구축하기 위해 MP-CNN(Multi-Part Convolutional Neural Network)을 이용하여 동물 품종을 분류하였다[12]. 이미지 분류는 동물을 개나 고양이로 분류하는 일반적인 분류와 더 세분화하여 개를 푸들 또는 퍼그로 분류하는 FGC(Fine-Grained Classification)으로 나뉜다. 해당 연구에서는 제안한 MP-CNN을 이용하여 동물의 머리, 상체, 앞다리, 뒷다리 등의 구별되는 부분을 검출하여 FGC 수준으로

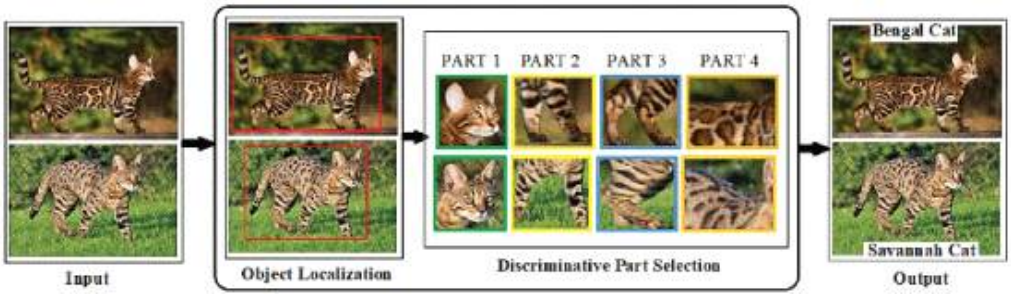


Fig. 2. Overview of proposed MP-CNN model [12].

동물 종을 분류한다. 따라서 해당 시스템은 MP-CNN을 이용해 개, 고양이, 코끼리 등의 일곱 가지 동물을 분류하며, 해당 범주 안에서 더 세분화하여 총 27종의 동물을 분류한다.

제안된 시스템의 검증 결과, 약 96%의 정확도로 동물의 전체적인 이미지를 기반으로 분류한 타 모델에 비해 좋은 결과를 보였다.

2.3 Faster R-CNN 기반 한글 글꼴 획요소 자동 추출 연구

Faster R-CNN은 딥러닝 객체 검출(Object Detection) 알고리즘 중 하나로 다른 알고리즘에 비해 작은 크기의 객체를 검출할 때 특히 정확도가 높다. 이 연구에서는 Tensorflow Object Detection API를 통해 Faster R-CNN Inception v2 모델을 파인 튜닝(Fine-Tuning) 하여 글자 이미지에서 한글 글꼴 획요소를 검출하는 학습을 진행하였다. 더하여 검출된 획요소 영역의 xmin, ymin, xmax, ymax 좌표 값을 기반으로 크롭하는 추출 알고리즘을 제안하여 검출된 한글 획요소의 영역을 자동으로 추출한다[10].

해당 연구는 한글 글꼴 간의 유사도 판단을 하고자 진행된 연구로 추출할 획요소를 선정하기 위해 한글 글꼴의 모양에 대한 속성을 정리한 후 폰트 업계 종사자 5명을 대상으로 유사도 판단을 위한 대표 한글 획요소를 선정하는 평가를 진행하였다. 그 결과 한글 글꼴 획요소를 '삐침', '꺾임', '돌출', '닫힘', '상투', '부리', '가지', '꼭지점' 총 8가지로 선정하였으며, 글꼴 범위는 정형글꼴 계층[17]으로 한정하였다.

자동 추출 학습의 검증하기 위해서 한국정보통신기술협회(TTA)와 글꼴 분류를 위한 한글 글꼴의 모양 특성 연구의 한글 모양 분류체계 표준[18-20]에 맞게 총 13종의 글꼴을 선별하였다. 정형계층의 글꼴을 Serif 및 Sans-serif 등 계열별로 고루 수집하였으며, 더하여 굵기별로도 구분하여 특정 모양의 글꼴에만 치중되는 분류 쏠림 현상을 방지할 수 있도록 Validation data를 선정하였다. 검증 결과, 여덟 가지의 획요소에서 13종의 글꼴로 추출하는 104번의 검증에서 모두 정확히 획요소가 검출되는 100%의 결과를 보였다. 따라서 해당 연구를 통해 제안된 시스템으로 한글 글꼴 획요소를 자동으로 추출하여 글꼴 유사도

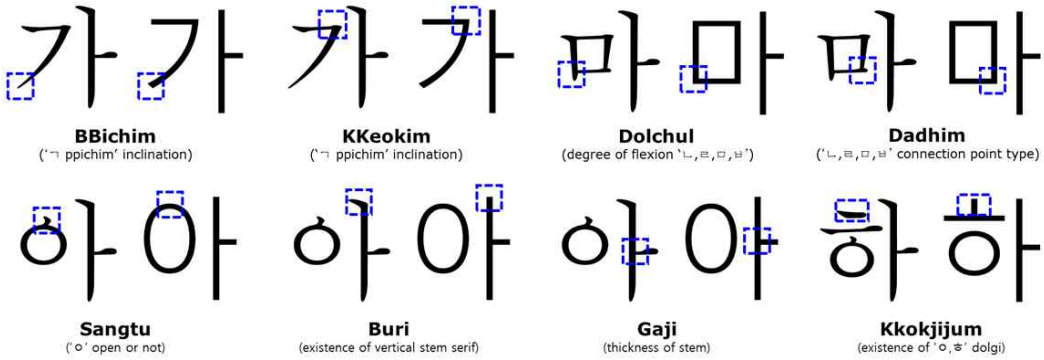


Fig. 3. Defining Stroke Elements in Section 2.4 [10].

판단에 사용이 가능하다는 결과가 도출되었다.

3. 한글 획요소 추출 적용 글자의 확장을 위한 실험 모델 설계

3.1 학습 및 적용 모델

한글 획요소 추출 적용 글자의 확장을 위한 실험을 위해 먼저 2,3절의 Faster R-CNN 기반 시스템을 이용하여 한글 획요소 추출 학습을 진행하였다. 글꼴에 대한 평가를 진행하여 정형계층의 글꼴 170종을 수집하였으며, 각 획요소를 검출할 글자는 삐침과 꺾임은 '가', 돌출과 닫힘은 '마', 상투와 부리, 가지는 '아', 꼭지점은 '하'로 선정하여 학습시킬 데이터를 구성하였다. 따라서 여덟 가지의 획요소당 170종의 글꼴에서 선정한 글자를 이미지 포맷으로 변환하여 총 1,360장의 이미지 데이터를 대상으로 한글 획요소 검출 학습을 진행하였다. 획요소 마다 학습을 진행하여 총 여덟 번의 학습을 하였으며, 모두 로스율이 0.05 아래로 확연히 감소될 때까지 진행하여 평균 43,109번의 학습을 하였다. 해당 시스템의 학습 모델은 Table 1에 자세히 표기되어 있다.

Table 1. Training model for expansion experiment of extracting Hangul stroke elements

No	Stroke elements	Train characters	Train fonts	Train steps
1	bbichim	가	170 of regular fonts	43,397
2	kkeokim	가	170 of regular fonts	43,267
3	dolchul	마	170 of regular fonts	44,085
4	dadhim	마	170 of regular fonts	42,147
5	sangtu	아	170 of regular fonts	43,516
6	buri	아	170 of regular fonts	42,356
7	gaji	아	170 of regular fonts	43,686
8	kkokjijum	하	170 of regular fonts	42,419

3.2 실험 태스크

획요소 추출 적용 글자의 확장 유형은 글자 범위의 확장, 그리고 글자와 글꼴 범위의 확장으로 정의하여 구분한다. 따라서 확장 글자에 따른 추출 성능 비교 실험을 위해 다음과 같은 태스크를 수행한다. 글자 범위를 확장시킨 첫 번째 태스크(Task 1)에서는 학습시킨 글꼴 170종으로 글꼴을 선택하며, 글자는 학습시키지 않은 글자 '훈', '민', '정', '음' 을 대상으로 획요소 자동 추출을 진행한다. 그리고 글꼴과 글자 범위를 확장시킨 두 번째 태스크(Task 2)에서는 학습하지 않은 글꼴에서 학습시키지 않은 글자 '훈', '민', '정', '음'을 대상으로 획요소 추출을 진행한다. 또한 글꼴 범위를 확장시킨 선행 연구의 검증 모델도 함께 비교해보고자 학습시킨 170종 이외의 정형글꼴 계층의 글꼴을 대상으로 자동 추출을 진행한다. 이때, 글자는 학습시킨 글자인 '가', '마', '아', '하' 를 대상으로 정한다. 따라서 Task 1과 Task 2 그리고 선행 연구의 검증 모델(Validation Model)의 세 번의 태스크를 수행한 후에 추출된 획요소에 대해 평가를 진행한다. 학습시키지 않은 글자인 '훈', '민', '정', '음'은 여덟 가지의 획 요소가 모두 들어있는 글자로 선

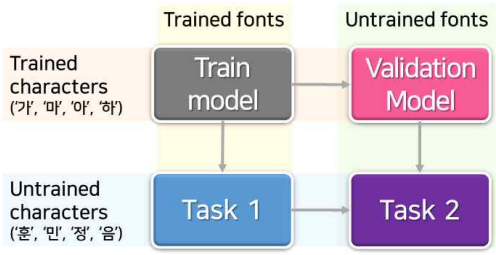


Fig. 4. Experimental task for expanding applied character.

Table 2. Selected font for evaluation for font expansion

Type	Thick-ness	Trained fonts	Untrained fonts
Serif	L	KBIZ한마음명조L	바탕체
Serif	L	조선일보명조	HY신명조
Serif	M	KoPupworld바탕M	나눔명조
Serif	M	순바탕 굵은체	산돌이야기M
Serif	B	타이포_씨명조 180	HY견명조
Serif	B	한컴소망B	산돌독수리B
San-Serif	L	Dx단선고딕Thin	굴림체
San-Serif	L	SpoqaHanSans	돋움체
San-Serif	M	10X10	맑은고딕
San-Serif	M	롯데마트드림B	나눔스퀘어
San-Serif	B	휴먼둥근헤드라인	빙그레체
San-Serif	B	Dx봄결ExB	휴먼옛체
San-Serif	B	Tmon몬소리B	타이포도담체

정하였으며, 초성, 중성, 종성의 자소들이 겹치지 않는 글자로 선정하였다.

실험을 위한 글꼴은 선행 연구의 검증 과정에서 사용하였던 한글 모양 분류체계 표준 기반의 글꼴을 사용하여 Serif, Sans-serif과 같은 글꼴의 계열과 Bold, Medium, Light와 같은 글꼴의 굵기별로 골고루 선정하였다. 따라서 다양한 모양의 한글 글꼴을 기반으로 실험이 가능하며 추가로 글꼴과 글자에 따른 확장 유형 외의 글꼴의 계열이나 굵기에 따라서도 한글 글꼴 획요소 추출 성능에 차이가 있는지 확인이 가능하다. 170종의 학습한 글꼴도 학습하지 않은 글꼴의 기준에 따라 맞추어 선정하여 글꼴 확장에 대한 실험 데이터를 구성하였으며, Table 2에 자세히 표기하였다.

3.3 실험 및 평가 방법

한글 획요소 추출 적용 글자의 확장을 위한 실험에서 추출이 제대로 되었는지 평가를 하기 위한 척도는 글자 내에서 정의한 획요소가 추출되었는가를 측정하는 성공률로 정의하였다. 성공률은 '1) 정의한 획요소가 제대로 추출되었다.', '2) 정의한 획요소와 모양은 같으나 정의하지 않은 획요소가 추출되었다.', '3) 정의하지 않은 획요소가 잘못 추출 되었다.' 라는 세 가지의 기준으로 구분하였으며 3점 리커트 척도 및 부분 점수로 측정한다. 이 때 두 번째 기준인 정의한 획요소와 모양은 같으나 정의하지 않은 획요소가 추출된 경우는 해당 시스템은 글자의 이미지 상에서 획요소의 모양만을 판단하여 획요소를 검출하기 때문에 Fig. 5와 같이 낱자 'ㅁ'과 'ㄴ'에서 획 요소 돌출의 모양이 같은 경우, 'ㄴ'과 'ㅣ'에서 획 요소 부리의 모양이 같은 경우 등 모양은 같지만 정의한 획요소가 아닌 경우가 존재한다. 따라서 시스템이 잘못 추출하였다고 단언할 수는 없지만 원하는 획요소는 아니므로 해당 경우를 위해서 정의된 기준이다.

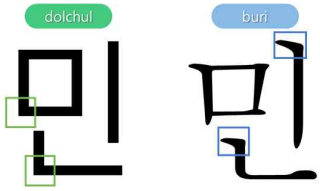


Fig. 5. Cases in which the defined stroke element and shape are the same, but undefined stroke elements are extracted.

글자별로 획요소 추출에 대한 점수를 측정한 후 각 글자별 획요소 추출 점수를 평균으로 계산하여 성공률을 측정하도록 정의한다. 획요소 추출에 대한 점수를 계산하는 방식은 객체 검출(Object Detection) 알고리즘의 성능을 평가하는 지표 중의 하나인 Precision[21] 기법을 활용한다. Precision은 모든 검출 결과 중 옳게 검출한 비율을 의미하며 옳은 검출은 TP(True Positive), 틀린 검출은 FP(False Positive)라 표기한다. Precision에 대한 계산식은 식 (1)에 나타나있다.

$$Precision = \frac{True\ \ Positive}{All\ \ Detections} \tag{1}$$

본 연구에서는 글자별 점수 계산 방식을 정의하기 위해 Precision 기법을 기반으로 한 후 Fig. 5와 같은 상황을 포함하기 위해 모양은 같지만 정의하지 않은 영역에서 추출된 획요소를 의미하는 Similar Positive 개념을 추가하였다. 따라서 글자에서 모든 획요소 추출 결과에서 옳게 검출한 비율을 계산한 후 모양은 같은데 정의하지 않은 획요소가 추출된 개수와 이에 대한 가중치를 곱하여 감점하고, 틀린 검출과 이에 대한 가중치를 곱하여 감점하여 글자별 추출 점수를 측정한다.

글자 기준의 획요소 추출 점수를 계산하는 식 (2)와 식 (2)에서 계산된 각 글자별 획요소 추출 점수의 평균으로 성공률을 계산하는 식 (3)으로 다음과 같이 정의하였으며, 점수 파악을 용이하게 하고 통계적 분석을 위해 글자당 획요소 추출 점수는 10점을 만점으로 정하였다. 이 때, w1는 False Positive에 대한 감점 가중치, w2는 Similar Positive에 대한 감점 가중치로 각각 1점과 0.5점으로 정의한다.

$$Score = \left[\frac{True\ \ Positive}{All\ \ Detections} - \{False\ \ Positive \times w_1\} - \{Similar\ \ Positive\, TIMES w_2\}\right] \times 10 \tag{2}$$

$$Success\ \ rate = \sum_{i=k}^{n} Score(k) \tag{3}$$

4. 한글 획요소 추출 적용 글자의 확장 유형 따른 실험 결과

4.1 실험 결과

Faster R-CNN 기반의 한글 글꼴 획요소 자동 추

출 시스템을 이용하여 3절에서 설계한 실험 모델대로 한글 획요소 자동 추출을 진행하였으며 min_score_thresh 값은 0.99로 두어 찾고자 하는 획요소일 확률이 99% 이상인 영역만 검출되도록 하였다. 실험 결과, 대체적으로 학습한 글자인 '가', '마', '아', '하' 에서는 여덟 가지의 획요소가 대체적으로 제대로 추출되었으나, 학습하지 않은 글자인 '훈', '민', '정', '음' 에서는 학습한 글자에 비해 제대로 추출되지 못한 경우가 많았다. 또한 낱자 'ㅁ' 이나 'ㄴ'에서 추출되는 획요소인 '꺾임', '돌출', '닫힘'은 Sans-serif 계열에서 'ㅇ'과 혼동하는 경우가 발생하는 등 학습하지 않은 글자에서 제대로 추출되지 않는 경우가 다수 존재하였다. 더하여 '부리'를 학습한 모델에서는 Serif 계열에서는 '훈'의 'ㄴ' 부분이나 Sans-serif 계열에서 '훈'이나 '하'의 꼭지점 부분의 모양이 똑같아 잘못 추출된 경우가 발생하였으며, '가지'를 학습한 모델에서는 'ㅏ'의 가지를 학습시켰으나 'ㅓ', 'ㅗ'의 가지도 검출하는 경우가 발생하였다. '삐침'을 학습한 모델에서는 'ㄱ'의 삐침을 학습시켰으나 'ㅈ'의 양쪽 줄기 모두 '삐침'으로 검출되는 경우가 발생하였으며, 가끔씩 'ㅣ'의 끝부분인 해당 연구에서 취급하지 않은 획요소 '맺음'을 '삐침'이라고 잘못 추출한 경우가 발생하였다.

각 획요소 별로 검출된 결과의 이미지 중 획요소의 우선순위가 높은 '부리'와 '가지', '삐침'과 상대적으로 제대로 추출되지 않던 경우가 많았던 '꺾임'의 태스크별로 구분하여 대표적으로 Table 3-6에 표기하였으며, 13종의 글꼴 중 Serif와 Sans-serif 계열별로 가장 정형글꼴에 가까운 글꼴 하나씩만 선별하여 표기하였다. 학습한 글꼴 중 선별한 Serif 계열의 글꼴은 'KBIZ한마음명조L'이며, Sans-serif 계열의 글꼴은 'SpoqaHanSans'이다. 또한 학습하지 않은 글꼴에서 선별한 Serif 계열의 글꼴은 '신명조'이며, Sans-serif 계열의 글꼴은 '맑은 고딕'이다.

4.2 성능 비교 평가

한글 획요소 추출의 적용 글자의 확장을 위해 진행한 실험은 독립 변수는 '글꼴의 학습 여부'와 '글자의 학습 여부'로 선정하였으며 더하여 '글꼴의 굵기', '글꼴의 계열'로도 선정하여 성능 비교 평가를 진행하였다. 이 때, 조절변수는 '획요소'이며, 종속변수는 '글꼴 모양 자동 추출 시스템의 성공률'이다.

자동 추출의 적용 대상을 확장한 태스크대로 여덟 가지의 정의한 획요소 모두 자동 추출을 진행한 후, 설계한 식을 토대로 성공률을 측정하였다. 그 결과, 태스크별로 평균 성공률을 살펴보면 검증 모델의 태스크가 8.88점, 태스크 1이 8.36점, 태스크 2의 평균 성공률은 8.46점으로 대체적으로 높은 점수의 결과가 나왔으며 검증 모델의 태스크, Task 2, Task 1 순으로 성공률이 높았다. 획요소별로 평균 성공률을 살펴보면 부리는 9.37점, 가지는 9.66점, 꼭지점은 9.21점, 상투는 9.25점, 삐침은 9.59점, 꺾임은 6.14점, 돌출은 7.36점, 닫힘은 8.52점으로 가지의 성공률이 가장 높고, 꺾임의 성공률이 가장 낮았다.

태스크별 성공률과 여덟 가지의 획요소별 성공률이 통계적으로 유의미한 차이가 있는지 알아보고자 도출된 평균들의 차이 분석을 진행하였다. 분석 방법으로는 이원배치 분산분석(ANOVA)을 수행하였으

Table 3. Experimental result of the stroke element "Gaji"

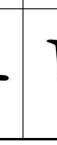
	Serif				Sans-serif			
Validation task								
task 1								
task 2								

Table 4. Experimental result of the stroke element "Buri"

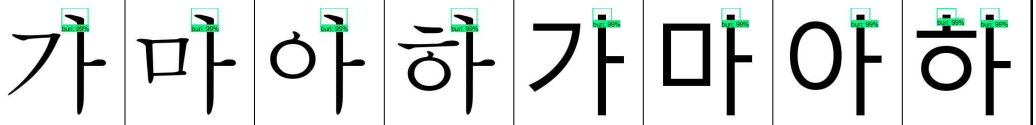
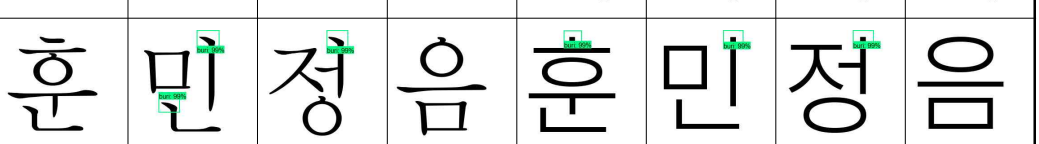
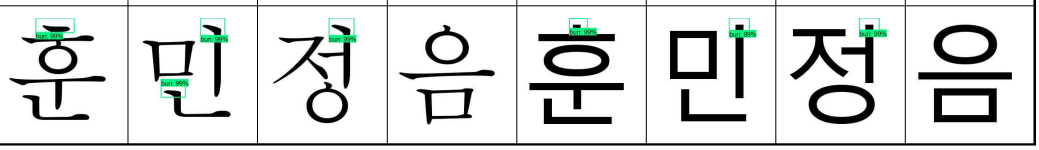
	Serif				Sans-serif			
Validation task								
task 1								
task 2								

Table 5. Experimental result of the stroke element "Bbichim"

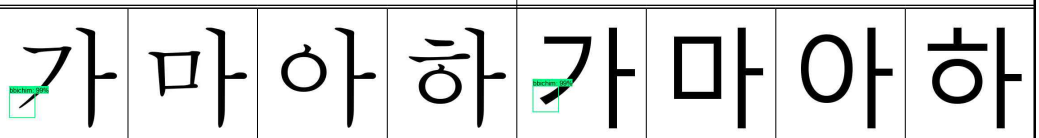
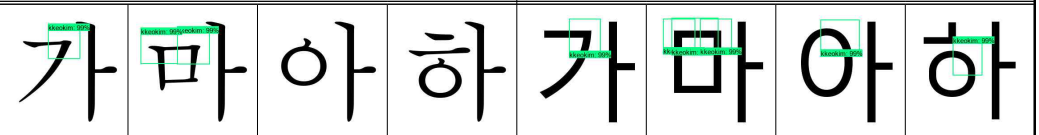
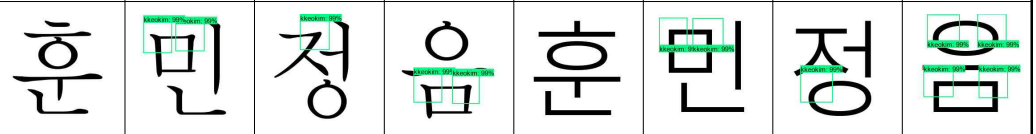
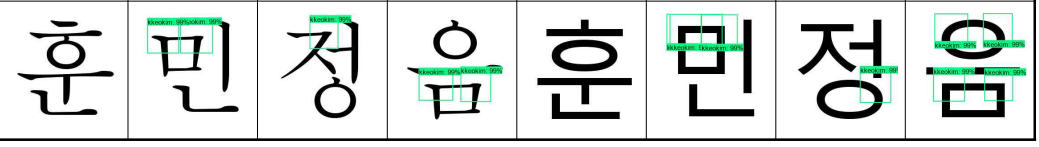
	Serif				Sans-serif			
Validation task								
task 1								
task 2								

Table 6. Experimental result of the stroke element "Kkeokim"

	Serif				Sans-serif			
Validation task								
task 1								
task 2								

며 유의수준은 5%로 설정하였다. 그 결과, 태스크별 F-통계량은 43.75이고, p-값은 0.049으로 유의수준 0.05보다 매우 근소하게 작은 값이 도출되었다. 따라서 '태스크별 평균은 모두 동일하다' 는 귀무가설이 기각되어 적용 대상을 글자로 확장하여 진행한 자동 추출과, 글꼴로 확장하여 진행한 자동 추출, 글자와 글꼴을 확장하여 진행한 자동 추출은 서로 차이가 있다고 할 수 있지만 그 값은 0.001으로 매우 근소하다고 할 수 있다.

또한 획요소별 F-통계량은 30.42이고, p-값은 2E-07으로 유의수준보다 매우 작으므로 여덟 가지의 획요소를 기반으로 자동 추출을 진행하여 도출되는 성공률의 평균의 차이는 통계적으로 매우 유의미하다고 할 수 있다. 태스크별 성공률과 획요소별 성공률의 그래프는 Fig. 6과 7에 나타나있다.

글꼴을 'Bold', 'Medium', 'Light'의 세 가지의 굵기로 구분한 후 정의한 획요소를 모두 자동 추출을 진행하여 성공률을 측정하였다. 그 결과, 굵기별로 평균 성공률을 살펴보면 Light가 8.77점, Medium이 8.66점, Bold의 평균 성공률이 8.48점으로 글꼴의 굵기가 얇을수록 평균 성공률이 높다는 결과가 도출되었다. 글꼴의 굵기 요인에 대한 세 집단의 평균 차이 분석을 위해 일원배치 분산 분석(ANOVA)을 수행하였다. 일원배치 분산분석의 귀무가설은 '글꼴 굵기별별 글꼴 획요소 자동 추출의 성공률은 모두 동일하다.' 이다. 분석 결과, F-통계량은 0.124이고 이에 대한 p-값은 0.884로 굵기별 성공률의 평균은 점수로 볼 때 얇을수록 더 성공률이 좋기는 하지만 이 차이는 통계적으로 무의미하여 결국 차이가 없다고 할 수 있다. (Fig. 8 참조)

Serif와 Sans-serif로 계열을 글꼴을 구분하여 한글 획요소 추출을 진행한 결과, Serif 계열의 평균 성공률은 8.67점, Sans-serif 계열의 평균 성공률은 8.48점으로 근소하게 Serif 계열의 평균 성공률이 높았다. 대응표본(쌍체 비교) T-검정 중 양측 검정을 수행한 결과, T-통계량은 2.365이고 유의확률인 p-값은 0.326으로 Serif와 Sans-Serif의 평균 차이는 없다는 결과가 도출되었다. (Fig. 9 참조)

4.3 결과 분석

한글 획요소 추출 적용 글자의 확장을 위한 실험을 진행한 결과, 각 태스크 모두 80%가 넘는 높은

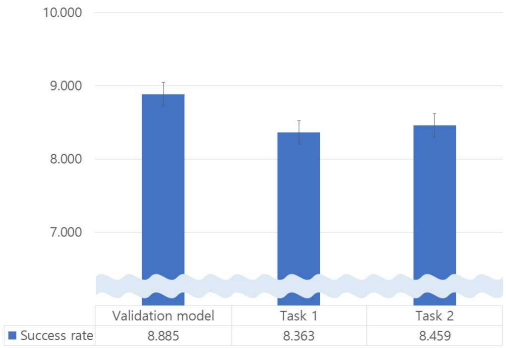


Fig. 6. Result chart of success rate by experiment task.

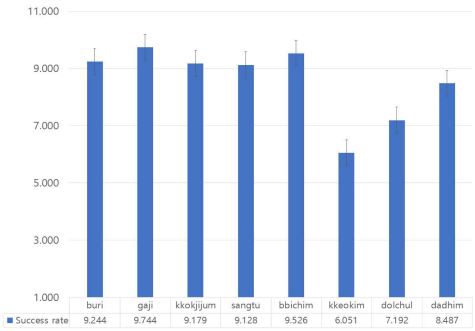


Fig. 7. Result chart of success rate by stroke elements.

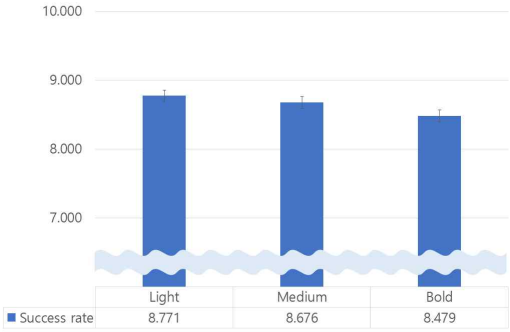


Fig. 8. Result chart of success rate by font thickness.

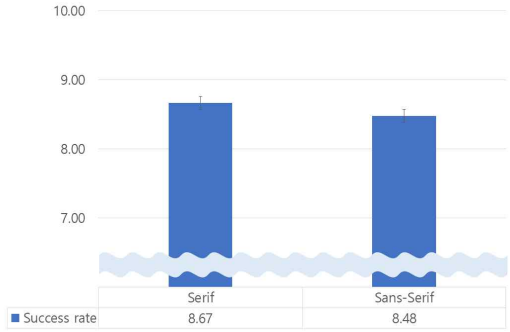


Fig. 9. Result chart of success rate by font family.

점수가 도출되었다. 실제 Object Detection 분야에서 발표된 Detector들의 성능(mAP@IoU=0.5)은 VOC07, VOC12, COCO 등 대표적 Benchmark에서 모두 70-80 수준으로 이에 견주어 봤을 때 전반적으로 성능이 높은 획요소 추출 모델이라 판단 가능하다.

통계적인 방법으로 분석해 본 결과, 학습한 글자로 진행한 검증 모델의 태스크가 학습하지 않은 글자로 진행한 Task 1과 Task 2보다 통계적으로 유의미하게 성공률이 높은 결과를 보이긴 하였지만, 그 차이는 매우 근소하였다. 따라서 글자 범위에서는 확장하지 않은 학습한 글자인 '가', '마', '아', '하' 에서 획요소 추출이 잘된다고 판단할 수는 있지만 확장한 글자 범위에서 획요소를 추출하였을 때에도 성능이 높다. 글꼴 범위를 대상으로 적용 대상을 확장하였을 때에는 학습하지 않은 글꼴에서 자동 추출을 진행한 Task 2이 학습한 글꼴인 Task 1보다 조금 더 높은 성공률을 보였다. 이는 학습하지 않은 글꼴은 한글 모양 표준 분류체계를 기반으로 선정하여 학습한 글꼴보다 좀 더 정형화된 모양인 글꼴이기 때문에 정형에 가까운 글꼴에서 추출 성공률이 좀 더 높다고 판단된다.

따라서 글꼴은 학습여부와 상관없이 정형글꼴계층의 모든 글꼴범위로 확장하여 성공률 높게 한글 획요소를 추출할 수 있어 처음 제안한 시스템의 목적인 유사도 판단에는 사용할 수 있는 소기의 목적을 달성하였다. 글자 범위를 확장했을 때에 학습한 글자에서보다 근소하게 성공률이 낮은 이유는 '꺾임', '돌출', '닫힘' 등의 낱글자 "ㅁ"에서 추출되는 획요소의 성공률이 낮았기 때문이라 판단된다. 이를 해결하기 위해서는 모델을 강화시킬 수 있는 추가적인 알고리즘을 도입해야 할 것이다.

글자와 글꼴 외에도 독립 변수로 선정하여 검증하였던 글꼴의 굵기와 계열은 모두 통계적으로 차이가 없어 굵기별, 계열별 차이에 따른 획요소 자동 추출의 성능의 차이는 없다고 할 수 있다. 하지만 점수로 보았을 때 굵기는 평균적으로 얇을수록 성공률이 높았고, 계열은 평균적으로 Serif 계열이 미세하게 더 높았다.

5. 결 론

본 논문에서는 선행 연구에서 제안한 Faster R-CNN 기반의 한글 획요소 추출 시스템을 이용하여 모든 글꼴과 글자에서 획요소를 추출할 수 있는지 확인해보고자 추출 적용 글자의 확장에 따른 실험 모델을 설계하고 그에 따른 평가 방법을 정의하여 성능을 비교하였다. 이때, 글자의 확장은 글꼴의 확장과 함께 비교하여 살펴보았으며, 이 외에도 추가로 글꼴의 계열이나 글꼴의 굵기에 따라서도 한글 글꼴 획요소를 추출하는 데 성능의 차이가 존재하는지 고려해보았다. 그 결과, 전반적으로 높은 점수가 도출되어 제안한 모델은 적용 대상을 확장하였을 때에도 성공률이 높게 획요소가 추출된다고 판단할 수 있었다. 특히 글꼴 범위를 확장하였을 때에는 학습 여부와 관계없이 성공률 높게 획요소가 모두 자동으로 추출되었다. 따라서 선행 연구의 본래 목적이었던 한글 글꼴 유사도 판단을 목적으로 한다면 다양한 글꼴을 필요로 할 뿐 다른 글자에서 획요소를 추출할 필요는 없으므로 해당 시스템을 이용하여 해결될 것이다. 하지만 본 연구에서 확인해보고자 했던 글자 범위를 확장하였을 때에는 학습하지 않은 글자보다 학습한 글자의 추출 성공률이 매우 근소하게 높았다. 한글은 조합형 글자로 초성과 종성 등의 글자의 위치에 따라 자소의 모양이 달라서 발생되는 문제로 추측되어 학습 데이터를 더 다양하게 수집한다면 해결될 것이라 판단한다. 글꼴의 계열과 굵기는 추출 성능에 대해서 영향을 주지 않았다.

따라서 향후 통계적으로도 확장 여부에 따라 성공률의 차이가 없는 추출 모델을 구현하기 위해추출된 데이터를 바탕으로 판단하여 새로운 글자의 획요소를 능동적으로 찾아주어 데이터 수를 늘려가 추출 모델을 강화시키는 반지도 능동 학습(Semi-Supervised Learning)을 도입하여 획요소 추출 학습을 진행하고자 한다. 이를 통해 11,172자의 글자에 대한 획요소 데이터를 점차적으로 늘려감으로써 모든 자소 단위의 글자를 지원할 수 있어 획요소 추출의 대한 성공률이 완벽하게 높은 모델이 될 것이라 판단한다.

REFERENCE

[1] E. Shin and J. Song, "Visual Characteristics of Korean Kinetic Typography According to Tone and Voice on Youtube," *Journal of the Korean Society Design Culture*, Vol. 26, No. 2, pp. 267-278, 2020.

[2] KISDI, *In the Era of the 4th Industrial Revolution, Media Content Survival Strategy,*

KISDI Premium Report 17-14, 2017.

[3] G. Chen, et al., "Large-scale Visual Font Recognition," *Proceeding of IEEE Conference on Computer Vision and Pattern Recognition*, pp. 3598-3605, 2014.

[4] Z. Wang, et al., "Deepfont: Identify your Font from an Image," *Proceeding of 23rd ACM International Conference Multimedia,* pp. 451-459, 2015.

[5] T. Chen, Z. Wang, N. Xu, H. Jin, and J. Luo, "Large-scale Tag-based Font Retrieval With Generative Feature Learning," *Proceeding of IEEE/CVF International Conference on Computer Vision,* pp. 9115-9124, 2019.

[6] S. Jiang, Z. Wang, A. Hertzmann, H. Jin, and Y. Fu, "Visual Font Pairing," *Proceeding of IEEE Transactions on Multimedia,* pp. 2086-2097, 2020.

[7] Korea National Hangul Museum, *Survey and Prospect of Font Industry in 2015*, Korea National Hangul Museum Research Report, 2015.

[8] J. Jeon and S. Lim, "Research on Hangul Stroke Element Classification and Automatic Extraction Using SVM and Sliding Window Algorithm," *Proceeding of the Fall Conference of the Korea Multimedia Society*, pp. 732, 2019.

[9] J. Jeon and S. Lim, "Machine Learning Based Automatic Extraction of Stroke Elements in Hangul Font," *Proceeding of the 8th Korea-Japan Joint Workshop on Complex Communication Sciences,* pp. 53-55, 2020.

[10] J. Jeon, D. Park, S. Lim, Y. Ji, and S. Lim, "Automatic Extraction of Hangul Stroke Element Using Faster R-CNN for Font Similarity," *Journal of Korea Multimedia Society*, Vol. 23, No. 8, pp. 953-964, 2020.

[11] B. Chakraborty, O. Rudovic, and J. Gonzalez, "View-invariant Human-body Detection with Extension to Human Action Recognition Using Component-wise HMM of Body Parts," *Proceeding of 8th IEEE International Conference on Automatic Face and Gesture Recognition*, pp. 1-6, 2008.

[12] S.D. Meena and L. Agilandeeswari, "An Efficient Framework for Animal Breeds Classification Using Semi-supervised Learning and Multi-part Convolutional Neural Network (MP-CNN)," *Proceeding of IEEE Access*, Vol. 7, No. 1, pp. 151783-151802, 2019.

[13] B. Heiselet, T. Serre, M. Pontil, and T. Poggio, "Component-based Face Detection," *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pp. 657-662, 2001.

[14] S. Lim, "Development Trend of Font Technology," *Typeface 1999 of Korean Typeface Development Research Center*, pp. 95-134, 1999.

[15] S. Lim, H. Kim, H. Chung, K. Park, and K. Choi, "Hangul Font Editor Based on Multiple Master Glyph Algorithm," *Korea Institute of Information Scientists and Engineers Transactions on Computing Practices,* Vol. 21, No. 11, pp. 699-705, 2015.

[16] J. Jeon, Y. Ji, D. Park, and S. Lim, "Evaluation of Criteria for Mapping Characters Using an Automated Hangul Font Generation System Based on Deep Learning," *Journal of Korea Multimedia Society*, Vol. 23, No. 7, pp. 850-861, 2020.

[17] D. Park, J. Jeon, Y. Ji, and S. Lim, "Design of Structure Map for Modeling Hierarchical Structure Information of Hangul," *Proceeding of the Fall Conference of the Korea Multimedia Society,* pp. 578-579, 2019.

[18] TTA, *Hangul Font Production Guides For Font Registration System Part 2. Design Based Classification*, e-Publishing Project Group 608, TTAK.KO-10.0906-part2, Telecommunication Technology Association, 2016.

[19] H. Kim and S. Lim, "Standardization Study of Font Shape Classification for Hangul Font

Registration System," *Journal of Korea Multimedia Society*, Vol. 20, No. 3, pp. 571-580, 2017.

[20] H. Kim and S. Lim, "Shape Property Study of Hangul Font for Font Classification," *Journal of Korea Multimedia Society*, Vol. 20, No. 9, pp. 1584-1595, 2017.

[21] D.L. Olson, D. Delen, *Advanced Data Mining Techniques*, Springer Publishers, Berlin, 2008.

전 자 연

2019년 숙명여자대학교 ICT융합공학부 IT공학전공(학사)
2019년~현재 숙명여자대학교 IT공학과 학석사연계과정
2020년~현재 소셜감성 창업

관심분야 : 기계학습, 딥러닝, 데이터분석, 멀티미디어(폰트, 키네틱 타이포그래피), UI/UX 등

임 순 범

1982년 서울대학교 계산통계학과(학사)
1983년 한국과학기술원 전산학과(석사)
1992년 한국과학기술원 전산학과(박사)

1989년~1992년 (주)휴먼컴퓨터 창업 (연구소장)
1992년~1997년 (주)삼보컴퓨터 프린터개발부 부장
1997년~2001년 건국대학교 컴퓨터과학과 교수
2001년~현재 숙명여자대학교 IT공학과 교수
2015년~현재 숙명여자대학교 ICT융합연구소 연구소장
2006년 University of Colorado 방문교수
2014년~한국멀티미디어학회 회장, 명예회장
2012년~한국전자출판학회 회장, 명예회장
관심분야 : 컴퓨터 그래픽스, 웹/모바일 멀티미디어 응용, 디지털 방송, 전자출판(폰트, 전자책, XML 문서), User Interface