

# 점자 인식 딥러닝 모델을 이용한 점자 문서 번역 애플리케이션 개발\*

김세연<sup>01</sup>, 김나연<sup>2</sup>, 양유진<sup>2</sup>, 조은호<sup>3</sup>, 지은경<sup>3</sup>

<sup>1</sup>고려대학교, <sup>2</sup>동국대학교, <sup>3</sup>한국과학기술원 전산학부

kimse2878@korea.ac.kr, nyn2265@mme.dongguk.edu, s2019112541@mme.dongguk.edu,  
ehcho@kaist.ac.kr, ekjee@se.kaist.ac.kr

## Development of Braille Document Translation Application Using Braille Detection Deep Learning Model

Se-Yeon Kim<sup>01</sup>, Na-Yeon Kim<sup>2</sup>, Yu-jin Yang<sup>2</sup>, Eunho Cho<sup>3</sup>, Eunkyong Jee<sup>3</sup>

<sup>1</sup>Korea University, <sup>2</sup>Dongguk University, <sup>3</sup>School of Computing, KAIST

### 요 약

정안인 교육자 혹은 시각 장애인 학부모는 시각 장애인 학생이나 자녀를 지도하기 위하여 점자를 직접 학습하고 번역하는 데 많은 시간을 소모한다. 그러나, 빠른 시간 내에 학습하기 어려운 점자의 특성으로 인해, 시각장애인 학생들은 정안인 학생들과 비슷한 수준의 지도를 받는 것을 기대하기 어렵다. 영어-점자 번역의 경우 다양한 양방향 번역 오픈소스 도구가 존재하여 사용자가 사용할 수 있는 다양한 번역 선택지가 있는 것에 비해, 한글은 완성된 오픈소스가 없다. 이 문제점에 기반하여 우리는 점자 인식 딥러닝 모델과 자체 개발 점자-한국어 번역 알고리즘을 활용한 점자 문서 번역 애플리케이션 ‘눈길’을 개발하였다. ‘눈길’은 시각장애인 학생들의 교육자와 학부모들이 오역을 확인하고, 점자를 알지 못하여도 시각장애인 학생들과 소통할 수 있도록 설계되었으며, 모바일 접근성 지침을 준수하여 모든 사용자가 불편함 없이 사용할 수 있도록 배리어 프리(Barrier-Free)를 실현하였다. 사용자 실험 결과, ‘눈길’이 시각장애인의 학습에 긍정적인 영향을 미친다는 것을 확인할 수 있었다.

### 1. 서론

점자 문자는 1824년에 발명되어, 시각장애인을 위한 글쓰기와 독서의 주된 방법으로서 사용되어 왔다. 기술의 발달로 인해 점자 이외의 정보 교환 수단이 발명되었지만, 그럼에도 불구하고 점자는 시각장애인들의 의사소통에 주된 매체가 된다. 점은 종이 위에 양각으로 새겨지며, 촉각으로 인식된다. 흰 종이 위에 별 다른 무늬 없이 새겨 지기 때문에, 점각을 눈으로 읽어내는 것은 쉽지 않다.

정안인 교육자와 전맹인 학생이 일반 학급에서 통합 교육을 하는 경우에도 점자가 사용되는데 정안인인 교육자가 점자로 쓰여진 학생들의 과제물, 교재를 다루어야 하는 경우가 있다. 이때, 교육자가 점자를 해석하지 못하는 경우가 대부분이기에 교육 프로그램이 원활하지 않은 경우가 빈번한 상황이다.

시각장애인들의 불편을 개선하기 위하여, 광학 점자 인식 분야는 1980년대부터 지속적으로 연구되고 있다. Zhang 외 연구 [1]가 역치 값을 활용하여 점을 추출 한 연구로 시작되어, 영상처리 기법 위주로 연구가 진행되었다. 최근에는, Li 외

연구 [2,3]와 Ovodov 외의 연구[4]와 같이 기계학습을 활용하여 SVM, 인공 신경망 등을 활용한 연구가 활발히 진행되고 있다.

점자 인식 기술과 함께 점자 번역 도구도 활발히 개발되고 있다. 현재 영미권에서의 점자 번역은 오픈소스 Liblouis[5]를 기반으로 딥러닝을 도입한 번역 웹 서비스 ‘abcBraille’[6]과 같은 사이트가 존재한다. 그러나 한글 번역 서비스는 웹과 애플리케이션 모두 출시되어 있지 않은 상황이다. 즉, 한국의 시각장애인들과 정안인을 위한 접근성이 좋은 번역 서비스의 부재라는 문제점이 존재한다.

본 연구에서는 모바일 기기로 촬영한 한국어-점자 시스템 기반의 점자 문서에서 점자를 한국어로 번역해내는 애플리케이션, ‘눈길’을 제작하였다. ‘눈길’은 영어-점자 번역의 점자 인식 기술을 기반으로 점자 문서의 이미지에서 점자를 인식하고, 국립국어원의 한글 점자 규정[7]에 따라 자체 제작한 한국어-점자 번역 알고리즘을 통해 번역하여 결과물을 도출한다. 모바일 애플리케이션의 장점인 휴대성과 이동성을 바탕으로 사용자는 해당 애플리케이션을 통해 번역하고자 하는 점자 문서를 촬영하여 즉각적으로 결과를 얻을 수 있다.

\* 본 연구는 과학기술정보통신부 및 정보통신기획평가원의 대학ICT연구센터지원사업의 연구결과로 수행되었음 (IITP-2022-2020-0-01795).

본 논문의 구성은 다음과 같다. 2장에서는 ‘눈길’의 설계와 알고리즘을 소개한다. 3장에서는 ‘눈길’의 사용자 실험과 그 영향을 평가한다. 4장에서는 결론을 제시한다.

## 2. ‘눈길’의 설계

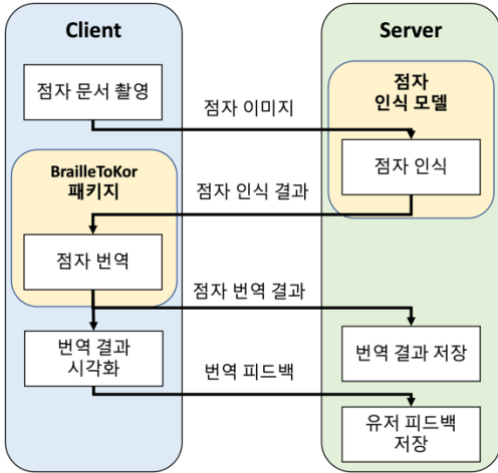


그림 1. 점자 번역 앱 눈길 아키텍처

### 2.1 애플리케이션 설계

점자를 한글로 번역하는 전체적인 흐름은 그림 1과 같다. 사용자가 모바일 기기에서 촬영한 점자 이미지를 서버로 전송한다. 서버에서는 사용자가 촬영한 이미지를 수신하여 점자 인식을 수행한다. 클라이언트로 전송된 점자 탐지 결과 데이터는 ‘BrailleToKor’ 번역 패키지<sup>1</sup>를 사용하여 번역된다. 번역 결과와 문서 이미지는 서버 내부 데이터베이스에 저장한다. 사용자는 번역 결과에 피드백을 요청할 수 있으며, 번역 결과 및 이미지는 모델 데이터셋으로 활용한다.

### 2.2 점자 인식 모델

‘눈길’의 점자 인식 모델은 컨볼루션(Convolution) 연산 기반의 물체 탐지 모델을 활용한다. 대부분의 점자 인식 모델은 1) 점자탐지, 2) 그리드 복구, 3) 점자 번역 세 가지 단계를 거쳐 완성된다. 하지만, 본 연구팀이 채택한 점자 인식 모델의 경우 One stage 물체 탐지 컨볼루션 신경망 모델을 사용하며, 점자 위치 인식과 라벨 예측의 병렬 처리로 프로세싱 속도를 단축하였다.

### 2.3 한국어-점자 번역 알고리즘

영어-점자 번역’ 알고리즘은 가장 대표적인 오픈소스 Liblouis[5]가 있다. 오픈소스 특성상 해당 프로젝트에 기여하는 사람들의 기호에 따라 개발이 달라지는데, 해당 오픈소스는 영미권 위주로 개발되어 왔기 때문에 한국어 지원이 미흡하다. 이러한 문제점을 토대로 새롭게 한국어 알고리즘을 개발했고, 이를 스위프트 패키지 매니저 (Swift Package Manager, SPM)로 패키징했다. SPM<sup>2</sup>은 국립국어원의 한글점자규정해설서[7]를 참고하여 약어, 약자, 문장부호, 특수문자, 숫자, 예외 처리 등을 모두 구현하였다.

개발한 번역 알고리즘의 흐름은 다음과 같다. 사용자가 입력한 점자 문장이 입력값으로 들어오면, 문장을 띄어쓰기 단위로 끊어준다. 나누어진 단어는 각각 초성, 중성, 종성 단위로 나누어져 순차적으로 한글로 번역한 후 최종적으로 음절로 결합된다. 결론적으로 완성된 단어들을 조합하여 완전한 한글 문장을 반환한다. 점자를 한글로 번역하면서 유의해야 할 점은 모음 ‘케’와 쌍시옷 받침이 같은 점형을 사용하기 때문에 순차적으로 번역 시 두 문자로 인해 오류가 나타날 수 있다는 점이다. 이와 유사한 상황에서는 단어 전체의 구조를 파악하여 두 가지 이상의 번역 결과 중 실제 사용 빈도를 고려하여 가장 적절한 번역 결과를 반환한다.

### 2.4 번역 결과 시각화

모바일 앱 접근성 지침[8]을 참고하여 번역 결과를 시각화하는 배리어프리 애플리케이션을 개발하였다. 기본적으로 애플리케이션 사용에 있어서 모호성을 최대한 제거하고 추가적으로 애플리케이션 사용법을 추가하였으며, 컴포넌트 영역을 명확히 구분하고 각 버튼 및 라벨의 크기를 9mm 이상으로 유지하였다. 최소 명암비를 4.5:1로 조정하여 뚜렷한 대비를 유지해 색 인식에 문제가 없도록 하였다. iOS 보이스오버 기능을 대비하여 모든 버튼, 라벨에 설명을 추가하여 사용자가 기능을 모두 파악하고 활용할 수 있게 했다. 또한 모바일 기기 사용자 지정 글씨 크기로 내용이 보일 수 있도록 하였으며, 글씨체 또한 동일하게 변경될 수 있도록 하여 혼란을 방지했다. 번역 시각화 결과는 그림2와 같다.



그림 2. 애플리케이션 점자 번역 결과 예시

### 2.5 유저 피드백 저장

점자 문서의 번역 결과를 받은 사용자는 1) 점자 인식, 2) 점자 번역 결과에 대한 피드백을 선택하여 작성할 수 있다. 유저 피드백과 번역에 사용한 점자 문서 이미지는 서버 내부 데이터베이스에 저장하여 ‘눈길’의 점자 인식 모델과 번역 알고리즘 개선 시 데이터 셋으로 활용한다.

## 3. 사용자 실험

### 3.1 점자 수기 해석 대비 애플리케이션 효율성 검증

‘눈길’의 속도를 측정하고, 효율성을 검증하기 위하여 점자 수기 해석과 ‘눈길’의 번역 시간을 비교하였다. A4 용지 1/8,

<sup>1</sup> <https://github.com/Bridge-NOONGIL/BrailleToKor>.

<sup>2</sup> <https://github.com/Bridge-NOONGIL/KorToBraille>.

1/4, 1/2, 1의 크기를 가진 tiny, small, medium, large 점자 문서 데이터를 각 3가지씩 준비하였다. 점자 번역 규정을 보고 번역해야 하는 비숙련 인원 3명과 점자 번역 규정을 보지 않고 번역이 가능한 숙련 인원 4명이 직접 번역하여 문서 번역에 소요되는 시간을 측정하였다.

표 1. 문서 번역 소요 시간

	비숙련 인원 (s)	숙련 인원 (s)	눈길 점자 번역 애플리케이션 (s)
Tiny	426.22	60.33	10.74
Small	956.67	143.42	11.13
Medium	1621.89	285.50	12.27
Large	2996.33	547.33	18.26

표 1은 측정된 번역 소요 평균 시간을 표로 나타내 정리한 것이다. 점자 번역에 소요되는 시간은 애플리케이션 기준으로 했을 때, 비숙련 인원의 경우, ‘눈길’에 비해 40~160배 가량의 시간이 소요되었으며, 숙련 인원도 5.6~30배 가량의 시간이 소요되었다. 비숙련 인원과 숙련 인원이 점자 문서의 점자 량에 비례하여 번역 시간이 증가한 것에 비하여 애플리케이션은 점자 량이 8배 가량 증가했음에도, 소요 시간이 약 2배 밖에 증가하지 않았다. 이를 통해, 애플리케이션의 사용이 직접 점역보다 시간을 획기적으로 단축시킬 수 있음을 확인하였다. 특히 한 장의 점자 문서에 존재하는 점자의 개수가 많을수록 애플리케이션을 사용하는 것이 유리하다는 것을 확인할 수 있다.

또한 인간이 직접 점역하였을 때에는 점자 사이의 간격을 육안으로 선명하게 구분하기에 어려움이 있어 이에 따른 점역 오류가 나타남을 확인하였다. 반면, 점자 인식 모델을 탑재한 해당 애플리케이션을 이용하였을 때에는 해당 사유에 의한 판단의 오류가 없음을 확인하였다.

### 3.2 애플리케이션 정확도 측정

‘눈길’의 번역 정확도를 확인하기 위하여 다양한 점자 문서 데이터셋을 활용해 정확도를 측정하였다. 모바일 기기로 직접 촬영한 서로 다른 점자 문서 11장의 점자 4,538개를 사용하여 실험하였으며, 서버는 vCPU 8개와 메모리 16GB를 활용하였다.

표 2. 눈길 점자 번역 애플리케이션 정확도

맞은 점자 개수	전체 점자 개수	정확도 (%)
4,466	4,538	98.24

실험 결과, 총 4,538개의 점자 중 4,466개의 점자를 정확하게 탐지하였으며, 약 98.24%의 정확도가 측정되었다. 점자 탐지 모델의 오류의 대부분은 점자 36점을 점자 25점으로 판단하면서 나타남을 확인하였는데 이는 연속적으로 등장하는 점자들의 경우 종이의 곡면을 잘못 인식하는 경우가 있기 때문으로 판단된다. 연속적으로 등장하는 점자들은 점자 문서 내의 페이지 분류나 밑줄로 해석된다. 이는 의미를 갖지 않는 점자들이기에 해석에 오류를 가져오지 않는다. 또한 볼록점 아래에 위치한 그림자가 흐린 점자를 분명하게 탐지하지 못한 점은 발견하였으며 이를 보완해야 할 사항으로 확인하였

다.

### 4. 결론

본 연구에서는 점자 딥러닝 모델을 이용한 점자 문서 번역 애플리케이션을 개발하였다. 웹사이트에서만 사용 가능하던 딥러닝 모델을 애플리케이션에 적용하여 사용 편의를 향상시켰으며, 모바일 앱 접근성 지침을 준수하여 특수 사용자 계층을 고려한 UI/UX를 개발하였다. 또한 한글-점자 번역 알고리즘을 자체 제작하여, 기존에는 영어-점자만 번역 가능하던 한계점을 해결하였다.

본 연구에서 개발한 애플리케이션을 통해 한국어 점자를 사용하는 시각 장애인과 그들의 교육자 및 보호자간 소통이 빠르고 효과적으로 이루어질 수 있도록 도울 수 있을 것으로 기대한다.

향후 연구에서는 딥러닝 모델의 처리 속도를 개선할 수 있는 방안을 개발하고, 다양한 사용자가 여러 상황에서 유용하게 사용할 수 있도록 애플리케이션에 영어-점자 문서 번역 기능을 추가하고자 하며, 현재 iOS 기반으로만 개발된 애플리케이션을 안드로이드 기반으로까지 확장하고자 한다.

### 참고문헌

[1] Zhang, Shanjun, and Kazuyoshi Yoshino. (2007). *A braille recognition system by the mobile phone with embedded camera*. [Second International Conference on Innovative Computing, Informatio and Control]. IEEE Xplore.

[2] Li, R., Liu, H., Wang, X., & Qian, Y. (2018). *DSBI: double-sided Braille image dataset and algorithm evaluation for Braille dots detection*. [Proceedings of the 2018 the 2nd International Conference on Video and Image Processing].

[3] Li, R., Liu, H., Wang, X., Xu, J., & Qian, Y. (2020). *Optical Braille Recognition Based on Semantic Segmentation Network With Auxiliary Learning Strategy*. [Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops].

[4] Ilya G. Ovodov. (2021). *Optical Braille Recognition Using Object Detection CNN*. [Proceedings of the IEEE/CVF International Conference on Computer Vision, Workshops, pp. 1741-1748].

[5] Liblouis. (2022). Liblouis - An open-source braille translator and back-translator. <http://liblouis.org>.

[6] *ABC Braille*. (2021). <https://abcbraille.com>.

[7] 국립국어원. (2017). 한글점자규정해설서. [https://www.korean.go.kr/front/etcData/etcDataView.do?m\\_n\\_id=46&etc\\_seq=603&pageIndex=1](https://www.korean.go.kr/front/etcData/etcDataView.do?m_n_id=46&etc_seq=603&pageIndex=1).

[8] 방송통신표준심의회. (2016). 모바일 애플리케이션 접근성 지침 2.0. [https://www.rra.go.kr/ko/reference/kcsList\\_view.do?nb\\_seq=1930&nb\\_type=6](https://www.rra.go.kr/ko/reference/kcsList_view.do?nb_seq=1930&nb_type=6).