

실시간 수어 AI 번역 프로그램 구현

이 동 욱¹ · 김 민 서¹ · 김 남 호² · 최 광 미^{2*}

¹호남대학교 소프트웨어학과 학사과정

²호남대학교 컴퓨터공학과 교수

^{2*}호남대학교 컴퓨터공학과 교수

Implementation of Real-Time Sign Language AI Translation Program

Dong-Uk Lee¹ · Min-Seo Kim¹ · Nam-Ho Kim² · Gwang-Mi Choi^{2*}

¹Bachelor's Course, Department of Software, Honam University, Gwangju 62399, Korea

²Professor, Department of Computer Engineering, Honam University, Gwangju 62399, Korea

^{2*}Professor, Department of Computer Engineering, Honam University, Gwangju 62399, Korea

[요 약]

세계 여러 나라에서 수어 언어를 중요시하며, 일부는 수어를 공식 언어로 지정하여 사회 참여 기회를 높이고 있지만 그럼에도 불구하고 음성 언어와 수어 간 의사소통 장벽은 여전히 존재한다. 본 논문에서는 이러한 문제들을 해결하기 위해 빠른 속도와 상대적으로 높은 정확도를 자랑하는 YOLO 알고리즘 중 YOLOv7 딥러닝 알고리즘을 사용하여 한국 수어를 실시간으로 한국어로 번역하여 수어와 음성 언어의 의사소통 격차를 해소하는 것을 목표로 하여 실시간 수어 번역 AI 프로그램을 구현하였다. 총 35개의 데이터 카테고리를 사용하였으며 학습 결과 mAP@0.5:0.95 값은 0.818로 나타났다. 테스트 데이터 셋을 사용한 인식률은 80%이상으로 높은 정확도를 보였다. 구현된 실시간 수어 번역 AI프로그램을 활용하여 수어와 음성 언어 간의 의사소통 격차를 해소하는데 기여할 것으로 기대된다.

[Abstract]

In many countries worldwide, sign language is highly valued, with some even designating it as an official language to enhance social inclusion opportunities. However, despite these efforts, communication barriers between spoken language and sign language persist. In this paper, we aim to address these issues by implementing a real-time sign language translation AI program using the YOLOv7 deep learning algorithm, known for its speed and relatively high accuracy. Our goal is to translate Korean Sign Language into Korean in real-time, thus bridging the communication gap between sign language and spoken language. We utilize a total of 35 data categories, and the training results yield an mAP@0.5:0.95 value of 0.818. The recognition rate using the test dataset showed an accuracy of over 80%. It is expected that the implemented real-time sign language translation AI program will contribute to reducing the communication gap between sign language and spoken language.

색인어 : 딥러닝, 객체 탐지, 수어, YOLO, YOLOv7

Keyword : Deep Learning, Object Detection, Sign Language, YOLO, YOLOv7

<http://dx.doi.org/10.9728/dcs.2023.24.10.2585>



This is an Open Access article distributed under the terms of the Creative Commons Attribution Non-Commercial License (<http://creativecommons.org/licenses/by-nc/3.0/>) which permits unrestricted non-commercial use, distribution, and reproduction in any medium, provided the original work is properly cited.

Received 31 August 2023; **Revised** 19 September 2023

Accepted 22 September 2023

***Corresponding Author: Gwang-Mi Choi**

Tel: +82-62-940-3665

E-mail: cgmi66@honam.ac.kr

I. 서론

청각장애는 소리나 음성을 인식하지 못하거나 제한적으로 인식하는 상태를 나타낸다. 이러한 상태는 선천적인 원인, 후천적인 원인, 노화 등 다양한 요인으로 인해 발생할 수 있으며, 청각장애로 인해 음성 언어를 이해하거나 발화하는 데 어려움을 겪는 청각장애인들은 시각적 정보를 기반으로 의사소통을 해야하는 독특한 환경에서 살아가고 있다. 이들은 언어적인 의미와 감정을 몸짓, 표정, 손의 움직임 등 시각적인 수단을 활용하여 전달하며, 이러한 의사소통 방식을 통칭하여 수어라고 한다. 수어는 언어의 본질적인 특성을 가지며, 다양한 문화와 사회적 의미를 지니고 있다. WED (World Federation of Deaf)에 따르면 현재 71개 국가에서 그 나라의 수어가 공용어 지위를 가지고 있으며, 우리나라도 2016년 한국수화언어법 제정을 통해 한국수어가 우리나라 공용어의 지위를 가지게 되었다[1]. 이러한 노력들은 청각장애인들이 의사소통과 사회 참여를 더욱 원활하게 할 수 있는 환경을 조성하며, 그들의 의사 표현과 사회적 참여를 지원하는 역할을 한다. 그러나 이러한 노력에도 불구하고, 청각장애인들은 여전히 수어와 음성 언어 간의 의사소통 장벽을 극복해야 하는 어려움을 겪고 있다. 수어는 음성 언어와는 완전히 다른 표현 방식을 가지고 있어 농인들이 비장애인들과 의사소통하는 과정에서 불편함을 겪는 경우가 많다. 특히 의료 분야에서는 의사와의 원활한 대화와 소통으로 환자의 증상을 정확히 전달하고 의료진의 지시를 올바르게 이해하는 데 필수적이다. 그렇지 않으면 오해나 오진으로 이어질 수 있다. 그래서 한국에서는 이러한 어려움을 해결하기 위해 의료 수어 통역사가 활동하고 있지만 청각장애인을 위해 병원 내에 수어 통역사를 배치하는 곳은 전국에 4개 병원에 불과한 현실적인 어려움이 있다. 이는 청각장애인이 진료나 수술 등의 의료서비스를 받기 위해 의료기관을 이용할 때 수어 통역 지원을 받기가 힘들다. 또한 개인 수어 통역사를 고용할 수 있지만 비용적인 부분이 문제가 되는 현실적인 어려움이 있다. 이러한 문제를 해결하고 일상적인 소통 상황에서도 청각장애인들이 의사를 전달하기 위해 실시간 수어 번역 AI 프로그램을 구현하는 것을 본 논문에서 제안한다.

II. 관련 연구

2-1 MediaPipe

Mediapipe는 Google의 오픈소스 라이브러리로, 실시간 객체 감지와 추적을 지원한다. 그림 1과 같이 실시간 영상에서 얼굴, 손, 진신과 같은 다양한 객체를 감지할 수 있다[2]. 현재 수어 인식을 위한 객체 인식 인공지능(AI) 모델들은 주로 Mediapipe를 기반으로 개발되어 있으며 이와 관련된 다양한 논문들이 존재한다. 그러나 CPU 추론 및 실시간 다중

객체 인식 프레임워크를 고려할 때, YOLO 알고리즘을 선택하는 것이 더 적절하다고 판단하여 YOLO를 수어 인식에 활용하기로 결정하였다.

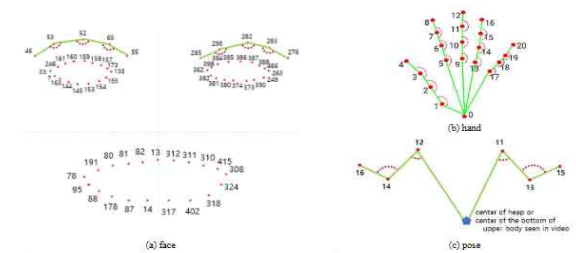


그림 1. 미디어파이프에서 손, 눈 등을 인식하는 방법
 Fig. 1. How to recognize hands, eyes, etc. in the media pipe

2-2 YOLO

YOLO (You Only Look Once) 알고리즘은 객체 감지 및 객체 분류를 위한 딥러닝 알고리즘으로, 입력된 이미지를 한번만 처리하므로 처리 속도가 빠르다는 장점이 있다. 하지만 이는 2-stage-detection 방식을 사용하는 네트워크에 비해 정확성에서 일부 차이가 있을 수 있지만 빠른 객체 감지가 필요한 경우에는 YOLO가 적합하다[3]. 또한 YOLO는 후보 영역을 추출하기 위한 별도의 네트워크를 적용하지 않기 때문에 Region-based Convolutional Neural Network (R-CNN)과 같은 다른 알고리즘보다 처리시간 측면에서 우수한 성능을 보인다[4]. YOLO는 그림 2와 같이 입력된 이미지나 영상을 19*19개의 그리드로 분할하고 각각의 그리드에 해당하는 물체의 위치와 범위 그리고 종류를 출력하는 네트워크다[5]. CNN은 이미지 인식 및 분류에 사용되는 일반적인 신경망 클래스인 반면, YOLO 모델은 CNN을 사용하여 이미지나 비디오와 같은 2D 데이터에서 공간적인 특징을 추출하고, 이러한 특징을 활용하여 객체 탐지, 분류, 위치 예측 등 다양한 작업을 수행하는 단일 인공신경망을 활용하여 학습하기에 다른 기법에 비해 인식속도가 매우 빠르다[6]. 이 과정에서 이미지나 비디오는 2D 데이터로 취급되며, 컬러 이미지의 경우 RGB 채널 정보가 함께 활용된다. 이러한 이유로 인식 속도가 빠른 YOLO 알고리즘을 선택하였다.

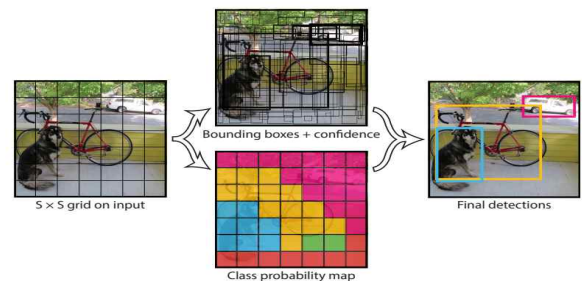


그림 2. YOLO 물체 검출 방법
 Fig. 2. YOLO object detection method

2-3 YOLOv7

본 논문에서 사용한 모델은 대표적인 객체탐지 알고리즘인 YOLO 시리즈 중 대만 연구소에서 개발한 YOLOv7을 적용하였다[7]. YOLOv7은 2022년 7월에 출시된 실시간 객체탐지 모델이다[8]. 그림 4[7]와 같이 5~160 fps 범위의 추론 속도를 가지면 현재까지 알려진 실시간 물체 검출 모델 중 가장 높은 정확도인 56.8% AP를 가지며 실시간 객체 탐지에 높은 효율을 보이는 것으로 발표되었다[9]. YOLOv7은 모델의 복잡도에 따라 그림2와 같이 YOLOv7-tiny, YOLOv7, YOLOv7x, YOLOv7w6, YOLOv7e6 등 다양한 변형이 있다[10]. 모델의 무게가 증가할수록 인식 성능은 향상되지만 데이터 분석 시간이 증가하며 컴퓨터 성능에도 민감해진다. 그러므로 본 연구에서는 학습용 PC의 컴퓨터 성능을 고려하여 YOLOv7-x 모델을 선택하여 개발하였다.

Model	Test Size	AP _{test}	AP _{50^{test}}	AP _{75^{test}}	batch 1 fps	batch 32 average time
YOLOv7	640	51.4%	69.7%	55.9%	161 fps	2.8 ms
YOLOv7-X	640	53.1%	71.2%	57.8%	114 fps	4.3 ms
YOLOv7-W6	1280	54.9%	72.6%	60.1%	84 fps	7.6 ms
YOLOv7-E6	1280	56.0%	73.5%	61.2%	56 fps	12.3 ms
YOLOv7-D6	1280	56.6%	74.0%	61.8%	44 fps	15.0 ms
YOLOv7-E6E	1280	56.8%	74.4%	62.1%	36 fps	18.7 ms

그림 3. YOLOv7 모델 종류
Fig. 3. Types of YOLOv7 models

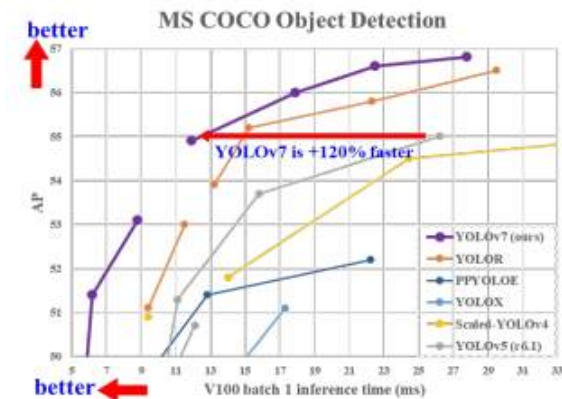


그림 4. 다른 모델과 비교한 YOLOv7 객체 탐지 결과
Fig. 4. Comparison of YOLOv7 object detection results with other models

III. AI 모델 구현

3-1 프로그램 설계

본 논문에서 구현하고자 하는 실시간 수어 번역 AI 프로그램의 프로세스는 그림 5와 같다. 먼저 수어 데이터를 수집하고, 이후 해당 데이터를 분류하고 라벨링하는 과정을 거치게

된다. 이후 AI 모델을 데이터로 학습시키고 학습 결과를 얻는다. 학습된 모델은 실시간 수어 번역 프로그램의 구현에 활용된다.

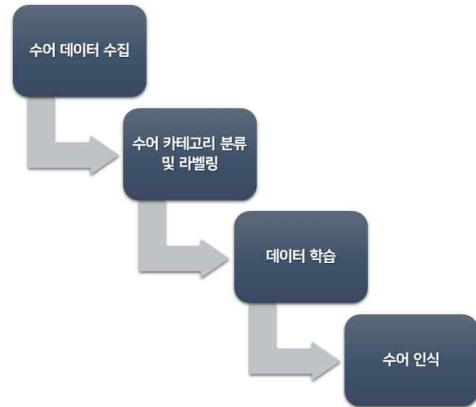


그림 5. 실시간 수어 번역 AI 프로그램 설계
Fig. 5. Design of real-time sign language translation AI program

3-2 데이터 수집 및 라벨링

AI 모델에 적용한 데이터는 자음 14개의, 모음 10개 및 7개의 복합 모음을 사용하였으며, 일반 단어로는 "나는", "너", "최고", "생각하다"라는 4개의 단어를 사용하여 총 35개의 데이터 카테고리에 해당하는 총 10,545장의 이미지 데이터를 활용하였다. 이 이미지 데이터는 labeling을 활용하여 수어에 해당하는 손 위치를 전처리하였으며, 이 중 7,700장을 학습 데이터로 사용하였다. labeling은 데이터 라벨링 작업을 수행하는 Python 기반의 도구이며 구체적으로 이미지가 무엇인지 입력하며 딥러닝 작업을 위해 필요한 과정 중 하나이다. 데이터 라벨링을 위해 먼저 각 클래스에 맞게 데이터 이름들을 수정한 후 라벨링을 진행하였다. 그림 6은 데이터 라벨링을 진행하는 과정을 보여주고 있다. YOLOv7은 txt 형식의 라벨링 데이터를 사용하는데, 이 파일에는 이미지에서 검출된 객체의 클래스 및 경계 상자 정보가 저장된다. 이 내용이 그림 7에 해당한다.

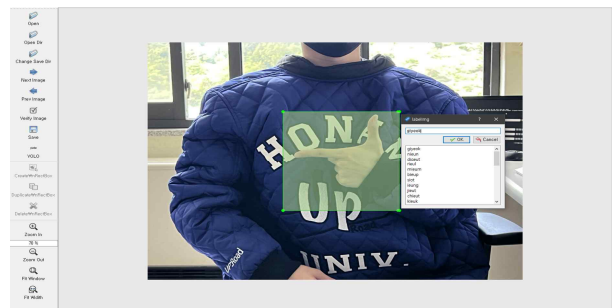


그림 6. Labelimg를 사용한 데이터 라벨링 과정
Fig. 6. The data labeling process using Labelimg

5 0.343750 0.517500 0.320000 0.525000
 14 0.286250 0.505833 0.335000 0.701667
 24 0.481250 0.417500 0.285000 0.605000

클래스 번호 중심 X좌표 중심 Y좌표 너비 높이

그림 7. 라벨링 데이터
 Fig. 7. Labeling data

3-3 데이터 학습

본 논문에서 제안하는 수어 AI 모델 학습은 표 1에 제시된 컴퓨터 사양을 기반으로 진행되었다. 이 컴퓨터 사양을 고려하여 학습 파라미터를 조정하였으며, 조정한 값은 표 2에 나열하였다. 총 7,700장의 학습 데이터를 YOLOv7-X 모델을 사용하여 100 Epochs, 200 Epochs 학습을 진행하였다. 그림 8과 그림 9는 Anaconda 환경에서 100 Epochs, 200 Epochs의 학습이 완료된 화면을 보여준다.

표 1. 학습용 PC 스펙 시트

Table 1. PC specification for training

PC specification for training	
OS	Windows 10
CPU	Intel(R) Core(TM) i7-10700
RAM	32GB
Graphic card	NVIDIA GeForce GTX 1660 SUPER

표 2. 학습 파라미터

Table 2. Training parameters

Training Parameters	
Class	35
Image Data	7,700
Batch-size	2
Epochs	100 / 200
Img-size	512
Model	YOLOv7 -X

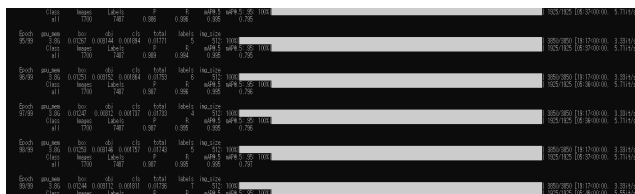


그림 8. 데이터 100번 학습
 Fig. 8. Training with 100 instances of data



그림 9. 데이터 200번 학습
 Fig. 9. Training with 200 instances of data

3-4 데이터 학습 결과

표 3과 표 4에 나오는 mAP@0.5는 IOU(Intersection Over Union) 임계치를 0.5 이상으로 설정한 것으로 Pascal VOC 방식의 정확도 표현 방식이다. 반면에 mAP@0.5:0.95는 COCO 데이터 셋에 사용되는 정확도 방식이며 IOU 임계치를 0.5부터 0.95까지 0.05씩 증가시킨 단계에서의 결과를 종합하여 평균을 나타내는 방식이며 모델의 다양한 성능과 객체 탐지 모델의 더 포괄적인 정확도를 제공하여 성능을 종합적으로 평가한다[11]. mAP@0.5:0.95는 mAP@0.5보다 더 엄격한 기준을 가지므로 결과가 mAP@0.5보다 높게 나올 수는 없다. 이러한 평가 방식에서는 값이 1에 가까울수록 모델의 성능이 높다고 평가된다. 총 35개의 클래스 중 자음 4개, 모음 4개, 단어 2개를 대표로 삼아 성능을 분석하였다. 이러한 클래스들을 기준으로 100Epochs와 200Epochs에서의 mAP@0.5과 mAP@0.5:0.95의 값을 살펴보면, 100Epochs와 200Epochs에서의 mAP@0.5 값은 큰 차이가 없는 것으로 보이지만, mAP@0.5:0.95에서는 100Epochs 대비 200Epochs 때 더 높은 성능을 보이고 있음을 확인할 수 있다. 이로써 모델의 학습이 잘 이루어진 것으로 판단된다. 특히

표 3. 학습 100번 mAP@0.5와 mAP@0.5:0.95 성능지표

Table 3. Learning 100 times mAP@0.5 and mAP@0.5:0.95 performance indicators

Class	mAP@0.5	mAP@0.5:0.95
giyeok (ㄱ)	0.995	0.827
nieun (ㄴ)	0.997	0.848
digeut (ㄷ)	0.995	0.843
rieul (ㄹ)	0.997	0.847
a (ㅏ)	0.996	0.869
ya (ㅑ)	0.997	0.839
eo (ㅓ)	0.996	0.845
yeo (ㅕ)	0.995	0.827
naneun (나는)	0.989	0.78
think (생각하다)	0.996	0.81

표 4. 학습 200번 mAP@0.5와 mAP@0.5:0.95 성능지표

Table 4. Learning 200 times mAP@0.5 and mAP@0.5:0.95 performance indicators

Class	mAP@0.5	mAP@0.5:0.95
giyeok (ㄱ)	0.996	0.862
nieun (ㄴ)	0.997	0.869
digeut (ㄷ)	0.996	0.855
rieul (ㄹ)	0.997	0.866
a (ㅏ)	0.997	0.886
ya (ㅑ)	0.997	0.865
eo (ㅓ)	0.996	0.852
yeo (ㅕ)	0.998	0.864
naneun (나는)	0.989	0.791
think (생각하다)	0.996	0.83

mAP@.5:0.95에서 200Epochs 결괏값이 과적합을 보이지 않고 더 우수하게 나왔다. 전체적인 mAP@0.5:0.95의 그래프는 그림 10과 같다. 또한 테스트 데이터 셋을 사용한 인식률은 평균 80% 이상으로 높은 정확도를 보였고 그림 11은 표 3 과 표 4에 설명한 35개 클래스 중 자음 4개, 모음 4개, 단어 2개를 포함한 테스트 데이터 셋의 결과를 보여준다.

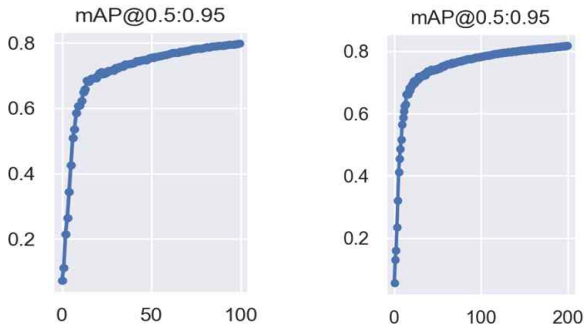


그림 10. (좌)100Epochs (우)200Epochs의 mAP@0.5:0.95 그래프

Fig. 10. (Left) mAP@0.5:0.95 graph for 100 epochs. (Right) mAP@0.5:0.95 graph for 200 epochs.

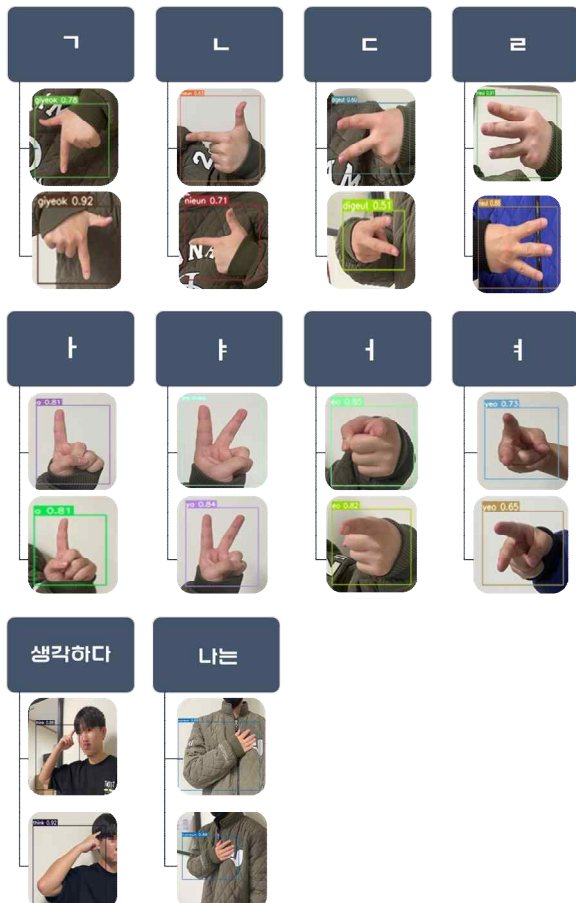


그림 11. 대표 클래스의 테스트 데이터 셋 결과
Fig. 11. Results of the representative class's test dataset

IV. 수어 번역 프로그램 구현

4-1 수어 번역 프로그램 실행

PyQt5는 Qt5 애플리케이션 프레임워크에 대한 파이썬 버전이다. 이 PyQt5를 사용하여 수어를 인식하고 그 결과를 텍스트로 변환해주는 Graphical User Interface 프로그램 개발을 진행하였다. 개발 환경은 PyCharm과 Anaconda를 사용하였다. 그림 12와 같이 PyQt Designer를 사용하여 GUI를 간단히 디자인하고 내부에 세부 기능을 개발하는 과정을 진행하였다. 이 프로그램이 실시간으로 수어를 인식하기 위해서는 YOLOv7의 detect.py 파일이 필요하다. 그림 13과 같이 detect.py 스크립트의 argument를 프로그램에 맞게 수정하여 소스코드를 변경하였으며, 표 4에 해당하는 200Epochs 학습한 최적 가중치 파일을 적용하였다.



그림 12. 수어 번역 프로그램

Fig. 12. Sign language translation program

```
if __name__ == '__main__':
    parser = argparse.ArgumentParser()
    parser.add_argument('--weights', nargs='+', type=str, default='best.pt', help='model.pt path(s)')
    parser.add_argument('--source', type=str, default='0', help='source') # 0: webcam, 1: 이미지 실용적인 질의 자동 논문
    parser.add_argument('--img-size', type=int, default=640, help='inference size (pixels)')
    parser.add_argument('--conf-thres', type=float, default=0.25, help='object confidence threshold')
    parser.add_argument('--iou-thres', type=float, default=0.45, help='IOU threshold for NMS')
    parser.add_argument('--device', default='', help='cuda device, i.e. 0 or 0,1,2,3 or cpu')
    parser.add_argument('--view-img', action='store_true', help='display results')
    parser.add_argument('--save-txt', action='store_true', help='save results to txt')
    parser.add_argument('--save-conf', action='store_true', help='save confidences in --save-txt labels')
    parser.add_argument('--nosave', action='store_true', help='do not save images/videos')
    parser.add_argument('--classes', nargs='+', type=int, help='filter by class: --class 0, or --class 0 2 3')
    parser.add_argument('--agnostic-nms', action='store_true', help='class-agnostic NMS')
    parser.add_argument('--augment', action='store_true', help='augmented inference')
    parser.add_argument('--update', action='store_true', help='update all models')
    parser.add_argument('--project', default='runs/detect', help='save results to project/name')
    parser.add_argument('--name', default='exp', help='save results to project/name')
    parser.add_argument('--exist-ok', action='store_true', help='existing project/name ok, do not increment')
    parser.add_argument('--no-trace', action='store_true', help='don't trace model')
```

그림 13. detect.py argument

Fig. 13. detect.py argument

개발된 수어 번역 프로그램의 Python 파일을 실행하게 되면 그림 10과 같은 GUI 프로그램이 실행되며, '수어 번역 시작' 버튼을 누르게 되면 그림 14와 같이 YOLO가 실행되며, 그 후에는 그림 15와 같은 학습된 모델이 적용된 웹캠이 실행된다.

```
YOLO: 2023-04 torch 2.0.0cpu (CPU)
Using layers...
Detect fuse
/Users/linu/anaconda3/lib/python3.10/site-packages/torch/functional.py:504: UserWarning: torch.meshgrid: in an upcoming
release, it will be required to pass the indexing argument. (Triggered internally at /Users/linu/anaconda3/lib/python3.10/
return _VF.meshgrid(tensors, **kwargs) # type: ignore[attr-defined]
Model Summary: 365 layers, 10384504 parameters, 0 gradients, 188.6 GFLOPS
Convert model to TracedModel...
traced_script_module saved
model is traced!
```

그림 14. 학습모델을 적용한 웹 카메라 실행

Fig. 14. Running a web camera with a learning model applied

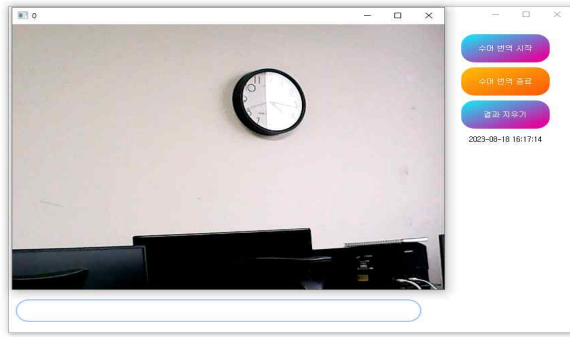


그림 15. 수어 번역 프로그램 실행
 Fig. 15. Run sign language translation program

4-2 수어 번역 프로그램 실행 결과

프로그램을 실행한 결과, 표 3과 표 4에 기재된 대표적인 수어 클래스를 정확하게 인식하였다. 이러한 인식결과는 표 5의 이미지들과 일치한다. 프로그램은 자음 'ㄱ', 'ㄴ', 'ㄷ', 'ㄹ'과 모음 'ㅏ', 'ㅑ', 'ㅓ', 'ㅕ'를 정확하게 인식하며, "나(는)", "생각(하다)"라는 단어들을 평균적으로 0.7 이상의 정확도로 인식하였다. 이러한 결과물은 텍스트 창과 Anaconda 환경에서도 제대로 출력되었다. 연속적인 동작도 정확하게 인식하여, 텍스트 창에도 "나(는)", "생각(하다)"라는 내용이 순차적으로 출력되었다. 또한 프로그램은 자음과 모음을 조합하여 단어를 생성하는 능력도 갖추고 있다. 표 5 그림에 'ㅅ', 'ㅈ', 'ㅇ', 'ㅣ', 'ㄷ', 'ㅏ'라는 자음과 모음을 인식하여 "사이다"라는 단어를 생성하였고, 이와 같은 방식으로 단어와 자음 모음을 인식하여 "나는 커피"라는 문장도 생성할 수 있었다. 특히 "사이다"라는 단어는 자음과 모음을 한번에 인식하여 단어를 생성하였다. 이러한 방식으로 프로그램은 미리 학습된 클래스에 없는 문장들도 자음과 모음을 조합하여 문장을 활용할 수 있음을 보여주고 있다.

V. 결 론

본 연구에서는 실시간 수어 인식을 위해 YOLOv7의 YOLOv7-X 모델을 활용하였다. 우선 수어 데이터에 라벨링을 진행하였고, 100Epochs와 200Epochs의 두 가지 시나리오로 모델을 학습시켰다. 결과적으로, 200Epochs 학습 시에 가장 우수한 0.86 mAP를 기록하면서도 과적합을 피할 수 있었다. 우수한 모델인 200Epochs의 최적 가중치 파일(best.pt)를 확보한 후 이를 활용하여 실시간 수어 번역 프로그램에 적용하였다. 실시간으로 전체 클래스 수어를 인식하였을 때 평균 0.7의 정확도로 수어를 인식하였다. 농인 분들이 일상생활에 어려움을 겪고 있는 여러 상황들을 실시간 수어 번역 프로그램으로 수어와 음성 언어 간의 소통 장벽을 획기적으로 극복하는 데 도움을 줄 것으로 기대한다. 불편한 상황

표 5. 실시간 수어 AI 번역 프로그램 실행 결과
 Table 5. Real-time sign language AI translation program execution result

을 해소하고 원활한 의사소통 과정을 제공함으로써, 농인분들에게 현실적이고 유용한 지원을 제공할 것으로 기대된다. 더욱이, 수어 통역사를 고용하는 데 따르는 비용적 부담도 해소할 수 있어서 많은 분들에게 도움을 전달할 수 있을 것이다. 추후 정확도를 높이기 위해 추가적인 수어 데이터 셋을 수집하는 작업을 진행할 계획이다. 더불어 수어 단어의 다양성을 확보하기 위해 다양한 수어 단어도 추가할 예정이다.

감사의 글

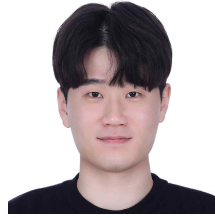
“본 연구는 과학기술정보통신부 및 정보통신기획평가원의 지역지능화혁신인재양성사업의 연구결과로 수행되었음.” (IITP-2023-RS-2022-00156287)

참고문헌

[1] I. Lee, “Sign Language Translation Technology Status and

- Improving Accessibility for the Deaf through Case Studies,” *Communications of the Korean Institute of Information Scientists and Engineers*, Vol. 40, No. 10, 14, October 2022.
- [2] S.-Y. Kim, S.-J. Urm, S.-Y. Yoo, S.-J. Kim, and K.-M. Lee, 2023, Application of Sign Language Gesture Recognition Using Mediapipe and LSTM, *Journal of Digital Contents Society*, Vol. 24, pp. 111-119. <https://doi.org/10.9728/dcs.2023.24.1.111>
- [3] I. Song and T. Park, “Automatic Emergency Detection Based on CNN for Pedestrians in CCTV Video,” *Journal of Digital Contents Society*, Vol. 23, No. 3, pp. 371-379, March 2022. <https://doi.org/10.9728/dcs.2022.23.3.371>
- [4] Y.-H. Lee and Y.-S. Kim. “A Comparative Experimental Study of CNN and YOLO for Object Detection,” *Journal of Semiconductor Display Technology*, Vol. 19, No. 1, pp. 85-92, 2020.
- [5] S.-H. Lee and N. I. Cho, “Pedestrian Detection using YOLO and Tracking” in *Proceedings of the Annual Conference of the Korean Society for Broadcasting and Media Engineering*, pp. 79-91, June 2018.
- [6] T. Park and H. Yi, “An Analysis of Adaptive Architectural 3D Printing with a Vision-Enabled Robot Arm - Real-time Monitoring Using YOLOv7 Algorithm Object Detection,” *Journal of the Architectural Institute of Korea Conference Proceedings*, Vol. 43, No. 1, p. 38, 2023.
- [7] C.-Y. Wang, A. Bochkovskiy, and H.-Y. M. Liao, “YOLOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors,” arXiv preprint arXiv: 2207.02696, 2022. <https://doi.org/10.48550/arXiv.2207.02696>
- [8] G. Park, J. G. Kang, S. Choi, Y. Youn, G. Kim, and Y.-W. Lee, “Detection of Active Fire Objects from Drone Images Using YOLOv7x Model,” *Journal of the Korean Society of Remote Sensing*, Vol. 38, No. 6, pp. 1737-1748, December 2022.
- [9] C.-S. Park, “Analysis of Inference Time Complexity of YOLOv7 Model in Various Computing Environments,” *Journal of Semiconductor & Display Technology*, Vol. 21, No. 3, pp. 7-11, 2022.
- [10] G. Park, Y. Youn, J. G. Kang, G. Kim, S. Choi, S. W. Jang, ... and Y.-W. Lee, “A Comparative Study on the Object Detection of Deposited Marine Debris (DMD) Using YOLOv5 and YOLOv7 Models,” *Journal of the Korean Society of Remote Sensing*, Vol. 38, No. 6, p. 1643, December 2022.
- [11] J. H. Lee, I. S. Kim, H. B. Kim, S. W. Lee, and S. K. Jung, “Case Study of YOLOv5 Model Optimization for Edge AI,” in *Proceedings of Symposium of the Korean Institute*

of Communications and Information Sciences, pp. 897-898, June 2022.



이동욱(Dong-Uk Lee)

2019년~현재: 호남대학교 소프트웨어학과 학사과정
 ※관심분야: 백엔드, 웹 프로그래밍, 모바일 프로그래밍



김민서(Min-Seo Kim)

2019년~현재: 호남대학교 소프트웨어학과 학사과정
 ※관심분야: 백엔드, 인공지능



김남호(Nam-Ho Kim)

1997년: 포항공과대학교 정보통신학과
 공학석사
 2013년: 전남대학교 전산통계 이학박사

1991년~1997년: 포스코 ICT(주) 연구원
 1998년~현재: 호남대학교 컴퓨터공학과 부교수
 ※관심분야: 사물인터넷, 인공지능, 응용 SW



최광미(Gwang-Mi Choi)

1990년: 조선대학교 전자계산학과
 이학학사
 1995년: 조선대학교 전자통계학과
 이학석사
 2003년: 조선대학교 전자통계학과
 이학박사

2018년~2021년: 조선대학교 SW중심대학사업단 초빙교수
 2021년~현재: 호남대학교 컴퓨터공학과 조교수
 ※관심분야: 스마트팩토리 지능화 시스템, AI 딥러닝,
 빅데이터 고도화 처리