International Journal of Internet, Broadcasting and Communication Vol.16 No.3 212-220 (2024)
http://dx.doi.org/10.7236/IJIBC.2024.16.3.212

IJIBC 24-3-22

Real time instruction classification system

Sang-Hoon Lee [1] , Dong-Jin Kwon [2]

[1] *Master, Korea Institute of Science and Technology, Korea*
[1] *lsh950223@kist.re.kr*

[2] *Associate Professor, Department of Computer Electronics Engineering, Seoil University, Korea*
[2] *djkwon77@seoil.ac.kr*

Abstract

A recently the advancement of society, AI technology has made significant strides, especially in the fields of computer vision and voice recognition. This study introduces a system that leverages these technologies to recognize users through a camera and relay commands within a vehicle based on voice commands. The system uses the YOLO (You Only Look Once) machine learning algorithm, widely used for object and entity recognition, to identify specific users. For voice command recognition, a machine learning model based on spectrogram voice analysis is employed to identify specific commands. This design aims to enhance security and convenience by preventing unauthorized access to vehicles and IoT devices by anyone other than registered users. We converts camera input data into YOLO system inputs to determine if it is a person, Additionally, it collects voice data through a microphone embedded in the device or computer, converting it into time-domain spectrogram data to be used as input for the voice recognition machine learning system. The input camera image data and voice data undergo inference tasks through pre-trained models, enabling the recognition of simple commands within a limited space based on the inference results. This study demonstrates the feasibility of constructing a device management system within a confined space that enhances security and user convenience through a simple real-time system model. Finally our work aims to provide practical solutions in various application fields, such as smart homes and autonomous vehicles.

Keywords: *Computer Vision, Speech Recognition, Machine Learning, Detection*

1. INTRODUCTION

Recently the advancement of machine learning technology is increasingly enhancing its applicability and importance in modern society. As a result, machine learning is being utilized in various fields, particularly in computer vision technology, where systems are being developed to automatically extract better results from image processing tasks such as object detection, tracking, image restoration, and image compression using algorithms.

Manuscript Received: June. 12, 2024 / Revised: June. 20, 2024 / Accepted: June. 26, 2024
Corresponding Author: djkwon77@seoil.ac
Tel: :+82-2-490-7349, Fax: +82-2-490-7802
Associate Professor , Department of Computer Electronics Engineering, Seoil University, Korea

In this study, we present a program that integrates a device management system within a confined space by building a User Interface (UI) through the Qt framework and combining it with a YOLO system that interfaces with cameras and microphones. Voice recognition technology, which enables computers to understand and process human speech, converts voice data collected from devices into digital signal data for interpretation. Modern AI technology utilize statistical machine learning and artificial neural network techniques to train models using large-scale voice datasets, continuously improving the accuracy of voice recognition systems. This technology is applied in various devices and applications, including smartphones, IoT devices, and vehicle navigation systems.

2. BACKGROUND KNOWLEDGE

2.1 Speech Recognition

Voice recognition is a technology that enables computers to understand and process human speech, and it is currently applied in various fields such as voice commands and control, voice search, automatic translation, and voice-based systems. Voice recognition systems convert voice signals into digital signals and then undergo preprocessing to extract features. the extracted features are used as data representing the verbal content of the voice signal and serve as input data for the model. recently, deep learning and artificial neural network technologies have been utilized to improve the accuracy of voice recognition. Neural network architectures such as Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM), and Convolutional Neural Networks (CNN) are primarily used, and these technologies maximize performance by training models on large-scale voice datasets [1-2]. additionally, state-of-the-art Natural Language Processing (NLP) models like BERT and GPT, which are transformer-based models, have contributed to enhancing the performance of voice recognition. these models can better understand the context and meaning of voice signals, enabling more sophisticated voice recognition. voice recognition technology is applied to various devices and applications, including smartphones, IoT devices, vehicle navigation systems, and smart speakers, and its applicability is expanding in numerous fields.

2.2 YOLO

YOLO is a neural network algorithm for real-time object detection, introduced by Joseph Redmon in 2015. Unlike traditional object detection methods, YOLO processes an image through a single forward pass to simultaneously predict bounding boxes and class probabilities for objects. this approach provides both fast processing speed and high accuracy. It is particularly used in combination with the OpenCV and Darknet frameworks to identify and classify objects in images and videos. YOLO's single network architecture divides the input image into a grid, with each grid cell predicting the probability of an object's presence. and, YOLO operates effectively regardless of the size of the input image or objects, enabling the detection of both small and large objects. By using a loss function that simultaneously optimizes the position and size of bounding boxes, as well as object class probabilities, YOLO enhances the accuracy of object detection. These features make YOLO applicable in various fields such as real-time video analysis, autonomous driving, and security monitoring systems. [3-5].

3. SYSTEM SUGGESTION

3.1 Setup Environment

The program developed in this study is based on the Windows OS and utilizes the Qt framework to provide a user-friendly and intuitive interface. Windows OS offers easy accessibility for a wide range of users, and Qt contributes to cross-platform development and improved usability. our program integrates essential libraries for image processing and running deep learning models. especially,

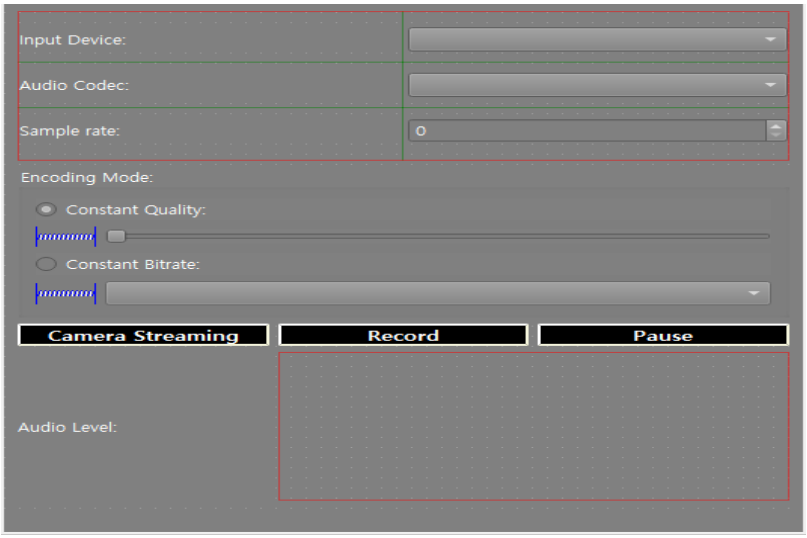


Figure 1. System UI

OpenCV provides powerful image processing capabilities to recognize users from image data captured via video or photographs and classify specific users. also, it collects voice data through a microphone to recognize commands. These results are displayed intuitively through Qt's UI, showing the camera viewer and recognized commands.

In particular, the program uses the SIMT (Single Instruction Multiple Threads) feature to execute multiple threads simultaneously for parallel processing with CUDA. This efficiently utilizes NVIDIA's GPU architecture, significantly reducing the inference time of deep learning models and providing enhanced performance. this accelerates deep learning models within the program. furthermore, our program is designed to be compatible with the deep learning framework TensorFlow. this provides users with the flexibility to easily integrate and execute various deep learning models.

3.2 Instruction and User classification

For voice recognition, the process of converting to digital signals involves the use of spectrogram transformation. As shown in Figure 2, a spectrogram is a graphical representation that visually depicts the changes in frequency over time. The frequency represents the pitch of the sound wave, while time denotes the elapsed time since the sound wave was generated. this method is widely used in the field of audio signal processing and voice recognition, as it helps in visualizing and analyzing the characteristics of voice signals by showing the variations in frequency and time. this visualization focus in applications such as word classification and music genre classification [6-7].

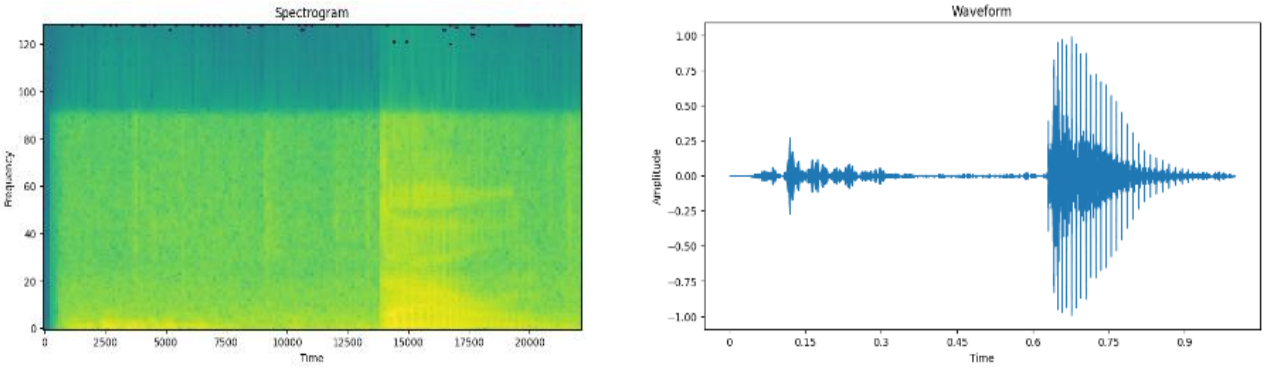


Figure 2. Speech recognition using spectrogram (Speech : Down)

A spectrogram intuitively displays the frequency components of the voice signal, energy distribution, and pattern changes in the frequency components, enabling users to understand and analyze voice data. This processed data is then used as input for machine learning systems, allowing the model to process, infer, and classify spoken commands by humans [6].

Figure 3 illustrates the process of classifying commands from voice data in this study. The spectrogram is generated using frequency analysis techniques such as FFT (Fast Fourier Transform).

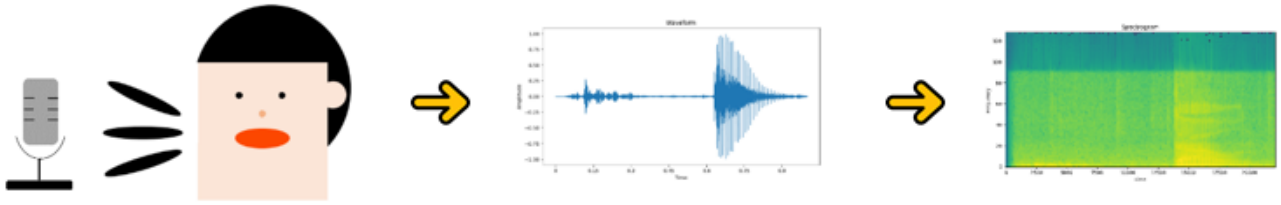


Figure 3. Instruction classification process

$$
\begin{aligned}
f(t) &= \lim_{T\to\infty}\frac{1}{T}\sum_{m=-\infty}^{\infty} F_m T e^{im\omega_0 t} \\
&= \lim_{T\to\infty}\frac{1}{2\pi}\sum_{m=-\infty}^{\infty} F_m T e^{im\omega_0 t}[(m+1)\omega_0 - m\omega_0] \\
&= \frac{1}{2\pi}\int_{-\infty}^{\infty} F(\omega) e^{i\omega t} d\omega
\end{aligned} \tag{1}
$$

In Equation (1), f(t) is the function of time, Fm is the Fourier coefficient, and ω0 is the fundamental angular frequency. This equation shows the Fourier transform relationship between the time domain and the frequency domain. Through FFT, the time-domain speech signal is transformed into the frequency domain and then arranged over time to generate spectrogram data; however, during this process of converting digital speech signal data, unnecessary noise may be introduced.

$$G(s) = \frac{a}{s+a} \tag{2}$$

In Equation (2) represents a low-pass filter, where G(S)is the transfer function of the filter, describing the output signal after the input signal's frequency components have passed through the filter. s is the complex

variable in the Laplace transform, determining the filter's frequency response. a is a constant that determines the filter's time constant or cutoff frequency; as this value increases, the cutoff frequency decreases, and the filter removes more high-frequency components. We need to use this to preprocess the initial voice signal data by applying a low-pass filter to eliminate noise data.

$$\bar{x}_k = \bar{x}_{k-1} + \frac{x_k - x_{k-n}}{n} \quad (3)$$

However, this process may result in unstable signal data conversion. To address this, the moving average filter (3).

In Equation (3) shows the formula for the moving average filter. xk represents the updated moving average at time k, xk-1 is the previous moving average, xk is the current data point, xk-n is the data point n steps back in time, and n is the number of data points used in the moving average calculation. We use a moving average filter to segment each voice interval with a specified window size, smoothing the signal by averaging the values within each segment. After this, the transformed spectrogram is subjected to classification tasks for commands using the EfficientNet B0 architecture model spectrogram

As shown in Table 1, We conducted experiments using the Speech Commands: A Dataset for Limited-Vocabulary Speech Recognition dataset, which has been previously researched in the field of speech recognition, and utilized the datasets provided by TensorFlow [8]. Particularly, when employing CNN models. to further improve our results, we conducted additional experiments using models with ResNet, DenseNet, and EfficientNet architectures. As a result, the EfficientNet model exhibits lower accuracy compared to the DenseNet model. but, it demonstrates a higher F1 Score. Based on these findings, we determined that the EfficientNet model is the most superior and thus applied it in our research [9-11].

Table 1. Instruction recognition model result

Parameter	CNN	Resnet	Densenet	**Efficientnet**
Learning rate	0.001	0.001	0.001	0.001
Epsilon	1e-07	1e-07	1e-07	1e-07
Loss function	Sparse Categorical Cross Entropy			
Max epochs	200	200	200	200
Batch size	32	32	32	32
Early stopping	Patience = 5, min_delta = 0.0001			
Optimizer function	Adam	Adam	Adam	Adam
Accuracy	84.86%	93.03%	93.75%	**93.51%**
Recall	0.8568	0.9372	0.9404	**0.9443**
Precision	0.9046	0.9351	0.9484	**0.9542**
F1 Score	0.8784	0.9360	0.9422	**0.9490**

The EfficientNet B0 model is based on Convolutional Neural Networks (CNNs). Unlike other CNN-based models, it optimizes the size and depth of each layer to increase accuracy while reducing the model's si ze. To achieve this, EfficientNet B0 employs Compound Scaling technology, adjusting the model's depth, wi dth, and resolution simultaneously to optimize performance. This approach optimizes all dimensions of the model to achieve higher accuracy with a smaller size compared to existing models. additionally, it reduces co mputational complexity, leading to superior performance in terms of model speed. The overall structure of th e EfficientNet B0 model used in our study is illustrated in Figure 4.

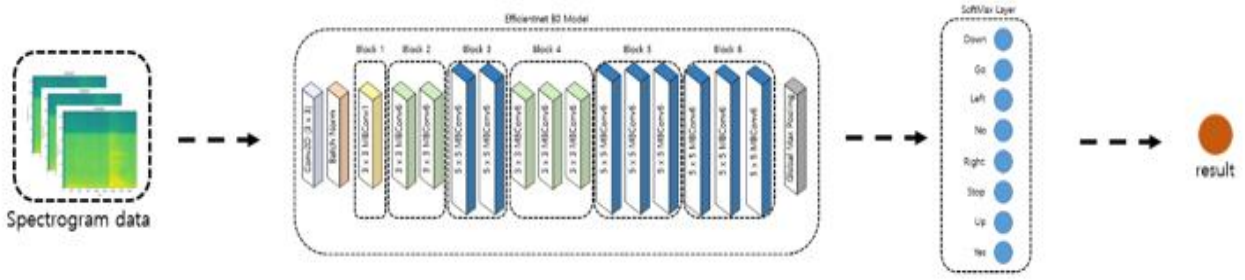
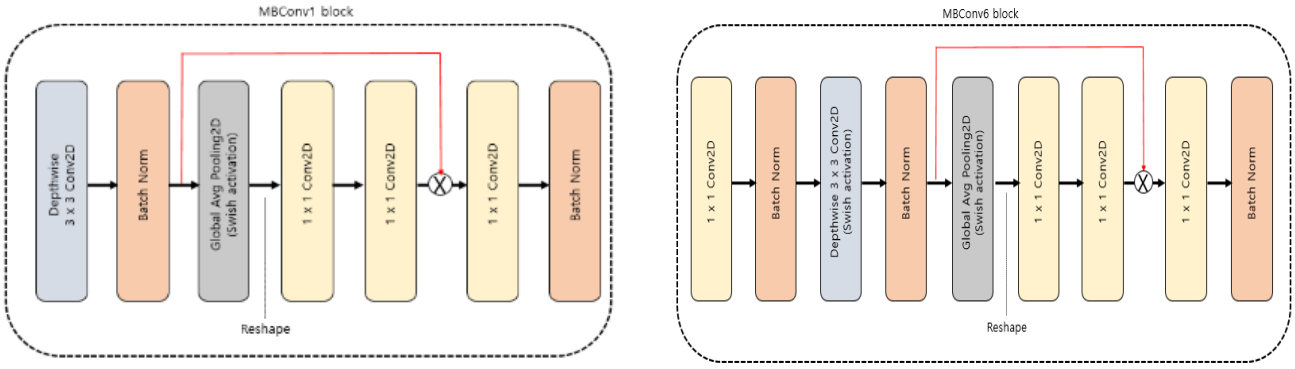


Figure 4. Speech recognition Efficientnet b0 model structure

Figure 5 depicts the structure of the MBConv1 and MBConv6 blocks used in EfficientNet. The MBConv1 block employs stride to reduce the size of input data and is used to increase the width and depth of the model in the initial layers.(a) The MBConv6 block, a core component of EfficientNet, is utilized in the middle or end of the model. This block consists of multiple convolutional, expansion, sampling, and reduction layers, increasing the model's depth and enabling it to learn more complex patterns for more accurate inference of input data(b) [12].

In equation (4), x represents the input value, while beta is an adjustable hyperparameter. The Swish function exhibits a similar form to ReLU when the input value is positive, gradually decreasing output values for negative inputs. it is one of the nonlinear activation functions used in artificial neural networks. It demonstrates better performance than the ReLU activation function.

$$f(x, beta) = x * sigmoid(beta * x) \tag{4}$$

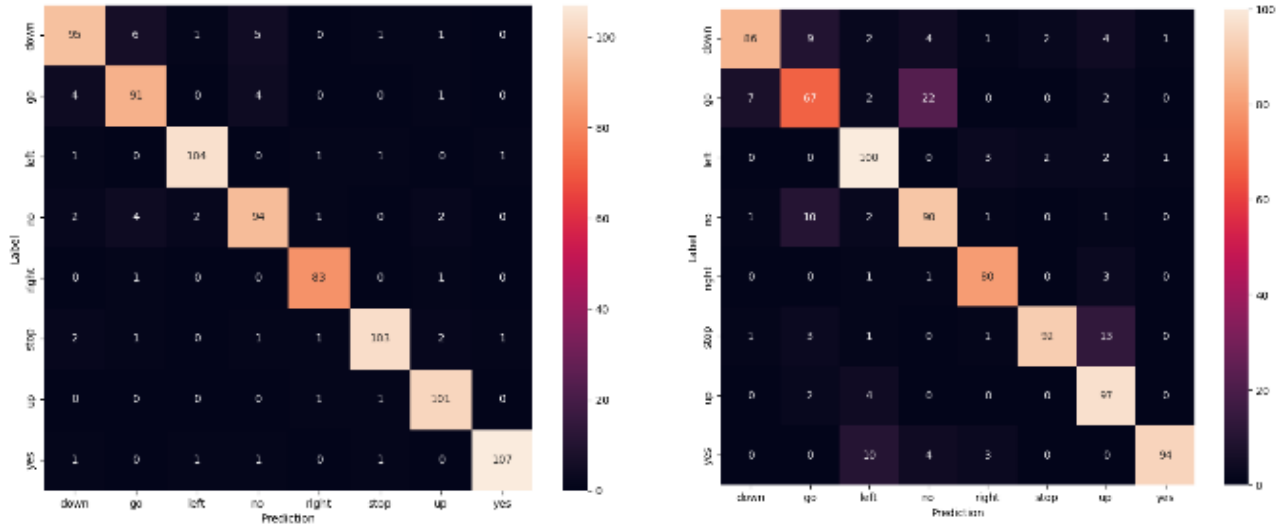


(a) **(b)**

Figure 5. (a) is MBConv1 Block process and (b) is MBConv6 block process

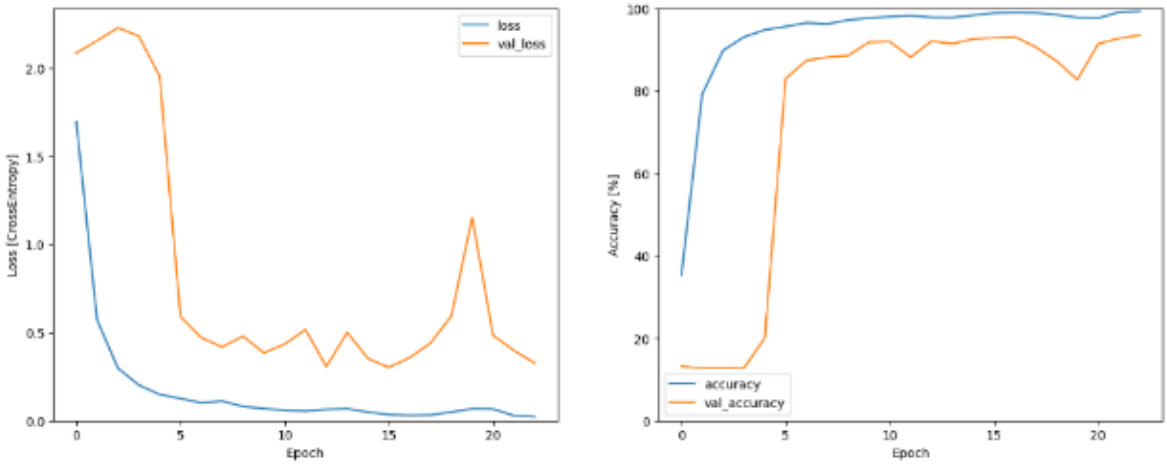
4. RESULT

Figure 6 presents the results of the command classification machine learning system used in this study, it was shown that the performance of the model intuitively through a confusion matrix. When comparing Efficientnet and CNN, there is a notable improvement in accuracy by 8.65%, along with a corresponding increase in F1 Score by 0.0706.(a) As this study is applied in real-time systems, the inference speed of the Efficientnet model demonstrates comparable performance to CNN, making it an overall superior model (b).



(a) **(b)**

Figure 6. (a) is Efficientnet and (b) is CNN result



(a) **(b)**

Figure 7. Instruction Efficientnet model result (a) Loss and (b) Accuracy

Figure 7 shows the loss (a) and accuracy (b) of the EfficientNet B0 model for the 8 words we trained on. The training was conducted using the test set, and the accuracy achieved was 93.51%.

Through YOLO, as shown in Figure 8, we are able to detect individuals. Upon detecting a person, we can classify eight simple commands (Down, Go, Left, No, Right, Stop, Up, Yes) via microphone input.

Figure 8. Person recognition using YOLO

5. CONCLUSION

In this study, we developed a deep learning object detection and tracking program in the Windows OS environment using the Qt framework. By integrating this program with a machine learning model, we created a user-friendly interface capable of real-time user recognition and simultaneous voice recognition. We constructed a high-performance and efficient deep learning system by leveraging the OpenCV and CUDA libraries. Moreover, we utilized an EfficientNet-based model for the voice recognition system, ensuring it is suitable for small devices with short inference times and low memory consumption. Additionally, we chose YOLO's simple neural network architecture for real-time object detection, providing an accurate object recognition-based system.

We aimed to combine computer vision and voice recognition systems to identify users, detect objects, and extract commands via a microphone, thereby integrating crucial technologies in the field of real-time systems. We particularly focused on situations requiring real-time processing, such as within confined spaces like vehicle interiors, to enhance user convenience and enable application on small devices. Furthermore, we aimed not only for simple command recognition but also for more complex conversational interactions, targeting the establishment of conversational systems. We anticipate that this research direction will be actively utilized in future real-time system fields, with the expectation that the performance of real-time systems will improve, enabling broader applications across various domains.

References

[1] O. A. Hamid, A. Mohamed, H. Jiang, L. Deng, G. Penn and D. Yu, " Convolution Neural Networks for Speech Recognition" , Transactions on Autio Speech and Language(TASLP) , vol. 22, pp. 1533-1545, 2014. DOI: https://doi.org/10.1109/TASLP.2014.2339736

[2] Juyoung Kim, Dai Yeol Yun, Oh Seko Kwon, Seok Jae Moon and CHio gon Hwang " Comparative Analysis of Speech Recognition Open API Error Rate" International Journal of Advanced Smart Convergence (IJASC), vol. 10, pp. 79-85, 2021

[3] J. Redimon, S. Divvala, R. Girshick and A. Farhadi, " You Only Look Once: Unified, Real-Time Object Detection" , Computer Vision and Pattern Recognition (CVPR) , ISSN. 1063-6919, pp. 779-788, 2016. DOI: https://doi.org/10.1109/CVPR.2016.91

[4] P. Warden, " Speech Commands: A Dataset for Limited-Vocabulary Speech Recognition" , arxiv:1804.03209, pp. 1-11, 2018.
DOI: https://doi.org/10.48550/arXiv.1804.03209

[5] Y. Zhang, B. Li, H. Fang and Q. Meng, " Spectrogram Transformers for Audio Classification" , International Sustainability Transitions(IST) , 2022
DOI:https://doi.org/10.1109/IST55454.2022.9827729

[6] J. Redmon, A. Farhadi, "YOLO9000: Better, Faster, Stronger", Computer Vision and Pattern Recognition (CVPR) , 2017.
DOI:https://doi.org/10.1109/CVPR.2017.690

[7] J. Redmon, Ali Farhadi, "YOLOv3: An Incremental Improvement", arxiv: 1804.02767 , pp. 1-6 , 2018.
DOI: https://doi.org/10.48550/arXiv.1804.02767

[8] J. Liang," Image classification based on RESNET" , International Conference on Computer Information Science and Application Technology (CISAT) , pp. 1-6 , 2020.
DOI: https://doi.org/10.1088/1742-6596/1634/1/012110

[9] Z. Zhong, M. Zheng, H. Mai, J. Zhao and X. Liu," Cancer image classification based on DenseNet model" , International Conference on Artificial Intelligence Technologies and Application (ICAITA) , pp. 1-6 , 2020.
DOI: https://doi.org/10.1088/1742-6596/1651/1/012143

[10] J. Wang, L. Yang, Z. Huo, W. He and J. Luo," Multi-Label Classification of Fundus Images With EfficientNet" ,Institute of Electrical and Electronics Engineers (IEEE) , vol. 8, pp. 212499-212508 , 2020.
DOI: https://doi.org/10.1109/ACCESS.2020.3040275

[11] M. Tan and Q. V. Le," EfficientNet: Rethinking Model Scaling for Convolutional Neural Network" ,arxiv: 1905.11946, pp. 1-11 , 2019.
DOI: https://doi.org/10.48550/arXiv.1905.11946

[12] M. Halle and K. Stevens," Speech recognition: A model and a program for research" , IRE Transactions on Information Theory, Vol. 8, pp. 155-159, 1962.
DOI: https://doi.org/10.1109/TTT.1942.1057686