SIGN LANGUAGE RECOGNITION SYSTEM USING PYTHON AND OPENCV

Er. Manisha Vaidya[1], Himanshu Tambuskar[2], Gaurav Khopde[3], Snehal Ghode[4], Sushrut Deogirkar[5],
*1Assistant Professor, *2,3,4,5 Students Department of Computer Science and Engineering,
Priyadarshini J L College of Engineering, Nagpur-440027,Maharashtra, India.

---***---

Abstract: Sign language recognition systems are becoming increasingly important for bridging the communication gap between hearing-impaired individuals and the hearing population. In this context, Python, OpenCV (Open-Source Computer Vision) and Mediapipe framework have emerged as powerful tools for developing sign language recognition systems. Python is a widely-used programming language with an extensive range of libraries and modules for computer vision, machine learning, and image processing, making it ideal for developing sign language recognition algorithms. OpenCV is an open-source computer vision library that provides a wide range of functionalities for image and video analysis, such as image filtering, feature detection, and object tracking. Mediapipe is a cross-platform framework for building machine learning pipelines, specifically designed for processing media data, including audio, video, and images. The main purpose of this paper is to demonstrate the methodology with the use of these technologies in sign language recognition systems involves capturing video or image data of sign language gestures and processing it using computer vision and machine learning algorithms to recognize the gestures and translate them into spoken or written language. Machine learning algorithms such as CNNs, SVMs, or RNNs can be used to train and classify the sign language gestures. However, these systems still face challenges, such as variability in sign language gestures and real-time processing. Ongoing research in this field aims to address these challenges and improve the accuracy and usability of sign language recognition systems.

Keywords: Data collection, Preprocessing, Hand landmarks detection, Training module, Sign recognition.

1. INTRODUCTION

Sign language recognition system is a technology that enables computers to understand and interpret sign language gestures. Sign language is a visual language used by people who are deaf or hard of hearing to communicate with others. However, not everyone understands sign language, which can be a barrier to communication. Sign language recognition technology helps to bridge this communication gap by enabling computers to recognize and interpret sign language gestures. The technology behind sign language recognition systems typically involves computer vision techniques such as image processing, feature extraction, and pattern recognition. In recent years, machine learning algorithms such as deep learning have also been used to improve the accuracy and robustness of sign language recognition systems. Overall, sign language recognition systems have the potential to break down communication barriers and enable greater inclusion and accessibility for deaf and hard of hearing individuals in a variety of settings.

These systems use computer vision and machine learning techniques to recognize and translate sign language gestures into spoken or written language. The recognition process involves capturing and analyzing sign language videos or images, detecting the hand and body movements, and matching them with the corresponding sign language gestures. Despite the progress made in this field, sign language recognition systems still face significant challenges due to the high variability in sign language gestures and the need for real- time processing. Ongoing research aims to overcome these challenges and improve the usability and effectiveness of these systems in various settings, including education, healthcare, and public communication.

The sign language recognition system is classified into two categories: 1. Sign language recognition systems typically use sensors or cameras to capture video or image data. These can include RGB cameras, depth sensors, or wearable devices such as gloves or bracelets with embedded sensor. 2.Sign language recognition systems use machine learning algorithms to analyze the captured video or image data and recognize the sign language gestures. These algorithms can include convolutional neural networks (CNNs), recurrent neural networks (RNNs), or support vector machines (SVMs). In public communication, these systems can be used to provide sign language interpretation for public speeches or news broadcasts.

Sign language recognition systems still face significant challenges due to the high variability in sign language gestures, lighting conditions, and occlusion (when a part of the body or hand is hidden from the camera's view). In addition, real-time processing is essential for effective communication, which requires fast and efficient algorithms. Overall, sign language recognition systems have the potential to improve the accessibility and inclusivity of communication for deaf and hard-of-hearing individuals. Ongoing research and development in this field will likely lead to more accurate and reliable systems that can beused in a variety of settings.

2. RELATED WORK

Sign language recognition systems have gained significant attention in recent years due to the increasing demand for assistive technologies for individuals with hearing and speech impairments. In this literature survey, we will review some of the recent research and developments in sign language recognition systems. Deep Learning Based Sign Language Recognition System:

Deep learning-based approaches have shown remarkable results in sign language recognition. In this approach, convolutional neural networks (CNN) and recurrent neural networks (RNN) are commonly used. For example, Kanchon podder (2022) proposed a Bangla sign language (BDSL) Alphabets and numerals classification using deep learning model [8]. The system achieved an accuracy of 91.2% on the Bangla Sign Language (ASL) dataset. Hand Gesture Recognition for Sign Language: Hand gesture recognition is a fundamental task in sign language recognition. Several methods have been proposed to recognize hand gestures, including color-based segmentation, template matching, and deep learning-based methods. For instance, the work by Ahmed Kasapbasi (2022) used a convolutional neural network approach to recognize hand gestures in sign language, and achieved an accuracy of 95.5% on their dataset [2]. Real-time machine learning and mediapipe identification of common sign languages The author of this model, Arpita Haldera (2021), provided a methodology with a 99% accuracy rate for sign language recognition using an open source framework and machine learning algorithm [4]. Real-time sign language for hand gesture identification was proposed by author Prashant Verma (2022) using OpenCV and Tensorflow, which are utilised for classification, perception, comprehension, discovery, and prediction [3].

3. METHODOLOGY

3.1 Dataset collection: It is necessary to create a suitable database of sign language gestures so that the images collected while communicating with this system can be compared. There are numerous pre-made datasets are available for the project but couldn't locate any that suited our needs in the form of raw photos. As a result, then decided to construct our own data set by using Open computer vision library. The dataset consists of images of the American Sign Language alphabet from A to Z. Eachfolder in the dataset has approximately 300 photos of a certain sign.

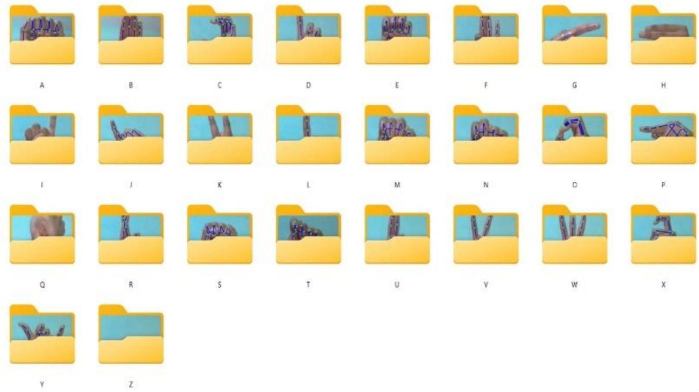
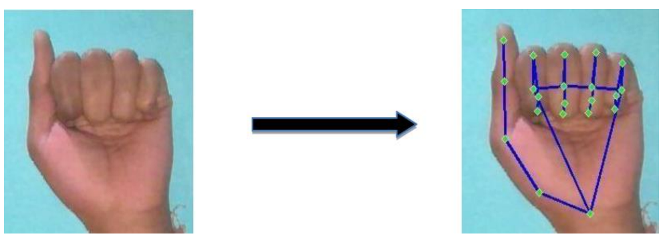


Fig 1: Images from the collected dataset

3.2 Preprocessing: Preprocess the dataset to remove noise, background, or unwanted objects, and convert the images into a standard format, such as grayscale or RGB. Media-Pipe is a framework that enables developers for building multi-modal. The proposed system (video, audio, any time series data) cross-platform applied ML pipelines. Media-pipe has a large collection of human body detection and tracking models which are trained on a massive and diverse dataset of Google. As the skeletonof nodes and edges or landmarks, they track key points on different parts of the body. All coordinate points are three-dimension normalized. Models built by Google developers using Tensorflow lite facilitate the flow of information easily adaptable and modifiable via graphs.

Recognition of sign language may involve dynamic movement of the palms, facial features, fingertips, and or entire body. The shapes, motions, and materials used in the hand gestures themselves are incredibly diverse. The feature must be effective and dependable to handle the diversity of these variances. Use Numpy or other libraries to extract features from the landmark coordinates, such as the distance, angle, curvature, or velocity of the fingers or hand. The features can be represented as a vector or matrix.



Normal Image

Processed Image

Fig 2: Image after applying ML pipeline

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 04 | April - 2023 Impact Factor: 8.176 ISSN: 2582-3930

3.3 Hand landmarks detection: The MediaPipe Hand Landmarks task allows you to detect hand landmarks in images. This Task can be used to localize important points on the hands and to produce visual effects on the hands. This task runs on picture data as static data or a continuous stream with a machine learning (ML) model and returns hand landmarks in image coordinates, hand landmarks in world coordinates, and handedness (left/right hand) of multiple detected hands. Figure 3 displays the Hand Landmark model's detection of 21 landmark locations.

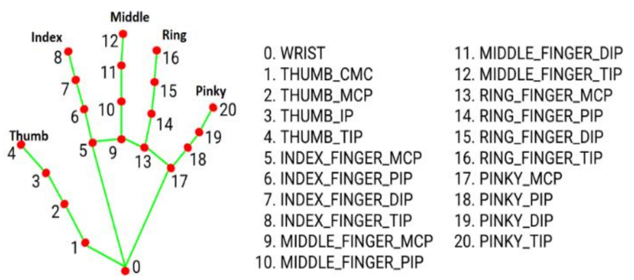


Fig 3: 21 knuckle of hand landmark

Now that the system has a working Palm and Hand identification model, there are A to Z alphabets in the American Sign Language dataset. As a result, the detection model run over each alphabet folder containing photographs and perform Hand detection, yielding the 21 landmark points displayed in Figure 1. The acquired landmark points are subsequently saved in a CSV file.

3.4 Model training: In our scenario, classifiers are used. The techniques or algorithms used to interpret data are called classifiers. The Hidden Markov Model (HMM), K Nearest Neighbor classifiers, Support Vector Machine (SVM), Artificial Neural Network (ANN), and Principal Component Analysis (PCA), Long Short-Term Memory networks (LSTM), Among others, are popular classifiers that recognize or comprehend sign language. However, CNN will be the classifier in this project. CNNs are utilized for picture classification and recognition because of their great degree of precision. Using a hierarchical architecture, CNN creates a network that resembles a funnel before processing the output in a fully-connected layer where all neurons are connected to one another. To extract the image's features, a series of convolution and pooling techniques are used. As we apply more filters, the size of the output matrix shrinks. Size of new matrix = (old matrix size - filter size) +1 As a classifier, a fully connected layer of the convolution neural networks will be used. The likelihood of the class will be predictedin the final layer.

The main steps of convolution neural networks are as follows:

1. Convolution 2. Pooling 3. Flatten 4. Full connection. In this architecture, two convolution layers are implemented followed by batch normalization and max pooling, followed by global average pooling with dense layer and batch normalization, and afinal dense layer for classification

```
Accuracy: 99.32
Model: "sequential"
_________________________________________________________________
 Layer (type)                Output Shape              Param #
=================================================================
 conv2d (Conv2D)             (None, 117, 157, 32)      1568

 dropout (Dropout)           (None, 117, 157, 32)      0

 max_pooling2d (MaxPooling2D  (None, 58, 78, 32)       0
 )

 conv2d_1 (Conv2D)           (None, 55, 75, 64)        32832

 dropout_1 (Dropout)         (None, 55, 75, 64)        0

 max_pooling2d_1 (MaxPooling  (None, 27, 37, 64)       0
 2D)

 flatten (Flatten)           (None, 63936)             0

 dense (Dense)               (None, 120)               7672440

 dense_1 (Dense)             (None, 24)                2904

=================================================================
Total params: 7,709,744
Trainable params: 7,709,744
Non-trainable params: 0
```

Fig 4: CNN Model Summary

Sign Recognition: The camera is used as a real-time input to collect the live feed of various signs and gestures in order to provide an accurate result of the given input. This can be accomplished utilizing computer vision techniques. The previous step's data may need to be preprocessed to remove noisy or unnecessary data that could interfere with recognition accuracy. Once the hand has been detected, the next step is to identify the hand landmark. This is accomplished by utilizing a landmark detection model such as Mediapipe, which identifies the coordinates of the hand by using features such as knuckles and edges. Following the identification of hand landmarks, characteristics are retrieved from these landmarks. This can include measurements such as the distance between fingers, finger angles, and even the curve of the hand. The retrieved features are then utilized to train a CNN machine-learning model. The model learns to link different hand gestures with different sign language words. Finally, the system recognizes and tracks hand landmarks in real time, extracting features from the landmarks and feeding them into the trained model for inference. The model then predicts the corresponding sign language word, which is displayed or spoken out.

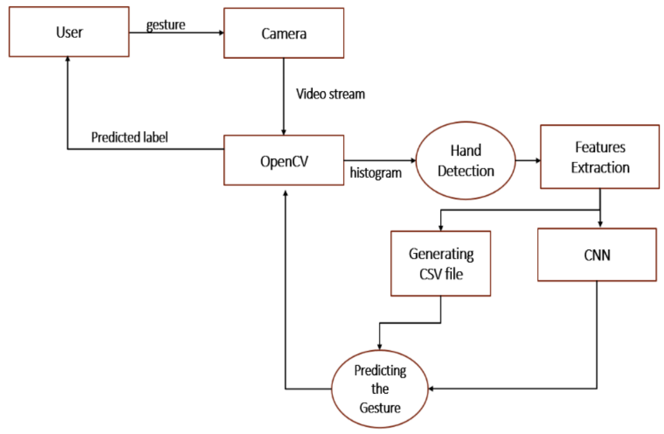


Fig 5: Dataflow diagram of sign language recognition System.

4. RESULT

Accuracy and loss of the model for validation data may vary with different scenarios as we train it. Loss should typically decrease and exactness should increase with each successive period. However, many scenarios, such as the following, can be conceivable with validation loss (keras validation loss) and validation accuracy.

1. As validation accuracy declines, validation loss increases. This suggests the model is memorizing values rather thanlearning.
2. As validation loss starts to rise, validation accuracy will follow suit. When softmax is employed in the output layer,this may be an instance of overfitting or different probability values.
3. Validation loss begins to decline and validation accuracy begins to rise. This is also okay because it shows that the model that was created is adapting and operating properly.

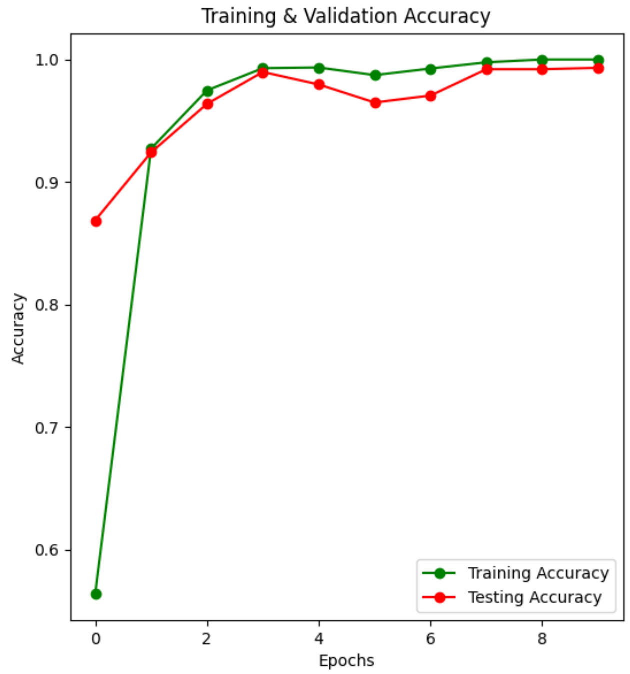


Fig 6: Training & Validation Accuracy graph

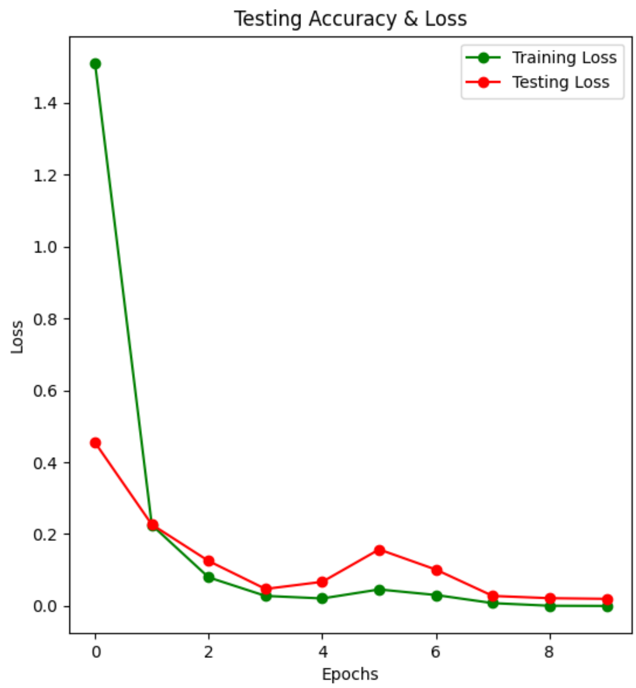


Fig 7: Testing & Training Loss graph

International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 07 Issue: 04 | April - 2023 Impact Factor: 8.176 ISSN: 2582-3930

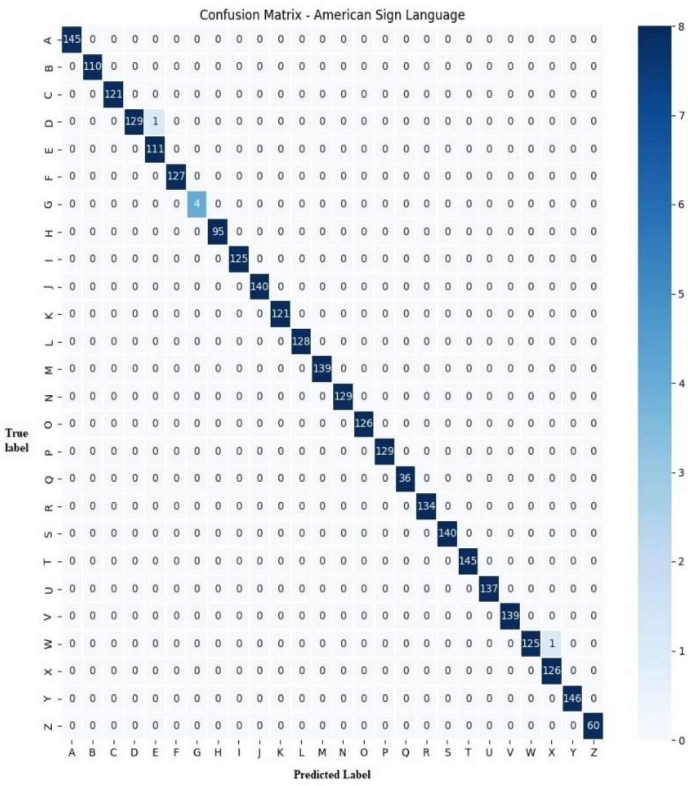
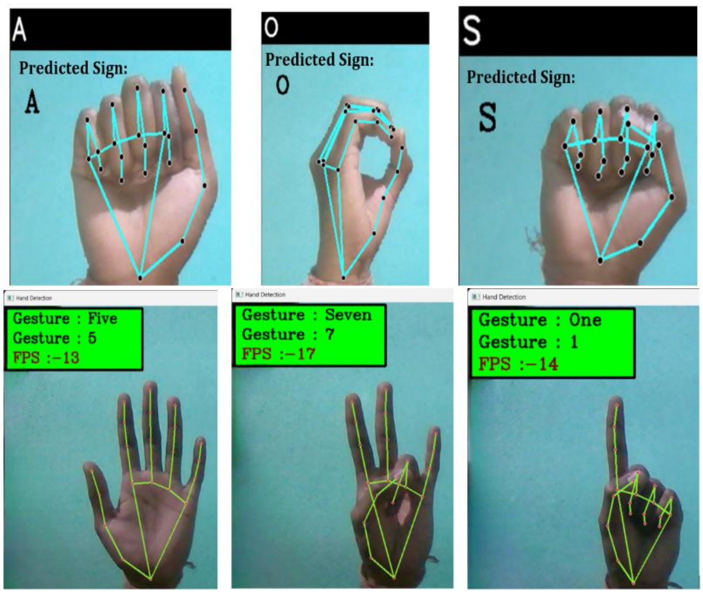


Fig 8: Confusion matrix



Fig,9: Real-time Predicted Result for alphabet A, O and S and number 5,7,1.

5. CONCLUSION

In conclusion, sign language recognition systems developed using Python, OpenCV, and MediaPipe offeran efficient and powerful solution for bridging the communication gap between the deaf and hard-of-hearing community. These systems leverage the capabilities of computer vision and machine learning algorithms to accurately recognize sign language gestures and translate them into spoken or written language. A CNN architecture choice provides accuracy of upto 99.2%. The database's image processing contributes to a reduction in computing complexity. Additionally, it aids in accelerating recognition while maintaining a manageable degree of complexity and size for the neural network model. There are some limitations are as follows:

1. The user needs to be in the range of 1 to 2 meter in-front of the camera.
2. The proper light is required.

6. FUTURE SCOPE

This system can now only recognize alphabets, numerals, and some gestures shown with hand gestures. with the help of continuous hand, body, and facial motions, to create sentences, and fully understand what the person is attempting to say and in what tone. Furthermore, one may communicate with the video camera using sign language, while the other could read the text generated from that sign language (sign to text translation). This could make it easier for persons who are deaf or hard of hearing to interact in video chats. People could learn sign language more easily.

ACKNOWLEDGEMENT

First and foremost, we would like to express our heartfelt sincere gratitude to Er. Manisha Vaidya of Priyadarshini JL College of Engineering for their excellent guidance and valuable suggestions that helped us throughout this project. The team members would like to thank the Department of Computer Science & Engineering of Priyadarshini JL College of Engineering for providing the necessary support and resources. In the last, we would like to thank all the members involved in the development of the system directly or indirectly. We hope that the project will serve the purpose for which it is developed thereby underlining its success of the project.

REFERENCE

[1] Sanket Bankar, Tushar Kadam, Vedant Korhale, Mrs. A. A. Kulkarni, “Real Time Sign Language Recognition Using Deep Learning” ,2022 .

[2] Ahmed Kasapbasi, Ahmed Eltayeb , Ahmed Elbushra, Omar Al-Hardanee, Arif Yilmaz “ A CNN based human computer interface for American Sign Language recognition for hearing impaired individuals”, 2022

[3] Prashant Verma , Khushboo Badli, “Real-Time Sign Language Detection using TensorFlow, OpenCV and Python”,2022

[4] Arpita Haldera , Akshit Tayadeb, “Real-time Vernacular Sign Language Recognition using MediaPipe and Machine Learning”, 2021

[5] Rachana Patil1, Vivek Patil1 , Abhishek Bahuguna1, and Mr. Gaurav Datkhile, “Indian Sign Language Recognition using Convolutional Neural Network” , 2021.

[6] Shruti Chavan, Xinrui Yu and Jafar Saniie, “Convolutional Neural Network Hand Gesture Recognition for American Sign Language” ,2021

[7] Ahmed Kasapbasi ,Ahmed Eltayeb ,Ahmed Elbushra, Omar Al-Hardanee , Arif Yilmaz “A CNN based human computer interface for American Sign Language recognition for hearing impaired individuals.”, 2022.

[8] Kanchon K. Podder Muhammad E.H. Chowdhury Anas M. Tahir Zaid Bin Mahbub Md Shafayet Hossain Muhammad Abdul Kadir, “Bangla Sign Language (BdSL) Alphabets and Numerals Classification Using a Deep Learning Model”,2022

[9] Pooja M.R Meghana M Praful Koppalkar Bopanna M J Harshith Bhaskar Anusha Hullali, “Sign Language Recognition System”, 2022. ijsepm.C9011011322.

[10] Bekhzod Olimov, Shraddha M. Naik, Sangchul Kim, Kil-Houm Park & Jeonghong Kim “An integrated mediapipe-optimized GRU model for Indian sign language recognition”, 2022.

[11] Satwik Ram Kodandaram, N.Pavan Kumar, Sunil Gl, “Sign Language Recognition ”, 2021.

[12] Mathieu De Coster, Mieke Van Herreweghe, Joni Dambre, “Isolated Sign Recognition from RGBVideo using Pose Flow and Self-Attention”,2021. CVPRW_2021

[13] Songyao Jiang, Bin Sun, Lichen Wang, Yue Bai, Kunpeng Li and Yun Fu, “Sign LanguageRecognition via Skeleton-Aware Multi-Model Ensemble”, 2021. 2110.06161v

[14] Ishika Godage, Ruvan Weerasignhe and Damitha Sandaruwan “Sign Language Recognition For Sentence-Level Continues Signing”, 2021. csit112305

[15] N. Mukai, N. Harada, and Y. Chang, "Japanese Fingerspelling Recognition Based on Classification.”