Faculty of Electrical and Electronic Engineering Technology

DEVELOPMENT OF SIGN LANGUAGE INTERPRETER USING COMPUTER VISION TECHNIQUE

NASHA ATHILAH BINTI ZAINAL

Bachelor Of Computer Engineering Technology (Computer Systems) With Honours

2022

DEVELOPMENT OF SIGN LANGUAGE INTERPRETER USINGCOMPUTER VISION TECHNIQUE

NASHA ATHILAH BINTI ZAINAL

A project report submitted
in partial fulfillment of the requirements for the degree of
Bachelor Of Computer Engineering Technology (Computer Systems) With Honours

Faculty of Electrical and Electronic Engineering Technology

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

2022

DECLARATION

I declare that this project report entitled “Development of Sign Language using Computer Vision Technique” is the result of my own research except as cited in the references. The project report has not been accepted for any degree and is not concurrently submitted in the candidature of any other degree.

Signature : NASHA

Student Name : NASHA ATHILAH BINTI ZAINAL

Date : 17/1/2023

APPROVAL

I hereby declare that I have checked this project report and in my opinion, this project report is adequate in terms of scope and quality for the award of the degree of Bachelor of Computer Engineering Technology (Computer Systems) with Honours.

Signature	:	
Supervisor Name	:	Dr.Suhaila Binti Mohd Najib
Date	:	23/2/2023

DEDICATION

I want to express my sincere thanks to my parents Roslim Binti Ali and Zainal Bin Amat for their constant words of support as I finished my final year 1 project. They are a huge support to me as I strive to carry out my project. They gave me some advice on how to complete it on time and without stress. They also set up a comfortable place where I could get inspiration and ideas to finish my project. Not to mention my sibling and friends, who provided me with a wealth of advice on how to make my work better. Thank you for all of your help and guidance during all of my inquiries about the project, Dr. Suhaila, my supervisor.

ABSTRACT

To assist the social interaction of deaf and hearing-impaired people, efficient interactive communication tools are expected. Gesture recognition forms the basis for translating sign languages where gesture recognition plays a critical role in Sign Language Recognition (SLR). The purpose of this project is to provide a sign language interpreter to ease the interaction with the hearing-impaired person based on the computer vision approach. The dataset from the website Roboflow Universe will be used which consists of 26 different hand sign gestures which include A-Z alphabet gestures. All the labeled images will be set as a training set. For the test set, the Python OpenCV library will be used to capture sign gestures from the computer's webcam. The training and test set images will be compared and classified based on the Support Vector Machine (SVM). The output of the system will predict the accuracy of the hand gesture captured by the webcam.

ABSTRAK

Untuk membantu interaksi sosial orang pekak dan bermasalah pendengaran, alat komunikasi interaktif yang cekap adalah diharapkan. Pengecaman gerak isyarat membentuk asas dalam menterjemah bahasa isyarat di mana pengecaman gerak isyarat memainkan peranan penting dalam Pengecaman Bahasa Isyarat (SLR). Tujuan projek ini adalah untuk menyediakan penterjemah bahasa isyarat untuk memudahkan interaksi dengan orang cacat pendengaran berdasarkan pendekatan penglihatan komputer. Set data daripada tapak web Roboflow Universe akan digunakan yang terdiri daripada 26 gerak isyarat tangan berbeza yang termasuk gerak isyarat abjad A-Z. Semua imej berlabel akan ditetapkan sebagai set latihan. Untuk set ujian, perpustakaan OpenCV Python akan digunakan untuk menangkap gerak isyarat tanda daripada kamera web komputer. Imej set latihan dan ujian akan dibandingkan dan dikelaskan berdasarkan Mesin Vektor Sokongan (SVM). Output sistem akan meramalkan ketepatan gerak isyarat tangan yang ditangkap oleh kamera web.

ACKNOWLEDGEMENTS

I would like to sincerely thank my supervisor, Dr. Suhaila Binti Mohd Najib, for her helpful guidance, words of advice, and patience although this project was being conducted.

I thank Universiti Teknikal Malaysia Melaka (UTeM) and my family for helping me with the project by providing me with financial support. In addition, I would like to thank my twin sister Nasha Athirah Binti Zainal, and my friends Nur Najihah Binti Mohd Musa, Ilya Iziana Binti Kamaruzaman, Nur 'Izzati Binti Khalid, Nur Alysha Binti Norazizan, Noor Shafina Binti Abdul Ghani, Chan Yoke Lin, and Nur Hanis Binti Rohaimi for their the willingness to share

Finally, my parents, friends, and family have my deepest gratitude for their support and prayers during this time of study. Many thanks for your inspiration and tolerance

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

1.1 Background

A sign language is a method of communication that involves the use of the hands and other body parts. It is not to be confused with nonverbal communication. Sign languages are an important means of communication for deaf individuals. They are frequently used by deaf individuals in place of spoken languages. Spoken languages are understood through the hearing and rely on sounds produced by the tongue. Sign languages are written with the hands and read with the eyes. Sign languages are easier for deaf and hearing persons to learn than spoken languages.

Communication is significantly more difficult when deaf and hearing people (who do not sign) meet. This problem is sometimes attributed to the deaf person, who is widely considered as impaired (unable to hear or speak clearly) and hence unable to communicate in the same way that hearing people do. Instead, the problem is one of a difference in comprehension between languages and cultures, as in spoken language interactions between different cultural groups. Deaf people exercise agency by using a variety of techniques to communicate with hearing people, such as iconic gestures, writing down words, and pointing.

During the COVID-19 epidemic, the world faced numerous obstacles in various sectors, including the education sector. Teachers and students have struggled to quickly adapt to distance learning at all levels and courses. For students with impairments, the experience may have been much more intense and difficult to adapt to a 'successful' online distance learning experience must involve a number of working components.

1.2 Problem Statement

Every human being need the ability to communicate. People with hearing and/or speech disabilities, on the other hand, require a method of communication other than vocal communication. However, understanding and learning Sign Language takes a lot of practice, and not everyone will grasp what the sign language movements represent. Because there is no good, portable technology for identifying sign language, learning sign language takes time. Hearing or speech-impaired people who know Sign Language will need a translator who also knows Sign Language to effectively communicate their views to others. This technology assists hearing or speech impaired people in learning and translating their sign language in order to help them overcome these issues.

1.3 Project Objective

The main aim project is to create a computer vision-based sign language interpreter to make interactions with hearing-impaired people easier. This is primarily for persons who are unable to communicate with others. Specifically, the objectives are as follows:

a) To recognize sign alphabets from the American Sign Language type using theYolov5 model computer vision technique.

b) To provide a real-time interface that allows a normal hearing-impaired person tocommunicate with a normal hearing-impaired person.

1.4 Scope of Project

In order to achieve the objectives of this project, the scope of project is:

a) The system's limitation is that it only enables the alphabet, not numbers.

b) The dataset will be used which consists of 26 different hand sign gestures.

c) The system will predict the accuracy of the hand gesture captured by the webcam.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

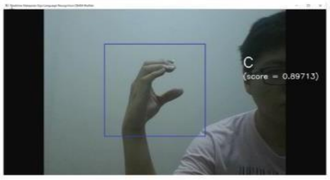
Those who have trouble hearing or speaking utilize sign language as a means of communication. They communicate well amongst themselves, but conversing with ordinary people is difficult. Malaysia Sign Language MSL was founded in 1998 by the Malaysia Federation of the Deaf (MFD). Approximately one million hard-to-hearing adults and nearly half a million hard-of-hearing children use by (Khan *et al.*, 2021) . The majority of hard hearing in town are unable to communicate via gestures. Several scholars proposed and created technologies for deaf or hard-of-hearing people in interacting with non-deaf or hard-of-hearing people. Furthermore, by teaching computers to understand human language, a user-friendly human-computer interface can be created.

In this chapter, I looked at some more similar studies that have been done in the field of sign language interpreters. The following are brief of the prproject'sarious works:

2.2 Malaysian Sign Language (MSL)

This research used a convolutional neural network (CNN) and a convolutional-based attention module (CBAM) to recognize Malaysian Sign Language (MSL) from images to tackle this challenge. For the project, CBAM-2DResNet (2-Dimensional Residual Network arebe used to "Within Blocks" and "Before Classifier" methods. The Python 3.6 programming language and Anaconda Spyder were utilized during he development period d Open CV(Khan *et al.*, 2021).

(a)



(b)

Figure 2.1 (a) “Null” classification result, (b) “C” correct classification with 0.897 confidencescore (Khan *et al.*, 2021)

As shown in figure2.2, to develop the real-time sign alphabets recognition application, the best trained CBAM-2DResNet “Before Classifier” was chosen as a classifier model. Using the OpenCV library, this real-time application provided a direct platform for evaluating the trained model using images taken from frames. Real-time sign images were retrieved from the blue box region for every four frames captured by a camera to feed as test inputs, and if the confidence score was greater than 0.5, the user was given the associated classification result.

Malaysia Sign Language (MSL) is a gesture-based communication system used by the deaf community in Malaysia. From this project, the data-glove approach, which used a specific glove-based device to extract hand posture and motion, is used for hand gesture identification. To identify hand poses that represent the alphabet, number, and numerous words from Malaysian Sign Language, the glove will use a microcontroller as the processor and tilt sensor and an accelerometer as the sensor. The gesture translation will be displayed on the phone(Shukor *et al.*, 2015).

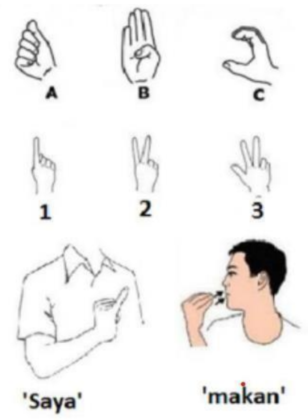


Figure 2.2 The gestures used for the system test (Shukor *et al.*, 2015)

As shown in figure 2.3, the glove also has an acceleration that detects gestures. The alphabets for the test are A, B, and C, selected from the standard Malaysian Sign Language database. 1, 2, and 3 are the numbers that need to be found. Three sets of gestures were examined for the motion: ‘saya,’ ‘Makan,’ and ‘Apa.’.

2.3 American Sign Language (ASL)

The purpose of this study was to describe methods for recognizing American Sign Language (ASL). The Support Vector Machine is a pattern recognition approach used to build sign language recognition systems. Furthermore, genetic algorithms are a sub-field of evolutionary computation, and the camera receives images of user input gestures, detection to determine whether it is hand or not, using a specific algorithm. As a result, the camera receives a user-input gesture image, and uses a specific algorithm, to determine whether it is hand-held or not. Image recognition is the following stage, where images collected from users are compared with photos in a data set to interpret the displayed gestures. The output is the following phase, the identified symbols are translated into text form (Dogra, Malik and Chowdary, 2018)

This study outlines modern ways to bridge the communication gap between the deaf and the hearing impaired. The Python programming language is used to create code. The data was collected from various ASL data set sites and included sign data for 26 letters and ten numbers. For each character input sample, 360 (10 samples per character) samples were used for testing various terms including the alphabet in them. This section outlines the "voice to sign" project proposal methodology. The system will receive speech input through a microphone, and the speech will be processed and converted to the appropriate text format(Patil *et al.*, 2019).

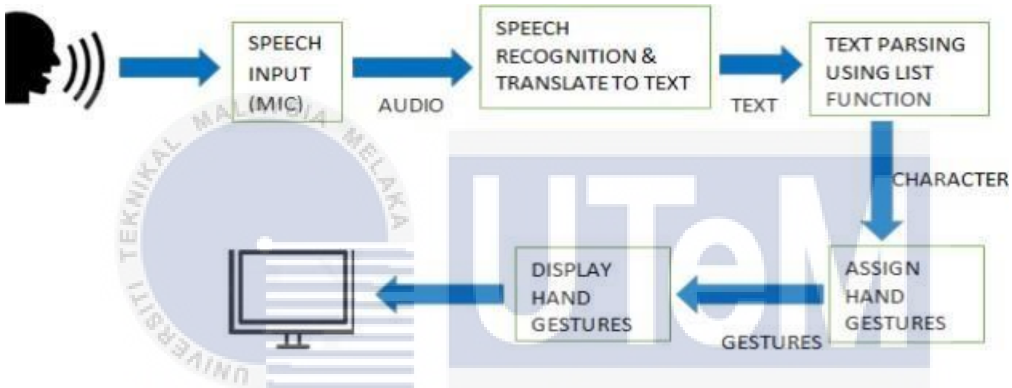


Figure 2.3 Speech to Sign module workflow (Patil *et al.*, 2019)

This report describes Sign Language Recognition using Media Pipes to recognize characters spelled in American Sign Language (ASL) for the deaf population, which can be used as a means of communication between the deaf and others. It provides a simple and accurate way to interact between humans and computers is one of the reasons for choosing a vision-based system. This research uses the Google Media tap. Media pipe solutions have improved their hand recognition models and now recognize 21 3D Palm Landmarks. Open Source Computer Vision (OpenCV) and the Python programming language are used to create the system. The method of taking pictures on the camera as touch data is used to collect sign language sign images. These sign language receivers will be able to recognize letters and detect hands as well as produce coordinators (A-Z). All alerts will be displayed in real-time (Gomase *et al.*, 2022).

This work reported the development of a dataset containing 26 English alphabets and described the deployment of a system that transforms Indian Sign Language into English. In this technique, a webcam is utilized to capture a still hand image frame. The image is then converted to grayscale before being converted to a binary image. The YCbCr model is used to identify skin color at the same time. Finally, to detect edges, the Canny edge detector is utilized. After that, the data was divided into 4800 training shots and 1200 testing photos. These images have been improved through post-production. Using feature extraction and classification algorithms, the sign language is subsequently translated into English text. This translation is converted to speech using the text to speech API (Apoorv, Kumar Bhowmick and Sakthi Prabha, 2020)

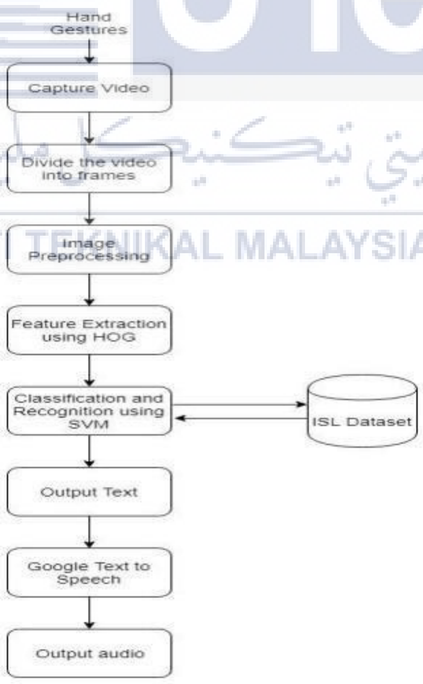


Figure 2.4 Sign Language interpreter flowchart (Apoorv, Kumar Bhowmick and SakthiPrabha, 2020)

The purpose of this paper is to show how to use Media Pipe open-source framework and machine learning algorithm to simplify Sign Language Recognition. Without any wearable sensors, real-time precise detection utilizing the Support Vector Machine (SVM) algorithm makes this technology more comfortable and simpler to use. On the American Sign Language (ASL), Indian Sign Language, and Italian Sign Language datasets, they constructed a simple image categorization model using logistic regression (Halder and Tayade, 2021)

This research looks at how deep learning can be used to classify American Sign Language into multiple classes (ASL). Every image in the ASL dataset will be assigned to one of 29 classes. On our ASL dataset, we design a fully convolutional neural network (CNN) to achieve this. The data set contains 87,000 photographs divided into 29 classes, with 26 for letters A-Z and three for SPACE, DELETE, and NOTHING, and they chose 21750 images, or 750 each class, to be used for the models. They built our model from scratch on a VGG16 network and used pre-trained weights to explore how transfer learning influences performance. They also wanted to test the effects of training the model with deeper and wider networks on performance, so we used InceptionNet and ResNet50 (Sood, 2022)

The goal of this project is to develop a vision-based application that provides sign language translation to text, allowing signers and non-signers to communicate more effectively. The suggested model extracts temporal and spatial characteristics from video sequences. Then, for identifying spatial characteristics, we utilize Inception, a CNN (Convolutional Neural Network). The RNN (Recurrent Neural Network) is then used to train on temporal information. The American Sign Language Dataset was used in this research (Bantupalli, 2018).

This research work shows an innovative context with the primary goal of converting 24 static motions from American Sign Language alphabets into human or machine read able English manuscript. More than ten thousand hand and face gesture signs exist to sign the various English words in ASL alphabets and numbers. This author also explains the system architecture, state of the art, data collecting for the proposed work, suggested system design, and thorough results evaluation by displaying a graphical representation of the proposed technique compared to existing techniques average recognition rate. Pre-processing the input image by an effective segmentation utilizing the BB Technique is distributed as part of the process of recognizing ASL Alphabets. Each data set comprises the 24 ASL alphabet movements. The alphabets 'J' and 'Z' have been left out since they need hand movement (Shivashankara and Srinath, 2019).

Figure 2.5: Translation of gestures of ASL alphabets(Shivashankara and Srinath, 2019)

The goal of this research is to create a sign language translation application utilizing OPEN CV on Android. The application will convert the movements of a finger into an alphabet letter using American sign language (ASL). Figure 2.8 depicts the flowchart. The application will first identify the background color on 7 section coordinates that are designated with 7 boxes. The threshold for acquiring binary imagery from RGBA data input is calculated using this color data. Following the acquisition of seven color data for the hand, seven upper and lower borders for the hand region are determined, which can be represented as a two-dimensional array. The binary images of the hand is then combined with the binary image of the background using the logic 'AND'. Next, is to use morphological operations (dilation and erosion) to remove noise (pixels that aren't in use) and achieve the best possible outcome. Following that, fingertip coordinates or Hu Moments values will be entered on an SVM data model with sequential labeling as part of the feature extraction process (Triyono *et al.*, 2018)

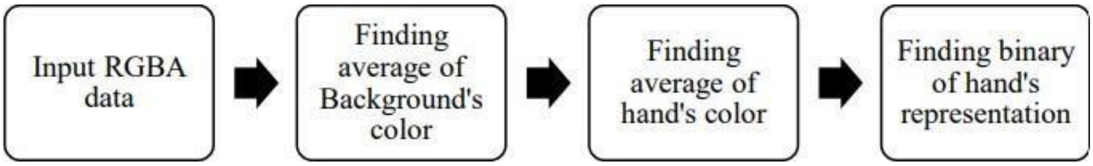


Figure 2.6: Background Subtraction Flowchart(Triyono *et al.*, 2018)

This difficulty is addressed by the system provided here. To recognize the sign made by a gesture, the proposed system leverages the American Sign Language (ASL) data collection. All of the hand shapes and actions are covered by these 70 samples of each symbol. All of the features are from the right hand. A camera is used in this system to collect various hand motions. After that, the image is processed using several techniques. Pre-processing of the image is the first step. The edges are then determined using an edge detection technique. Finally, the sign is identified and the text is displayed using a template-matching algorithm. Because the output is text, it is simple to decipher the meaning of a given sign. The system is implemented using OpenCV and Python language (Shrenika and Madhu Bala, 2020)

In this paper, this proposed system is to recognize ASL Alphabets and Numbers, which mainly depend only on hand and fingers. The process of identifying ASL Alphabets and Numbers is distributed as pre-processing the input image, computation of the region properties of the pre-processed image, and transliteration from treated image to text. 70.83 percent of the time (ASL alphabet movements M, N, Q, V, W, X, and Y are not recognised) and 97.5 percent of the time (ASL alphabet motions G, P, and Q are not recognised in some data sets (S and S, 2018).

2.4 Indian Sign Language (ISL)

The recognition of gestures and sign languages is a well-studied topic in American Sign Language (ASL), but few studies on Indian Sign Language have been published (ISL). Based on the data they collected, they divided their technique for tackling the categorization task into three parts. The first part is to remove the skin from the image, as the remainder can be considered noise in terms of character classification. The next part is to extract key characteristics from the skin segmentation images that will be useful in the learning and classification stages. As previously stated, the extracted features are used as input into various supervised learning models for training, and then the trained models are used for classification (Dhavale, 2019)

The authors presented an Open CV-based system for converting Indian sign language to text. The initial stage in Image Acquisition is to acquire an image through an integrated camera during runtime. Using the SIFT method, photos are recorded in the database for a certain letter. The images will be captured using basic code to access a webcam using OpenCV, which can find hand location, determine image orientation (right or left), and build a hand mask image. Then, using image processing, the digital computer will process the collected images, which are digital in nature. This improves the quality of a photograph so that it appears better (Goyal and Singh, 2014)

2.5 Table of Comparison

Author /Date	Theoretical/Conceptual Framework	Research Hypothesis	Methodology	Analysis & Results	Conclusions	Implications for Future research	Implication for practice
Khan, R.U. et al. (2021)	Design a real-time sign recognition system for translating from sign language to text and from text to sign language in an easy means of communicati	Using the CBAM-ResNet approach, create a real-time MSL Recognition System based on human gestures.	• CBAM-2DResNet • Python with Anaconda Spyder • OpenCV	Real-time signs images were retrieved from the blue box region for every four frames captured by a camera to feed as test inputs, and if the confidence score was greater than 0.5.	Image recognition and video recognition approaches were used to study static and dynamic signs, respectively.	Creating entire phrases from recognized signs via video or in real-time.	CBAM-2DResNet has been selected to create a real-time sign recognition system for translating from sign language to text and from text to sign language, allowing deaf mutes and others

	on between deaf mutes and other individuals.						to communicate more easily.

Author /Date	Theoretical/Conceptual Framework	Research Hypothesis	Methodology	Analysis & Results	Conclusions	Implications for Future research	Implication for practice
Ahmad Zaki Shukor (2015)	To test the system's accuracy in reading sign language	To create a sign language translation system to help persons who are deaf or hard of hearing interact with people who are not deaf or hard of hearing	• Flex sensor • Accelerometer • Microcontroller • Custom made glove • IR optic sensor • Tilt sensor	After tilt sensor test, ten tilt sensors were installed onthe data glove, two for each finger. The glove is equipped with an accelerometer that detects gestures. A, B, and C were the alphabets used in the test. 1, 2, and 3 were the numbers to be found.	The results of the testing reveal that the system correctly detects the alphabets, numerals, and words that were examined. The tilt sensor's bend aids in the detection of alphabets/numbers, while the accelerometer aids in the detection of words/gestures	It can be done by adding a database of different Malaysian Sign Language stances and gestures.	The tilt sensor is mounted vertically (with 0 degrees assumed as the vertical axis)

Author /Date	Theoretical/Conceptual Framework	Research Hypothesis	Methodology	Analysis & Results	Conclusions	Implications for Future research	Implication for practice
Dr.P.R. Patil (2019)	This paper outlines a modern technique to bridging the communication gap between deaf and hearing people.	The stage of the American-language text generation through the speech recognition module.	• Modeling • Gesture animation • Speech to Text conversion	Speech Recognition are some of the python libraries utilised in this project. The accuracy rate of deaf talk utilising 3d animated sign language was determined to be 87 percent.	This application built on different Python modules and dataset generated provides a way for persons with hearing impairments to communicate naturally.	More animation and user-friendly model will be developed.	These animations are shown by 2D hand gestures in real time and a corresponding text is displayed on the screen to help the user in judging the system.

Author /Date	Theoretical/Conceptual Framework	Research Hypothesis	Methodology	Analysis & Results	Conclusions	Implications for Future research	Implication for practice
Dogra, A., Malik, K. and Chowdary, V. (2018)	The goal of developing this system is to make online communication easier for the deaf and mute communities.	The input mode for this project is static hand gesture recognition.	• Camera capture • Support Vector Machine	The input gesture image is captured by the camera. The SVM, or Support Vector Machine, model is used to recognise images. The output is the following phase, in which the identified symbol is translated to text form.	This application, which is built on different Python modules and dataset generated as part of this project.	This system takes into consideration the various drawbacks in the existing system and also their advantages for better working of this system.	Support Vector Machine is used to analyze data and classify them.

Author /Date	Theoretical/Conceptual Framework	Research Hypothesis	Methodology	Analysis & Results	Conclusions	Implications for Future research	Implication for practice
Gomase, K. et al. (2022)	To develop a program to translate sign language into OpenCV.	Electronic recognition of sign language deals from signalling to touch and continues until text / speech production.	• Mediapipe • CMOS sensor • OpenCV • KNN	• This sign language receiver can detect hand and produce coordinator and will be able to recognize letters (A-Z). All signs will appear in real time.	Sign Language using Mediapipe and recognition through Computer vision was partially successful and accurate an average of 17 FPS with an average accuracy of 86 to 91%.	To improve Human Computer Interoperability (HCI) using a very powerful and fast algorithm	Mediapipe Hands is a reliable hand and finger tracking device solution. It uses machine learning (ML) to understand 21 3D local hand marks from just one frame.

Author /Date	Theoretical/Conceptual Framework	Research Hypothesis	Methodology	Analysis & Results	Conclusions	Implications for Future research	Implication for practice
Apoorv, S., Kumar Bhowmick, S. and Sakthi Prabha, R. (2020)	To create an application that will convert sign language to English in the form of text and voice, hence facilitating sign language communication. .	Although it is an efficient mode of communication, communicating with speech disabled people remains a difficulty for those who do not understand sign language.	• Webcam • Histogram of Oriented Gradients(HOG) • Support Vector Machine(SVM)	A webcam is used to capture the hand picture frame. These frames are enhanced by post-processing. The sign language is then translated into English text using feature extraction and classification techniques. The text to speech API was used to turn this translation to speech.	To describe the implementation of an Indian Sign Language to English translation system. We've talked about how important an ISL translator is when communicating with the deaf and mute.	Will focus on developing a mobile applicationbased on this paradigm.	HOG states that intensity distribution gradients or edge directions can be used to represent an object or form inside an image.

Author /Date	Theoretical/Conceptual Framework	Research Hypothesis	Methodology	Analysis & Results	Conclusions	Implications for Future research	Implication for practice
Halder, A. and Tayade, A. (2021)	To develop a program to translate sign language into OpenCV.	To develop a variety of automatic sign language recognition algorithms that can understandably interpret sign motions.	• K-Fold Cross-Validation • Support Vector Machine (SVM) • KNN • Artifical Neural Network(ANN) • Multi-Layer Perceptron	• SVM outperformed other machine learning algorithms in terms of accuracies • Deep learning methods such as Artificial Neural Network (ANN) and Multi-Layer Perceptron exceeded	Our proposed methodology shows that MediaPipe may be efficiently utilised as a tool to recognise complicated hand gestures exactly, with an average accuracy of 99 percent in most of the sign language dataset	The study can be expanded by using Mediapipe's state-of-the-art and best-possible classification algorithms to detect words in sign language from videos.	Cross-validation is a resampling approach used to test machine learning models on a small set of data.

				SVM in terms of accuracies (MLP)	utilising MediaPipe's technology and machine learning.		

Author /Date	Theoretical/Conceptual Framework	Research Hypothesis	Methodology	Analysis & Results	Conclusions	Implications for Future research	Implication for practice
Sood, D. (2022)	To classify every image in the ASL dataset to one of 29 classes.	To explores the application of deep learning to the task of multi-class classification of American Sign Language (ASL).	• convolutional neural network (CNN) • VGG16 • ResNet50 • Inception Net	• ResNet50 achieved the best results training from scratch, while using pretrained weights for VGG16 proved the effectiveness of transfer learning.	The aim of this project was to find a model with highest accuracy for the task of multi-class classification of American Sign Language.	These classification networks can be used, built on further, and even combined with temporal data and recurrent neural networks to learn sequences of words and sentences	ResNet uses batch normalization that adjusts the input layer to increase the performance.

Author /Date	Theoretical/Conceptual Framework	Research Hypothesis	Methodology	Analysis & Results	Conclusions	Implications for Future research	Implication for practice
Bantupalli, K. (2018)	To develop a vision-based programme that translates sign language into text.	Assisting non-signers and signers in communicating	• CNN (Convolutional Neural Network) • •Recurrent Neural Networks (RNN) • •Long short-term memory (LSTM) networks • •OpenCV	ResNet50 achieved the best results training from scratch, while using pretrained weights for VGG16 proved the effectiveness of transfer learning.	The dataset was made up of videos into frames using Python's OpenCV. The dataset was randomly divided into two parts: 80 percent for training and 20% for testing. ..	It's a method for dealing with the issues that persons with hearing and speech impairments confront. It consists of two primary components: analysing and classifying images from motions.	Employing two separate models to feed into one other to improve CNN results in data loss and increased training time, whereas using a single ensemble allows for thorough monitoring of input data and accurate model modifications

Author /Date	Theoretical/Conceptual Framework	Research Hypothesis	Methodology	Analysis & Results	Conclusions	Implications for Future research	Implication for practice
Shivashankara, S. and Srinath, S. (2019)	To convert 24 static motions from the American Sign Language alphabets into a human or computer readable writing in the English language	After clear segmentation and preprocessing steps, the gesture recognition procedure begins.	• Preprocessing and segmentation • Feature extraction and transliteration	The proposed technique was evaluated on 28 data sets of ASL Alphabet motions. Each data set comprises the 24 ASL alphabet movements.	This research article uses an innovative framework to attain a 98.21% average recognition rate.	This research may be expanded to recognise ASL Alphabets, Numeric motions, and some complicated gestures in both static and real-time situations, taking into account plain and complex backgrounds with varying lighting conditions.	Canny Edge Detector (CED), also called optimal detector.

Author /Date	Theoretical/Conceptual Framework	Research Hypothesis	Methodology	Analysis & Results	Conclusions	Implications for Future research	Implication for practice
Triyono (2018)	American Sign Language (ASL) to explaining the words and concepts..	The creation of an Android-based sign language translator application using OpenCV.	• Finding background color • Finding hand color • Binary of the hand representation • Finding contour and feature extraction • SVM Model File	Gesture image. Features File SVM Model File .	The colour difference method can be used to detect skin colour as a recognised object in sign language. .	Use OpenCV's Motion Templates to your advantage.	The silhouette or contour of an item in a photograph is usually used to calculate Hu Moments.

Author /Date	Theoretical/Conceptual Framework	Research Hypothesis	Methodology	Analysis & Results	Conclusions	Implications for Future research	Implication for practice
Shrenika and Madhu Bala, (2020)	Works on American Sign Language.	A camera is used in this system to collect various hand motions	• Gesture Image • Pre-Processing • Feature Extraction • Template Matching	To begin, use a camera to capture an image. Then, for further processing, transform it to a grayscale image. The sign in the image was detected using an edge detection method. The sign alphabet is displayed as the final stage.	The purpose of this document is to assist and serve our society's deaf in communicating with regular people. The system is implemented utilising image-processing techniques in this case..	They can create a two-way system that allows for the conversion of signs to text and text to signs.	The edge picture obtained via edge tracking is the template image.

Author /Date	Theoretical/Conceptual Framework	Research Hypothesis	Methodology	Analysis & Results	Conclusions	Implications for Future research	Implication for practice
Dhavale, 2019	Being done in Indian Sign Language .	To take a step ahead in this subject by gathering a dataset and then feature extraction approaches, which is then fed into various supervised learning techniques.	• Image Capture • Image Preprocessing • Image Preprocessing • SVM is a Support Vector Machine	They conduct experiments on the same (30% or 70%) dataset for both training and testing the KNNclassifier.	They have been used many technologies to go through an autonomous sign language gesture recognition system in real-time in this project.	ISL alphabets, words, and sentences must be included. These signs could be used in the future.	SVM is a supervised machine learning model with a learning method for classification and regression analysis.

Author /Date	Theoretical/Conceptual Framework	Research Hypothesis	Methodology	Analysis & Results	Conclusions	Implications for Future research	Implication for practice
Goyal and Singh, 2014	conversion of Indian sign language .	to create a human-computer interaction system capable of reliably recognising deaf and dumb language	• Image Acquisition • Feature Extraction • Orientation Detection • Gesture Recognition	The procedure includes passing a one-dimensional array of 26 characters corresponding to alphabets, with the image number stored in the database providedin the array. .	The system will have a Sign Language Recognition interface that will make it simple to communicate with deaf people.	will create a simple gesture recognition system using the OpenCv toolkit and incorporate it into the Visionary framework.	OPEN CV is a machine learning and computer vision software library that is open source.

2.6 Review of Comparison Table

We compare authors, framework, research hypothesis, methodology, and result based on a table of study comparisons. From the researches, The Support Vector Machine (SVM) is a strong choice for classification problems such as hand gesture recognition. This is demonstrated in a study by (Dhavale, 2019), in which the system uses SVM to categorize one-handed and two-handed alphabets (with 56 percent accuracy) (60 percent accuracy). It also proved in the (Apoorv, S., Kumar Bhowmick, S. and Sakthi Prabha, R. 2020). With the implementation of SVM, the system's accuracy was increased to 88 percent. The majority of the researchers, Malaysian Sign Language (MSL), American Sign Language (ASL), and Indian Sign Language are some of the most frequent sign languages (ISL).

2.7 Summary

Sign languages are naturally occurring manual languages that develop in deaf populations. These manual communication systems are fully expressive, systematic human languages, not just traditional pantomime systems or manual codifications of spoken languages. In this research, we can conclude that most of the projects use the same software which is OpenCV (Open Source Computer Vision Library) is a machine learning and computer vision software library that is free to use. OpenCV was created to provide a common infrastructure for computer vision applications and to let commercial goods incorporate machine perception faster using a python programming language. As a result, the goal of this work is to discover a systematic and successful process for developing sign language interpreters using computer vision techniques, and, the main focus of this work by using American Sign Language.

CHAPTER 3

METHODOLOGY

3.1 Introduction

This chapter provides a description and explanation of the whole project from the beginning until the end and also the flow of steps involved in the development of the Sign language Interpreter using the Computer Vision technique. Besides that, this chapter will discuss about the tools used in the project. For software development, PyCharm is an integrated development environment (IDE), designed exclusively for the Python programming language. Pycharm is a Python IDE toolkit designed for writing programs and/or creating software.

3.2 Methodology

The first step is to collect data by hand movements. Many researchers have employed sensors or cameras. Effective interactive communication technologies are expected to enhance deaf and hearing-impaired people's social involvement in our system. Gesture recognition is important in Sign Language Recognition because it sets the groundwork for translating sign languages.

The goal of this project is to create a computer vision-based sign language interpreter to make interactions with hearing-impaired people easier. The Roboflow universe dataset will be used, which contains 26 classes that consist of 4589 images consists of A-Z letter gestures. The training set will be created from all of the labeled images that can be used to recognize sign language using YOLO (You Only Look Once).

The Python OpenCV library will be used to capture sign gestures from the computer's webcam for the test set. The model is then evaluated and the system would then be able to predict the alphabet. The system's output will forecast the accuracy of the webcam-captured hand motion.

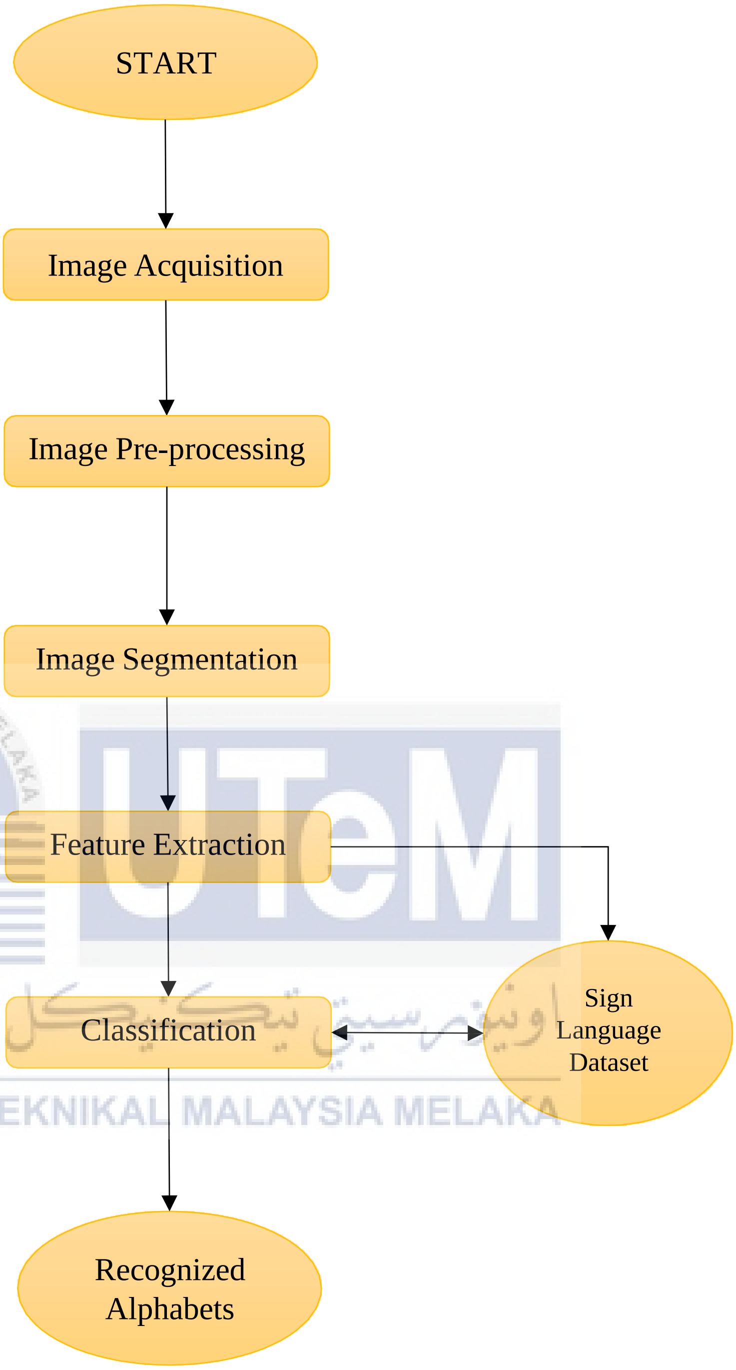


Figure 3.1 Flowchart of a Sign language interpreter

3.2.1 YOLO (You Only Look Once)

"You Only Look Once" (YOLO) is one of the deep learning approaches that has lately become popular for object detection *(YOLO). YOLO* can recognize objects quickly and effectively. It is notable for its quickness. The detection of sign languages using YOLOv5 can be performed by training the system in a network based on pre-labeled images that appropriately categorize the image's gesture.

3.2.2 Image Acquisition

The act of retrieving a picture from an external source for subsequent processing is known as image acquisition. Because no method is accessible before getting an image, it is always the first step in the workflow. A camera or webcam is the most commonly used device by many academics because it allows for better and more natural human-computer interaction without the necessity of extra equipment. The images will be acquired using basic code that opens a webcam using OpenCV and then captures the image using frames per second, which will be kept in a separate directory from all of the input images.

3.2.3 Image Pre-processing

Preprocessing techniques are used to increase the quality of an input image. This can be done by color images, greyscale images, binary images, and multispectral images are all types of digital images. Each pixel in a colour image has its own colour information. Greyscale images have only shades of grey as their only colour, whereas binary images have only two colours, generally black and white pixels. Image data spanning the electromagnetic spectrum within a specific wavelength is captured in multispectral images. The outcome of this process can have a substantial impact on accuracy if the preprocessing techniques are well chosen.

3.2.4 Image Segmentation

Image segmentation is the next step in the process of partitioning a digital image into several image segments, also known as image regions or image objects, in digital image processing and computer vision (sets of pixels). The purpose of segmentation is to make an image more understandable and easier to evaluate by simplifying and/or changing its representation. For training, we used the roboflow universe website, which has roughly 4000 images of sign languages.

3.2.5 Feature Extraction

Feature extraction is a sort of dimensionality reduction in which a large number of pixels in an image are efficiently represented in such a way that the image's most interesting sections are effectively captured. In this study, feature extraction was employed to recognize the ASL alphabet. The number of obtained joint coordinates in 3D space with X, Y, and Z-axis values is 21, and these coordinates have been used to extract additional features. Even if the hand on the right edge of the camera or image has the same signature as the hand on the left edge, the output will be presented as a different value.

3.2.6 Classification

The support vector machine (SVM) method has been used to classify the data. For unstructured and semi-structured high-dimensional datasets, SVM works well. Because SVM models have generality in practice, the risk of over-fitting is reduced. SVM is utilized in text and hypertext classification, as well as in the recognition of handwritten characters. They are employed in the classification of images.

3.2.7 Dataset

ASL or American sign language is a sign language used in English-speaking nations such as the United States. It consists of 26 letters of the alphabet from A to Z that may be articulated with one hand. The ASL data was the first dataset used in this project. The dataset for this project was taken from the Roboflow Universe website, and it contains more than 1500 images of sign languages, divided into three datasets: training validation and testing, as seen in figure 3.2. About 4589 and 340 images, respectively, from the training and validation dataset, will be utilized to train and validate the YOLOv5 model.

Figure 3.2 Dataset Images from Roboflow Universe website.

The function of each dataset:

1. **Training dataset:** A collection of images or videos of people signing would make up a training dataset, coupled with labels identifying the words or phrases being signed. A machine learning model would be trained using the dataset to identify and categorize various signs and gestures in the photos or videos. In order to train the modelto correctly predict the labels for future cases, a large number of labeled examples would be provided. The model's parameters would then be adjusted. In order to train amodel that can generalize effectively to new data, it is crucial to have a diverse and representative dataset. Both the quality and quantity of the training data can have a substantial impact on the performance of the trained model.

2. **Validation dataset:** The dataset would be used to assess a machine learning model's performance during training and optimize its hyperparameters. To fine-tune the model's hyperparameters, such as the learning rate and regularisation coefficients, the validation dataset, which is often a subset of the training dataset, is employed. The model can be tested on a different test dataset after it has been trained and its hyperparameters have been tuned to determine how well it performs on data that has not yet been observed. The validation dataset would demonstrate that the model is capable of precisely recognizing and classifying a wide variety of signals and gestures in the setting of a sign language interpreter.

3. **Testing dataset:** The dataset used for testing should be distinct from those used for training and validating the model, and it should be a good representation of the data that will be used in actual applications. You can get a sense of the model's performance on the test dataset and how it will perform when used to solve real-world issues. The

testing dataset would ensure that the model is capable of precisely recognizing and classifying a wide variety of signals and gestures in the setting of a sign language interpreter.

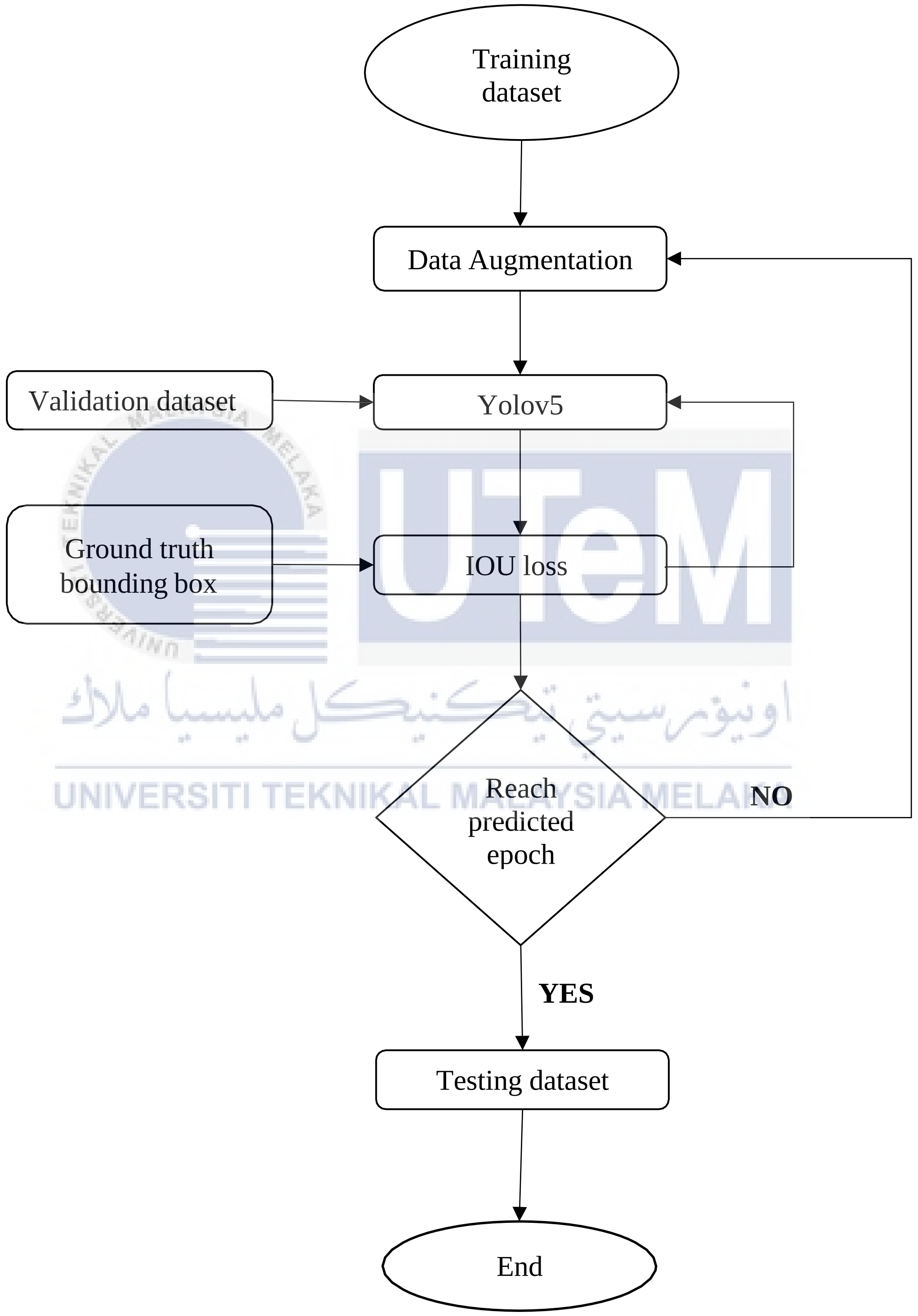
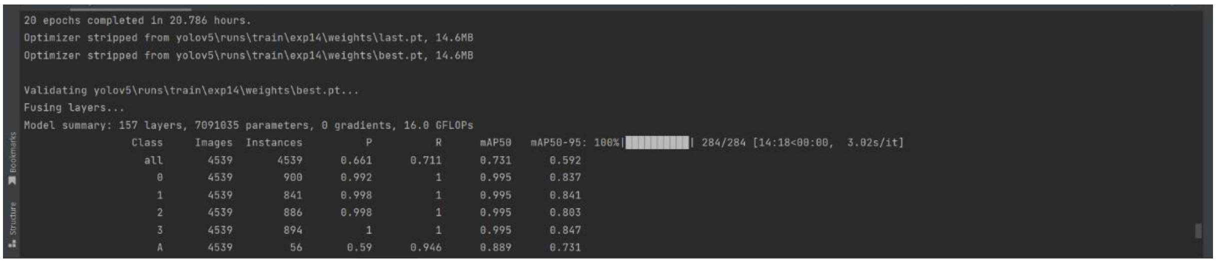


Figure 3.3 Training flowchart of Yolov5

The procedure required for the training system is shown in Figure 3.3. The dataset will be trained and validated based on YOLOv5 object detection model to extract the expected box from the images following the dataaugmentation. Data augmentation is a technique that allows practitioners to greatly broaden the variety of data available for training models. Techniques for enhancing data such as cropping, padding, and horizontal flipping. The projected bounding box and the actual bounding box will then be compared. A total IOU loss will be realized by comparing these two as a result of the overlap area, distance, and aspect ratio. Imagesfrom the training dataset and the testing dataset must be distinct from one another. The output value of the bounding boxes can then be used to determine how well the trained model performed. Images of the identified sign hand with bounding boxes should be the output.

3.2.8 Epoch (training cycle)

A trained machine learning model's performance can be greatly influenced by the number of epochs. The accuracy of the model can be increased by increasing the number of epochs, although this can lengthen training time. To obtain the appropriate degree of model performance, it is crucial to strike the proper balance between the number of epochs and the training time. The system has a sizable training dataset of 4589 images, thus the batch size is set at 16. Before the system can compare the real-time sign language with the testing dataset,it must first be trained to detect a wide range of shapes and hand signs. The training system was taken about 20.786 hours to complete the training system with 20 epochs as shown in Figure 3.4 while for 50 epochs, around 60 hours were taken.



```
20 epochs completed in 20.786 hours.
Optimizer stripped from yolov5\runs\train\exp14\weights\last.pt, 14.6MB
Optimizer stripped from yolov5\runs\train\exp14\weights\best.pt, 14.6MB

Validating yolov5\runs\train\exp14\weights\best.pt...
Fusing layers...
Model summary: 157 layers, 7091035 parameters, 0 gradients, 16.0 GFLOPs
                 Class     Images  Instances          P          R      mAP50   mAP50-95: 100%|██████████| 284/284 [14:18<00:00,  3.02s/it]
                   all       4539       4539      0.661      0.711      0.731      0.592
                     0       4539        900      0.992          1      0.995      0.837
                     1       4539        841      0.998          1      0.995      0.841
                     2       4539        886      0.998          1      0.995      0.803
                     3       4539        894          1          1      0.995      0.847
                     A       4539         56       0.59      0.946      0.889      0.731
```

Figure 3.4 Summary of model training using 20 epochs

3.2.9 Testing Flowchart

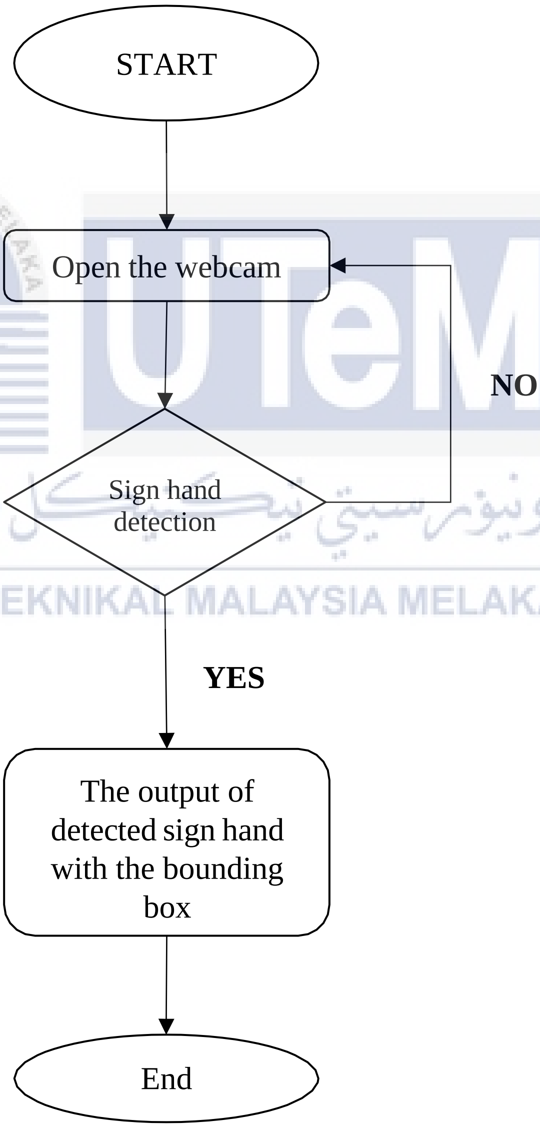


Figure 3.5 Testing flowchart of a sign language interpreter

Figure 3.4 shows the flowchart process of a sign language interpreter. This process start with opening the webcam using the code "python C:\Users\Nasha\PycharmProjects\yolov5ASL\yolov5\detect.py--weights C:\Users\Nasha\PycharmProjects\yolov5ASL\yolov5\runs\train\exp22\weights\best.pt --source 0 --data C:\Users\Nasha\PycharmProjects\yolov5ASL\yolov5\data\data.yaml --conf-thres 0.75" and then it will detect the sign hand. The output should be the sign hand of detected with bounding box and alphabets.

3.3 Equipment

The main component to build the system is Pycharm. This software will use a Pycharm for Python and Open CV. The hardware and software are used in the project the following:

Table 3.1 Hardware Tool

Hardware	Specification
Acer Laptop -3AVB7DEC	• 11th Gen Intel(R) Core(TM) i5-11400H @ 2.70GHz 2.69 GHz • RAM 8.00 GB (7.77 GB usable) • Graphic NVIDIA GEFORCE RTX

Table 3.2 Software Tool

Software	Function	Specification
Pycharm	• Python programming is done using the integrated development environment (IDE) PyCharm. In addition to supporting Django web	• 64-bit versions of Microsoft Windows 10, 8, 7 (SP1) • GB RAM minimum, 8 GB

	development, it offers code analysis, a graphical debugger, an integrated unit tester, integration with version control systems, and more. The Czech business JetBrains is the one that creates PyCharm.	RAM recommended. • 1.5 GB hard disk space + at least 1 GB for caches. • 1024 × 768 minimum screen resolution. • Python 3.7 or newer.
OpenCV	• A computer vision and machine learning software library called OpenCV is available for free use. The primary areas of focus are image processing, video capture, and analysis, which include tools for object and face detection.	• 200MB • Python,Java ,C++, and C • Windows, Linux, Mac OS, iOS, and Android.
Matplotlib	• A Python library for 2D data visualization is called Matplotlib. It offers Python creation	• 3.2.2 version

	tools for a wide variety of static, animated, and interactive visualizations.	
PyTorch	• Python and the Torch library are thefoundation of PyTorch, an open-source machinelearning (ML) framework.	• 1.12.1 version

3.4 Summary

The major goal of this project is to develop a vision-based system for recognizing spelled characters in American Sign Language (ASL). The fact that it provides a straightforward and accurate means to interact between a human and a computer is one of the reasons for choosing a vision-based system. This project demonstrates how the theories of image segmentation and the image acquisition system may be applied to the development of a sign language interpreter using Python and OpenCV. The dataset for this project was taken from the Roboflow Universe website, and it contains more than 1500 images of sign languages, divided into three datasets: training validation and testing.

3.5 Gantt Chart

Table 3.3 Gantt Chart for PSM 1

PROJECT PLANNING (PSM 1)														
Listdown the main activity for the project proposal. State the time frame needed for each activity.														
	2022													
Project Activity	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**
BDP 1 Briefing by JK, PSM, FTKEE														
Ttile discussion														
Progress Literature Review														
Research for articles and journals														
Submission Chapter 2: Literarure Review	**BP Briefing**							**Mid Term Break**						
Research for articles and journals														
50% finish Chapter 3: Methodology														
Submission Chapter 3: Methodology														
Write up Chapter 1 : Introduction														
Submission Chapter 1: Introduction														
Finish Write up Chapter 4														
Submit Report to Panel and Supervisor														
PSM1 Presentation														

Table 3.4 Gantt Chart for PSM 2

PROJECT PLANNING (PSM 2)														
Listdown the main activity for the project proposal. State the time frame needed for each activity.														
	2022													
Project Activity	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**
BDP 2 Briefing by JK, PSM, FTKEE	**BP Briefing**							**Mid Term Break**						
Research Project														
Meeting with Supervisor														
Fix the error on the pycharm														
Train the system using 1 epoch														
Train and test the system using 20 epochs														
Train and Test the system using 10 epoch														
Train and Test the system using 30 epoch														
Train and Test the system using 25 epoch														
Editing report														
Train and Test the system using 50 epoch														
Thesis draft submission to supervisor														
Testing & analysis														
PSM2 Documentation														
PSM2 Presentation														

CHAPTER 4

RESULTS

4.1 Introduction

In this chapter, the analysis and outcomes based on the system from software will be discussed. Before reviewing the results, testing and training are essential to reveal any problems and facilitate troubleshooting. Many well-known image processing libraries are used in the approach prototype, including OpenCV (Open Source Vision). The YOLOv5 model will be trained and validated using the dataset for this project, which was acquired from the Roboflow Universe website and contains roughly 4589 images of training data, 340 images of validation data, and 72 images of testing data.

4.2 Training, Validation, and Testing Result

The outcome of training with epochs is that the model improves in accuracy as more epochs are applied. It's essential to keep in mind that after a certain number of epochs, the accuracy and precision may start to decrease. Additionally, training the model for an excessively long time may cause it to become overfit, which prevents it from generalizing successfully to new samples. Figure 4.1 shows that the system had been trained by using 20 epochs taken for about 20.786 hours and 50 epochs taken for around 60 hours.

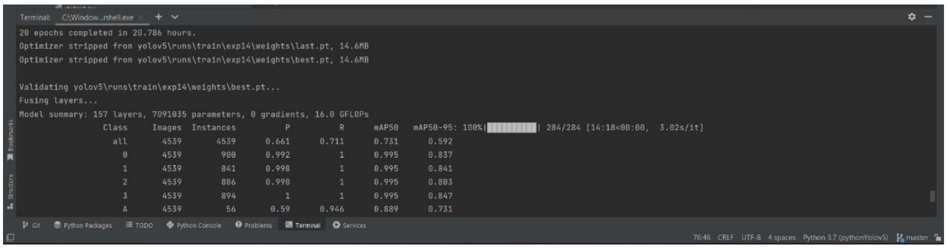


Figure 4.1 Train using 20 epochs

Figure 4.2 Output of training batch

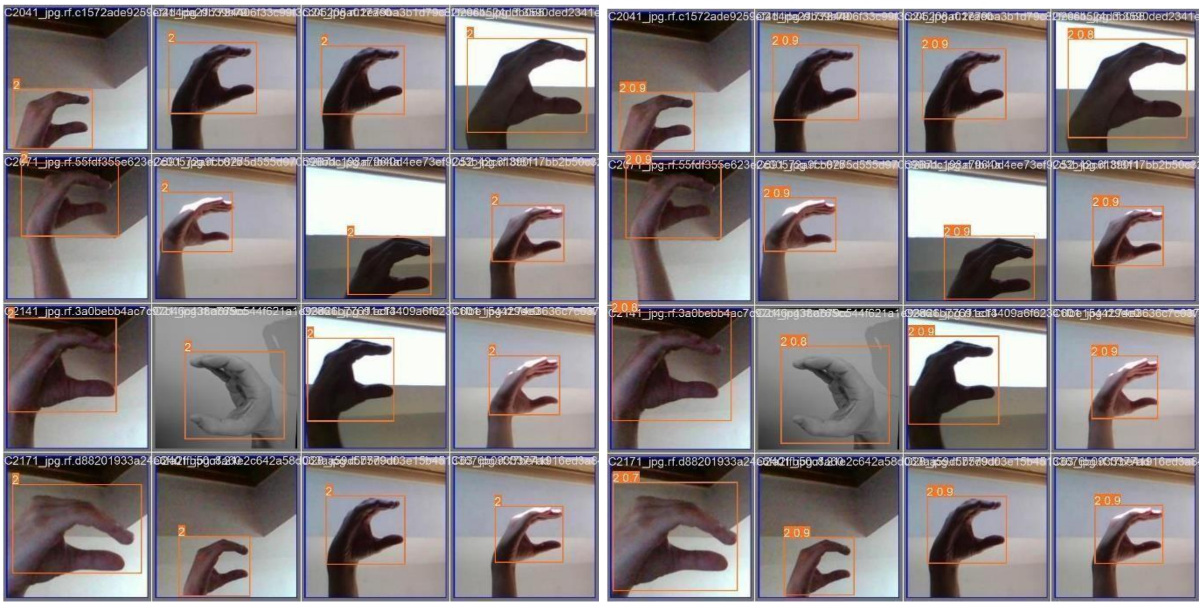


Figure 4.3 (a) validation label and (b) validation prediction

Figure 4.2 represents the training batch, which was utilized to update the model's weight. The size of the training batch may have an impact on both the model's performance and the training's pace. According to the loss function that is calculated for each training batch, the system will update the model's weight. The validation label and validation prediction process, which follows after model training and involves comparing the trained model with a test set of data, are shown in Figure 4.3. The label by each image in the validation dataset is indeed contained in the validation label. The model's predictions for each case are part of the validation prediction.

4.2.1 Loss Function of Training Model

A loss function is a comparison function between the desired and expected output values. We try to reduce this difference in output between the predicted and the target during training. Box loss, object loss, and class loss are the three loss functions that have been defined in this project for use in both training and validation.

Box loss function: Measurement of the size of the discrepancy between the bounding box of an object predicted by a model and the actual bounding box.

Object loss function: Calculates the difference between the model's output predictions and the real ground truth labels connected to the discovered object in the images. The precision increases as the loss value decrease.

Class loss function: A measure of how far a model's estimates depart from the actual ground-truth labeling of an image's main subject.

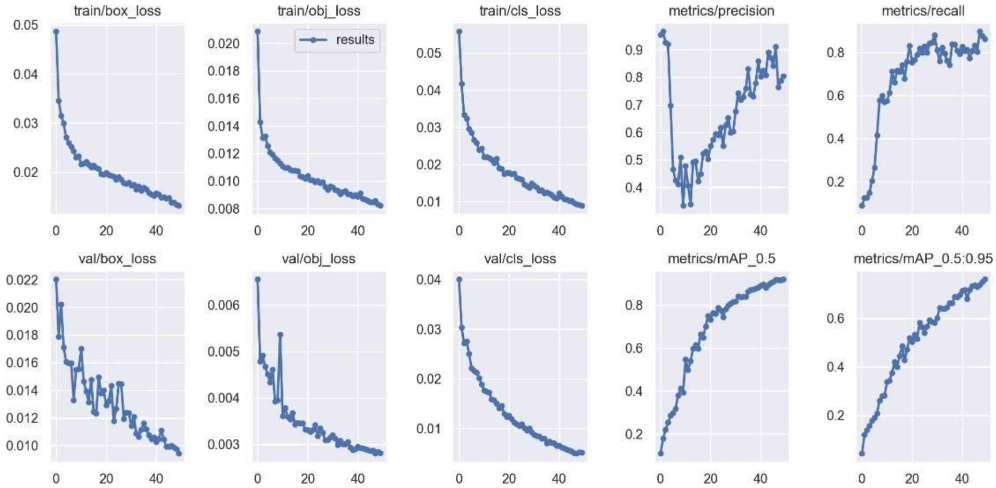


Figure 4.4 Loss Function of Training using 50 Epoch

According to Figure 4.4, the loss function value's bar chart indicates a tendency to decrease during training and validation. For the training process, the value of box loss function is 0.013194 and the value of object loss function and class loss function are 0.008218 and 0.008944. For the validation process, the value of box loss function and object loss function are 0.009422 and 0.002815. For value of class loss function is 0.005158. But the worth of recall and precision increased significantly.

4.2.2 Evaluation of Trained Model

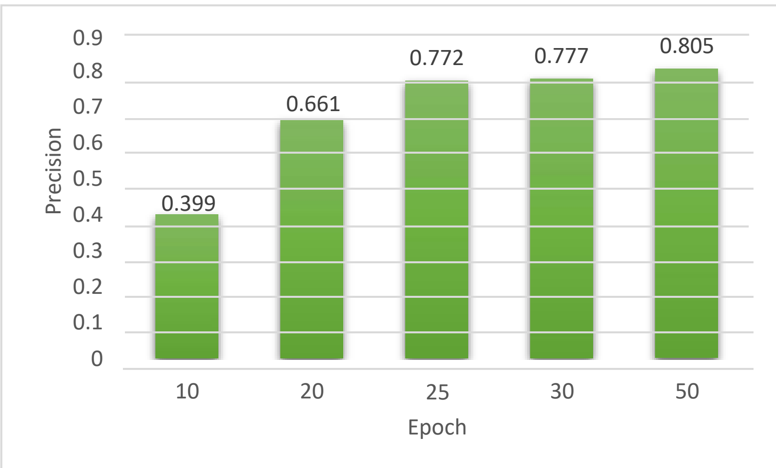


Figure 4.5 Precision Bar Chart

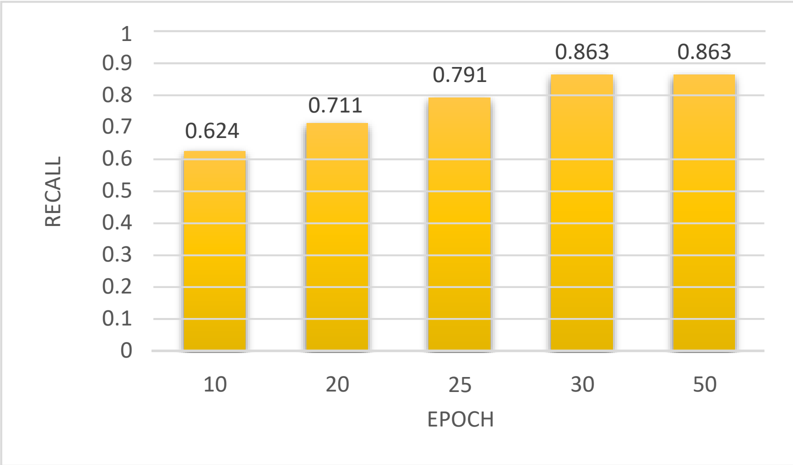


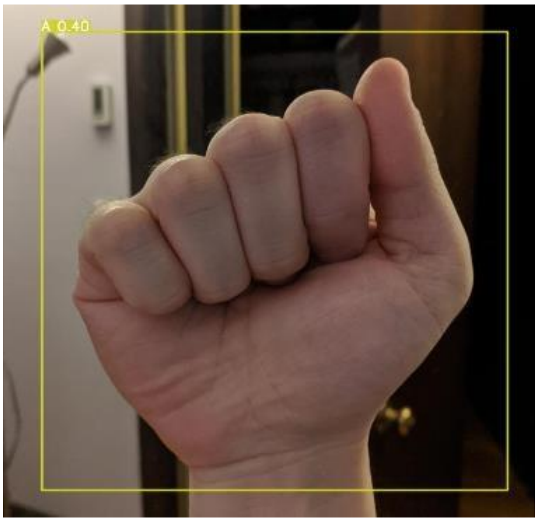
Figure 4.6 Recall Bar Chart

Figure 4.5 and 4.6 show the bar chart of precision and recall based on the epoch of 10, 20, 25, 30, and 50. Precision is to the model's capacity to correctly locate and identify items in an image. It is a percentage-based measurement of the model's precision in identifying and locating objects. High accuracy indicates that the model can reliably locate objects inside the image and has a low rate of false positives. Recall is the model's capacity to identify every instance of an object present in an image. It is a percentage-based indicator of how accurately the model has identified things. High recall indicates that the model is able to identify every occurrence of an object in the image and has a low rate of false negatives. As you can see, in epoch 10, the precision value is 0.399 and the recall value is 0.624, showing that the precision value is lower. Next, at the 20 epochs, the value of precision and recall is 0.661 and 0.711 while 0.772 is the value of precision while the recall value is 0.791 at epoch 25. At epoch 30, the value of precision is 0.777 and the value of recall is 0.791. Finally, at epoch 50, the precision value is 0.805 and the recall value is 0.863. An ideal system is one that has a high recall value and can identify a variety of hand signals while identifying boundary boxes.

4.2.3 Testing Result

The trained YOLOv5 model will be tested using around 340 images out of 4589 images in the testing dataset. The bounding box output value can be used to assess how effectively the trained model is performing. The process, which begins with the testing images as input, is complete when the system reaches the final image. The output should be images of sign-hand detections with bounding boxes.

Table 4.1 Table of a different epoch using the testing dataset

Epoch	**Output**
10	
20	

25	
30	
50	

According to Table 4.1, the output image can be seen at epoch 30, where the confidence level was lower than it is at epoch 50. When confidence level was close to 1.0, the system with the most accurate result is 0.93 at epoch 50.

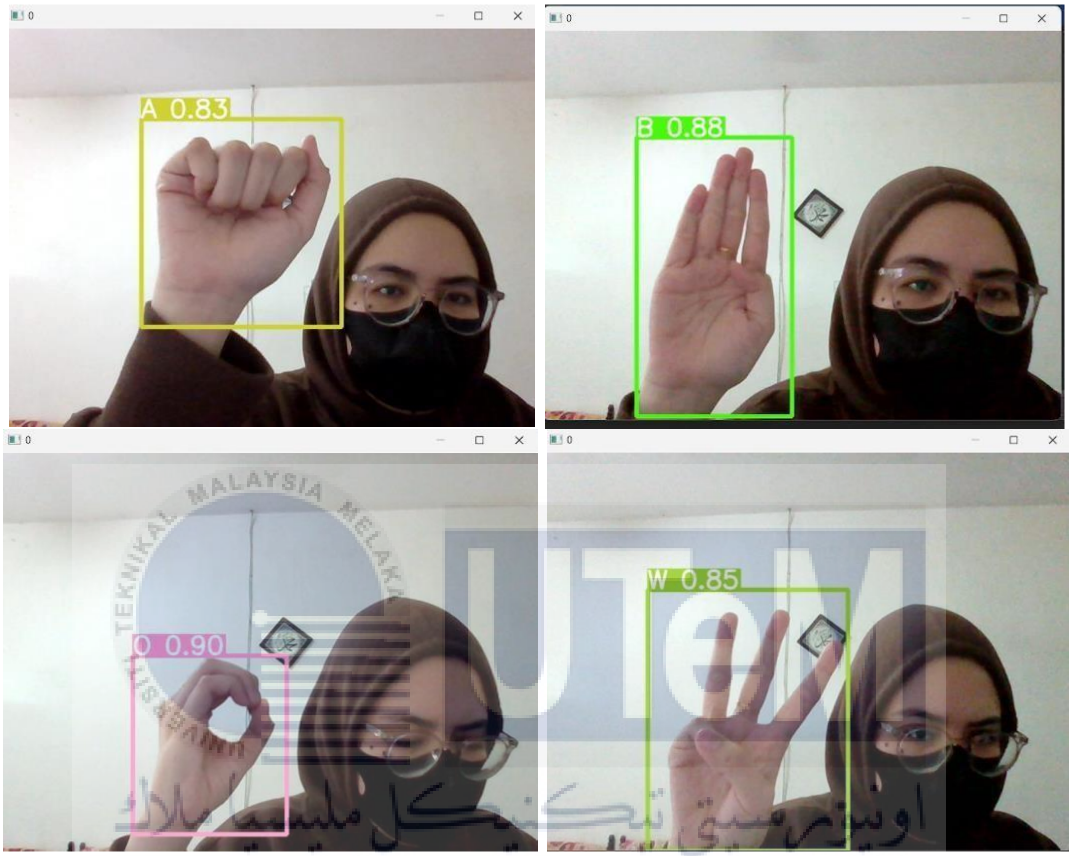


Figure 4.7 Test using a webcam

.

Figure 4.7 shows the result using a webcam, where the confidence level of A is 0.83, B is 0.88, O is 0.90 and W is 0.85 at epoch 50, where the confidence level is nearing the value of 1.0.

4.3 Summary

This chapter presents the outcome of the system and analysis for the development of sign language interpreters when the model was trained based on 50 epochs, recall and precision are respectively 80% and 86% where the system produced a high confidence level. We identified the sign hand using OpenCV and Python in the PyCharm software. The system is able to accuratelytrained classes that were as collected from the webcam and label their position with the bounding box by using the YOLOv5 object identification model. The goal is to provide society with a tool that will make it easier for deaf and mute people to communicate with each other by utilizing this system.

CHAPTER 5

CONCLUSION

5.1 Conclusion

The purpose of this study was to look into sign language recognition, which will be valuable in a variety of technological breakthroughs, and a lot of research has been done to assist deaf and dumb individuals. Deep learning and computer vision can also be utilized to help with the cause. This can be extremely useful for deaf and dumb people in interacting with others, as understanding sign language is not something that everyone has. Furthermore, this can be extended to building automatic editors, where a person can easily write using only their hand gestures. Based on the project's results, it can be concluded that the theories of hand segmentation and the hand detection system may be used to construct sign language recognition using computer vision. To sum it up, this system has achieved the following project objective: (1) To recognize sign alphabets from the American Sign Language type using the Yolov5 model computer vision technique and (2) to provide a real-time interface that allows a normal hearing-impaired person to communicate with a normal hearing-impaired person. We develop a sign detector in this sign language recognition project that detects the alphabets A to Z. Using the PyCharm IDE, we created this software project. The photos will be captured withsimple code that uses OpenCV to open a webcam. YOLOv5 was used to recognize sign languages, and Python was used to detect language.

5.2 Future Recommendation

For future works, the accuracy of the development of sign language interpreters using computer vision techniques could be enhanced as follows:

1) Plan to look at a larger dataset or a more in-depth area of sign language.
1) To improve the effectiveness of translating sign language into text or speech.
2) Creating real-time sign language recognition systems, such as by processing sign language recognition on a smartphone using edge-based computing.

5.3 Project Potential

This system has bright potential to be implemented as an early tool for hearing-impaired students to on American Sign Language (ASL). By providing real-time translation of sign language into spoken or written language, it would be easier for people who use sign language as their primary means of communication to communicate with those who do not understand it. The following are some possible uses for this technology:

- Real-time translation of sign language during meetings, lectures, and presentations to increase accessibility in academic and professional settings.
- Enabling communication between medical professionals and people who use sign language in hospital settings.

REFERENCES

Apoorv, S., Kumar Bhowmick, S. and Sakthi Prabha, R. (2020) ‘Indian sign language interpreter using image processing and machine learning’, *IOP Conference Series: Materials Science and Engineering*, 872(1), pp. 1907–1909. doi:10.1088/1757-899X/872/1/012026.

Bantupalli, K. (2018) ‘American Sign Language Recognition Using Machine Learning and Computer Vision’.

Dhavale, P. (2019) ‘Sign Language Recognition using Machine Learning Approach’, *International Journal for Research in Applied Science and Engineering Technology*, 7(4), pp. 853–861. doi:10.22214/ijraset.2019.4154.

Dogra, A., Malik, K. and Chowdary, V. (2018) ‘Sign Language Interpreter’, *International Journal of Engineering & Technology*, 7(3.12), p. 990. doi:10.14419/ijet.v7i3.12.17619.

Gomase, K. *et al.* (2022) ‘Sign Language Recognition using Mediapipe’, *International Research Journal of Engineering and Technology (IRJET)*, 9(1), pp. 744–746. Available at: https://www.irjet.net/archives/V9/i1/IRJET-V9I1133.pdf.

Goyal, E.K. and Singh, A. (2014) ‘Indian Sign Language Recognition System for Differently-able People’, *Journal on Today’s Ideas - Tomorrow’s Technologies*, 2(2), pp. 145–151. doi:10.15415/jotitt.2014.22011.

Halder, A. and Tayade, A. (2021) ‘Real-time Vernacular Sign Language Recognition using MediaPipe and Machine Learning’, *International Journal of Research Publication and Reviews*, (2), pp. 9–17. Available at: www.ijrpr.com.

Khan, R.U. *et al.* (2021) ‘Convolutional-Based Attention Module with Residual Network’, 2021.

Patil, P.R. *et al.* (2019) ‘Deaf Communicator : - A Python based ASL tool’, VIII(Iv), pp. 1100–1106.

S, Shivashankara and S, Srinath (2018) ‘American Sign Language Recognition System: An

Optimal Approach', *International Journal of Image, Graphics and Signal Processing*, 10(8), pp. 18–30. doi:10.5815/ijigsp.2018.08.03.

Shivashankara, S. and Srinath, S. (2019) 'An american sign language recognition system using bounding box and palm FEATURES extraction techniques', *International Journal of Recent Technology and Engineering*, 7(4), pp. 492–505.

Shrenika, S. and Madhu Bala, M. (2020) 'Sign Language Recognition Using Template Matching Technique', *2020 International Conference on Computer Science, Engineering and Applications, ICCSEA 2020*, pp. 5–9. doi:10.1109/ICCSEA49143.2020.9132899.

Shukor, A.Z. *et al.* (2015) 'A New Data Glove Approach for Malaysian Sign Language Detection', *Procedia Computer Science*, 76(Iris), pp. 60–67. doi:10.1016/j.procs.2015.12.276.

Sood, D. (2022) 'Sign Language Recognition using Deep Learning', *International Journal for Research in Applied Science and Engineering Technology*, 10(3), pp. 246–249. doi:10.22214/ijraset.2022.40627.

Triyono, L. *et al.* (2018) 'Sign Language Translator Application Using OpenCV', *IOP Conference Series: Materials Science and Engineering*, 333(1). doi:10.1088/1757-899X/333/1/012109.