Interim Report: Unified ASL and Emotion Recognition System

COMP3025

Individual Dissertation Single Honours

FYP Supervisor:

Yasir Hafeez

yasir.hafeez@nottingham.edu.my

FYP Student:

20398888

Mohammed Ismail Abusirdaneh

hfyma13@nottingham.edu.my

30th December, 2024

Acknowledgment

I am deeply grateful to my FYP supervisor, Dr. Yasir Hafeez, for his support, guidance, and encouragement during this project. His contribution, through insight and experience, has been helpful to shape this research in order to overcome the various problems that arose during its preparation. I am also thankful for his constructive feedback and the valuable critiques he offered, which played a very important role in refining the objectives and methodology of the project.

I would also like to thank my colleagues, my family, and my institution for their support and facilities that allowed me to work on this project

Abstract

The integration of gesture and emotion recognition technologies holds the capability to bridge major communication gaps, therefore benefiting especially those with speech or hearing impairments. The project deals with the integration of a system that performs both ASL and facial emotion detection using state-of-the-art frameworks for hand gesture tracking and CNNs for facial emotion recognition. It also intends to provide real-time performance for ensuring smooth interaction in dynamic environments.

This paper highlights certain challenging factors that lead to recognition: variability in the gesture style, subtle facial expression for emotion detection, and real-time optimization in a resource-constrained environment. Training datasets for the various ASL gestures were implemented through custom-made datasets; emotion detection used the FER-2013 dataset. The integration will also make use of lightweight and efficient frameworks so that compatibility is attained within real-time systems.

Preliminary results indicate the accurate detection of both gestures and emotions under controlled conditions, while the scalability for real-world applications like education, accessibility, and augmented reality is yet possible.

Table of Contents

Introduction

Communication skills form a cornerstone in human life and a basis for relations, society, and development; however, for millions across the globe with hearing or speech impairments, societal barriers severely limit their ability to connect with others. Sign language has emerged as the transformative medium to bridge this gap, offering a visual-spatial linguistic system where the impaired individuals in terms of hearing or speech can fully express themselves. However, for most, the major problem is that this is not widely understood and fluent enough. This is perpetuated by the inability to convey and grasp the subtlety of emotions communicated by signing, which creates a profound gap in mutual understanding and connection.

Technology, in the last couple of decades, has striven to address these challenges. With the introduction of gesture and emotion-recognizing systems, computer vision, machine learning, and AI also play their role in connecting communication for inclusivity, as these will be able to implement real-time interpretation of sign language and the emotions forming human interaction. Despite such bright prospects, many of the existing systems fall short of these expectations. They also do not integrate information, on which their recognition of gestures is based, and the expression of emotions, which is critical communication. Furthermore, technical challenges in real-time processing of information and adapting to environmental variability (e.g., lighting conditions and user positioning) outline the needed necessity for further development of a practical solution.

The following report discusses and explores the development of a single unified system that would incorporate both American Sign Language (ASL) recognition and facial emotion recognition (FER). Therefore, in this research, using state-of-the-art gesture recognition frameworks, like MediaPipe, alongside deep learning-based approaches to detect facial emotion using convolutional neural networks (CNN) to contribute to solving issues of accessibility and communication. It also examines the social, historical, and technological contexts in which this work is particularly important, aside from focusing on the technical development of a dual recognition system.

1.1 Historical Background

American Sign Language in the United States stems from a blend of French Sign Language and the indigenous sign systems of the American Deaf community. Eventually, ASL evolved to be one of the most widely used and complex signed languages in the world (see Fig. 1 in the Appendix).

The scientific community started to investigate the possibility of automating sign language interpretation in the 1990s. Early systems used simple pattern recognition strategies that were not very accurate or scalable. The rise of machine learning and deep learning in the 2010s led to

even more accurate systems. Among the most prominent breakthroughs for tracking real-time gestures is Mediapipe by Google.

1.2 The Emotional Context of Communication

Emotion is an important form of communication; it may expose the intentions and emotional states that are not captured in the signal. Human communication is heavily reliant on facial cues, tone cues, and non-verbal cues. Since the publication of Charles Darwin's landmark book on emotional evolution, The Expression of the Emotions in Man and Animals (1872), emotions are understood as both universal and subtle, shaped by individual and cultural contexts. Computer vision and machine learning are recent research domains that play a significant role in training FER systems to classify emotions such as happiness, sadness, anger, and surprise. Specifically, CNN has played a key role in improving FER systems by enabling a wide range of applications, from analyzing customer experiences to monitoring mental health by extracting features from the human face.

1.2.1 Classification of Facial Expressions

Facial expressions are a key aspect of emotional signaling and are carefully studied in psychology and technology. A FER system usually involves face detection, facial expression analysis, and emotional classification to identify a target human emotion. The facial expressions serve to be building blocks of automated emotion recognition, providing real-time interaction with human emotions. Several examples of key emotions consist of happiness, with a smile and upward cheek muscles; anger, marked by furrowed brows and tightened lips; sadness, characterized by downward lip corners and lost eye focus; a neutral expression, often subtle with minimal facial movement; and surprise, identified by raised eyebrows and an open mouth (see Fig. 2 below).

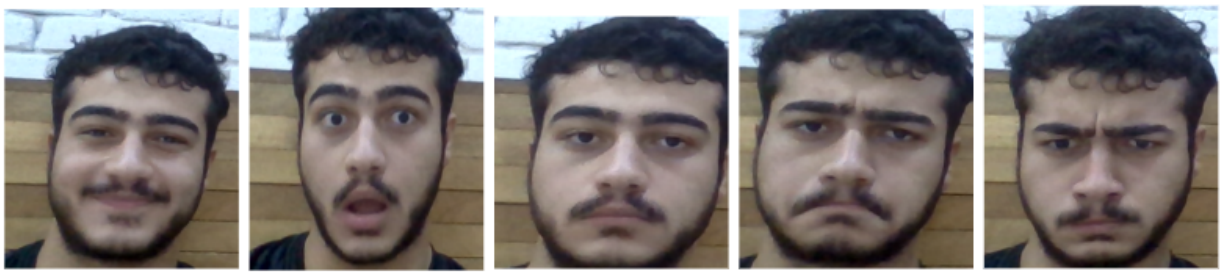


Fig. 2 Facial Expressions (happiness, surprise, neutral, sadness, anger)

1.3 Integration Challenges

Despite individual advancements in sign language recognition and emotion detection, their combination into a unified system remains underexplored. The unique challenges entailed are that with the complexity of hand gestures, facial expressions, and contextual variability, overlap in gestures or incomplete visibility of hands is common in regular gesture recognition systems,

whereas variations in lighting, occlusions, and subtlety in the way emotions are expressed are common for FER systems. A lack of integrated systems also raises very serious questions concerning the inclusivity and accessibility of current technological solutions. This research tries to bridge the gaps through the development of a unified approach to communication, effectively integrating ASL recognition with emotion detection to overcome the limitations of existing systems.

1.4 The Role of Advanced Technologies

The proposed system uses Mediapipe for gesture recognition and CNN-based frameworks for emotion detection. For a wide range of ASL recognitions, the Mediapipe solution depends on a trained hand tracking and landmark detection model. This is lightweight in terms of architecture, cross-platform in nature, and perfect for real-time applications. On the other hand, the CNNs are used with their hierarchical feature extraction power to classify facial expressions into discreet emotional categories. Integrating these technologies into one pipeline not only creates more functionality but also provides insight into how gestures and emotions interact.

Motivation

This research is motivated by societal, technological, and personal factors. For most people who have complete linguistic and sensory capabilities, communication is a rather unconscious process. However, the process of communication can be fraught with barriers for people with impaired hearing or speech, limiting their social interactions, access to education, and professional advancement. This work is motivated by an attempt to overcome these barriers by exploiting modern technology in devising a system that bridges the communicative gap between the Deaf and hearing communities.

2.1 Societal Need for Inclusive Communication

Globally, there are close to 466 million people experiencing disabling hearing loss worldwide who use sign language as part of daily communication. To this date, very few children learn any sort of formal signing methodically, hence limiting this section of accessibility and integration. This will tend to keep the communicators away from mainstream circulation and, consequently, be restricted to partial participation within society.

Facial expressions and emotional undertones provide context and depth to conversations, which help participants connect on a more meaningful level. In the absence of tools that recognize and convey these emotions alongside gestures, communication can feel mechanical and incomplete. This research is motivated by the need to create a communication system that recognizes not just the gestures but the emotions of the gestures that would further facilitate deeper empathy and understanding.

2.2 Improvements in AI and Their Potential

The motivation to leverage these technologies is their demonstrated success in addressing isolated challenges. This research integrates hand tracking from Mediapipe with CNN-based emotion recognition to present a system that combines the strong points of both approaches. Such a system has the potential to redefine how technology can support inclusive communication.

2.3 Future Implications

The applications will go well beyond the improvement of Deaf communication and include immediate feedback for sign language students, enhanced access in customer service, healthcare, and legal proceedings. An integrated system for gesture and emotion recognition leads to new visions for human computer interaction in areas like virtual reality.

Related Work

3.1 Sign Language Recognition (SLR)

Sign language recognition has become one of the essential sub-areas in developing assistive technologies. Previous works were based on handcrafted features and traditional algorithms; for example, Starner and Pentland suggested Hidden Markov Models for recognizing gestures from visual input. Although these approaches were innovative, they could not face the real challenges, such as the great variability in signing styles or complex backgrounds.

The advent of deep learning has significantly transformed this field. CNNs have manifested much better accuracy and adaptability, especially when combined with their 3D extensions and RNNs. Recently, an example has been to analyze the dynamic hand gestures using spatiotemporal data employing 3D CNNs combined with RNNs by Molchanov et al. This resulted in state-of-the-art performances. Recently, real-time gesture recognition performance was enhanced by many frameworks, such as Google's Mediapipe, via offering accurate hand tracking. The studies such as [1] show that the integration of Mediapipe with CNN-based classifiers can enable detection of static gestures.

3.2 Facial Emotion Recognition (FER)

With the advancement in deep learning technologies, the facial emotion recognition has also evolved. Traditional systems of FER are based on feature extraction methods comprising Local Binary Patterns (LBPs) and Histogram of Oriented Gradients (HOGs), which were effective only for basic emotion recognition and failed to be robust across more complex expressions

Deep learning-based FER systems have utilized CNNs, such as those by Tang et al., to extract hierarchical features of faces with high accuracy on datasets like FER-2013 (see Fig. 3 in the

Appendix). Hybrid architectures that combined CNNs with Long Short-Term Memory networks were explored to improve temporal emotion detection in video sequences. For example, the researchers at Microsoft combined the facial action units with their temporal dynamics to classify emotion with improved accuracy in naturalistic settings. These are now lightweight and can operate in real-time with frameworks like TensorFlow Lite, enabling their viable use in edge devices for healthcare and human-computer interaction applications [2].

3.3 Integrated Gesture and Emotion Recognition

M. N. Anwar et al. demonstrated the possible applications of gesture and emotion recognition systems in augmented reality by combining Mediapipe and CNN-based emotion models. However, these systems had limitations to specially defined situations and lacked the flexibility required to handle real-world unpredictability [3]. Recent works focus on the need for systems to integrate contextual information. For example, frameworks using Mediapipe for gesture tracking combined with emotion classifiers trained on AffectNet have shown potential for scalability and naturalistic applications. However, high computational complexity remains one of the major challenges that face real-time performance with those approaches, especially in resource-constrained environments [4].

Methodology

The proposed methodology of the project represents a step-by-step process for developing a real-time integrated system for SLR and FER. It ensures the use of enhanced technologies, and state-of-the-art neural network architectures are used. Further, this section elaborates on the components: data acquisition, preprocessing, model training, and integration of subsystems.

The development process was split into two major pipelines: the Sign Language Recognition pipeline, powered by the hand tracking functionality of Mediapipe, and the Facial Emotion Recognition pipeline, which is based on a convolutional neural network for emotion classification. This work integrates these two modules into one integral system that may provide detailed insight into human communication.

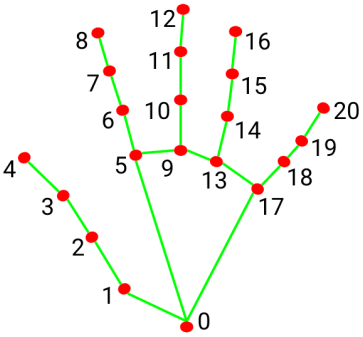
4.1 Sign Language Recognition

Sign Language Recognition uses the Mediapipe Hands framework for real-time tracking of the hands and landmark detection. The algorithm by Mediapipe detects 21 landmarks in every hand, thus locating joint positions accurately. These form a 2D representative of the different gestures by the responsibility of feature extraction (see Fig. 4 below)

First, some preprocessing was applied to the landmark data in order to make them more meaningful. The coordinates of each landmark were normalized with respect to the wrist for normalization regarding hand size, position, and orientation not to impact the gesture

classification process. These normalized coordinates were then converted into feature vectors that could be fed into a machine learning model for classification (see Fig. 5 below).

It consisted of convolutional layers to extract the spatial patterns of hand landmarks and dense layers mapping those patterns to predefined gesture classes. The network was trained on a highly varied dataset of ASL gestures, augmented by changes in lighting conditions and positioning of the hand. In this example, the training process used cross-entropy as a loss function and the Adam optimizer for faster convergence. The performance was evaluated by the metrics of accuracy and precision.



0. WRIST
1. THUMB_CMC
2. THUMB_MCP
3. THUMB_IP
4. THUMB_TIP
5. INDEX_FINGER_MCP
6. INDEX_FINGER_PIP
7. INDEX_FINGER_DIP
8. INDEX_FINGER_TIP
9. MIDDLE_FINGER_MCP
10. MIDDLE_FINGER_PIP
11. MIDDLE_FINGER_DIP
12. MIDDLE_FINGER_TIP
13. RING_FINGER_MCP
14. RING_FINGER_PIP
15. RING_FINGER_DIP
16. RING_FINGER_TIP
17. PINKY_MCP
18. PINKY_PIP
19. PINKY_DIP
20. PINKY_TIP

Fig. 4 21 key points for wrist and joints of the thumb, index, middle, ring, and pinky.

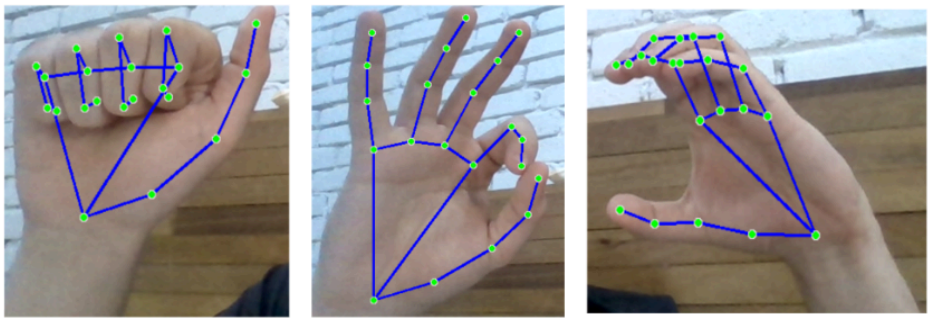


Fig. 5 Visualization of the hand landmarks using Mediapipe for ASL letters "A", "F", and "C" showcasing the 21 detected keypoints.

4.2 Facial Emotion Recognition

For that reason, face detection in the FER pipeline used OpenCV's Haarcascade Classifier to make sure only the region of interest (the face) was being treated for emotion recognition (see Fig. 6 below). For this reason, the face is resized to a 48x48 pixel grayscale image because that is what the neural network takes as input.
The neural network for emotion recognition was a CNN that had been trained on the FER-2013 dataset. The FER-2013 is a large dataset of facial expressions with five categories of emotions, including angry, happy, neutral, sad, and surprised. It contained convolutional layers performing feature extraction, pooling layers to reduce dimensions, and dense layers that carry out classification. Transfer learning was applied to enhance the accuracy of the network, since real-world conditions are rarely perfect and differ from the training data. This model was

intensively trained with augmented data in order to simulate changes in facial expressions due to lighting, occlusion, or head tilt.

The FER pipeline was integrated with the mechanism to handle multiple face detections in a frame, each face to be processed independently and its corresponding emotion to be decided. This ensures the system will be able to accommodate group settings where multiple individuals may be present.

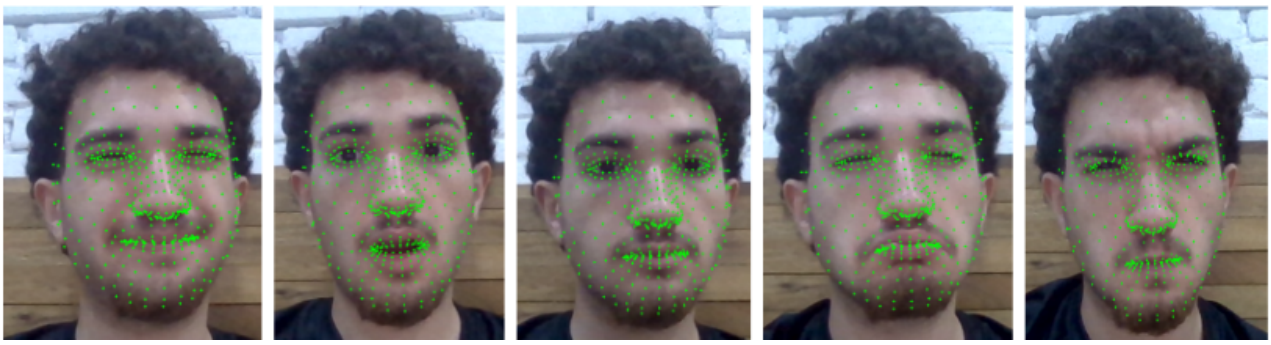


Fig. 6 Facial key landmark using Mediapipe (happiness, surprise, neutral, sadness, anger)

4.3 Challenges Encountered

The methodology showcased was functional; however, the process did present several challenges. Issues occurred when the system was trying to recognize gestures of overlapping hands or when there was a partially occluded face from poor lighting-the conditions Mediapipe could not recognize.. All these pointed toward a number of problems: which leads to the necessity of enhanced training datasets and preprocessing techniques.

Design

5.1 System Architecture

The system architecture was designed to integrate two major functionalities—American Sign Language (ASL) recognition and facial emotion detection—into a single, unified framework. The real-time processing starts with a video feed from a webcam, which serves as the input source. Then, various machine learning models and frameworks processed the video frames to fetch the required information. Further, hand landmarks were detected and tracked by using MediaPipe; the custom-trained KeyPoint Classifier for ASL recognition; face detection was performed by a Haar Cascade Classifier; and emotion detection has been carried out through CNN-based on the FER2013 dataset. All components are connected with each other to show smooth performance and representation of results (see Fig. 7 in the Appendix).

5.1.1 Integration of Pipelines

The SLR and FER pipelines had to be highly time-synchronized so that both outputs were corresponding to exactly the same moment in time. The video feed processed each frame, and with every processed frame, a timestamp was available, enabling the alignment of gestures with facial expressions. Such synchronization was particularly important in situations that call for combined interpretations, like identifying a happy gesture with a smiling face.

5.2 Hand Detection and ASL Recognition

The recognition system is based on MediaPipe Hands, a high-performance framework that identifies and tracks in real time 21 keypoint hand landmarks. It recognizes those landmarks along with fingers and the palm, making it possible to track every tiny motion of the hands accurately. These are preprocessed as normalized data and fed to the designed KeyPoint Classifier, which maps the gestures into letters. This modular design maintains the high accuracy even when using in real time (see Fig. 8 below).

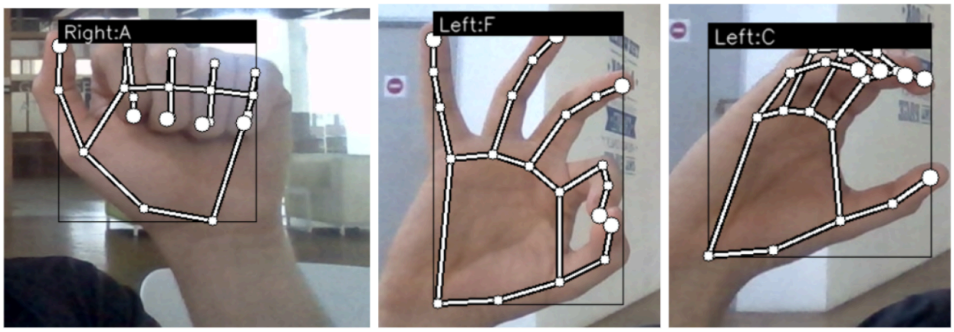


Fig. 8 Unified system's real-time output for ASL letters "A", "F", and "C"

5.3 Face Detection and Emotion Recognition

Haar Cascade was implemented for face detection and CNN for facial expression categorization. This Haar Cascades detect the face region from the video feed (see Fig. 9 below). A convolutional neural network trained on FER2013 database acts on the cropped face, thereby categorizing the expression as angry, happy, neutral, sad, or surprised. This framework provides reliable real-time emotion detection, important for the interpretation of non-verbal cues.

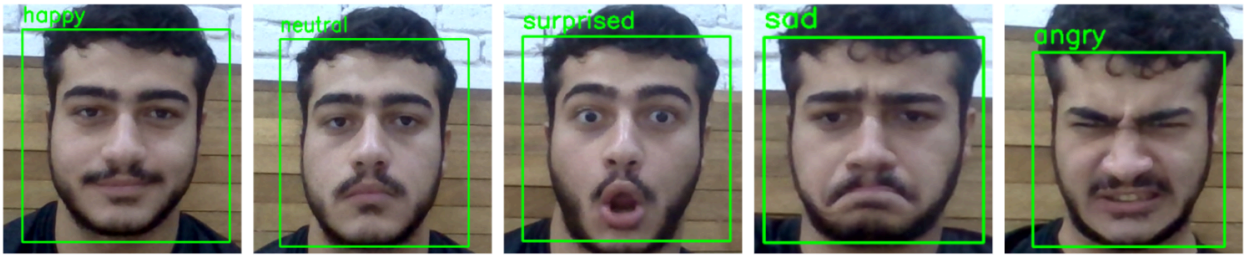


Fig. 9 Detected facial expression with emotion label

(happiness, neutral, surprise, sadness, anger)

5.4 Unified Output

A single frame is used to show the outputs from both ASL recognition and emotion detection. The integration to a unified system will allow to interpret the sign language gestures with their emotional context of communication at the same time.

This integrated system will be helpful in scenarios where both forms of communication are of paramount importance, such as teaching environments, accessibility tools, and therapeutic settings. The real-time video frame overlays the detected ASL letter and the recognized facial emotion for clear visual output (see Fig. 10 below).

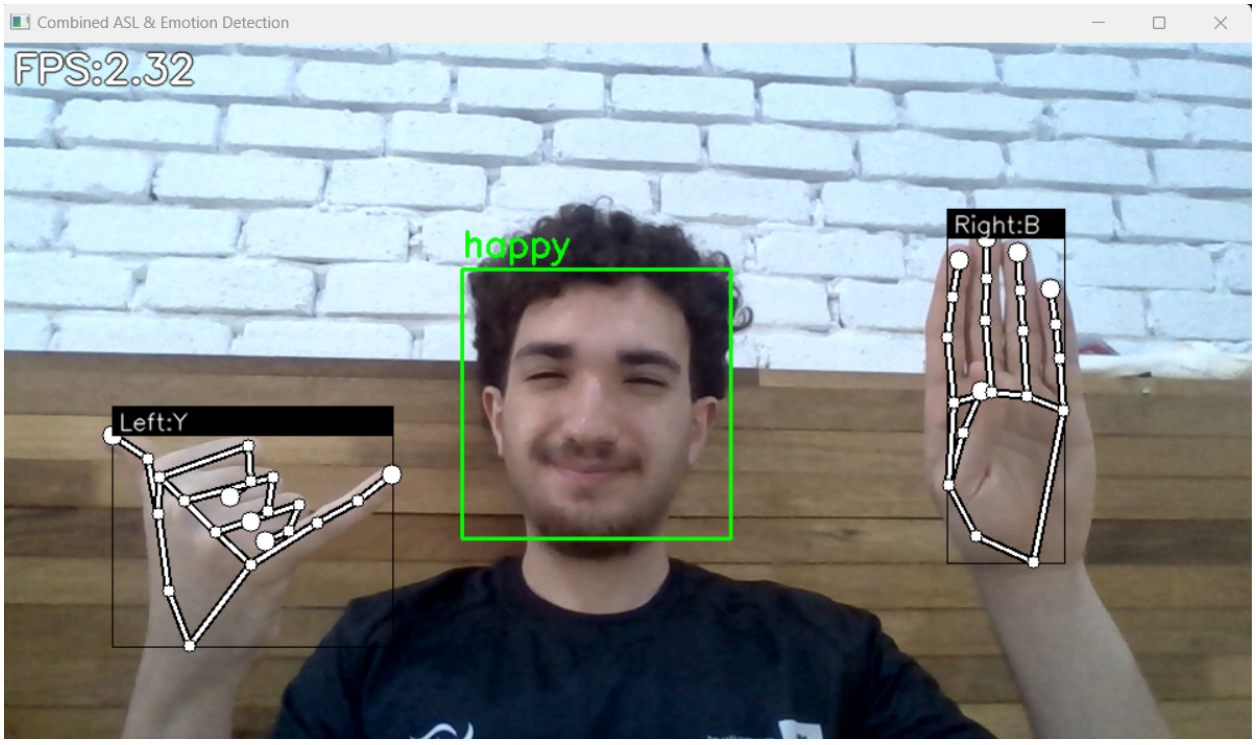


Fig. 10 Unified System Output: The system demonstrates simultaneous recognition of gestures and emotion. The left hand is identified as 'Y,' the right hand as 'B,' and the detected facial emotion is 'Happy,' showcasing the integration of ASL and emotion recognition in real-time.

5.5 Project Structure

The project structure follows the organization of code, models, and datasets which would be required for a unified system of real-time American Sign Language recognition and emotion detection. It should be structured in a modular fashion for ease of maintenance and future enhancements.

Project Directory

```
UnifiedSystemFYP/
│
├── app.py                          # Unified system implementation
├── haarcascade_frontalface_default.xml # Pre-trained Haar Cascade for face detection
├── utils/                          # Utilities for performance tracking
│   ├── cvfpscalc.py                # FPS calculation utility
│
├── model/
│   ├── keypoint_classifier/        # ASL keypoint classifier
│   │   ├── keypoint_classifier.py  # KeyPointClassifier implementation
│   │   ├── keypoint_classifier_label.csv # ASL gesture labels
│   │   ├── keypoint_classifier.keras # Pre-trained ASL model in Keras
│   │   ├── keypoint_classifier.tflite # TensorFlow Lite version of ASL model
│   │   ├── keypoint.csv            # Dataset of preprocessed keypoints ASL
│
├── Emotion-Recognition-main/
│   ├── emotion_detection_model.h5  # Pre-trained emotion classification CNN
│   ├── emotion_detection.py        # Implementation of emotion detection
├── notebooks/                      # Notebooks for training models
│   ├── keypoint_classification.ipynb # ASL classifier training
│   ├── emotion_classification.ipynb # Emotion classification training
│
└── requirements.txt                # Python dependencies
```

Implementation

6.1 Training the ASL Process

The unified system consists of two major training processes: the training of the ASL keypoint classifier and the training of the emotion detection CNN. Both models rely on supervised learning methods; hence, they require an accurately labeled dataset for high performance.

Classifier of ASL Keypoint: The training starts with creating a dataset. Hand landmarks for various ASL signs are tracked using the MediaPipe Hands framework, giving the exact coordinates of 21 key points on the hand. Information regarding these key points has been stored in structured form in the keypoint.csv file, where every row represents a class label followed by the normalized keypoint coordinates. This formatted dataset will have the model trained proficiently to identify multiple ASL signs.

The classifier is built using a dense neural network architecture implemented in TensorFlow/Keras. This network will take as input a vector of 42 inputs, namely, the x and y coordinates of the 21 key points derived from hand landmarks. With iterations of training, this network will be able to learn labeling gestures with high accuracy, thereby allowing the ability to interpret ASL gestures in real time.

6.1.1 Model Architecture

```
model = tf.keras.models.Sequential([
    tf.keras.layers.Input((21 * 2, )),  # Input layer
    tf.keras.layers.BatchNormalization(),
    tf.keras.layers.Dense(128, activation='mish', kernel_regularizer=tf.keras.regularizers.l2(0.01)),
    tf.keras.layers.Dropout(0.5),
    tf.keras.layers.Dense(64, activation='mish', kernel_regularizer=tf.keras.regularizers.l2(0.01)),
    tf.keras.layers.Dropout(0.5),
    tf.keras.layers.Dense(32, activation='mish', kernel_regularizer=tf.keras.regularizers.l2(0.01)),
    tf.keras.layers.Dense(NUM_CLASSES, activation='softmax')
])
```

6.1.2 Evaluation

This training and evaluation process helps to optimize the models for real time applications. Later, the dataset is divided into two subsets, 75% and 25%, for the purpose of training and testing, respectively. This allocation provides sufficient training data while conserving enough test sets for generalization.

The loss metric for the training process will be sparse categorical cross-entropy. This loss metric will help solve the problem of multi-class classification. The optimization factor chosen for effective convergence is the Adam optimizer. To keep the track of how the training has been going, use validation data to prevent overfitting by guiding an early stop if needed.

After training, the model is tested for correct classification for the inputs fed into it. While accuracy and the confusion matrix are in wide use, for this system, the need arises to observe the performance when tested in real scenarios. Thus, the trained model in TensorFlow is saved in.h5 format.

6.2 Training the Emotion Detection Model

The emotion detection model was built using the CNN on the FER2013 dataset. This preprocessed database, FER2013, includes grayscale images, each of size 48 × 48 pixels in size and labeled as one out of five categories: happiness, sadness, anger, surprise, and neutral.

The dataset preparation includes loading the images, normalizing the pixel intensity values in a range between 0 and 1, and reshaping them for the input of the CNN. Normalization enhances the training efficiency of the model by keeping the scaling of the pixel values consistent.

The model uses a CNN architecture for feature extraction and classification. The convolutional layers extract features such as edges and textures from the input images, which are processed into pooling layers. Extracted features further pass on to fully connected dense layers mapping them to certain classes of emotions. This has been optimized to provide the best balance between accuracy and computational efficiency, hence providing a real-time application.

6.2.1 Model Architecture

```
model = Sequential([
    Conv2D(32, (3, 3), activation='relu', input_shape=(48, 48, 1)),
    MaxPooling2D((2, 2)),
    Dropout(0.25),
    Conv2D(64, (3, 3), activation='relu'),
    MaxPooling2D((2, 2)),
    Dropout(0.25),
    Flatten(),
    Dense(128, activation='relu'),
    Dropout(0.5),
    Dense(7, activation='softmax')
])
```

6.2.2 FER Evaluation

In the process of training the model for emotion detection, it utilizes a loss function as categorical cross entropy combined with the Adam optimizer for efficiency in learning and convergence. Its performance was closely monitored to make sure that it produces high accuracy in classification while training until it successfully completed its training. The model is then saved to the file emotion_detection_model.h5, thus preparing the model to be easily applied to various real-life use cases or further distributed with ease.

Progress

The structured and well-planned timeline is being followed in the project, as reflected in the Gantt chart (see Fig. 11 in the Appendix). Each stage has further been subdivided into smaller tasks so that the project attains its milestones in a step-by-step manner.

7.1 Research Phase

Two important aspects under consideration during the research process are the exploration of the ASL recognition techniques and the study of the emotion recognition models. The project sets up a technical backbone with the search for datasets of emotion detection, such as FER2013 and those of ASL regarding the gesture recognition and evaluation of available frameworks like MediaPipe for hand tracking and TensorFlow for designing neural networks.

7.2 ASL Model Development

The development of the ASL model began with preprocessing and analyzing hand gesture datasets. MediaPipe extracts the key points and sends them to a trained classifier that can identify all 26 letters from the ASL alphabet. Then data augmentation was used to enhance the model to improve prediction accuracy

7.3 Emotion Detection Development

Emotion detection utilized the CNN model trained on FER2013 enhanced in a number of ways to pre-process input data, such as resizing and normalizing facial images. For face detection, a Haar Cascade classifier was applied, while real-time emotion classification was enabled by using TensorFlow. The FER system was integrated with the ASL recognition system in such a way that the frames process facial emotions along with hand gestures.

7.4 Future Expansion: Web Application Design

While the current system works as a stand-alone application, the future work involves developing a web-based platform to facilitate online learning, mentoring, and parental engagement. It integrates all the functionalities of the current system with a web-based conferencing feature, dashboards, and a database for storing the interaction of each user. Scaling and flexibility for this would need Docker containerization and management of databases.

7.5 Gantt Chart

The project is running as scheduled, with all phases until 2.3 successfully completed (see Fig. 11 in the Appendix). The third phase will develop and refine the contextual emotional model, CEM, while the fourth phase will design and implement a website portal that will be used for user interaction and system demonstration. These phases further the core system by making it smooth and user-friendly.

Contributions

The real-time ASL and emotion recognition system was successfully developed in this project using MediaPipe for hand tracking and a TensorFlow neural network for classifying ASL letters. The preprocessing pipeline aids the system to be consistent against varying conditions. Also, an emotion recognition system built with a CNN trained on FER2013 and Haar Cascade for face detection reliably classified emotions such as anger and happiness in real time.

Key contributions include optimizing the system for real-time performance, enabling the simultaneous processing of gestures and emotion detection in one video frame. Its lightweight models and FPS feedback mechanisms provided key elements for smooth operation. The architecture being modular further allows the scope for scalability, and the possibility of future features like e-learning.

Reflections

The entire project was a learning curve, with the realization that it is equally important to balance technical difficulty with system performance. Issues with real-time efficiency, integration of ASL, and emotion recognition pipelines were resolved through iterative testing and optimization. These highlight the ability to adapt while solving problems.

Personally, this work has deepened my knowledge and expertise in machine learning and computer vision but has highlighted the necessity for designing with the user in mind. The rewarding experience would be learning ASL as a way to effectively test the system. It will be not only the accurate way to validate the recognition model, but it will also foster my understanding of the language. The system will meet the initial objectives but also allow for future enhancements, such as real-time educational communication tools.

Bibliography

[1] P. Molchanov, X. Yang, S. Gupta, and J. Kautz, "Online Detection and Classification of Dynamic Hand Gestures with Recurrent 3D Convolutional Neural Networks," *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 8, no. 1, pp. 4207–4215, Jul. 2016. [Online]. Available: IEEE Xplore.

[2] W. Tang, S. H. Liu, and D. Huang, "Facial Emotion Recognition Based on Convolutional Neural Networks and Long Short-Term Memory Networks," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. 10, no. 2, pp. 45–56, Jun. 2021. [Online]. Available: IEEE Xplore.

[3] M. N. Anwar, A. F. Alenezi, and T. M. Shah, "Integrating gesture and emotion recognition systems for augmented reality applications using Mediapipe and deep learning," Journal of

Advanced Multimedia and Signal Processing, vol. 12, no. 2, pp. 78–85, Mar. 2023. [Online]. Available: ResearchGate.

[4] D. P. Sharma, S. Roy, and H. Singh, "Context-aware emotion and gesture recognition with Mediapipe and AffectNet: Challenges and advancements," *IEEE Transactions on Affective Computing*, vol. 14, no. 3, pp. 300–313, Sept. 2023. [Online]. Available: IEEE Xplore.

Appendix

Figure 1

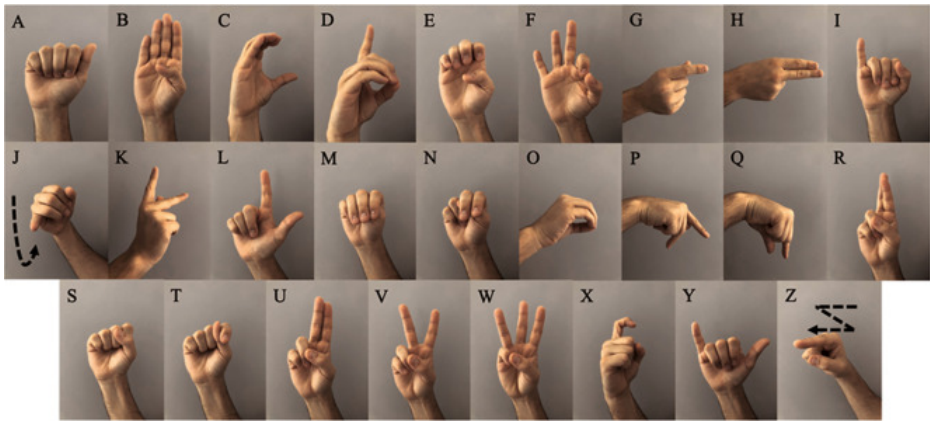


Fig. 1 The 26 Letters of the English Alphabet Represented in American Sign Language

Figure 2

Fig. 3 FER 2013 Dataset Images

Figure 7

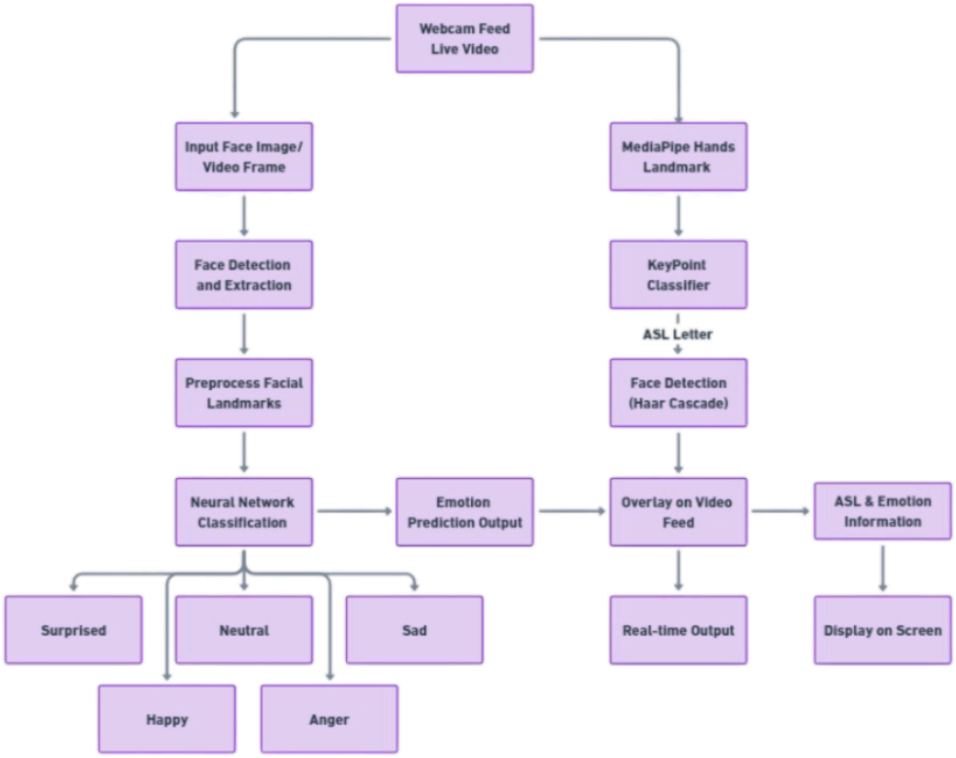


Fig. 7 A System Architecture flowchart illustrating unified system

Figure 11

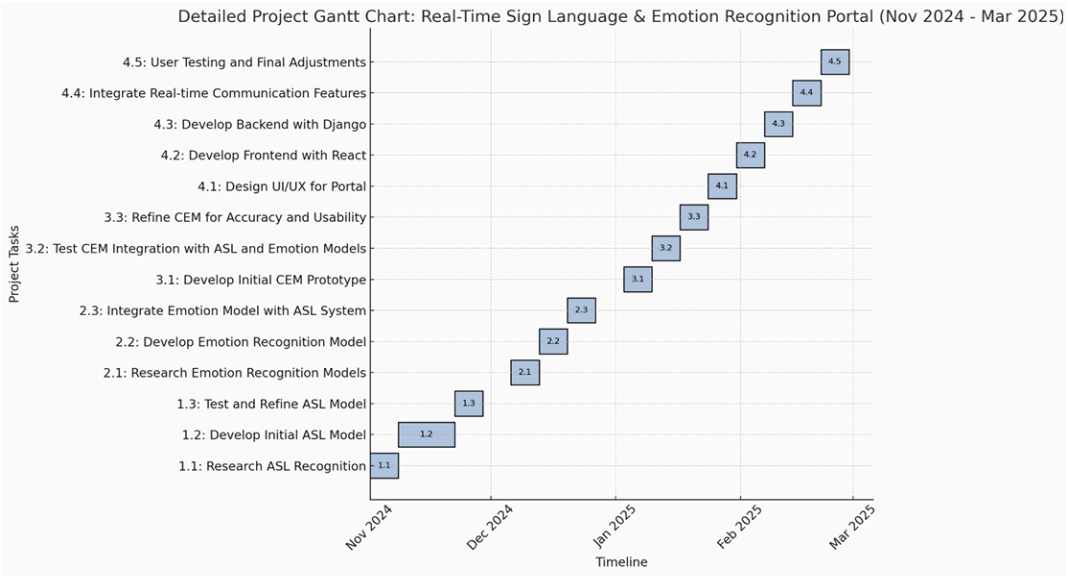


Fig. 11 Gantt chart showcasing project's timeline, spanning from Nov 2024 to Mar 2025.

Code Snippet 1 Webcam Feed Processing

The system uses OpenCV to capture and process webcam frames in real-time. Each frame is processed for both ASL detection and emotion recognition.

```
cap = cv.VideoCapture(0)
while True:
    ret, frame = cap.read()
    if not ret:
        break
    frame = cv.flip(frame, 1)
    debug_image = copy.deepcopy(frame)
```

Code Snippet 2 ASL Detection with MediaPipe Hands

MediaPipe extracts hand keypoints, which are processed to classify ASL gestures.

```
results = hands.process(cv.cvtColor(frame, cv.COLOR_BGR2RGB))
if results.multi_hand_landmarks:
    for hand_landmarks in results.multi_hand_landmarks:
        brect = calc_bounding_rect(debug_image, hand_landmarks)
        landmark_list = calc_landmark_list(debug_image, hand_landmarks)
        pre_processed_landmark_list = pre_process_landmark(landmark_list)
        hand_sign_id = keypoint_classifier(pre_processed_landmark_list)
        draw_landmarks(debug_image, landmark_list)
```

Code Snippet 3 Emotion Recognition Using CNN

A pre-trained CNN detects emotions from face regions detected using Haar Cascades.

```
gray = cv.cvtColor(debug_image, cv.COLOR_BGR2GRAY)
faces = face_cascade.detectMultiScale(gray, 1.1, 5)
for (x, y, w, h) in faces:
    face_roi = gray[y:y + h, x:x + w]
    resized = cv.resize(face_roi, (48, 48))
    reshaped = np.reshape(resized / 255.0, (1, 48, 48, 1))
    emotion_index = np.argmax(emotion_model.predict(reshaped))
    emotion_label = emotion_labels[emotion_index]
    cv.putText(debug_image, emotion_label, (x, y - 10), cv.FONT_HERSHEY_SIMPLEX, 0.9,
(0, 255, 0), 2)
```