A logo of a person

Description automatically generatedA logo with text and a person's head

Description automatically generated

Minya University

Faculty of Computers and Information Minya, Egypt

Final Year Project Thesis in Preparation for graduation

**Submitted by:**

**Ahmed Abd El-Aziz Ahmed Saber Ragab**

**Kerolos Nabil Fahim Ahmed Amer Nasr El-Dien**

**Malak Mohammed Abd El-Ghani**

**Academic Year 2023-2024**

logo

**[Signify: Translate Sign Language to Arabic]**

**Supervised by:**

**Dr. Rehab Emad El-Dein T.A. Al-Zahraa Ahmed**

* **Abstract:**

Deafness is a disability that affects a person’s ability to hear, while muteness affects their ability to speak. Both conditions impact communication, but they do not limit a person’s capacity in other areas of life. The main barrier between deaf or mute individuals and the hearing-speaking community is effective communication. With tools to bridge this gap, such as sign language, people with these disabilities can navigate life just as easily as others.

Sign language is a critical tool for deaf and mute individuals to communicate with both each other and hearing people, yet it often receives minimal attention from the wider community. Many only recognize its importance when they have loved ones affected by hearing or speech disabilities. With the growing focus on accessibility, however, sign language recognition has emerged as a vital area of research within computer vision, promising advancements that could enhance communication and inclusion for all.

Our app is designed to bridge the communication gap, making it easier for deaf and mute individuals to communicate effectively. We offer an affordable, portable solution that uses computer vision to recognize and translate sign language into Arabic. Additionally, our system converts spoken audio directly into text. Through Natural Language Processing, each word or phrase is then matched with corresponding signs or images from our dataset, creating a seamless two-way communication tool for everyone.

* **Acknowledgement**

Table of Contents

**Abstract 1**

Acknowledgement 2

Contents3

List of Tables4

List of Figures5

List of Abbreviations 6

Introduction7

1.1 Problem Definition (or Motivation) 9

* **List of Tables**
* **List of Figures**
* **Figure 2.5:** Object detection and bounding box regression modules.
* **Figure 2.6:** Hand landmarks with key points.
* **Figure 2.7:** The ASR system’s main architecture.
* **List of Abbreviations**

**ArSL:** Arabic Sign Language.

**Introduction**

Individuals with special needs often face discrimination and obstacles that restrict their participation in various societal activities. The lack of effective communication can result in the denial of their rights to live independently, pursue employment opportunities, or even move freely within their communities.

According to UNICEF there are 72 million deaf people worldwide, according to statistics from the World Federation of the Deaf. 80% of these people live in developing countries and use more than 300 sign languages.  In Egypt, there are many challenges facing the deaf, mute and hearing-impaired.

In today's world, communication technologies and tools such as Imo and WhatsApp have become integral to our daily lives. These platforms can significantly facilitate communication between the deaf community and the hearing majority. While deaf individuals can effectively communicate with each other using these technologies, they often face challenges when interacting with those who do not know sign language. Therefore, the development of automatic sign language translation systems is essential to provide equal communication opportunities and enhance public welfare.

In recent years, there has been a transformative shift worldwide aimed at bridging the gap and ensuring that individuals with disabilities enjoy the same standards of equality and rights as everyone else. Information technology has played a pivotal role in this new approach, with various assistive systems being developed to support deaf communities globally.

* 1. Problem Definition (or Motivation)

Deaf individuals in Arabic countries often rely on human interpreters for communication with the hearing majority, which can be both challenging to find and compromise privacy. To bridge this gap, there's a pressing need for the development of an automated system that can translate between Arabic Sign Language (ArSL) and Arabic speech or text. The focus here is on creating technology to seamlessly convert ArSL to Arabic and vice versa.

Given the rapid advancements in mobile technology, we can anticipate highly capable handheld and wearable devices in the near future. These devices should enable intuitive and natural interactions, possibly through 3D hand and body gestures, replacing traditional touchscreens and trackpads. This shift towards more intuitive interfaces is rooted in humans' natural experiences and physical interactions with their environment.

Technical challenges remain, including developing media technologies that can accurately detect, recognize, and track complex hand gestures. Achieving this will require innovative approaches and significant research to overcome the limitations of current technologies.

According to the World Health Organization (WHO), over 1.5 billion people globally live with hearing loss, and 430 million of them have disabling hearing loss. This population faces significant educational and employment disparities, with only 48% of individuals with hearing loss securing employment. These challenges often lead to social withdrawal, limited access to services, and emotional issues stemming from reduced self-esteem and confidence.

To address these issues, it's crucial to raise awareness about the capabilities and rights of individuals with hearing loss. Promoting understanding and empathy can help reduce discrimination and stigma, creating a more inclusive society.

Furthermore, governments and organizations should prioritize the development and implementation of policies that support the rights and well-being of people with hearing loss. This includes improving access to education, healthcare, and employment opportunities, as well as fostering an environment that encourages social inclusion and participation.

By addressing these key areas, we can help create a more supportive and equitable world for individuals with hearing loss, ensuring they have the same opportunities and rights as everyone else.

* 1. **Project Objectives**

Our primary goal is to develop a solution that enables deaf, hearing, and speech-impaired individuals who use sign language to communicate effectively and efficiently with others in their natural language. This will help them lead normal lives. Below are the objectives of our project:

**1. Translating Arabic Sign Language:**

The project allows users to use a camera to perform sign language, which will then be translated into Arabic text.

**2. Translating Arabic Speech into Text:**

The project enables users to record Arabic speech and translate it into Arabic text.

**3. Promoting Independence:**

The application empowers users to communicate freely without needing a translator or interpreter, allowing them to use it anytime, anywhere.

* 1. **Contributions of our study**

Deaf individuals often face psychological challenges such as depression and anxiety due to the communication barrier between those who use sign language and those who use written or spoken language. A significant percentage of deaf individuals, particularly those born deaf, may choose to withdraw from the hearing world, despite the availability of current or future technologies. Our aim is to help prevent this isolation by offering support and solutions.

Through our education and research, we have acquired valuable knowledge in several areas:

- Utilizing technology to develop our own projects.

- Learning about artificial intelligence and its subfields, such as computer vision algorithms and tools, to create hand gesture recognition and detection systems.

- Understanding Human-Computer Interaction (HCI) principles to design user interfaces that are flexible, easy to use, and interactive.

* 1. **Project Scope**

"Signify" is a mobile application that functions as a real-time translator, detecting Arabic sign language made by a speech-impaired individual using the mobile camera and converting it into Arabic text for the hearing person. Additionally, it converts the spoken words of the hearing person into Arabic text for the hearing-impaired individual to read. This application ensures a high level of convenience by allowing both users to communicate in their everyday language.

* 1. **Project Timeline**
  2. **Document Organization**

**Chapter 2: Literature Review**

**2.1 Introduction**

**2.2 Background**

In this section we provide brief background information on the materials needed for our project.

**2.2.1 Arabic Sign Language Datasets**

**2.2.2 Hand Detection Methods**

**2.2.2.1 Bounding Boxes**

In state-of-the-art object detection algorithms, bounding box regression is crucial for achieving high localization accuracy. Nearly all popular deep learning-based object detection algorithms utilize bounding box regression to fine-tune object locations.

Bounding boxes are typically used to describe the spatial location of an object. These rectangular boxes are defined by the x and y coordinates of their upper-left and lower-right corners. Another common representation is the (x,y) coordinates of the bounding box center, along with the box's width and height.

Data annotators draw these rectangles over images to outline the object of interest, specifying its X and Y coordinates. This helps machine learning algorithms identify objects, determine collision paths, and conserve computing resources. The best-fit bounding box is the smallest rectangle that can contain the object in all possible orientations.

A black background with white doors

Description automatically generated

Figure 2.5: Object detection and bounding box regression modules.

This diagram represents a module commonly used in object detection systems. It shows two parallel paths, one for object classification (softmax output) and another for bounding box regression (bbox reg).

**Components:**

1. **rf-map n:**
   * Represents the input feature map derived from a neural network (e.g., a convolutional backbone).
   * n denotes the depth or number of channels in this input feature map.
2. **Bounding Box Regression Path (Top Path):**
   * **Conv 3×3, 512:**
     + A convolutional layer with a 3×3 kernel and 512 filters, extracting features specifically for bounding box regression.
   * **Conv 3×3, 4×A:**
     + Another convolutional layer that outputs bounding box predictions. Each anchor produces 4 regression values (center x, y, width, height).
     + A is the number of anchor boxes per location.
3. **Object Classification Path (Bottom Path):**
   * **Inception Modules:**
     + Two inception-like modules for extracting diverse features, emphasizing multiscale feature extraction.
   * **Conv 3×3, (k+1) × A:**
     + Outputs class predictions using a 3×3 convolutional layer.
     + (k+1) corresponds to k object classes plus a background class.
     + A anchors per location multiply the output.
4. **Outputs:**
   * **Softmax (Classification):**
     + Computes probabilities for each object class (including background) for every anchor box.
   * **Bounding Box Regression:**
     + Produces the coordinates for bounding boxes, refining the positions of anchor boxes.

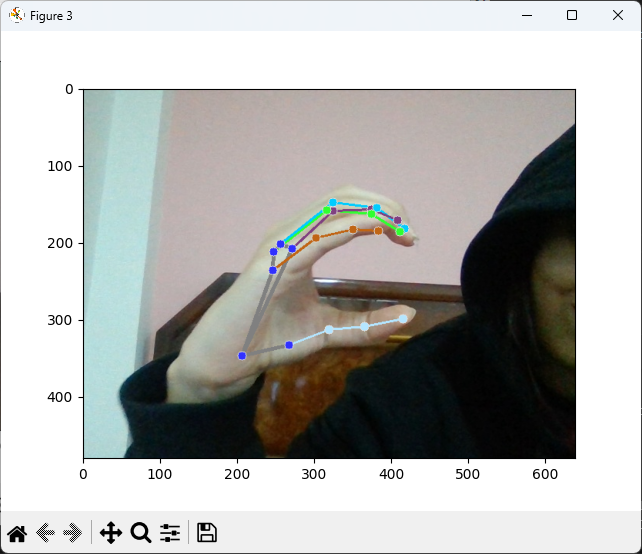
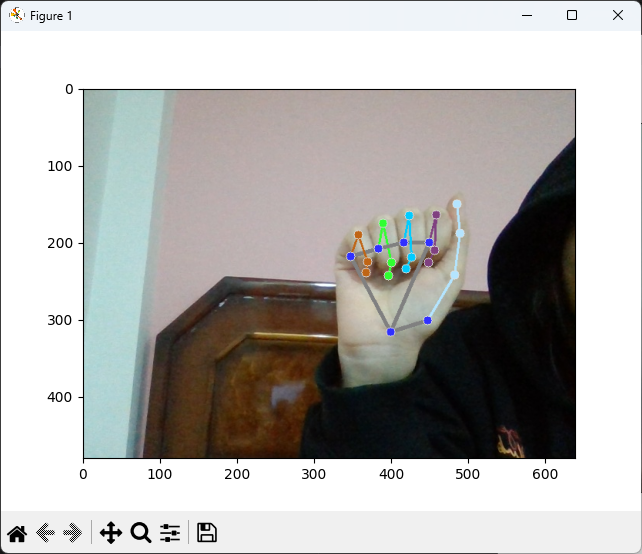
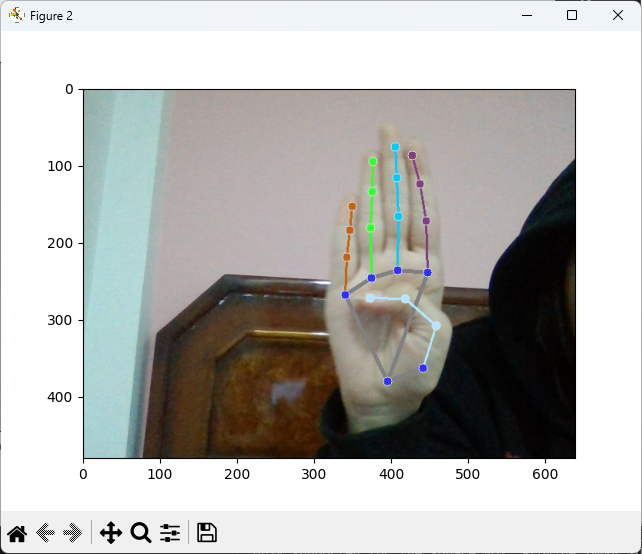
**2.2.2.2 Hand Pose Estimation**

After testing the box detection method, we discovered a superior detection technique that overcomes the limitations of the previous method.

Pose estimation is a computer vision technique used to predict and track the location of a person or object by analyzing a combination of pose and orientation. This process typically involves identifying, locating, and tracking several key points on the given subject.

Our goal is to track these key points in images and videos, focusing specifically on hand key points in our app.

It's important to differentiate between 2D and 3D pose estimation. 2D pose estimation involves predicting the location of key points in 2D space relative to an image or video frame, providing X and Y coordinates for each key point. In contrast, 3D poses estimation transforms a 2D image into a 3D representation by adding a Z-dimension to the prediction.



(Letter Aa) (Letter Bb) (Letter Cc)

Figure 2.6: Hand landmarks with key points.

The method illustrated in figure 2.6 involves **hand landmark detection** using **MediaPipe Hands** illustrated in those steps:

1. **Input Image Processing:**
   * A webcam or camera captures a real-time image of the hand. This frame is fed into the detection model.
2. **Hand Detection:**
   * The model identifies the region of the image containing a hand using a **bounding box**. This step isolates the hand from the rest of the scene to focus computational resources on the hand area.
3. **Landmark Localization:**
   * Once the hand is detected, the model predicts the positions of **key hand landmarks**. These landmarks correspond to predefined points on the hand, such as:
     + Fingertips
     + Knuckles
     + Joints
     + Wrist
   * The detected landmarks are returned as 2D coordinates in the image frame (and optionally as 3D coordinates in space, depending on the model used).
4. **Connections Between Landmarks:**
   * The model also visualizes the skeletal structure of the hand by connecting key points (landmarks) with lines, making it easier to interpret the gesture.
5. **Gesture or Pose Recognition** 
   * Once the landmarks are extracted, additional processing can be performed to identify specific hand gestures (e.g., "hello," "stop," "thumbs up").
   * This is done by analyzing the relative positions and orientations of the landmarks.

**Key Features of the Method:**

* **Robustness to Variations:** The model can track hand positions and orientations in real-time, even under challenging lighting and background conditions.
* **Real-Time Performance:** Modern implementations (like MediaPipe) are optimized for real-time processing, enabling fluid and responsive hand tracking.
* **Multi-Hand Support:** Many models can detect and track multiple hands simultaneously, though the images provided focus on one hand.

**Applications:**

* **Sign Language Recognition:** Detecting gestures to translate them into text or speech.
* **Human-Computer Interaction (HCI):** Controlling devices or applications through hand gestures.
* **Augmented Reality (AR):** Integrating hand tracking for immersive experiences.
* **Gaming:** Using hand gestures as an input method.

**2.2.2.3 Hand Gesture Recognition**

Gesture recognition is a topic in computer science and language technology that aims to interpret human gestures using mathematical algorithms. It is a subdiscipline of computer vision. Gestures can originate from any bodily motion or state, but they most come from the face or hand. Current focuses in the field include emotion recognition from facial expressions and hand gesture recognition. Users can use simple gestures to control or interact with devices without physically touching them. Many approaches have been made using cameras and computer vision algorithms to interpret sign language. However, the identification and recognition of posture, gait, proxemics, and human behaviors are also subjects of gesture recognition techniques.

Gesture recognition can be seen as a way for computers to begin to understand human body language, thus building a richer bridge between machines and humans than primitive text user interfaces or even graphical user interfaces (GUIs), which still limit the majority of input to keyboards and mice and allow for natural interaction without any mechanical devices.

Using gesture recognition, it is possible to point a finger, and the system will respond accordingly. This could make conventional input devices redundant.

The gestures presented in diverse backgrounds need to be accurately processed and segmented for precise classification by the hand gesture recognition system. This study compares the performance of the proposed Image Segmentation Algorithm with a standard Canny Edge Detection Algorithm by comparing the statistical values of the features obtained from the feature extraction stage, thus validating the importance of having a robust preprocessing stage for hand gestures.

In computer interfaces, two types of gestures are distinguished:

**- Offline gestures:** These are processed after the user interaction with the object, such as a gesture to activate a menu.

**- Online gestures:** These are direct manipulation gestures used to scale or rotate a tangible object.

There are many challenges associated with the accuracy and usefulness of gesture recognition software. For image-based gesture recognition, there are limitations on the equipment used and image noise. Images or videos may not be under consistent lighting or in the same location. Items in the background or distinct features of the users may make recognition more difficult.

The variety of implementations for image-based gesture recognition may also cause issues for the viability of technology for general usage. For example, an algorithm calibrated for one camera may not work for a different camera. The amount of background noise also causes tracking and recognition difficulties, especially when occlusions (partial and full) occur. Furthermore, the distance from the camera, and the camera's resolution and quality, also cause variations in recognition accuracy.

**2.2.3 Deep Learning Frameworks**

After we discussed the methods to detect, we will continue talking about the frameworks that are able to implement this, we had four options:

● **TensorFlow (TF)** is an end-to-end machine learning framework from Google that allows you to perform an extremely wide range of downstream tasks. With TF2.0 and newer versions, more efficiency and convenience were brought to the game.

● **Keras** is built on top of TensorFlow, which makes it a wrapper for deep learning purposes. It is incredibly user-friendly and easy to pick up. A solid asset is its neural network block modularity and the fact that it is written in Python, which makes it easy to debug.

● **PyTorch** is TensorFlow’s direct competitor developed by Facebook and is widely used in research projects. It allows almost unlimited customization and is well adapted to running tensor operations on GPUs (actually, so is TensorFlow).

● **Scikit-learn** is another user-friendly framework that contains a great variety of useful tools: classification, regression and clustering models, as well as preprocessing, dimensionality reduction and evaluation tools.

Later, we found that Keras is a high-level API capable of running on top of TensorFlow, CNTK and Theano. It has gained favor for its ease of use and syntactic simplicity, facilitating fast development. TensorFlow is a framework that provides both high- and low-level APIs. Pytorch, on the other hand, is a lower-level API focused on direct work with an array of expressions.

**2.2.4 Speech Recognition**

Speech is the most natural form of human communication, and speech processing has been one of the most exciting research areas of signal processing.

Speech recognition, also known as automatic speech recognition (ASR), computer speech recognition, or speech-to-text, is a capability that enables a program to process human speech into a written format.

The Arabic language is the largest still-living Semitic language in terms of the number of speakers. Around 300 million people use Arabic as their first native language, making it the fourth most widely used language based on the number of first language speakers.

The vagaries of human speech have made development challenging. It’s considered to be one of the most complex areas of computer science, involving linguistics, mathematics, and statistics. Speech recognizers consist of several components, such as speech input, feature extraction, feature vectors, a decoder, and a word output. The decoder leverages acoustic models, a pronunciation dictionary, and language models to determine the appropriate output.

Various algorithms and computational techniques are used to recognize speech into text and improve the accuracy of transcription. Below are brief explanations of some of the most used methods:

**- Natural language processing (NLP).**

**- Hidden Markov Model (HMM).**

**- N-grams.**

**- Neural networks (NN).**

**- Speaker Diarization (SD).**

**2.2.4.1 Speech Recognition Components**

The goal of an Automatic Speech Recognition (ASR) system is to determine the most probable sequence of words

from a fixed vocabulary, given a set of acoustic observations

The optimal word sequence can be estimated using the following equation derived from Bayes’ theorem:

Since is independent of the word sequence, it can be ignored in the maximization, reducing the equation to:

W = arg max P(W|O) = arg max P(o|𝑤) P(𝑤) / P(o)

* is the probability of the word sequence, independent of the acoustic observations. This is modeled using the language model, which captures the syntactic and semantic structure of the language.
* is the likelihood of the acoustic observations given the word sequence. This is modeled using the acoustic model, which connects sound features to potential words.

Steps in ASR Processing:

1. **Feature Extraction:**  
   Extract acoustic features (observations) from the spoken utterance, such as Mel-Frequency Cepstral Coefficients (MFCCs) or spectrograms.
2. **Language Model ():**  
   Estimate the probability of a word sequence based on the rules and structure of the language. The language model predicts how likely a sequence of words is, independent of the actual sounds.
3. **Acoustic Model ():**Estimate the probability of the observed acoustic features given a hypothesized word sequence. This models how words sound in different contexts or conditions.

These components work together to map the acoustic input (speech) to a text output, optimizing for the most likely word sequence .

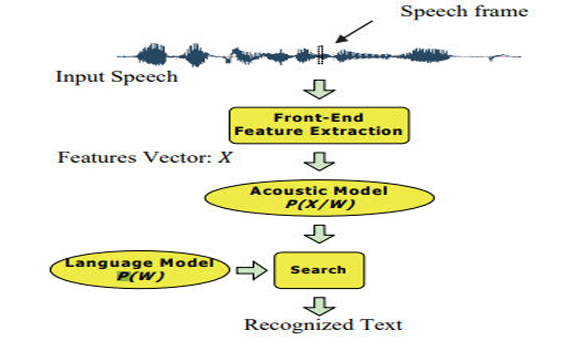


Figure 2.7: The ASR system’s main architecture.