# 

# 

**MULTIMEDIA UNIVERSITY OF KENYA**

**FACULTY OF COMPUTER AND INFORMATION TECHNOLOGY**

In Partial Fulfillment of

**CSE 2400: SOFTWARE PROJECT II**

**SIGN LANGUAGE RECOGNITION SYSTEM**

Submitter to:

**MR. James Adunya**

**Ms Yvette Otukana**

Submitted by:

**Faith Sang**

Date Submitted:

3/11/2022

# DECLARATION

I hereby declare that this Project [Proposal] is my own work and has, to the best of my knowledge, not been submitted to any other institution of higher learning.

**Student :** FAITH JEBET SANG **Registration Number:**CIT-227-042/2019

**Signature: ............................................... Date:.....................................................**

This project has been submitted as a partial fulfillment of requirements for the Bachelor of Science in Software Engineering of Multimedia University of Kenya with my approval as the University supervisor.

**Supervisor:** MR.JAMES ADUNYA **Signature: ……………………………………**

**Supervisor:** Ms Yvette Otukana **Signature: …………………………………….**

# **DEDICATION**

To all the individuals who are deaf or hard of hearing, whose lives have been impacted by the lack of accessibility and understanding in our society, this project is dedicated to you. May this project serve as a reminder of the importance of sign language and the need for increased support and recognition of the Deaf community. May it inspire greater efforts to promote inclusivity, break down barriers, and foster greater understanding and acceptance of all individuals.

We dedicate ourselves to this cause with the hope that our efforts will contribute to a more equitable and just world,where everyone has the opportunity to fully participate and thrive, regardless of their abilities or differences.

I also dedicate this project to my classmates and students of Multimedia university who were an inspiration to do this project .I would also like to thank my supervisors Mr James Adunya and Ms Yvette Otukana for giving me support throughout this project.

# 

# **ACKNOWLEDGMENTS**

I acknowledge the almighty God for seeing me through the entire research. I thank my supervisors, Mr James Adunya and Ms Yvette Otukana, for the guidance they have given me during my entire study time and the project development period. Their valuable recommendations have guided me to produce a Sign Language Recognizer system that I am proud to be part of. I also appreciate the support from my family, friends, my

lecturers and classmates for all the support that made it possible for me to complete this project. I would also like to appreciate those who took part in this project process for their time, patience and support and to be specific I would like to thank Elijah Wandimi, Felix Obuya and Oyamo Brian for helping me especially with the debugging process. Without their support this work wouldn’t have been a success.Lastly I thank the University of Multimedia administration, staff,and the students for their support.

# **ABSTRACT**

There are learning resources for the deaf and speech-impaired, but their use is limited. The proposed system will be a real-time system where live signal gestures are processed using image processing. A classification is then used to differentiate between the different tags and the text appears in the translated output. Machine learning algorithms are used to train the dataset. The purpose of this system is to improve the current field system in terms of response time and accuracy by using efficient algorithms, higher quality datasets and better sensors. Existing systems can recognize gestures with high latency because they only use image processing. Our project aims to develop a highly responsive and robust cognitive system for daily use by people with hearing and speech impairments.

Interpreting deaf-mute people has always been a problem for people as they primarily rely on sign language for communicating. Active participation of the deaf-mute community still remains at an elementary stage, despite multiple nations providing resources for the same, like a sign language interpreter and communicator of news in the country of New Zealand. Perturbing situations such as kidnapping, deception, fire breakout or any other situations of general agony could further exacerbate this barrier of communication, as the mute people try their best to communicate, but the majority remains oblivious to their language. Therefore, bridging the gap between these two worlds is of utmost necessity.

# 

# TABLE OF CONTENTS

[**DECLARATION 2**](#_heading=h.3coi00f31wvs)

[**DEDICATION 3**](#_heading=h.rlyyitgda4f2)

[**ACKNOWLEDGMENTS 4**](#_heading=h.mwsn48sigl26)

[**ABSTRACT 5**](#_heading=h.nin32jw12mcs)

[**TABLE OF CONTENTS 6**](#_heading=h.1fob9te)

[**List of Figures 8**](#_heading=h.3znysh7)

[**List of Tables 8**](#_heading=h.q7tz0nncy0up)

[**CHAPTER ONE:INTRODUCTION 9**](#_heading=h.2et92p0)

[1.1 BACKGROUND OF THE STUDY 9](#_heading=h.tyjcwt)

[1.2 STATEMENT OF THE PROBLEM 10](#_heading=h.3dy6vkm)

[1.3 OBJECTIVES OF THE STUDY 10](#_heading=h.1t3h5sf)

[1.4 SCOPE AND LIMITATIONS 10](#_heading=h.4d34og8)

[1.5 SIGNIFICANCE OF THE STUDY 11](#_heading=h.2s8eyo1)

[**CHAPTER TWO: REVIEW OF RELATED STUDIES AND LITERATURE 11**](#_heading=h.17dp8vu)

[2.1 INTRODUCTION 12](#_heading=h.36s7d4atw269)

[Figure 2.1 Sign Chart for the American Sign Language(ASL) 13](#_heading=h.21fzjk12pj6)

[2.2 RELATED SYSTEMS 13](#_heading=h.68nb45y90urv)

[2.3 PROPOSED SOLUTION 13](#_heading=h.pdsi5bhiczlq)

[**CHAPTER THREE:METHODOLOGY 15**](#_heading=h.aa1j99jkwaww)

[3.1 INTRODUCTION 15](#_heading=h.ssm337lhsars)

[3.2 DATA COLLECTION METHODS AND TOOLS 16](#_heading=h.6iyptadghspr)

[3.3 PROJECT RESOURCES 16](#_heading=h.xiq3pppuaxz9)

[3.4 PROJECT SCHEDULE 17](#_heading=h.5ul7cbixrmo8)

[Table 3.4 Project Schedule 18](#_heading=h.ir23xtdmzr50)

[3.5 PROJECT BUDGET 18](#_heading=h.8188j8zbnnt4)

[Table 3.5 Project Budget 18](#_heading=h.23pgj3cof5p4)

[**CHAPTER FOUR: SYSTEM ANALYSIS 19**](#_heading=h.casynxtoy4l0)

[4.1 DETAILED ANALYSIS OF CURRENT SYSTEM 19](#_heading=h.73yebzexad8w)

[4.1.1 DATA FLOW DIAGRAM 19](#_heading=h.4nyi5vjarjal)

[Figure 4.1 Data flow diagram for sign language recognition 21](#_heading=h.hj1ed6o04ono)

[4.1.2 UML DIAGRAMS 21](#_heading=h.dv5p1ti22o2f)

[4.1.2.1 USE CASE DIAGRAM 22](#_heading=h.8syqxsjz5ju4)

[Figure 4.1.2.1 Use case diagram of sign language recognition system 23](#_heading=h.1g3imeianopk)

[Table 3.1 Use case scenario for sign language recognition system 24](#_heading=h.duplfmp1dam7)

[4.1.2.2 SEQUENCE DIAGRAM 24](#_heading=h.yyh28s11iqz0)

[Figure 4.1.2.2 Sequence Diagram 26](#_heading=h.68xui8y0obbq)

[4.1.2.3 STATE CHART 26](#_heading=h.scbsjbm86vzm)

[Figure 4.1.2.3 State Chart 27](#_heading=h.x6im8tetnnw6)

[4.2 SYSTEM REQUIREMENTS 27](#_heading=h.ci5ebfldqbf0)

[4.2.1 Functional Requirements 28](#_heading=h.v81vk42xxpny)

[4.2.2 Non-functional Requirements 28](#_heading=h.egab2nj4w2zx)

[**CHAPTER FIVE: SYSTEM DESIGN 29**](#_heading=h.agqlmyxf5s5a)

[5.1 Architectural Design 29](#_heading=h.7fwvqsrmxtzo)

[Figure 5.1 System Architecture 30](#_heading=h.8anzc7914u5q)

[5.2 User Interface Design 30](#_heading=h.octgtqf3ukeu)

[5.2.1 Starting the webcam 30](#_heading=h.pxq2ifoa3y7z)

[Figure 5.2.1 starting webcam 31](#_heading=h.khj2zofv5oie)

[5.2.2 After prediction 31](#_heading=h.7ounivqmbp6b)

[Figure 5.2.2 after prediction 31](#_heading=h.lhqe9qyurebr)

[**CHAPTER SIX:IMPLEMENTATION AND TESTING 32**](#_heading=h.yr60pwonwqw7)

[6.1 DEVELOPMENT ENVIRONMENT 32](#_heading=h.31nlesrxcne8)

[6.2 SYSTEM COMPONENTS 32](#_heading=h.byff0o2eqoo6)

[6.3 TEST DATA 33](#_heading=h.frchxrx2jd)

[6.4 TEST RESULTS 33](#_heading=h.dj6jhnl61ewy)

[**CHAPTER SEVEN: CONCLUSION 35**](#_heading=h.gxdzkc3jmth)

[7.1 Achievements and Lesson LearntAchievements: 35](#_heading=h.5l0mfvwfhx2d)

[7.2 Conclusion 36](#_heading=h.gkitj1sa5csr)

[7.3 Recommendation 37](#_heading=h.9ggptdf0jai1)

[**REFERENCES 38**](#_heading=h.ewwf7p1r6afj)

# 

# 

# 

# **List of Figures**

Figure 2.1 Sign Chart for the American Sign Language(ASL)

[Figure 4.1 Data flow diagram for sign language recognition](#_heading=h.hj1ed6o04ono)

[Figure 4.1.2.2 Sequence Diagram](#_heading=h.68xui8y0obbq)

[Figure 4.1.2.3 State Chart](#_heading=h.x6im8tetnnw6)

[Figure 5.1 System Architecture](#_heading=h.8anzc7914u5q)

[Figure 5.2.1 starting webcam](#_heading=h.khj2zofv5oie)

[Figure 5.2.2 after prediction](#_heading=h.lhqe9qyurebr)

# **List of Tables**

Table 3.4 Project Schedule

Table 3.5 Project Budget

[Table 3.1 Use case scenario for sign language recognition system](#_heading=h.duplfmp1dam7)

# **CHAPTER ONE:INTRODUCTION**

## **1.1 BACKGROUND OF THE STUDY**

Accessibility is a factor that is usually emphasized when developing any software. It is based on the fact that with whatever you are coming up with, always keep in mind a special group of users , people with disabilities. This whole study keeps focus on this fact and the desire for a computer-based solution is significant in this age of technology for deaf people. Researchers have been working on the problem for quite some time and the results are promising. Although interesting technologies for voice recognition are becoming available, there is currently no commercial solution for sign recognition on the market.

The gesture is a vital and meaningful mode of communication for the visually impaired person.In the world of the deaf that is basically their only way to connect with the rest of the world. But for the rest of the world,about 0.1% of the whole population in any given country make up the deaf sign language users,this figure is very low. The immediate effect is that it causes a disconnect between the deaf world and the rest of the world, a communication barrier.

For instance, in our home we have a housekeeper who happens to be deaf. Only one person in the house happens to know sign language and we all depend on that person to communicate with the housekeeper. If for instance the only person who can communicate to her is not around, apart from hello and goodbye there is no way we can communicate with her.In this era of technology this should not be an issue.

This miscommunication is caused by people not taking sign language as important as we should, a necessity for communication this in turn contributes to the barrier and thus greatly inconveniencing the deaf community.

The proposed solution here is a computer based method for regular people to understand what the differently abled individual is trying to say.In our system, the user will perform the hand gestures or signs by turning on their camera, and the system will detect the sign and display it to the user.

## **1.2 STATEMENT OF THE PROBLEM**

In a bid to solve the communication barrier that exists in the community, the research has an eye to introducing a system that will act as a mediator between the regular people and the differently abled. In the life of a deaf person, he or she never knows if the person they will encounter will understand what they say or not.They will communicate in the form of gestures and the other people will either understand what they are saying and respond back or be confused with what the other person is trying to say.It so happens that majority of the population lie in the category where they do not understand and hence will not communicate back. If no one is around to translate the conversation ends up hanging there and this causes the problem the study is trying to solve.

## **1.3 OBJECTIVES OF THE STUDY**

The general objective of this study is to take sign language gestures and generate their associated meaning.

The specific objectives include:

i)To learn the sign language gestures that are commonly used.

ii)To interpret the gestures and convert them into English.

iii)Distinguish between closely related gestures.

## **1.4 SCOPE AND LIMITATIONS**

The study focuses on a community of both regular people and people with disabilities.The data collection will be limited to only a small percentage of around 50 people who will represent the community of Rongai. The project is estimated to cover 8 months.

The study will not be able to Interpret English and give the corresponding sign.And the system could have minor glitches due to the quality of camera that could be used in demonstration.

## **1.5 SIGNIFICANCE OF THE STUDY**

The main purpose of this study is to design, develop and implement a system that will be adopted by the community in general when they encounter a person with disability.

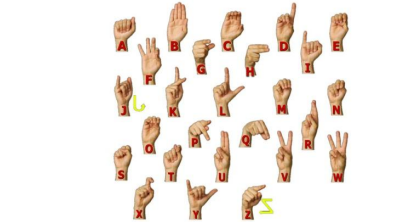
The research will also be beneficial to the person with disability when they can be able to be understood with what they might be communicating.

# **CHAPTER TWO: REVIEW OF RELATED STUDIES AND LITERATURE**

## **2.1 INTRODUCTION**

A sign language is a language which mainly uses actions or gestures to convey meaning as opposed to acoustically conveyed sound patterns. There are significant differences between signed and spoken languages, because of the constraints offered by visual gestures.Yet the two are fundamentally similar as both have their own syntax and semantics. Groups of hearing speech impaired people have used signs to communicate for many years and so sign language is developed among them.

American sign language(ASL) substantially facilitates communication in the hearing impaired community; there are only ~250,000 - 500,000 speakers which limits the number of people that they can communicate with[7]. In order to diminish this obstacle and to enable better communication, I would like to propose an ASL recognition system that uses MediaPipe Holistic and LSTM model to translate a user’s ASL signs into text in real time.



#### Figure 2.1 Sign Chart for the American Sign Language(ASL)

## **2.2 RELATED SYSTEMS**

Sign language recognition (ASL) is not a new problem in computer vision.Over the past 20 years, researchers have used several categories of classifications, which can be loosely grouped into linear classifications and Bayesian networks.

A real-time sign language translator is an important milestone in facilitating communication between the deaf community and the general public. Brandon Garcia and Sigberto Alarcon Viesca[1] were able to implement a robust model for the letters a-e and a modest one for the letters a-k. The Surrey and Massey datasets on ASL along with the GoogleNet architecture are used to train the system. The system takes the input of the user’s sign language video, classifies the actions for each letter and then tries to come up with the most accurate words. The factors such as lighting conditions and sign language border detection are considered while designing the project. They attained a validation accuracy of nearly 98% with five letter and 74% with ten letters

Hardie Cate, Fahim Dalvi and Zeshan Hussain[2] use machine learning techniques for temporal classification, specifically the multivariate case. Although the results obtained are not very high, we believe that a more efficient implementation of the algorithms can yield more complex models that will perform well.

## **2.3 PROPOSED SOLUTION**

This system uses the University of South Wales dataset which has 95 unique signs.It suggests the use of baseline SVM for high quality data and Temporal Classification techniques for the lower quality data. It also considers implementing a custom LSTM model that removes the assumptions of the technique that do not apply to the data. A device called Myo armband is also used to collect new data. They achieved an accuracy of 78.6% on a high quality data set.

Machine learning and computer vision techniques are used extensively in sign language recognition research. Using these approaches, we can model the complex dynamics and variations of sign language gestures and extract relevant features from sign language videos.Some specific techniques that have been used in sign language recognition include:

* Deep neural networks (DNNs): DNNs, such as convolutional neural networks (CNNs) and recurrent neural networks (RNNs), have been used to automatically learn characteristics of sign language videos and classify sign language gestures. They are used in different modes such as RGB, depth data and skeleton.
* Neural Networks Neural networks are inspired by the biological arrangement of processing elements called neurons in the brain. These neurons allow for parallel processing of computational tasks. This allows neural networks to solve complex pattern recognition problems better than procedural algorithms. A CNN is a neural network capable of calculating the responses of neurons through convolution operations. The first level of CNN can be used to match images to a fixed pattern and the next level can be used to detect differences between selected images, improve accuracy and generate models or models.
* Its Cascade is a computer vision learning technique that trains a cascade function on a large number of positive and negative images. So can be used to detect objects from other images. First, the algorithm was trained to recognize faces and trained using many images. Once the model is trained, features can be extracted.
* Keras and TensorFlow Python Library Keras is a high-level deep learning library that can be used as an interface to TensorFlow . TensorFlow was developed by Google and is used as a modeling framework for back-end neural networks.

# **CHAPTER THREE:METHODOLOGY**

## **3.1 INTRODUCTION**

A research methodology is the elaboration of a clear strategy for gathering evidence,

including the specific data collection methods to be used, the kinds of evidence to be

collected, and the approach for analyzing the evidence (Darian-Smith & McCarthy,

2017). It is the path to solve a research problem.Hence it must be planned according to the objectives of the study

The proposed methodology to be adopted when developing the system is the agile methodology. The phases of this methodology include:

1. Concept- Here we determine the scope of the project and the key requirements are discussed and a documentation to outline them. The requirements should be minimum as they can be added at later stages. Estimation of the time and cost of the project. All this information will determine the feasibility of the project.
2. Inception- Once the concept is outlined, a project team is developed and are provided with the tools and resources. They then can start the design process. They will come up with a mock-up of the user-interface and build the project architecture. The requirements will be represented on a diagram to determine the product functionality
3. Iteration- also known as construction phase. It’s the longest phase and all the bulk work is carried out here. Work is done to turn design into code.The goal is to build the bare functionality of the product by the end of the first iteration.Additional features can be added in later iterations.
4. Release- the product is almost ready for use but first the agile team will test the system to ensure the code is clean. If potential bugs are detected then developers will address them swiftly.Then the software quality assurance team will test to ensure the software is fully functional. User training also takes part in this phase and also requires extra documentation.
5. Maintenance- the software will now be fully developed and made available to customers. During this phase the team will provide ongoing support to keep the system running smoothly and fix any new bugs.There will also be additional training to users and ensure they know how to use the product.Overtime, new iterations can take place to refresh the existing product with upgrades and additional features.
6. Retirement- A product will enter the retirement phase in two circumstances. Either it is being replaced by new software or the system itself has become obsolete and incompatible with the organization overtime. The development team will notify the users that the system is being retired and if there is a replacement the users will migrate. Finally the developers will carry out any remaining end of life activities and remove support from the existing software.

Overall, Agile methodology can be an appropriate approach for a sign language recognition system as it allows for rapid prototyping, continuous testing and evaluation, and flexible adjustments in response to new research findings and changing requirements.

## **3.2 DATA COLLECTION METHODS AND TOOLS**

The techniques employed by the researcher to get in depth information about the menstrual cycle included:

* Observation- This is achieved by observing and monitoring how the existing systems work and identifying both the strengths and weaknesses
* Secondary data- This data was obtained from journals and articles from the internet and also websites that provided in depth information on sign language.

## **3.3 PROJECT RESOURCES**

Project resources are anything that is necessary for the project to be completed. Without them, it’s impossible for a project to be completed successfully.

Human resources:

* Supervisor

Software requirements include:

* Operating system: Windows 2000 Xp
* Programming language, Python 3.10(32/64 bit)

Hardware requirements:

* 4GB RAM or more
* Processor:Standard processor with a speed of 1.6 GHz or more
* Hard Disk :20 GB or more
* Monitor:Standard color monitor
* Keyboard:Standard keyboard
* Mouse:Standard mouse

## **3.4 PROJECT SCHEDULE**

| **PROJECT ACTIVITY** | **PROJECT DURATION** | | | | | | | |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **SEMESTER 1 - 2022** | | | | **SEMESTER -2 2023** | | | |
| **Sep** | **Oct** | **Nov** | **Dec** | **Jan** | **Feb** | **Mar** | **April** |
| Project Identification |  |  |  |  |  |  |  |  |
| Research on the Problem |  | |  |  |  |  |  |  |
| Design |  |  | |  |  |  |  |  |
| Drafting of Proposal |  |  |  |  |  |  |  |  |
| Revision of Proposal |  |  |  |  | |  |  |  |
| Obtaining resources |  |  |  |  |  |  |  |  |
| Implementation of the System, Unit Testing |  |  |  |  |  | |  |  |
| Testing the System |  |  |  |  |  |  | |  |
| Drafting the Final Document |  |  |  |  |  |  | |  |
| Preparing presentation |  |  |  |  |  |  |  |  |
| Binding Final Report |  |  |  |  |  |  |  |  |

##### Table 3.4 Project Schedule

## 3.5 PROJECT BUDGET

| **PROJECT ITEM** | **AMOUNT(KSH)** |
| --- | --- |
| Laptop | 30,000 |
| Printing | 500 |
| Binding | 300 |
| Internet | 3000 |
| Storage | 2000 |
| Miscellaneous | 1500 |
| **TOTAL** | **37300** |

##### Table 3.5 Project Budget

# **CHAPTER FOUR: SYSTEM ANALYSIS**

## **4.1 DETAILED ANALYSIS OF CURRENT SYSTEM**

## **4.1.1 DATA FLOW DIAGRAM**

The DFD is also known as a bubble chart. It is a simple graphical formality that allows the system to perform input data, various processes on this data, and represent the system to generate output data through the entire system. It maps out the flow of information for any process or system, how data is processed in terms of inputs and outputs. It uses rectangles, circles, and arrow segmentation symbols to indicate data input, output, storage points, and paths within each target. They can be used to analyze an existing system or model of a new one. A DFD can often visually “say” things that would be hard to explain in words and they work for both technical and non- technical.

There are four components in DFD:

* External Entity.
* Process.
* Data Flow.
* Data Store.

1) External Entity :- It is an external system that sends or receives data, communi-cating with the system. Losing access to a system is a source of information and a destination. They can be external entities or individuals, computer systems or business systems. They are known as Terminator, Source and Sync or Actor. They are usually drawn on the edge of the figure. These are the sources and targets of the system’s inputs and outputs.



2) Process: - It is just like a function that changes the data, producing an output. It might perform computations to sort data based on logic or direct the dataflow based on business rules.

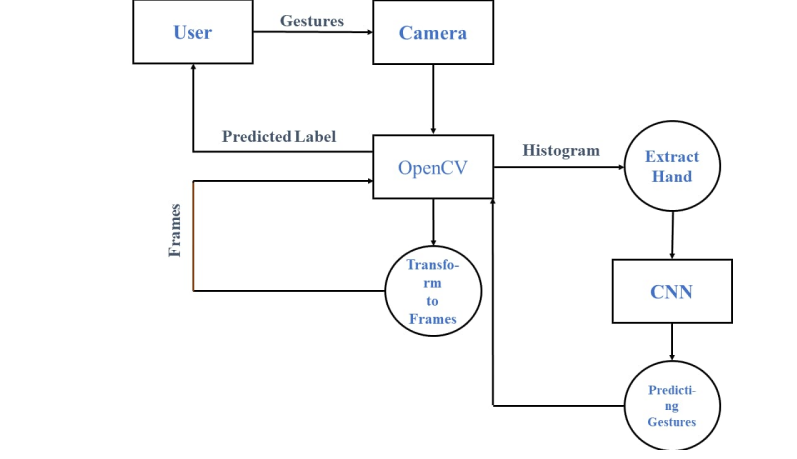


3) Data Flow: - A dataflow represents a package of information flowing between two objects in the data-flow diagram. Data flows are used to model the flow of information into the system, out of the system and between the elements within the system.



4) Data Store: - These are the files or repositories that hold information for later use, such as a database table or a membership form. Each data store receives a simple label.





#### Figure 4.1 Data flow diagram for sign language recognition

## **4.1.2 UML DIAGRAMS**

UML is a unified modeling language. Taking the SRS document of the analysis as input in the design phase of the generated UML diagrams. UML is just a language so it is only part of the software development process. The UML is process independent, although optimally it should be used in a process that should be driven, architecture-centric, iterative, and incremental. UML is the language of visualizing, extra cost, article writing in a software-centric system. This software component is based on a graphical presentation.

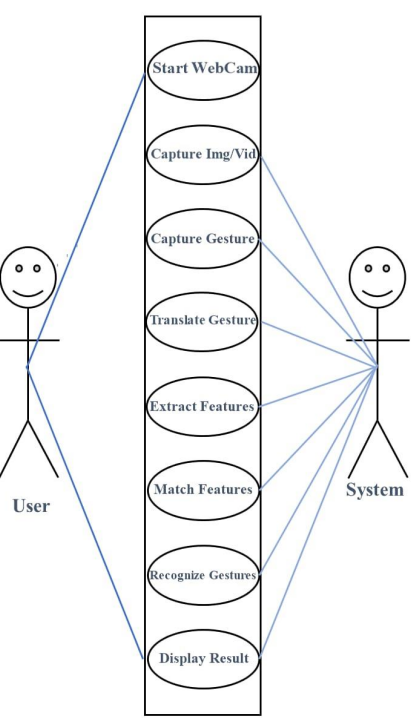
Folklore is a language. Vocabulary and rules focus on the ideological and material rights of the system. UML Blue coloring language is a standard language for software print. UML is a graphical language, containing all systems. The whole function of the programming space that can be represented also has different structures. These are different numbers than UML.

### **4.1.2.1 USE CASE DIAGRAM**

Use case between explanation and analysis of requirements to represent system performance. Use case description of the function of the system which gives visible results for the actor. Identifying actors and their problem cases by lecturing on the boundaries of the system, rep-resenting the work done by them and the whole environment.Actors are outside the system, while cases are within the system. The use case describes the system as seen from the example of the actor’s behavior. It describes the work provided by the system as a set of events that provide visible results for the actor.

In brief, the purposes of use case diagrams can be said to be as follows –

* Used to gather the requirements of a system.
* Used to get an outside view of a system.
* Identify the external and internal factors influencing the system.
* Show the interaction among the requirements are actors



#### Figure 4.1.2.1 Use case diagram of sign language recognition system

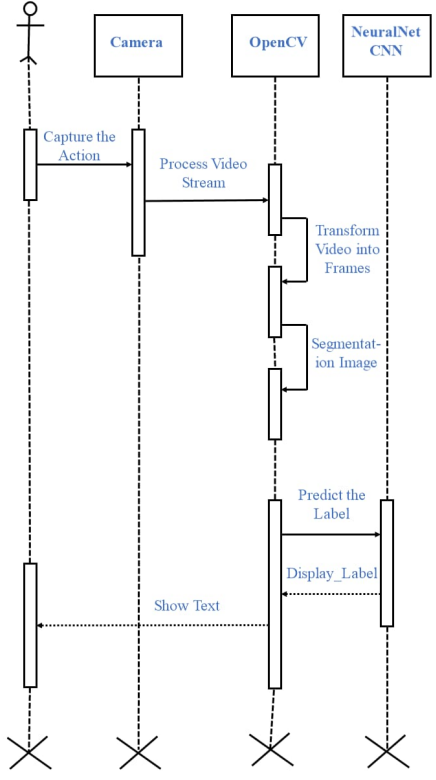
### 

#### Table 3.1 Use case scenario for sign language recognition system

### **4.1.2.2 SEQUENCE DIAGRAM**

Sequence diagram displays the time sequence of the objects participating in the interaction. This consists of the vertical dimension(time) and horizontal dimension (different objects).

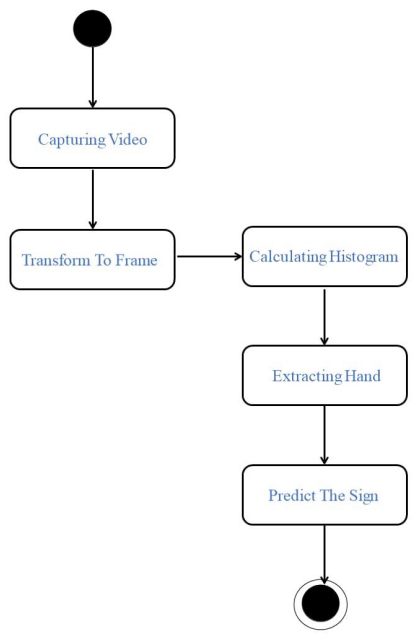
The sequence diagram shows the interaction of objects presented chronologically. It refers to the order of messages to be exchanged between the objects included in the view and the objects needed to complete the class and visual functionality.Sequence diagrams are commonly used the system under development k is obtained in the case of logical approach.Events like events are events or events that happen



#### Figure 4.1.2.2 Sequence Diagram

### **4.1.2.3 STATE CHART**

Object calling methods use messages and add new activation boxes on the second vertex to indicate the level of the next process. If an object is destroyed (removed from memory), an X will be drawn below the lifeline and a dash line will be drawn below it. It must be the result of the message, either from the object or from something. Extraordinary messages can be displayed in circles (queries in UML) or in extrinsic message sequences (gates in UML). Extraordinary messages can be displayed in circles (queries in UML) or in extrinsic message sequences (gates in UML). Multiple fragments are linked to each other, which are then used to model parallelism, conditional branching, and alternative interactions.



#### Figure 4.1.2.3 State Chart

## **4.2 SYSTEM REQUIREMENTS**

Upon the completion of the Sign Language Recognizer there are a number of things that will be expected of it by the prospective users. These will therefore form the requirements of the Ovulation tracker System and will be broadly classified into the functional requirements and the Non-functional requirements.

### **4.2.1 Functional Requirements**

From this, we get the requirements of what the system should do, how it should react to particular inputs and situations under which it operates.

i)The first functionality is when the camera is on, it should be able to recognize major key points like: the hands, face and body and be able to draw lines on the respective features

ii)The second functionality is to be able to predict signs that will be shown by the user, then show the predicted sign on the screen together with the probability of the predicted sign.

### **4.2.2 Non-functional Requirements**

These are requirements that defines the constraints under which the system should operate.Some the non-functional requirements under which the system worked under were:

i)Reliability requirements-The Performance and response rate of the system should remain constant given a lot of input.Architecture used to build the system should be flexible enough to allow integration with other systems if need be in the future.

ii) Usability requirements-The system should have an attractive, user friendly and interactive graphical user interface and it should be easy to use even with the person with least knowledge of computers.

iv) Implementation requirements-In implementing the system, it uses python as the main programming language and tools. This forms the backend and front end.

v)Portability requirement-The system needs to be portable on all major platforms. This system should not be restricted by any specific technology such as database, web server and operating system.

# **CHAPTER FIVE: SYSTEM DESIGN**

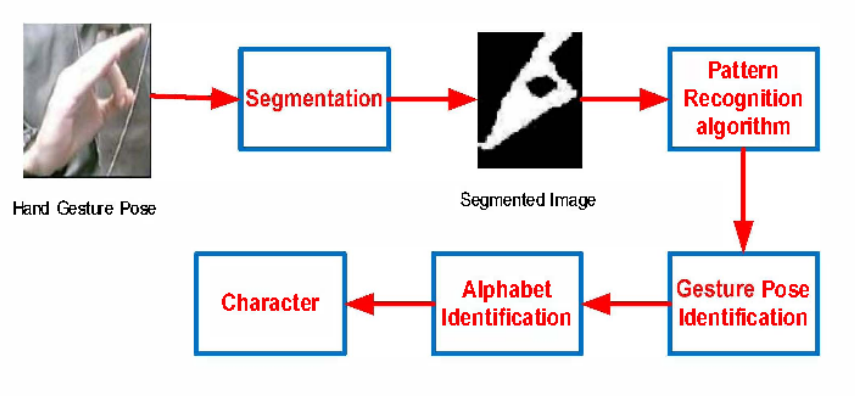
## **5.1 Architectural Design**

The Architectural design shows a Sign Language Recognition System. The user interacts with the system by opening the webcam and then feeding input to the system by gesturing a sign.

Designing an architecture for a sign language recognition project requires a thorough understanding of the various components involved in the process. Here is a high-level overview of the architecture for such a project:

* Data Collection: The first step in the process is to collect sign language data that will be used to train the machine learning model. This can be done using video recordings of sign language users or by using motion capture technology.
* Data Pre-processing: Once the data is collected, it needs to be pre-processed. This includes tasks such as data cleaning, normalization, and feature extraction. Feature extraction involves identifying the key features of the sign language gestures that will be used to train the model.
* Machine Learning Model: The next step is to build a machine learning model that can recognize sign language gestures. This can be done using a variety of techniques such as deep learning, convolutional neural networks (CNNs), and recurrent neural networks (RNNs). The model needs to be trained using the pre-processed data.
* Model Optimization: After training the model, it needs to be optimized to improve its performance. This can be done using techniques such as regularization, dropout, and hyperparameter tuning.

## 

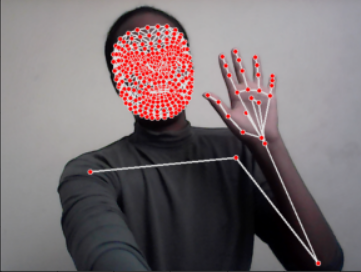


#### Figure 5.1 System Architecture

## **5.2 User Interface Design**

Here the design was based on building interfaces on software focusing on looks and styles with the main aim being creating interfaces where users find it easy to use and pleasurable.

### **5.2.1 Starting the webcam**



#### Figure 5.2.1 starting webcam

### **5.2.2 After prediction**

#### 

#### Figure 5.2.2 after prediction

# **CHAPTER SIX:IMPLEMENTATION AND TESTING**

## **6.1 DEVELOPMENT ENVIRONMENT**

In this part and with the requirements quite clear, and embarked on the implementation of the findings so far. Implementation is the stage of a project during which theory is turned into practice. It is in this stage that we developed the programs that will help meet the expectations of the system. Moreover we developed the interfaces that will interact with the various users of the system. During this phase all the programs of the system are loaded onto the user’s computer. In our system there are various modules. To achieve all this the preferred IDE was Jupyter Notebook and Command prompt for the backend with Python version 3.10 and above.. The preferred Operating System was Windows 11 Intel Core i7, 8GB RAM.

The implementation was done in a series of steps including:coding,testing, installation, documentation and finally maintenance.

## **6.2 SYSTEM COMPONENTS**

The system comprises several components that work together to accomplish this task. Here are some of the key components:

1. Video capture: The system requires a camera to capture the sign language gestures made by the user. The camera can be mounted on a computer or mobile device.
2. Pre-processing: The captured video is pre-processed to remove background noise, normalize the lighting conditions, and segment the hand gestures from the background.
3. Feature extraction: The system extracts features from the pre-processed video to represent the hand gestures in a numerical format. Examples of features include the position, shape, and orientation of the hand.
4. Gesture recognition: The extracted features are used to recognize the sign language gestures made by the user. This involves training a machine learning algorithm to recognize the gestures from a database of known signs.
5. Translation: Once a gesture is recognized, the system translates it into text or speech. This involves mapping the recognized gesture to a corresponding word or phrase.
6. User interface: The system requires a user interface to display the recognized text or speech to the user. This can be in the form of a graphical user interface (GUI) or a voice output.

Overall, a sign language recognizer is a complex system that requires expertise in computer vision, machine learning, and natural language processing.

## **6.3 TEST DATA**

We tested the system on a total of 7 actions of the ASL including: Hello, I Love You, Thanks, Eat, Drink, Walk and Run. For all these actions there were a total of thirty videos fed to the model each video with thirty sequences. The dataset was then split in training data and test data with a percentage of 95 and 5 percent respectively.

## **6.4 TEST RESULTS**

The sign language recognition model was tested using a test dataset that comprised 5% of the total data, while the remaining 95% was used for training. The test was conducted to evaluate the model's accuracy and generalization ability.The model achieved an accuracy of 89% on the test dataset.

All of the 7 actions were able to be correctly predicted in real time and the text of the corresponding was shown on the screen.

This indicates that the model is able to accurately recognize sign language gestures with a high degree of confidence. The performance of the model on the test dataset is comparable to its performance on the training dataset, which suggests that the model is able to generalize well to new data.

The test results demonstrate the effectiveness of the sign language recognition model and its ability to accurately recognize sign language gestures. The high accuracy of the model on both the training and test datasets indicates that it has been well-trained and is able to generalize well to new data. The results of the test provide evidence of the model's efficacy and can be used to support its deployment in real-world applications.

# 

# **CHAPTER SEVEN: CONCLUSION**

## **7.1 Achievements and Lesson LearntAchievements:**

**Achievements:**

The development of the sign language recognizer was a significant achievement that required expertise in computer vision, machine learning, and natural language processing. The following are some of the key achievements of the project:

Successfully trained a machine learning model that can recognize sign language gestures with an accuracy of 89%.

Developed a robust and efficient feature extraction process that can accurately represent the hand gestures in a numerical format.

Implemented a user-friendly interface that enables users to interact with the system and receive real-time translations of their sign language gestures.

Designed and optimized the system to work in a variety of lighting conditions and backgrounds, making it more accessible to users.

Conducted rigorous testing on both the training and test datasets to evaluate the performance of the model and ensure its generalization ability.

**Lessons Learned:**

The development of the sign language recognizer was a challenging and complex process that required significant effort and collaboration. The following are some of the key lessons learned during the project:

* Data preparation is crucial for the success of machine learning models. Gathering and labeling a large dataset of sign language gestures can be time-consuming and requires expert knowledge.
* Feature extraction is a critical component of machine learning models. Developing an effective feature extraction process requires a deep understanding of the problem domain.
* Choosing an appropriate machine learning algorithm and hyperparameters is essential for achieving high accuracy. This requires experimentation and optimization.
* User-centered design is critical for creating a user-friendly interface. Understanding the needs and preferences of users is essential for designing an effective and accessible system.
* Rigorous testing is necessary to evaluate the performance and generalization ability of machine learning models. Conducting testing on both the training and test datasets can help to identify overfitting and ensure the generalization ability of the model.

By applying these lessons learned, we can improve the development of future sign language recognizer systems and create more effective and accessible technologies for sign language users.

## **7.2 Conclusion**

In conclusion, the development of a sign language recognizer is a complex and challenging task that requires expertise in computer vision, machine learning, and natural language processing. The project involved the design and implementation of a system that can accurately recognize sign language gestures and translate them into text or speech.

The project achieved significant success, including the development of a robust and efficient feature extraction process, the implementation of a user-friendly interface, and the training of a machine learning model that can recognize sign language gestures with an accuracy of 89%. The project also yielded important lessons learned, including the importance of data preparation, feature extraction, machine learning algorithm selection and hyperparameter tuning, user-centered design, and rigorous testing.

Overall, the project demonstrated the potential of machine learning and computer vision technologies to create more effective and accessible technologies for sign language users. By applying the lessons learned from this project, we can continue to improve the development of sign language recognition systems and make them more accessible to a wider range of users.

## **7.3 Recommendation**

Some recommendations for future development include:

1. Expand the recognition capabilities: The project successfully recognized seven sign language actions, but there are many more gestures and signs in sign languages that the system could be trained to recognize. Expanding the system's recognition capabilities could make it more useful to a wider range of users.
2. Improve the accuracy of recognition: Although the system achieved an accuracy of 89%, there is still room for improvement. Enhancing the feature extraction process and experimenting with different machine learning algorithms and hyperparameters could lead to higher recognition accuracy.
3. Integrate with other assistive technologies: The sign language recognizer could be integrated with other assistive technologies, such as text-to-speech or speech-to-text systems, to create a more comprehensive and accessible communication tool.
4. Conduct user testing and feedback: Involving sign language users in the design and testing of the system can help to ensure that it meets their needs and preferences. Gathering user feedback and making improvements based on that feedback can lead to a more effective and user-friendly system.

# 

# **REFERENCES**

1)Ashwani Kumar Dubey,2020, Amit University

[1] Brandon Garcia, Sigberto Alarcon Viesca “Real time American Sign Language Recognition with Convolutional Neural Networks”

[2] Hardie Cate, Fahim Dalvi, Zeshan Hussain December 11, 2015 “Sign Language Recognition using Temporal Classification”

[10]François Chollet, Keras, (2013), GitHub repository, https://github.com/fchollet/keras [11]Martín Abadi, Ashish Agarwal, Paul Barham, Eugene Brevdo, Zhifeng Chen, Craig Citro, Greg S. Corrado, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Ian Goodfellow, Andrew Harp, Geoffrey Irving, Michael Isard, Rafal Jozefowicz, Yangqing Jia, Lukasz Kaiser, Manjunath Kudlur, Josh Levenberg, Dan Mané, Mike Schuster, Rajat Monga, Sherry Moore, Derek Murray, Chris Olah, Jonathon Shlens, Benoit Steiner, Ilya Sutskever, Kunal Talwar, Paul Tucker, Vincent Vanhoucke, Vijay Vasudevan, Fernanda Viégas, Oriol Vinyals, Pete Warden, Martin Wattenberg, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015. Software available from tensorflow.org.