**SIGN LANGUAGE RECOGNITION USING COMPUTER VISION**

**BY:**

**AUGUSTINE ADEIZA AROKE**

**SCI16CSC034**

**DEPARTMENT OF COMPUTER SCIENCE**

**FACULTY OF SCIENCE**

**FEDERAL UNIVERSITY LOKOJA,**

**KOGI STATE.**

**SEPTEMBER, 2021.**

**SIGN LANGUAGE RECOGNITION USING COMPUTER VISION**

**BY:**

**AUGUSTINE ADEIZA AROKE**

**SCI16CSC034**

**A PROJECT SUBMITTED TO THE DEPARTMENT OF COMPUTER SCIENCE, FACULTY OF SCIENCE, FEDERAL UNIVERSITY LOKOJA,**

**KOGI STATE.**

**IN PARTIAL FULFILLMENT OF THE REQUIREMENTS**

**FOR THE AWARD OF THE BACHELOR**

**OF SCIENCE (HONOURS) DEGREE**

**IN COMPUTER SCIENCE**

**SEPTEMBER, 2021**

**CERTIFICATION**

I hereby certify that I am solely responsible for the work submitted in this project to the department of computer science Federal University Lokoja, that the original work is mine, except as specified in acknowledgement and references and that neither the study nor the original work contained therein has been submitted to this University or any other institution for the award of a degree.

…………………………………

AUGUSTINE ADEIZA AROKE

(SCI16CSC034)

**APPROVAL**

This project, **SIGN LANGUAGE RECOGNITION USING COMPUTER VISION** has been approved for the Department of Computer Science, Faculty of Science, Federal University Lokoja, Lokoja, Nigeria

Supervisor ……………………………

PROF. SUNDAY ERIC ADEWUMI Signature

Head of Department ……………………………

PROF. FRANCISCA O. OLADIPO

External Examiner …………………………….

PROF. FRANCIS S. BAKPO Signature

**DEDICATION**

I dedicated this research to God almighty for all His mercies upon me and to my friends and family for always believing in me.

**ACKNOWLEDGMENTS**

I wish to express gratitude to my supervisor Prof. Sunday E. Adewumi for not just being a supportive supervisor but also being a father figure. I also wish to appreciate Mr AbdulMalik Rufai for his support all through my period as an undergraduate student and for always being there for me. I will be ingrate not to mention my wonderful lecturers Mr Olalekan Ihinkalu, Mr Ahmad Shehu, Dr Taiwo Kolajo, Dr Emeka Ogbuju and my wonderful Head of Department Prof. Francisca O. Oladipo for always challenging me to be a better student.

Lastly, I want to appreciate my family, the entire Lawals for all the support and love throughout this program.

TABLE OF CONTENTS

TITLE PAGE i

CERTIFICATION ii

APPROVAL iv

DEDICATION v

ACKNOWLEDGEMENTS vi

TABLE OF CONTENTS vii

LIST OF FIGURES x

ABSTRACT xi

[1.1 BACKGROUND OF THE STUDY 1](#_Toc85114754)

[1.2 STATEMENT OF PROBLEM 2](#_Toc85114755)

[1.3 AIM AND OBJECTIVES 2](#_Toc85114756)

[1.4 SIGNIFICANCE OF THE STUDY 2](#_Toc85114757)

[1.5 SCOPE OF THE STUDY 3](#_Toc85114758)

[1.6 LIMITATIONS OF THE STUDY 3](#_Toc85114759)

[LITERATURE REVIEW 4](#_Toc85114760)

[2.1 AUTOMATIC SIGN LANGUAGE RECOGNITION 4](#_Toc85114761)

[2.1.2 DEFINITION OF SPECIFIC RESEARCH CONCEPTS. 4](#_Toc85114762)

[2.3 REVIEW OF RELATED RESEARCH WORKS AND SYSTEMS ON AUTOMATIC SIGN LANGUAGE RECOGNITION. 5](#_Toc85114763)

[CHAPTER THREE 24](#_Toc85114764)

[SYSTEM ANALYSIS AND METHODOLOGY 24](#_Toc85114765)

[3.0 INTRODUCTION 24](#_Toc85114766)

[3.1 DESCRIPTION OF EXISTING SYSTEM 24](#_Toc85114767)

[3.2 REVIEW OF THE EXISTING SYSTEM 25](#_Toc85114768)

[3.2.1 PROBLEMS OF THE EXISTING SYSTEM 25](#_Toc85114769)

[3.3 DESCRIPTION OF THE PROPOSED SYSTEM 25](#_Toc85114770)

[3.3.1 ADVANTAGES OR JUSTIFICATION OF THE PROPOSED SYSTEM 25](#_Toc85114771)

[3.3.2 FEASIBILITY STUDIES OF THE PROPOSED SYSTEM 26](#_Toc85114772)

[3.4 REQUIREMENT ELICITATION 26](#_Toc85114773)

[3.4.1 FUNCTIONAL REQUIREMENTS OF THE PROPOSED SYSTEM 26](#_Toc85114774)

[3.4.2 NON-FUNCTIONAL REQUIREMENT OF THE PROPOSED SYSTEM 27](#_Toc85114775)

[3.5 HIGH-LEVEL MODEL OF THE SIGN LANGUAGE RECOGNITION SYSTEM 27](#_Toc85114776)

[3.7 METHODOLOGY 30](#_Toc85114777)

[3.7.1 SPECIFICATION AND JUSTIFICATION FOR THE SELECTED METHODOLOGY 30](#_Toc85114778)

[3.7.2 PROJECT PLAN 32](#_Toc85114779)

[SYSTEM DESIGN AND IMPLEMENTATION 33](#_Toc85114780)

[4.1 INTRODUCTION 33](#_Toc85114781)

[4.2 OBJECTIVES OF THE DESIGN 33](#_Toc85114782)

[4.3 ARCHITECTURAL DESIGN OF SIGN LANGUAGE RECOGNITION SYSTEM 34](#_Toc85114784)

[4.4.1 PHYSICAL DESIGN 35](#_Toc85114785)

[4.4.1.1 INPUT DESIGN 35](#_Toc85114786)

[4.4.1.2 OUTPUT DESIGN 36](#_Toc85114787)

[4.4.2 LOGICAL DESIGN 36](#_Toc85114788)

[4.4.2.1 USE CASE DIAGRAM 36](#_Toc85114789)

[4.5 APPLICATION ALGORITHM 37](#_Toc85114790)

[4.6 THE USER INTERFACE DESIGN 38](#_Toc85114791)

[4.7 SYSTEM IMPLEMENTATION 38](#_Toc85114792)

[4.7.1 SYSTEM REQUIREMENTS 40](#_Toc85114793)

[4.8 IMPLEMENTATION TOOLS USED 40](#_Toc85114794)

[4.9 DEVELOPMENTAL PROCESS FOR THE SIGN LANGUAGE RECOGNITION SYSTEM. 41](#_Toc85114795)

[4.10 PROGRAM MODULE SPECIFICATION 42](#_Toc85114796)

[CHAPTER 5 43](#_Toc85114797)

[SUMMARY, RECOMMENDATION AND CONCLUSION 43](#_Toc85114798)

[5.1 SUMMARY 43](#_Toc85114799)

[5.1.1 REVIEW OF CONTRIBUTIONS AND ACHIEVEMENTS 43](#_Toc85114800)

[5.2 RECOMMENDATIONS 43](#_Toc85114801)

[5.2.1 FUTURE RESEARCH AND WORK 44](#_Toc85114802)

[5.3 CONCLUSION 44](#_Toc85114803)

[APPENDIX A 53](#_Toc85114804)

[APPENDIX B 59](#_Toc85114805)

**TABLE OF FIGURES**

Fig 3.1: High-level Model of the Proposed System 28

Fig 3.2: Dataflow of the Proposed System 29

Fig 3.3: Connection of Neurons in CNN 31

Fig 3.4 Gantt chart depicting project plan 32

Fig 4.1 Architectural design of the system 34

Fig 4.2 Input design 35

Fig 4.3 Output design 36

Fig 4.4 Use Case diagram of sign language recognition system 37

Fig 4.4 Use Interface design 38

Fig 4.6 Development phase of sign language recognition system 41

**ABSTRACT**

Communication has always been a basic part of human. Without communication, the world would all be chaotic because man will lack understanding of his neighbor. This research was carried out to make communication better on social platforms by making it possible for the physically disabled (deaf and dumb) to communicate effectively with the non-disabled through the use of a sign to text converter. This research was taken majorly for the deaf. There is an exclusion they face in communication especially while on social media as they are forced to type everything every time they wish to convey a message. This can be stressful and draining on their part. To achieve this task, machine learning was used. We use the convolutional neural network to achieve this task while levering on the TensorFlow object detection API. Currently, we were able train for just 10 signs and achieve an accuracy of 94.6% while training on an Intel Core i3-2330M CPU with a processor with a clock speed of 2.20 GHZ.

**CHAPTER ONE**

**INTRODUCTION**

# **1.1 BACKGROUND OF THE STUDY**

Communication can be seen as the foundation of society. It has assisted man to have a greater knowledge of his neighbor, and hence its significance cannot be overstated. Despite the significance of communication in modern culture, there is still a significant divide between the hearing community and the hearing impaired or deaf community.

According to Beena and Namboodiri (2017), in our daily lives, communication across different cultures is heavily reliant on human-based translation, which is not always easily available and can be fairly costly. This challenge is exacerbated by the fact that only a small percentage of Nigerians understand and communicate in sign language. An automatic sign recognition system can help bridge the communication gap. Beena and Namboodiri (2017) claim that the sign language to text converter minimizes or bridges the gap between normal people and deaf and dumb individuals.

This method will make it easier for deaf and hard-of-hearing persons to communicate on social media platforms by providing a way to convert sign language to text. Through the use of computer vision, an automatic sign language recognition system can be built.

An automatic sign recognition system can help bridge the communication gap. The sign language to text converter decreases or bridges the gap between hearing and deaf (Beena and Noodiri, 2017).

This method will aid deaf and dumb people in communicating on social media platforms by providing an alternative to converting sign language to text. It will also improve automatic sign language identification through the use of computer vision.

In 1977, a robotic hand was developed that could translate alphabets into finger spellings, paving the way for automatic sign language recognition. Gloves with motion sensors became a possibility as a result of further technological advancements. Cyber Glove and VPL Data Glove were both born as a result of this innovation. With the help of computer hardware, this wearable hardware allowed for the capturing of signers' hand forms and movements.

Nonetheless, with the advancement of computer vision, wearable devices were supplanted with cameras, which were more efficient and imposed fewer physical restrictions on the signer. The data collected from the camera is now processed using neural networks.

# **1.2 STATEMENT OF PROBLEM**

The challenge of communication between the physically challenged (deaf and dumb) and non-physically challenged (speaking and hearing) is of paramount concern as communication from the side of the physically challenged is quite slow since they have to manually type everything they wish to communicate with the other party. Thus, the need to enhance this method of communication by creating a model that accepts signs from the physically challenged and converts them to text. Automatic sign language recognition technology is needed to make communication easier for deaf and dumb individuals on social media platforms.

# **1.3 AIM AND OBJECTIVES**

This study aims to design a system that recognises sign language with the following objectives.

1. To identify data sets that can be used to simulate sign language.
2. To develop a model that recognises sign language using computer vision.
3. Evaluate the data and give results in textual form.

# **1.4 SIGNIFICANCE OF THE STUDY**

1. The study introduces a system that will help in easing communication for deaf and dumb people on social media platforms.
2. It also improves sign language recognition by using just computer vision.

# **1.5 SCOPE OF THE STUDY**

This study is for the development of a system is to recognize basic words in sign language and not sentences.

# **1.6 LIMITATIONS OF THE STUDY**

1. Inability to integrate the system and the model.
2. The limited time frame for the work.
3. Limited access to data for internet connectivity.

**CHAPTER TWO**

# **LITERATURE REVIEW**

# **2.1 AUTOMATIC SIGN LANGUAGE RECOGNITION**

Manual and non-manual signing movements are used in sign languages. Non-manual signs consist of broader movements of the head and torso, as well as facial expressions, whereas manual signs consist of gestures that are isolated simply to the hand and arm. The majority of lexical meaning can be conveyed through manual signs.

The hand's shape, orientation, location, and movements are the essential components of manual signs.

Furthermore, there are movements between individual signs that are not part of the sign and do not always transmit any meaning when signs are composed to form sentences. More specifically, some signals may include situations where one finger obscures another, or if both hands are employed, one may obscure the other. It's also worth noting that signs aren't flat; they exist in three-dimensional locations, which must be taken into account while collecting data for an ASLR system.

# **2.1.2 DEFINITION OF SPECIFIC RESEARCH CONCEPTS.**

Machine learning is a branch of artificial intelligence that aims at making machines to be able to perform their jobs skillfully, without the aid of intelligence software. It is an interdisciplinary branch.

According to (Mohammed et al., 2016) machine learning constitutes of different learning for various research problems. These learning methods include supervised learning, unsupervised learning, semi-supervised learning, and reinforcement learning. For this research work, we will be using supervised learning.

Machine learning has been widely used in a variety of fields. Market segmentation, credit risk analysis, consumer profiling, fraud detection, picture recognition, and medical diagnosis are some of the most common (Tzanis, Vlahavas and Katakis, 2016).

Computer vision is also required for this research. Computer Vision is the science of using images to make helpful judgements about real-world objects and scenes. It has to do with extracting world descriptions from images or sequences of images. It also examines the image and generates descriptions that might be used to interact with the real world. Optical character recognition, facial detection, grin detection, object recognition, and vision-based biometrics are only a few of the applications of computer vision.

# **2.3 REVIEW OF RELATED RESEARCH WORKS AND SYSTEMS ON AUTOMATIC SIGN LANGUAGE RECOGNITION.**

As stated earlier, machine learning has a wide range of applications which includes sign language recognition, sentiment analysis, and various types of classification and detection cases.

**2.3.1 Towards an Automatic Annotation of French Sign Language Videos: Detection of Lexical Signs.**

This study outlines a method for developing an automatic annotation system for French sign language (LSF). This annotation aims to reduce the processing time and subjectivity of manual annotations used by linguists to examine and index signs for automatic recognition. The researchers were more concerned with the sign frequencies and lengths in this publication. It was determined that lexical signs make up 66.99 per cent of the signs in the database. According to the researcher, we should test our approach on different sign languages to see how versatile it is. *(Chaaban et al., 2019)*

**2.3.2 Sign Language Recognition and Translation with Kinect**.

This research looks at how Chinese sign language (CSL) is recognized and translated into text. It focuses on vision-based sign language identification and aims to provide rapid and accurate 3D SL recognition using Kinect's depth and colour data *(Chai et al., 2013)*

**2.3.3 American Sign Language Recognition with Kinect.**

This research looks into the potential of the Kinect depth mapping camera for sign language recognition and verification in educational games. This solution eliminates the requirement for users to wear gloves and strap sensors to their wrists, which is the most major disadvantage of the present copycat system in use. The dataset for this project was gathered from three sources: the kindest while the signer was seated, Kinect while the signer was standing, and with gloves on. The Kinect data with the signer standing produced the highest accuracy (88.02) for words and (76.12) for sentences. Future work can be done to create a real-time version of the copycat game that employs a Kinect sensor and is appropriate for deaf children. *(Zafrulla et al., 2011)*

**2.3.4 Sign Language Recognition using a Combination of new Vision-Based Features.**

This study describes an experimental system for recognizing manual movements in American Sign Language. The system is made up of modules for hand detection, tracking, feature extraction, and an HMM classifier. The highest error rate in this study was 10.90%. *(Zaki and Shaheen, 2011)*

**2.3.5 Vision-Based Portuguese Sign Language Recognition System.**

The researchers developed a real-time system capable of reading Portuguese Sign Language. Although the implemented system was simply trained to recognize vowels, it was able to understand Portuguese sign language and recognize Portuguese vowels in real-time with an accuracy of 99.4 per cent with one dataset of features and an accuracy of 99.6 per cent with a second dataset of features. The Support Vector Machine algorithm was used (SVM). More work can be done on occlusion because the model is unable to recognize a sign when there is an object behind the hand. (*Trigueiros et al., 2014)*

**2.3.6 An Assertive Framework for Automatic Tamil Sign Language Recognition System Using Computational Intelligence.**

According to the findings of this study, researchers used unsupervised learning to construct an automatic Tamil sign language recognition system for the acquisition and exchange of knowledge among the hearing-impaired population. The fundamental motivation for this research project stems from the discovery by the researcher that a significant proportion of the Tamil community suffers from hearing impairment. Mat lab was utilized in the construction of the system. *(Krishnaveni et al., 2019)*

**2.3.7 An Automatic Arabic Sign Language Recognition System Based on Deep CNN: An Assistive System for the Deaf and Hard of Hearing.**

According to this study, the deep convolutional neural network architecture that has been developed will be used to construct an Arabic sign language recognition system. Using a collection of over 50 thousand Arabic sign photos gathered from random volunteers of all ages, the deep CNN architecture was trained and tested, and the system was able to recognize the signs of the Arabic alphabet based on real-time input from the user. The proposed deep CNN architecture, according to the researchers, generated exceptional accuracy of 97.6 percent, which is higher than the accuracy of past studies in a similar context. The researcher believes that future studies could focus on developing algorithms that improve the results while also increasing the size of the data, which would increase the variety of images available in terms of illumination, noise, and other factors, as well as developing a better method to translate words and sentences between different languages.. *(Latif et al., 2020)*

**2.3.8 Bengali Sign Language Recognition Using Deep Convolutional Neural Network.**

This research presents a novel method for Bengali sign language recognition that makes use of a deep convolutional neural network. This method was used to recognize static hand signs of 37 Bengali alphabet letters using a deep convolution neural network and utilizing learn futures from a pre-trained network as well as fine-tuning the top layer of this network, yielding a high overall recognition rate of 96.33 percent on the training dataset and 84.68 percent on the validation dataset, resulting in a high overall recognition rate of 96.33 percent on the training dataset and 84.68 percent on the validation dataset. *(Hossen et al., 2018)*

**2.3.9 Word-Level Deep Sign Language Recognition from Video: A New Large-Scale Dataset and Methods Comparison.**

The researchers provide a new large-scale word-level American sign language dataset encompassing more than two thousand words performed by more than 100 signers, which they call the American Sign Language Dataset. A large-scale dataset allows the researcher to experiment with a variety of deep learning algorithms for word-level sign language recognition study because of the large-scale dataset. A further statement by the researcher is that future work will explore the possibility of using word-level annotations to aid in machine sign language identification at the sentence and story level. *(li et al., 2019)*

**2.3.10 Vision-Based Approach for American Sign Language Recognition Using Edge Orientation Histogram.**

In this work, a system for detecting American Sign Language is developed that does not require the use of colored gloves or expensive sensory gloves, as was previously done. In tough backgrounds with varying lighting, the system was able to recognize 88.36 percent of the objects in 0.5 seconds on average. *(Pansare and Ingle, 2016)*

**2.3.11 Recurrent Convolutional Neural Network for Continuous Sign Language Recognition by Staged Optimization.**

In this study, a novel approach for real-world sign language identification is developed from continuous image streams, which is described in detail elsewhere. In addition, the researcher devised a stage optimization approach for training the deep neural network architecture, which was implemented. They successfully used the representation capabilities of CNN by adjusting on a large number of gross level segments and avoided overfitting by using a deep architecture that was not overfit. On a challenging benchmark, the approach's effectiveness was further shown, with the results being comparable to those obtained by the state of the art. *(Cui et al., (2016)*

**2.3.12 Automatic Recognition of Sign Language Structures in Rgb Videos: The Detection of Pointing and Lexical Signs.**

This paper proposes a generic technique to continuous sign language detection in conventional RGB sign language movies, which can be applied to a variety of situations. This research, in contrast to previous studies, does not only examine sign language at the lexical level; rather, it makes use of both lexical and SL-specific elements that were accurately recognized and localized. After that, a recurrent neural network was trained on the generic features to recognize vocabulary and SL-specific structures such as pointing, which was subsequently tested. After being applied to the French sign language Corpus Dicta sign, pointing detection obtained a sequence-wise F1 score of 78 percent on four-second chunks, and the network also obtained a sequence-wise F1 score of 70 percent for detection of lexical signs with less than 50 instances in the training set. The researchers went on to say that they believe that fine-tuning the network, adding new pre-computed features, and pre-processing techniques can all help to enhance performance. *(Valentin et al., 2019)*

**2.3.13 A Survey on Mouth Modeling and Analysis for Sign Language Recognition.**

The researcher discusses the results of the first survey on mouth non-manuals in automatic sign language recognition in this paper. It goes on to demonstrate the significance of lip motion and relevant methodologies employed in automatic sign language identification in the course of the research. On which this research is predicated is the notion that sign languages are composed of fundamental semantic components known as phonemes, which are largely represented through combinations of physical features such as hand form and position, as well as location and motion. Most of the prosody in sign languages is transmitted through non-manual signs, such as head and torso position, facial expression (combination of eyes, eyebrows, cheeks, and lips), which plays an important part in lexical articulation and is utilized to transmit most of the prosody. *(Antonakos et al., 2015)*

**2.3.14 Sign Language Recognition Based on Hand and Body Skeletal Data.**

According to the findings of this work, the researcher has developed a deep learning-based system for identifying sign language from video sequences that is both accurate and robust. This approach makes use of the body and skeletal traits that may be extracted from RGB videos. As a result, there is no need for additional features such as gloves, which could restrict the signer's ability to move. The fact that the methodology was able to classify both one-handed and two-handed signs, despite the fact that the proposed system does not employ data from the left hand, is worth pointing out. (Konstantinidis *et al., 2018)*

**2.3.15 A Novel Approach to American Sign Language (ASL) Phrase Verification Using Reversed Signing.**

This work presents a novel technique for American Sign Language phrase verification that incorporates confidence measures and emerges from aligning reverse sign models to the same output. The strategy is based on aligning reverse sign models to the same output. It was decided to construct a game clone for children's death signs that required American Sign Language verification in order to allow involvement. According to the study, the most significant advantage of using the new method is that the alignment chosen from the sign matches the ground truth more exactly than the previous method. Most previous recognition algorithms are just concerned with whether or not an alignment can be produced; they are not concerned with whether or not the alignment is proper. *(Brashaer et al., 2010)*

**2.3.16 Sign Language Recognition Using Convolutional Neural Networks.**

An object recognition system based on the Microsoft Kinect, a convolutional neural network, and GPU acceleration is presented in this research. The process of feature building is depicted here. The researcher was able to distinguish 20 Italian gestures with great accuracy, and the model was able to generalize to users and surrounding environments that were not present during training with an accuracy of 91.7 percent, according to the results. *(Pigou et al., 2015)*

**2.3.17 Learning Motion Disinfluencer for Automatic Sign Language Segmentation.**

In this work, the researcher introduces a novel technique for automatically detecting word boundaries inside continuous phrase expressions in Japanese sign language by exploiting three-dimensional body joint locations. A boundary random forest model was trained on the extracted attributes in order to automate the synthesis of words from motion sequences. According to the findings, the algorithm is capable of distinguishing between dynamic and static motion phases and achieves segmentation accuracy that is comparable to that of previous state-of-the-art systems. This study should be extended to more data sets to determine how well it performs on different sign languages and data sizes; it should also be utilized in a two-stage classifier for continuous sign motion data because it promises to enhance recognition accuracy in this difficult situation*. (Farag and Brock, 2018)*

**2.3.18 Experimental Framework Design for Sign Language Automatic Recognition**

In this work, the authors examine the difficulties associated with developing an experimental framework for machine learning-based Automatic Sign Language Recognition. The framework was developed in order to analyze the consistency of current data sets as well as CNNs that have been trained experimentally in order to improve their performance. The system achieved greater than 50% precision for tough data sets recorded in both natural and artificial environments. Other possible future studies could focus on improving system training to match or exceed accuracy rates for both depth and RGB images separately, using all available data sets, and on the construction of two computer apps, one for recording and the other for real-time gesture detection. *(Santiago et al., 2018)*

**2.3.19 Datasets of Pakistani Sign Language and Automatic Recognition of Hand Configuration of Urdu Alphabet through Machines Learning**

There is a communication gap between the deaf and the deaf and non-deaf communities, as discovered by the researcher in this case. The researcher was able to construct datasets from images taken using a camera and uploaded to a computer. They even went so far as to develop an Android mobile application based on machine learning to help bridge the communication gap between the two parties. The Support Vector Machine (SVM) methodology was used in this investigation. *(Imran et al., 2021).*

**2.3.20 Large Scale Sign Language Interpretation.**

In his research, the researcher claims to have amassed one of the world's largest data sets to date, consisting of 50000 video clips chosen from a pool of 10,000 signers and compiled into a single database. The sequence-to-sequence deep learning approach was also utilized to automatically convert Chinese sign language into written Mandarin and English text, which was a first for the field. This model takes into account elements such as body joint position, facial expression, and finger articulation. Existing deep learning algorithms can overfit the training split, but they are unable to generalize sufficiently on the more difficult Chinese sign language data, as revealed by the analysis component of the datasets. *(Yuan et al., 2019)*

**2.3.21 An Automatic Arabic Sign Language Recognition System (ArSLRS)**

A visual sign language recognition system that automatically transforms isolated Arabic word signals to text is described in this research. Among the four primary stages of the proposed system are hand segmentation and tracking as well as feature extraction and classification. Hand segmentation is accomplished by the use of a dynamic skin detector that is dependent on the color tone of the face. In the following step, the hands are detected and tracked using a skin-blob tracking technique that has been proposed. When analyzing the suggested system, a dataset of 30 isolated words that are used in the daily school life of hearing-impaired children was produced, taking into consideration that 83 percent of the words exhibit different occlusion statuses. Experimental results demonstrate that the proposed system has a recognition rate of 97 percent when used in the signer-independent mode. *(Nada et al., 2017)*

**2.3.22 Study of Vision-Based Hand Gesture Recognition Using Indian Sign Language**.

This study project examines the historical basis, the necessity for, the breadth, and the concerns associated with the Indian sign language. This paper goes on to discuss a research of sign language interpretation systems in relation to vision-based hand gesture recognition, which is the subject of this work. The Bayesian classifier was used to segment the skin, although the researcher claimed that other unsupervised methods, such as K-means clustering, might also be employed to do the task. In their research on computer vision, the researchers present a variety of methods for improving accuracy. *(Ghotkar & Kharate, 2014)*

**2.3.23 Automatic Arabic Sign Language Recognition: A Review, Taxonomy, Open Challenges, Research Roadmap, And Future Directions.**

This paper discusses the research background and foundations of Arabic Sign Language Recognition (ArSLR), as well as the applications of ArSLR. The taxonomies for ArSLR research, databases, open questions, future research trends and directions, and a roadmap to ArSLR research are all offered. The primary taxonomy investigated in this review is related to the capturing mechanism of the gestures for ArSLR, which can be either a Vision-Based Recognition (VBR) approach or Sensor-Based Recognition (SBR) approach, and the second taxonomy investigated is related to the type and task of the gestures for ArSLR, which can be either the Arabic alphabet, i.e., writing, or pointing, or the Arabic numerals, i.e., drawing.

*(Al-Shamayleh et al., 2020)*

**2.3.24 Sign Language Video Analysis for Automatic Recognition and Detection**

This research looks at collecting essential information from a Sign Language (SL) video and building methods to detect lexical signs or even high-level linguistic aspects. The researcher concluded that by using image processing methods such as CNNs and sequence processing tools such as LSTMs, it is now possible to detect high-level linguistic elements in continuous sign language movies in an automated manner. They also suggested that additional work might be done to create an RNN-based SLR system that would allow them to search through a large number of sign language videos. *(Valentin, 2019*)

**2.3.25 American Sign Language Recognition Using Microsoft Kinect**

A new method for American Sign Language (ASL) alphabet identification is introduced in this work, which makes use of a low-cost depth camera, such as Microsoft's Kinect. A segmented hand configuration is initially created by applying a depth contrast feature-based per-pixel classification method to a depth contrast feature-based per-pixel classification algorithm. Once this was accomplished, a hierarchical mode-seeking algorithm was designed and implemented in order to localize hand joint positions while adhering to kinematic restrictions. Finally, a Random Forest (RF) classifier was developed to recognize American Sign Language (ASL) signs based on joint angles. Based on the results, it appears that our system is capable of achieving greater than 90% accuracy in detecting 24 static ASL alphabet signs. *(Dong et al, 2015)*

**2.3.26 Real-Time Sign Language Recognition Using a Customer Depth Camera**

A highly exact approach for recognizing static gestures using depth data, provided by sensing technologies such as time of flight and structured light cameras, is proposed in this work by the researchers. The depth pictures are utilized to derive rotation-, translation-, and scale-invariant features that are not affected by rotation, translation, or scale. It is subsequently trained to classify the feature vectors with the use of a multi-layered random forest (MLRF), which results in an identification of the hand signs. By validating their algorithm on simulated data, such as 24 signs from American Sign Language (ASL), and on a new dataset, which was obtained using the recently released Intel Creative Gesture Camera, the researchers hoped to demonstrate the advantages of their technique to other researchers. *(Kuznetsova et al., 2013)*

**2.3.27 Automatic Finger Spelling using Convolutional Neural Network: Analysis**

The researchers in this study are primarily concerned with the recognition of static movements in American Sign Language that have been collected using the Microsoft Kinect sensor. The researchers went on to say that the creation of a good classifier that can accurately identify the input gestures is the most difficult component of the design of an autonomous sign language recognition system, which they believe is the most difficult aspect of the design. The CNN architecture from Kinect depth pictures is used in the system design for this work, and the classifier for sign language identification is based on this architecture. *(Beena & Namboodiri, 2017).*

**2.3.28 Bangla language modelling algorithm for automatic recognition of hand-sign-spelt Bangla sign language.**

There are two phases to the algorithm presented in this study, which is a Bangla language modeling approach for the automatic recognition of hand-sign-spelled Bangla sign language. The first phase is intended for hand-sign classification, and the second phase is intended for the Bangla language modelling algorithm (BLMA), which is intended for automatic recognition of hand-sign-spelled Bangla sign language (Hand-sign spelling). Based on 52 hand-signs of Bangladesh Sign Language, the researcher devised the Bangla language modelling algorithm (BLMA), which uncovers all "hidden characters" by comparing them to "recognized characters." Work on this project was carried out in collaboration with CNN and KNN with 500 words, 100 composite numerals, and 80 sentences in Bangladesh Sign Language as input, the system is put through its paces for the BLMA, with mean accuracy rates of 93.50 percent, 95.50 percent, and 90% achieved in the tests. *(Rahaman et al, 2019).*

**2.3.29 Quantitative Survey of the State of the Art in Sign Language Recognition.**

In this work, the researchers present a meta study covering around 300 published sign language recognition papers with over 400 experimental results. It includes most papers between the start of the field in 1983 and 2020. The analyzed papers have been manually labeled with a set of categories. The data reveals many insights, such as, among others, shifts in the field from intrusive to non-intrusive capturing, from local to global features and the lack of non-manual parameters included in medium and larger vocabulary recognition systems. Conclusively, this paper tries to present the best quantitative approach to sign language recognition. *(Oscar, 2020).*

**2.3.30 Finger Detection for Sign Language Recognition**

This paper introduces an efficient and fast algorithm for identification of the number of fingers opened in a gesture representing an alphabet of the American Sign Language. Finger Detection is accomplished based on the concept of Boundary Tracing and Finger Tip Detection. The system does not require the hand to be perfectly aligned to the camera or use any special markers or input gloves on the hand. The system had a 5% error in the gesture recognition algorithm which the researchers believe is okay since the dictionary can be used to correct any error resulting from the system. The researchers also believe that sensor-based contour analysis can be employed to detect which fingers are open in the future.  *(Ravi et al., 2009*).

**2.3.31 The Significance of Facial Features for Automatic Sign Language Recognition.**

This paper describes a vision-based recognition system that employs both manual and facial features, extracted from the same input image. For facial feature extraction an active appearance model is applied to identify areas of interest such as the eyes and mouth region. Afterwards a numerical description of facial expression and lip outline is computed. An extensive evaluation was performed on a new sign language corpus, which contains continuous articulations of 25 native signers. The obtained results proved the importance of integrating facial expressions into the classification process. The recognition rates for isolated and continuous signing increased in signer-dependent as well as in signer independent operation mode. *(Agris et al., 2008).*

**2.3.32 Real-Time American Sign Language Recognition Using Desk and Wearable Computer Based Video**.

In this research, two real-time hidden Markov model-based systems for recognizing sentence-level continuous American Sign Language (ASL) using a single camera to track the user's unadorned hands is presented. The first system observes the user from a desk mounted camera and achieves 92%-word accuracy. The second system mounts the camera in a cap worn by the user and achieves 98% accuracy (97% with an unrestricted grammar). Both experiments use a 40-word lexicon.

*(Starner et al., 1998).*

**2.2.33 Handshapes and Movements: Multiple-Channel American Sign Language Recognition**

In this paper the researchers present a framework for recognizing American Sign Language (ASL). They further discuss the main challenges in developing scalable recognition systems which are to devise the basic building blocks from which to build up the signs, and to handle simultaneous events, such as signs where both the hand moves and the handshape changes. The latter challenge is particularly thorny, because a naive approach to handling them can quickly result in a combinatorial explosion. The researchers loosely follow the Movement-Hold model to devise a breakdown of the signs into their constituent phonemes, which provide the fundamental building blocks. They also show how to integrate the handshape into this breakdown, and discuss what handshape representation works best. To handle simultaneous events, they split up the signs into a number of channels that are independent from one another. We validate our framework in experiments with a 22-sign vocabulary and up to three channels. *(Vogler & Metaxas, 2014).*

**2.3.34 Sign language recognition using sensor gloves.**

This paper examines the possibility of recognizing sign language gestures using sensor gloves. Previously sensor gloves are used in games or in applications with custom gestures. This paper explores their use in Sign Language recognition. This is done by implementing a project called “Talking Hands”, and studying the results. The project uses a sensor glove to capture the signs of American Sign Language performed by a user and translates them into sentences of English language. Artificial neural networks are used to recognize the sensor values coming from the sensor glove. These values are then categorized in 24 alphabets of English language and two punctuation symbols introduced by the author. So, mute people can write complete sentences using this application. *(Medhi & Khan, 2002).*

**3.3.35 Sign Language Transformers: Joint End-to-End Sign Language Recognition and Translation**

In this paper the researchers proposed Sign Language Transformers, a novel transformer-based architecture to jointly learn sign language recognition and translation in an end-to-end manner. They utilized Connectionist Temporal Classification (CTC) loss to inject gloss level supervision into the transformer encoder, training it to do sign language recognition while learning meaningful representations for the end goal of sign language translation, without having an explicit gloss representation as an information bottleneck. They further stated that future work could be done to expand the approach to model multiple sign articulators, namely faces, hands and body, individually to encourage our networks to learn the linguistic relationship between them. *(*Camgöz et al., 2020*)*

**2.3.36 Sign language recognition using sub-units**

This paper discusses sign language recognition using linguistic sub-units. It presents three types of sub-units for consideration; those learnt from appearance data as well as those inferred from both 2D or 3D tracking data. These sub-units are then combined using a sign level classifier; here, two options are presented. The first uses Markov Models to encode the temporal changes between sub-units. The second makes use of Sequential Pattern Boosting to apply discriminative feature selection at the same time as encoding temporal information. This approach is more robust to noise and performs well in signer independent tests, improving results from the 54% achieved by the Markov Chains to 76%. *(Cooper et al., 2012)*

**2.3.37 Heterogeneous Hand Gesture Recognition Using 3D Dynamic Skeletal Data.**

In this work, the researchers explore a way to classify dynamic hand gestures using hand skeletal representation. They proposed a method using three gestural features representing the hand shape and the motion information computed on these new data in addition to a temporal encoding of the gesture dynamics. The evaluation of our approach shows a promising way to perform hand gesture recognition with a skeletal based approach. Experiments are carried out on three hand gesture datasets, containing a set of fine and coarse heterogeneous gestures captured in different scenarios. *(Smedt et al., 2019).*

**2.3.38 A Review of Hand Gesture and Sign Language Recognition Techniques**

This paper provides a thorough review of state-of-the-art techniques used in recent hand gesture and sign language recognition research. The techniques reviewed are suitably categorized into diﬀerent stages: data acquisition, pre-processing, segmentation, feature extraction and classification, where the various algorithms at each stage are elaborated and their merits compared. Further, the researchers also discuss the challenges and limitations faced by gesture recognition research in general, as well as those exclusive to sign language recognition. *(cheok et al., 2019)*

**2.3.39 SubUNets: End-To-End Hand Shape and Continuous Sign Language Recognition**

Here, the researchers propose a novel deep learning approach to solve simultaneous alignment and recognition problems (referred to as "Sequence-to-sequence" learning). We decompose the problem into a series of specialised expert systems referred to as SubUNets. The spatio-temporal relationships between these SubUNets are then modelled to solve the task, while remaining trainable end-to-end. The approach mimics human learning and educational techniques, and has a number of significant advantages. SubUNets allow us to inject domain-specific expert knowledge into the system regarding suitable intermediate representations. They also allow us to implicitly perform transfer learning between different interrelated tasks, which also allows us to exploit a wider range of more varied data sources. In our experiments we demonstrate that each of these properties serves to significantly improve the performance of the overarching recognition system, by better constraining the learning problem. The proposed techniques are demonstrated in the challenging domain of sign language recognition. We demonstrate state-of-the-art performance on hand-shape recognition outperforming previous techniques by more than 30%). *(Camgoz et al., 2017).*

**2.3.40 Exploiting 3D Hand Pose Estimation in Deep Learning-Based Sign Language Recognition from RGB Videos**

In this paper, we investigate the benefit of 3D hand skeletal information to the task of sign language (SL) recognition from RGB videos, within a state-of-the-art, multiple-stream, deep-learning recognition system. As most SL datasets are available in traditional RGB-only video lacking depth information, we propose to infer 3D coordinates of the hand joints from RGB data via a powerful architecture that has been primarily introduced in the literature for the task of 3D human pose estimation. *(Parelli et al., 2020).*

**2.3.41 Recent Developments in Visual Sign Language**

This paper described a comprehensive approach to robust visual sign language recognition which reﬂects recent developments in this ﬁeld. The proposed recognition system aims to signer-independent operation and utilizes a single video camera for data acquisition to ensure user friendliness. In this paper, sophisticated algorithms were developed that robustly extract manual and facial features for mobile operation in uncontrolled environments in order to cover all aspects of sign languages.

*(Agris et al., 2008).*

**2.3.41 Speech Recognition Techniques for a Sign Language Recognition System**

The paper presented a vision-based approach to continuous automatic sign language recognition. It showed that appearance-based features, which have been proven to be a powerful tool in many image recognition problems, are also well suited for the recognition of sign language. Furthermore, the paper presented that several principles known from Sign language recognition, such as pronunciation and language modelling can be directly transferred to the new domain of vision-based continuous Automatic Sign Language Recognition. *(Drew et al., 2007)*

**2.3.42 Real-Time American Sign Language Recognition from Video Using Hidden Markov Models**

This paper shows an unencumbered, vision-based method of recognizing American Sign Language (ASL). Through use of hidden Markov models, low error rates were achieved on both the training set and an independent test set without invoking complex models of the hands. The paper proposes that with a larger training set and context modeling, lower error rates are expected and generalization to a freer, user independent ASL recognition system should be attainable. *(Starner & Pentland, 1997)*

**2.3.43 Sign Language Recognition Systems: A Decade Systematic Literature Review**

This paper identifies one hundred and seventeen research articles related to sign language recognition, and published between 2007 and 2017. It aimed to present the research summary on the basis of sign language which is further categorized in different dimensions like data acquisition technique, static/ dynamic signs, signing mode, single/double handed signs, classification technique and recognition rate. Although this review cannot claim to be comprehensive, it does provide reasonable insights and shows the incidence of research on this subject. Three hundred and ninety-six research articles were identified and reviewed for their direct relevance to sign language recognition systems. One hundred and seventeen research articles were subsequently selected, reviewed and classified in the research. *(Kumar et al., 2016).*

**2.3.44 Artificial Neural Network Based Method for Indian Sign Language Recognition**

This paper presents a neural network-based method for automatically recognizing the fingerspelling in Indian sign language. The features extracted from the hand shapes identify the signs. Skin colour based segmentation was used for extracting the hand region from the image. A new shape feature based on the distance transform of the image is proposed in this work. The features extracted from the sign image are used to train a feed forward neural network that recognizes the sign. The method was implemented completely by utilizing digital image processing techniques so the user does not have to wear any special hardware device to get the features of the hand shape. The proposed method has low computational complexity and very high accuracy when compared to the existing methods*. (Adithya et al., 2013).*

**2.3.45 Continuous Sign Language Recognition: Towards Large Vocabulary Statistical Recognition Systems Handling Multiple Signers**

This work presents a statistical recognition approach performing large vocabulary continuous sign language recognition across diﬀerent signers. The work shows the recent advances in system design for ASLR. The approach was evaluated on two large publicly available continuous sign language datasets representing lab-data (SIGNUM database: 25 signer, 455 sign vocabulary, 19k sentence) and unconstrained ’real-life’ sign language (RWTH-PHOENIX-Weather database 9 signer, 1081 sign vocabulary, 7 thousand sentences) reﬂecting the community’s moving from artiﬁcial lab-generated data to real-life data. *(Koller et al., 2015)*

**2.3.46 Iterative Alignment Network for Continuous Sign Language Recognition**

In this paper, an alignment network with iterative optimization for weakly supervised continuous sign language recognition was proposed. The framework consists of two modules: a 3D convolutional residual network (3D-ResNet) for feature learning and an encoder-decoder network with connectionist temporal classiﬁcation (CTC) for sequence modelling. *(Pu et al., 2019)*

**2.3.47 Robust Person-Independent Visual Sign Language Recognition**

In this paper, high recognition performance was developed for person-dependent classiﬁcation. The presented system is also suitable for person-independent real-world applications where small vocabularies suﬃce, such as controlling interactive devices. Two main challenges can be identiﬁed for robust person-independent recognition of larger vocabularies: Accurate feature extraction in real-world conditions, and handling inter-personal variance in feature processing. *(Zieren & Kraiss, 2005)*

**2.3.48 Indian Sign Language Recognition Using SVM**

This paper proposes a system that targets to facilitate disabled people who are not able to hear and there are not many people who can understand their sign language. The mobile system can convert their signs in language. The system was implemented and four gestures were classified using SVM. Results were 97.5% accurate. (Raheja et al., 2016).

# **CHAPTER THREE**

# **SYSTEM ANALYSIS AND METHODOLOGY**

# **3.0 INTRODUCTION**

System analysis is simply the act of observing or analyzing a system critically for the aim of making improvements, whereas methodology is simply the pattern or method of carrying out a task.

Throughout this chapter, we will examine the features and functionalities of the present system as well as those of the proposed system in greater depth. Additionally, we compare and contrast the present system with the proposed system, as well as investigate the design technique used to create the suggested system. In this chapter, we would have finished creating the system design that will be used in the Implementation section of the following chapter.

# **3.1 DESCRIPTION OF EXISTING SYSTEM**

A sign language recognition system consists of capturing hand signs, sending the captured image to the model which then gives the textual output of the captured image.

This research builds on certain existing systems of online communication that are currently in use. However, despite the fact that these existing systems do not include sign language, they have been extremely effective and efficient in making communication easier. Including sign language recognition technology in these systems would go a long way toward closing the communication gap that exists between the disabled and the non-disabled. The following characteristics can be found in these existing systems:

1. Registration of users on the system.
2. User authentication with phone number.
3. Provision of voice to text typing.
4. Interface for manual typing.
5. The simplicity of usage.

# **3.2 REVIEW OF THE EXISTING SYSTEM**

The techniques described above in online communication simply serve to provide an interface for inputting text from a manual keyboard, as well as an interface for voice to text typing, among other functions. The existing system does not provide for the inclusion of physically challenged individuals. Because it does not give an effective mechanism for the disabled to communicate readily, the current system is deemed insufficient.

# **3.2.1 PROBLEMS OF THE EXISTING SYSTEM**

The existing system is confronted with the issue of exclusion for people with disabilities. The existing system does not provide an alternative means of communication for the disabled, but rather forces them to conform to the conventional method of manual typing, which can be extremely stressful in certain situations and also prevents people from fully expressing themselves, especially when they consider the stress they would experience manually typing everything they wish to communicate. More so, as people type, the communication process becomes slower and tiring.

# **3.3** **DESCRIPTION OF THE PROPOSED SYSTEM**

The proposed system is intended to make the most of the current online communication platform by providing a means for people with disabilities to communicate more expressively and efficiently through the internet. There will be a model that is housed online and then blended with the system, which will result in a beautiful interface for the camera capture that will be easy to use. The system will primarily consist of a camera that will take hand signs and a model which then interprets the meaning of the hand sign captured by the camera.

# **3.3.1 ADVANTAGES OR JUSTIFICATION OF THE PROPOSED SYSTEM**

1. The system provides real-time sign language recognition.
2. The system’s model is hosted online which makes it lighter.

# **3.3.2 FEASIBILITY STUDIES OF THE PROPOSED SYSTEM**

A feasibility study is an analysis that takes all of a projects relevant factor into account to measure it the viability of the project, to help define the goals and objectives of the project, to help in developing a plan and even in the execution of that plan.

In this work, the following feasibility studies are analysed.

**Economic feasibility:**  This analysis has to do with the cost-effectiveness of the proposed system. Since the intended system will be built on an open-source framework, it is feasible.

**Operational feasibility**: This analysis deals with the user’s perception and behavior of the intended system. The system is feasible operationally with a simple and attractive user interface to enhance ease of use and operation.

**Behavioral feasibility:** This analysis checks if a system solves the problem. In this aspect, our project is feasible as its successful converts sign language to its textual format.

# **3.4 REQUIREMENT ELICITATION**

Requirement elicitation is the process of gathering the requirement needed for developing a system probably from users, customers and other stakeholders. Requirements elicitation methods involve the use of interviews, questionnaires, user observation, workshops, brainstorming, use cases, etc. for gathering the necessary requirement for the development of a system. The requirement elicitation techniques used in this work are observation and domain analysis.

# **3.4.1 FUNCTIONAL REQUIREMENTS OF THE PROPOSED SYSTEM**

The functional requirements of a system are requirements expected of the functional and operational aspects of the proposed system. This refers to the service which the end-users will get from the proposed system. The functional requirement of this system is:

* To convert sign language to its textual form.

# **3.4.2 NON-FUNCTIONAL REQUIREMENT OF THE PROPOSED SYSTEM**

The non-functional requirements of a system are the requirements of a system that are not a part of its functionalities. The non-functional requirements for this system include:

* Reliability
* Usability
* Correctness
* Performance
* Robustness

# **3.5 HIGH-LEVEL MODEL OF THE SIGN LANGUAGE RECOGNITION SYSTEM**

The high-level model an architectural structure that is used to give an overview of a system. In a high-level model, it is possible to easily identify the basic components in a software or system and how they relate with each other to bring the overall system. With the high-level model of a system, it is possible for software engineers to partition the job involved in building a system.

The high-level system of this research work undergoes some processes which include the following:

1. Data acquisition or data gathering of images of interest.
2. Data training.
3. Testing.

The high-level model of the sign language recognition system is depicted in fig 3.1 showing how the components of the system interact to attain the complete system.

**Sign Language Recognition System**

**Capture module**

**Sign Recognition module**

**Training module**

**Obtain model accuracy**

**Detect sign**

**Fit model**

**Label image**

**Capture image**

***Fig 3.1: High-level Model of the Proposed System***

**3.6 DATA FLOW DIAGRAM OF THE PROPOSED SYSTEM**

Data visualization is a very important aspect of computer science as it is very important to see how the data in our system flows through various channels. There are various diagrams that are used to visualize data in a system. One of these diagrams is the data flow diagram popularly known as the DFD. The DFD is diagram that is used to show how data moves or flows through a system. These entails its movement from its entry point, through the processes and finally the database.

A data-flow diagram is simply a graphical representation of the flow of data through an information system. The data flow diagram of the sign language recognition system is shown in fig 3.2.

Image Input

1.1

Image Preprocessing

Image Database

D1

1.2

Feature Extraction

1.3

Classification

Identify (result)

***Fig 3.2: Dataflow of the Proposed System***

# **3.7 METHODOLOGY**

When selecting a suitable methodology for a project work, one has to put certain factors such as the source of the data, the area of application of the project etc. The selected research process for this work is Crisp-DM (Cross-industry standard process for data mining). This methodology is chosen because of its iterative nature in problem-solving. The research methodology adopted for this work is Convolutional Neural Network (CNN).

Furthermore, this project is developed using two programming languages which include JavaScript and Python. The model will be built with Python and Jupyter notebook and the model will also leverage on the TensorFlow object recognition model while the system will be built with JavaScript using the react.js framework.

In this research work, we will train the model with 10 signs using a total of 7794 dataset for all the 10 classes.

# **3.7.1 SPECIFICATION AND JUSTIFICATION FOR THE SELECTED METHODOLOGY**

When compared to other image classification techniques, the Convolutional Neural Network takes up a comparatively small amount of time during the pre-processing phase. For every input, each neuron does a weighted sum over the other inputs, runs the result through an activation function, and then reacts with an output. Simply put, CNN are used in analyzing visual imagery. Below is a figure for describing the input and occurrences in a Convolutional Neural Network. Fig 3.3 shows the basic structure of the convolutional neural network.

Σ

Ϝ

W1

W2

W3

Wn

⁞

**output**

Activation function

Sum

**Inputs X**

Output

***Fig 3.3: Connection of Neurons in CNN***

*Source: medium.com*

Here,

x = Input data

w = Randomly initialized weight matrix

Mathematically, the sum is

The CNN is comprised of four (4) layers with four (4) activation functions. Three of these activation functions were Relu. Relu is expressed mathematically as

It is actually the most used activation function in Neural Networks. The other activation function used was SoftMax. SoftMax is expressed mathematically as

# **3.7.2 PROJECT PLAN**

The project plan is a plan that is usually used to show when different activities of a project were carried out, and if possible, it shows who executed these events. Fig 3.4 shows the project plan for the sign language recognition system.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Weeks | | | | | | | | | | | | | | | | |
| Project Activities | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 |
| Project Initiation |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Planning |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Design |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Coding |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Testing |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Delivery |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

*Fig 3.4 Gantt chart depicting project plan the Rule Engine for the Java Platform*

*onal Diagnosis ()()s Internationale industry ()lway Sectors ()97898999999999*

**CHAPTER FOUR**

# **SYSTEM DESIGN AND IMPLEMENTATION**

# **4.1 INTRODUCTION**

In this section of the work, an in-depth discussion is held about transforming the stated user requirements into a concrete and meaningful description that will guide the project's implementation phase.

The fundamental advantage of project system design is that it provides efficient and equitable information about the system and the data contained inside the system.

Creating the aspects of a system, such as its modules, architecture, components, and their interfaces and data, in accordance with the system's needs is referred to as system design. It usually results in a standard approach to problem-solving being used in the future.

The benefits of system design include cost savings, the elimination of design inconsistencies, and the acceleration of the implementation process.

The process of actually developing a system based on detailed design specifications is known as system implementation. It also entails writing code for the system in a programming language, testing it, and producing documentation on how to use the developed system.

# **4.2 OBJECTIVES OF THE DESIGN**

# As common with every human endeavor, they are always goals that we wish to attain for everything we do. Thus, even for this design there are certain goals we wish to attain and these goals include:

1. To accurately achieve and satisfy user requirements
2. Ease system usability
3. Easy to understand
4. Good domain centred user experience
5. Simple and attractive user interface

# **4.3 ARCHITECTURAL DESIGN OF SIGN LANGUAGE RECOGNITION SYSTEM**

The architectural design of a system simply has to do with the fundamental structure of that system. This is how the components are connected to each other and relate to attain the desired goal of the system. It shows the relations among the components, their properties and other relationship with every component that they relate with.

Architectural design is a system design that depicts the higher components of a system and how they interact with one another. It offers a structured approach to developing the desired system. The architectural design for the sign language recognition system is given below. The architectural design of our system is shown in fig 4.1.

**Hand and sign recognition**

**Feature extraction**

**Database**

**Neural network**

**Sign language to text**

***Fig 4.1 Architectural design of the system***

**4.4.1 PHYSICAL DESIGN**

The physical design of the system is related to the actual input and output processes of the system. It primarily focuses on how to enter data into the system, validate it, process it, and display the outcomes of the processes. It produces a workable system by defining design requirements that precisely outline the functions of the candidate system under consideration.

It consists of three parts: user interface design, process design, and data design.

Thus, it is related to this work because it involves the user interface where inputs are received and how images are moved through the CNN.

# **4.4.1.1 INPUT DESIGN**

Aspects of input design include receiving inputs that are raw data, which must then be processed in order to attain the required output. When developing an input design, it is critical to ensure that the design is flexible across a variety of platforms, including mobile, desktop computers and display devices. The design should be able to adequately get the needed information in an efficient and effective manner. The following is an example of the input data that was provided to the sign language recognition system and was processed by it. The input design of our system is shown in fig 4.2.



***Fig 4.2 Input design***

# **4.4.1.2 OUTPUT DESIGN**

The output design of a system is essentially one of the most important aspects of any system as it is very necessary to know the representation of data that have been processed. When creating the output design of a system, the developers must know the type of output needed. The objective of an output design is not just to deliver the appropriate quantity of output but to also meet the end user's requirement.

Below is an output design for our project work. The input design of our system is shown in fig 4.3.



***Fig 4.3 Output design***

# **4.4.2 LOGICAL DESIGN**

The logical design of the system describes operational aspects of the system not visible to the user explicitly. It is the abstract representation of data flows, database design, inputs and outputs of the system.

# **4.4.2.1 USE CASE DIAGRAM**

This diagram defines the system behaviour based on the requirements that have been stated by the user. It goes further to also state a set of functionalities that the system should perform for a set of entities that it relates with. It however does not describe the implementation of functionalities. Fig 4.4 shows the use case diagram of the proposed system.

RESULT OF THE ANALYSIS

SELECT/CAPTURE IMAGE

VERIFICATION OF THE IMAGE

***Fig 4.4 Use Case diagram of sign language recognition system***

# **4.5 APPLICATION ALGORITHM**

Algorithms are the step by step means of solving a problem. Thus, relating it with this project, we can say our application algorithms are step by step means of moving through the sign language system so as to achieve the desired task which in this case is to move achieve a converted sign. In moving through our system, we have five steps. I.e.

**STEP 1:** Start

**STEP 2**: Click on capture image

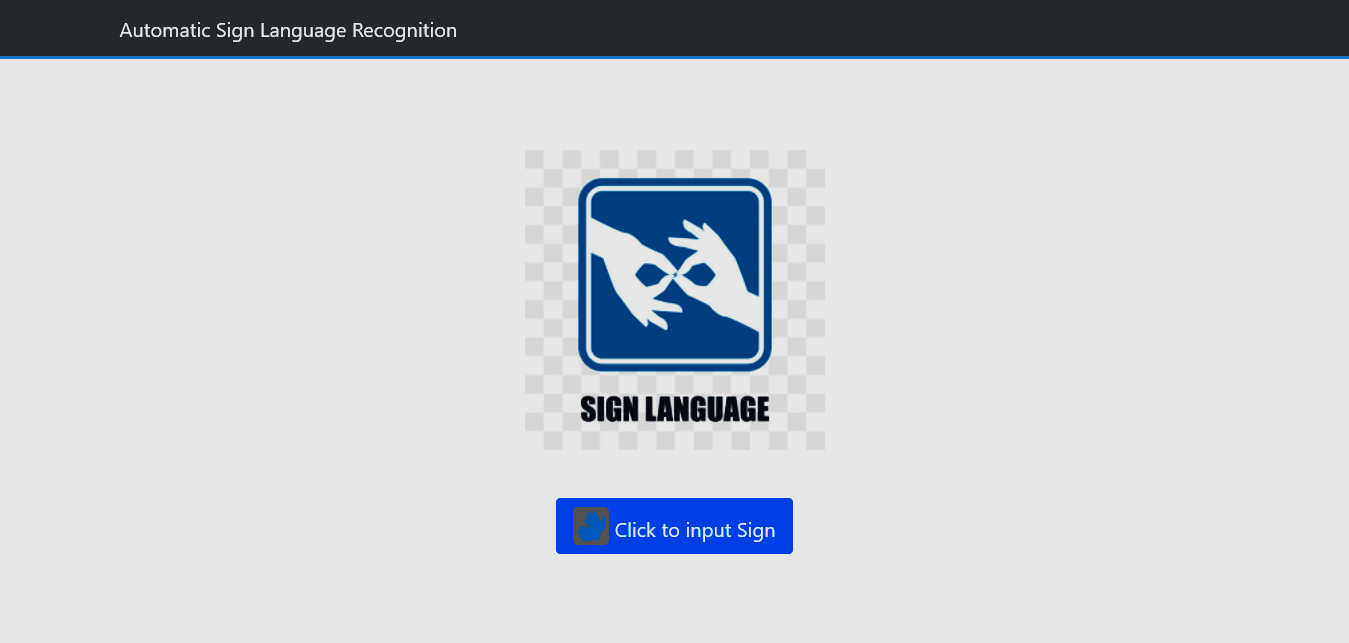
**STEP 3**: Verify the sign by performing sign language recognition by our algorithm

**STEP 4**: Output the corresponding textual format of the sign

**STEP 5**: End

# **4.6 THE USER INTERFACE DESIGN**

The sign language recognition system has a user interface that grants users access to input sign which is then received and processed by the model. Fig 4.5 shows the user interface design of our sign language recognition system.



***Fig 4.5 user interface design***

# **4.7 SYSTEM IMPLEMENTATION**

System implementation is the process of defining how a system should be built while ensuring that the system is operational and meets the requirement that it was intended for. System implementation is basically concerned with how operational a system will be and how maintainable it will be.

# **4.7.1 SYSTEM REQUIREMENTS**

According to the Merriam-webster dictionary, a requirement is something that is needed or wanted. In relation to this work, system requirements are the items or features that are needed to successfully make use of a system.

The system requirements for a specific project is simply the required specification a device must have to use the software. Thus, for the sign language recognition system, we have the following software and hardware requirements.

* + - 1. **Minimum Hardware Requirements**

1. Smart Mobile Device
2. 4GB RAM and above
3. 320GB ROM and above.
4. Quad-core Intel Core i3
   * + 1. **Software Requirements**
5. Python 3
6. JavaScript
7. Jupyter notebook
8. TensorFlow Object Detection API

# **4.8 IMPLEMENTATION TOOLS USED**

1. Python 3
2. Tensor Flow
3. Bootstrap
4. Html
5. CSS
6. Jquery

# **4.9 DEVELOPMENTAL PROCESS FOR THE SIGN LANGUAGE RECOGNITION SYSTEM.**

The developmental process for the sign language recognition system involves five basic steps which include planning, analysis, design, implementation and testing.

In the planning phase, the requirements for the system is put down and the developers of the system lay down a road map to achieving the desired system. This plan entails the whole items that would be needed in achieving this task. Planning is one of the most crucial aspects of a project.

Analysis is the second stage of this process; it involves the analysis of all the acquired requirements of the systems.

The design phase involves the structuring of the system into basic components, showing how data would flow through this system. It makes the implementation phase of the project easier as every component has already been documented. It further aids in code maintenance.

The implementation phase has to do with the writing of codes to perform the proposed functions.

The testing phase has to do with using the proposed system or program to find and correct errors. This is the final stage of any project as it prepares the project for the use of the public. The development process of the sign language recognition system is shown diagrammatically in fig 4.6.

Implementation

Planning

Design

Analysis

Test

***Fig 4.6 Development phase of sign language recognition system***

# **4.10 PROGRAM MODULE SPECIFICATION**

In this system, we have two modules

1. Image capture
2. Model

The image capture module consists of the following:

1. Image retrieval from the operator

The model comprises of the following modules

1. Object detection
2. Training

**Object detection:** This is simply a computer vision technique that allows us to identify and locate objects in an image or video.

**Training:** The training module comprises of learning good values for all the weights and basis from a set of labelled examples. It consists of a set of input data that have an influence on the output data.

**4.11 Application Manual**

The manual for the application is as follows

1. Go to the URL
2. Press the “click to input sign” button

# **CHAPTER 5**

# **SUMMARY, RECOMMENDATION AND CONCLUSION**

# **5.1 SUMMARY**

Sign language is a very important means of communication, especially among the non-speaking community. When a sign language recognition system is applied to the current means of communication on the various social platforms it will greatly improve the flexibility of communication among the non-speaking community. To create the sign language recognition system, several machine learning algorithms were put to use.

In this research, the TensorFlow object detection API was used to train the CNN with 10 classes which represented 10 different signs.

This system was built to serve as an extension for the current systems of communication on social media platforms and not as an improvement because it introduces a new method in the social media communication platforms.

# **5.1.1 REVIEW OF CONTRIBUTIONS AND ACHIEVEMENTS**

On carrying out this research work, we were able to:

1. Obtain a dataset of 7797 images with 10 classes of 10 signs (hello, hurts a lot, I am okay, I love you, I want to talk, love, mine, no, wrong, yes).
2. Prepare the dataset for training, validation and testing using TensorFlow.
3. Create labels for the dataset
4. Train and validate the model on our dataset.
5. Test the model on a custom dataset.

# **5.2 RECOMMENDATIONS**

The sign language recognition model is highly recommended for use in social media platforms such as WhatsApp, Facebook etc. when merged with such social media platforms, this model will help the non-speaking community to communicate with the speaking community more easily and effectively.

# **5.2.1 FUTURE RESEARCH AND WORK**

Taking the achievements and limitations of this research work into consideration, we intend to make improvements to the model by increasing the dataset to 20 classes of signs and training on a Graphical Processing Unit (GPU). Furthermore, we could not integrate the model into a system in the future as this could not be done due to time constraints.

# **5.3 CONCLUSION**

At the end of this work, we successfully developed a machine learning model that could recognize 10 different hand signs with an accuracy of 96.5%. This model was trained with a total of 7,797 datasets for all the 10 different signs or classes.

**REFERENCES**

Adithya, V.K., Vinod, P., & Gopalakrishnan, U. (2013). Artificial neural network- based method for Indian sign language recognition. 2013 IEEE CONFERENCE ON INFORMATION AND COMMUNICATION TECHNOLOGIES, 1080-1085.

Agris, U. V., Knorr M. & Kraiss, K. (2008). The significance of facial features for automatic sign language recognition. 2008 8th IEEE International Conference on Automatic Face & Gesture Recognition, 1-6, doi: 10.1109/AFGR.2008.4813472.

Agris, U., Zieren, J., Canzler, U., Bauer, B., & Kraiss, Karl-F. (2008). Recent developments in visual sign language recognition. Universal Access in the Information Society. 6. 323-362.

Al-Shamayleh, A. S., Ahmad, R., Jomhari, N., & M. Abushariah, M. A. (2020). Automatic Arabic Sign Language Recognition: A Review, Taxonomy, Open Challenges, Research Roadmap And Future Directions. *Malaysian Journal Of Computer Science*, *33*(4), 306– 343. Retrieved from https://ejournal.um.edu.my/index.php/MJCS/article/view/26902

Antonakos, E., Roussos, A & Zafeiriou, S.(2015).A Survey on Mouth Modeling and Analysis for Sign Language Recognition. 2015 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG), 1-7.

Beena M. V. & Namboodiri M.N. A. (2017). Automatic Sign Language Finger Spelling Using Convolution Neural Network: Analysis. International Journal of Pure and Applied Mathematics, 117(20), 9-15.

Brashaer H., Starner T., Hamilton H. and Zafrulla Z (2010). A Novel Approach to American Sign Language (ASL) phrase verification reversed signing. 2010 IEEE Computer Society Conference on Computer Society Conference on Computer Vision and Pattern Recognition Workshop (CVPR Workshops), San Francisco, CA, 1, 48-55.

Camgoz, N. C., Hadfield, S. Koller O. & Bowden, R. (2017). SubUNets: End-to-End Hand Shape and Continuous Sign Language Recognition. 2017 IEEE International Conference on Computer Vision (ICCV), 3075-3084, doi: 10.1109/ICCV.2017.332.

Camgöz, N.C., Koller, O., Hadfield, S., & Bowden, R. (2020). Sign Language Transformers: Joint End-to-End Sign Language Recognition and Translation. 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 10020-10030.

Chabaan, H., Gouiffes, M., & Braffort, A. (2019). Towards an Automatic Annotation of French Sign Language Videos: Detection of Lexical Signs. CAIP 2019: Computer Analysis of Images and Patterns, 11679, 402-412.

Chai, X., Li, G., Lin, Y., Xu, Z., Tang, Y.B., & Chen, X. (2013). Sign Language Recognition and Translation with Kinect.

Cheok, M.J., Omar, Z. & Jaward, M.H. A review of hand gesture and sign language recognition techniques. *Int. J. Mach. Learn. & Cyber.* **10,** 131–153 (2019). https://doi.org/10.1007/s13042-017-0705-5

Cooper, H. & Ong, E., & Pugeault, N., & Bowden, R. (2012). Sign Language Recognition using Sub-Units. Journal of Machine Learning Research. 13. 2205-2231.

Cui R., Lui H., & Zhang C. (2016). Recurrent Convolutional Neural Network for Continuous Sign Language Recognition by Staged Optimization. Computer Vision Foundation (7361-7369).

Dreuw, P., Rybach, D., Deselaers, T., Zahedi, M., & Ney, H. (2007). Speech recognition techniques for a sign language recognition system. INTERSPEECH.

Dong C., Leu M. C. & Yin Z. (2015). American Sign Language Alphabet Recognition Using Microsoft Kinect. Computer Vision Foundation, 44-52.

Farag I. & Brock, H. (2019). Learning Motion Disfluencies for Automatic Sign Language Segmentation. ICASSP 2019 – 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 7360-7364.

Ghotkar A. S. & Kharate G. K. (2014). Study of vision-based hand gesture recognition using Indian sign language. International journal on smart and intelligent systems, 7(1).

Hossen M. A., Govindaiah A., Sultana S. & Bhuiyan A. (2018). Bengali Sign Language Recognition Using Deep Convolutional Neural Network. 2018 Joint 7th International Conference on Informatics, Electronics & Vision (ICIEV) and 2018 2nd International Conference on Imaging, Vision & Pattern Recognition (iciVPR), 2018, 369-373.

Imran A., Razzaq A., Baig I. A., Hussain A., Shahid S. & Rehman T. (2021). Dataset of Pakistan Sign Language and Automatic Recognition of Hand Configuration of Hand Configuration of Urdu Alphabet through Machine Learning. Data, in brief, 36.

Koller, O., Forster, J., & Ney, H. (2015). Continuous sign language recognition: Towards large vocabulary statistical recognition systems handling multiple signers. Comput. Vis. Image Underst., 141, 108-125.

Konstantinidis, D., Dimitropoulos K. & Daras, P. (2018). Sign Language Recognition Based on Hand and Body Skeletal Data. 2018-3DTV- Conference: The True Vision – Capture, Transmission, and Display of 3D Video (3DTV-CON), 1-4.

Krishnaveni M., Subashini P. & Dhivyaprabha T. T. (2019). An Assertive Framework for Automatic Tamil Sign Language Recognition System Using Computational Intelligence. Springer International Publishing, 55- 87.

Kumar, A., Thankachan, K., & Dominic, M.M. (2016). Sign language recognition. 2016 3rd International Conference on Recent Advances in Information Technology (RAIT), 422-428.

Kuznetsova, A. Leal-Taixe, L. Rosenhahn B. (2013). Real-Time Sign Language Recognition Using a Consumer Depth Camera. 2013 IEEE International Conference on Computer Vision Workshops. 83-90.

Lang S., Block M., Rojas R. (2012) Sign Language Recognition Using Kinect. In: Rutkowski L., Korytkowski M., Scherer R., Tadeusiewicz R., Zadeh L.A., Zurada J.M. (eds) Artificial Intelligence and Soft Computing. ICAISC 2012. Lecture Notes in Computer Science, vol 7267. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-29347-4\_46.

Latif, G., Mohammed, N., Alkhalaf, R., Alkhalaf, R., Alghazo, J., & Khan, M. A. (2020). An Automatic Arabic Sign Language Recognition System based on Deep CNN: An Assistive System for the Deaf and Hard of Hearing. International Journal of Computing and Digital Systems, 90(4), 715-724.

Li D., Opazo C. R., Yu X. & Li H. (2020). Word-level Deep Sign Language Recognition from Video: A New Large-scale Dataset and Methods Comparison. 2020 IEEE Winter Conference on Application of Computer Vision (WACV), 1448-1458.

Mehdi, S., A. & Khan, Y., N. (2002). Sign language recognition using sensor gloves. Proceedings of the 9th International Conference on Neural Information Processing, ICONIP .5(2), 2204-2206 doi: 10.1109/ICONIP.2002.1201884.

Mohammed, M.M., Khan, M.B., & Bashier, E. (2016). Machine Learning: Algorithms and Applications.

Nada, I. B, Mazen M. S & Hala H. Z. (2017). An Automatic Arabic Sign Language Recognition System (ArSLRS). Journal of King Saud University – Computer and Information Sciences, 30(4), 470-477.

Oscar, K. (2020). Quantitative Survey of the State of the Art in Sign Language Recognition. Speech and Language Microsoft.

Pansare, J., R. & Ingle M. (2016). Vision-based approach for American Sign Language recognition using Edge Orientation Histogram. 2016 International Conference on Image, Vision, and Computing (ICIVC), 86-90.

Parelli, M., & Papadimitriou, K, Potamianos, G., Pavlakos, G., & Maragos, P. (2020). Exploiting 3D Hand Pose Estimation in Deep Learning-Based Sign Language Recognition from RGB Videos. 10.1007/978-3-030-66096-3\_18.

Pigou, L., Dieleman, S., Kindersmans, PJ. & Schrauwen, B. (2015). Sign Language Recognition Using Convolutional Neural Networks. Computer Vision-ECCV 2014 workshops. 8925, 572-578.

Pu, J., Zhou, W., & Li, H. Iterative Alignment Network for Continuous Sign Language. 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), 4160-4169, doi: 10.1109/CVPR.2019.00429.

Rahaman, M., A., Jasmin, M., Ali, H. Md. & Hasanuzzaman, Md. (2019). Bangla language modeling algorithm for automatic recognition of hand-sign-spelled Bangla sign language. Frontiers of Computer Science, 14(3).

Raheja, J.L., Mishra, A. & Chaudhary, A. (2016). Indian sign language recognition using SVM. Pattern Recognit. Image Anal. 26

Ravikiran, J., Kavi, M. Suhas, M., Dheeraj, R., Sudheender, S. & Nitin, V.P. (2009). Finger Detection for Sign Language Recognition. Proceedings of the International MultiConference of Engineers and Computer Scientists 2009. I.

Santiago, T. D., Benderitter, I., & Mateo, C. G. (2018). Experimental Framework Design for Sign Language Automatic Recognition. IbeerSPEECH 2018.

Smedt, Q., Wannous, H. & Vandeborre, J. (2019). Heterogeneous hand gesture recognition using 3D dynamic skeletal data. Computer Vision and Image Understanding. 181. 10.1016/j.cviu.2019.01.008.

Starner T., Pentland A. (1997) Real-Time American Sign Language Recognition from Video Using Hidden Markov Models. In: Shah M., Jain R. (eds) Motion-Based Recognition. Computational Imaging and Vision, vol 9. Springer, Dordrecht. https://doi.org/10.1007/978-94-015-8935-2\_10

Starner, T. Weaver J. & A. Pentland. (1998) Real-time American sign language recognition using desk and wearable computer-based video. IEEE Transactions on Pattern Analysis and Machine Intelligence, 20(12). 1371-1375, Dec. 1998, doi: 10.1109/34.735811.

Trigueiros P., Ribeiro F. & Reis L. P. (2014). Vision-based Portuguese Sign Language Recognition System. Advances in Intelligent Systems and Computing, 275.

Valentin, B. (2019). Sign Language Video Analysis for Automatic Recognition and Detection. 14th IEEE International Conference on Automatic Face and Gesture Recognition May 2019.

Valentin, B., Michele, G. & Annelies, B. (2019). Automatic Recognition of Sign Language Structures in Rgb Videos: The Detection of Pointing and Lexical Signs. Hal archives ouvetes

Vogler C., Metaxas D. (2004). Handshapes and Movements: Multiple-Channel American Sign Language Recognition. In: Camurri A., Volpe G. (eds) Gesture- Based Communication in Human-Computer Interaction. GW 2003. Lecture Notes in Computer Science, vol 2915. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-540-24598-8\_23.

Yuan, T., Sah, S., Ananthanarayana, T., Zhang, C., Bhat, A., Gandhi, S., Ptucha, R. (2019). Large Scale Sign Language Interpretation. 14th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2019).

Zafrulla, Z., Brashear, H., Starner, T., Hamilton, H., & Presti, P. (2011). American sign language recognition with the kinect. ICMI'11 - Proceedings of the 2011 ACM International Conference on Multimodal Interaction. 279-286. 10.1145/2070481.2070532

Zaki, M., M. & Shaheen, S. I. (2011). Sign Language recognition using a combination of new vision-based features. Pattern Recognition Letters, 32(4), (572-577).

Zieren, J., & Kraiss, K. (2005). Robust Person-Independent Visual Sign Language Recognition. 3522. 520-528. 10.1007/11492429\_63.

.

.

# **APPENDIX A**

**/\* importing our necessary libraries \*/**

**import cv2**

**import numpy as np**

**import tensorflow as tf**

**from tensorflow import keras**

**from tensorflow.keras.models import Sequential**

**from tensorflow.keras.layers import Activation, Dense, Flatten, BatchNormalization, Conv2D, MaxPool2D**

**from tensorflow.keras.optimizers import Adam**

**from tensorflow.keras.preprocessing.image import ImageDataGenerator**

**from tensorflow.keras.metrics import categorical\_crossentropy**

**from sklearn.metrics import confusion\_matrix**

**import itertools**

**import os**

**import shutil**

**import random**

**import glob**

**import matplotlib.pyplot as plt**

**from tensorflow.keras.models import load\_model**

**import warnings**

**warnings.simplefilter(action='ignore', category= FutureWarning)**

**%matplotlib inline**

**from tensorflow.keras.preprocessing import image**

**os.chdir("C:\\Users\\Augustine Aroke\\Desktop\\RSL\\Dataset")**

/\* defining our data splits\*/

train\_size =4902

test\_size = 795

valid\_size = 2100

/\* setting the paths for our data set \*/

train\_path = 'C:\\Users\\Augustine Aroke\\Desktop\\RSL\\Dataset\\training set'

test\_path = 'C:\\Users\\Augustine Aroke\\Desktop\\RSL\\Dataset\\testing set'

valid\_path = 'C:\\Users\\Augustine Aroke\\Desktop\\RSL\\Dataset\\validation set'

/\* pre processing our images \*/

train\_batches =ImageDataGenerator(rotation\_range=15, width\_shift\_range=0.2, height\_shift\_range=0.2, shear\_range=0.15,

zoom\_range=0.2, channel\_shift\_range=0.1, horizontal\_flip=True,

preprocessing\_function=tf.keras.applications.vgg16.preprocess\_input) \

.flow\_from\_directory(directory=train\_path, target\_size=(200,200), classes=['Hello','Hurtsalot','Iamokay','Iloveyou','Iwanttotalk','Love','Mine','No',' Wrong','Yes'],batch\_size=10)

test\_batches = ImageDataGenerator(rotation\_range=15, width\_shift\_range=0.2, height\_shift\_range=0.2, shear\_range=0.15,

zoom\_range=0.2,channel\_shift\_range=0.1,horizontal\_flip=True,

preprocessing\_function=tf.keras.applications.vgg16.preprocess\_input) \

.flow\_from\_directory(directory=test\_path,target\_size=(200,200), classes=['Hello','Hurtsalot','Iamokay','Iloveyou','Iwanttotalk','Love','Mine','No',' Wrong','Yes'],batch\_size=10)

valid\_batches = ImageDataGenerator( rotation\_range=15, width\_shift\_range=0.2, height\_shift\_range=0.2, shear\_range=0.15,

zoom\_range=0.2, channel\_shift\_range=0.1, horizontal\_flip=True,

preprocessing\_function=tf.keras.applications.vgg16.preprocess\_input) \

.flow\_from\_directory(directory=valid\_path,target\_size=(200,200), classes=['Hello','Hurtsalot','Iamokay','Iloveyou','Iwanttotalk','Love','Mine','No',' Wrong','Yes'],batch\_size=10)

assert train\_batches.n == 4902

assert test\_batches.n == 795

assert valid\_batches.n == 2100

imgs,labels = next(train\_batches)

print(labels)

def plotImages(images\_arr):

fig, axes = plt.subplots(1,10,figsize=(20,20))

axes = axes.flatten()

for img, ax in zip(images\_arr,axes):

ax.imshow(img)

ax.axis("off")

plt.tight\_layout()

plt.show()

plotImages(imgs)

print(labels)

/\* setting up our model \*/

model = Sequential([

Conv2D(filters=16,kernel\_size=(3, 3),activation='relu',padding='same',input\_shape=(200,200,3)),

MaxPool2D(pool\_size=(2,2),strides=2),

Conv2D(filters=32,kernel\_size=(2,2),activation='relu',padding='same'),

MaxPool2D(pool\_size=(2,2),strides=2),

Conv2D(filters=64,kernel\_size=(2,2),activation='relu',padding='same'),

MaxPool2D(pool\_size=(2,2),strides=2),

Flatten(),

Dense(units=10,activation='softmax')

])

model.summary()

model.compile(optimizer=Adam(learning\_rate=0.00001),loss='categorical\_crossentropy',metrics=['accuracy'])

model.fit(x=train\_batches,validation\_data=valid\_batches,epochs=10,verbose=2)

[{"metadata":{"trusted":false},"id":"28921b50","cell\_type":"code","source":"model.fit(x=train\_batches,validation\_data=valid\_batches,epochs=10,verbose=2)","execution\_count" :14,"outputs":[{"name":"stdout","output\_type":"stream","text":"Epoch 1/10\n491/491 - 1299s - loss: 2.3195 - accuracy: 0.6363 - val\_loss: 0.8239 - val\_accuracy: 0.8005\nEpoch 2/10\n491/491 - 1661s - loss: 0.4020 - accuracy: 0.8860 - val\_loss: 0.4153 - val\_accuracy: 0.9157\nEpoch 3/10\n491/491 - 1255s - loss: 0.1749 - accuracy: 0.9480 - val\_loss: 0.4497 - val\_accuracy: 0.9357\nEpoch 4/10\n491/491 - 1651s - loss: 0.1130 - accuracy: 0.9678 - val\_loss: 0.4733 - val\_accuracy: 0.9200\nEpoch 5/10\n491/491 - 74194s - loss: 0.1020 - accuracy: 0.9714 - val\_loss: 0.3083 - val\_accuracy: 0.9462\nEpoch 6/10\n491/491 - 1218s - loss: 0.0649 - accuracy: 0.9796 - val\_loss: 0.4730 - val\_accuracy: 0.9438\nEpoch 7/10\n491/491 - 1278s - loss: 0.0427 - accuracy: 0.9843 - val\_loss: 0.4314 - val\_accuracy: 0.9590\nEpoch 8/10\n491/491 - 1315s - loss: 0.0377 - accuracy: 0.9884 - val\_loss: 0.4451 - val\_accuracy: 0.9643\nEpoch 9/10\n491/491 - 1296s - loss: 0.0183 - accuracy: 0.9935 - val\_loss: 0.3940 - val\_accuracy: 0.9671\nEpoch 10/10\n491/491 - 1299s - loss: 0.0487 - accuracy: 0.9876 - val\_loss: 0.3160 - val\_accuracy: 0.9614\n"},{"data":{"text/plain":""},"execution\_count":14,"metadata":{},"output\_type":"execute\_result"}]}]

/\* retrieve and print the score of the model \*/

score = model.evaluate(test\_batches, verbose =1)

print(score)

/\* print the model for future use \*/

model.save('model.h5')

save\_model = load\_model('model.h5')

filepath = 'C:\\Users\\Augustine Aroke\\Desktop\\RSL\\Dataset\\newdata\\Hurtsalot\\hurtsalot (1).jpg'

def img\_preprocessed(filepath):

img = image.load\_img(filepath, target\_size=(200,200,3),color\_mode='rgb')

img\_array = image.img\_to\_array(img)

img\_array = np.expand\_dims(img\_array, axis=0)

return img\_array

np.round(model.predict(img\_preprocessed(filepath)))

print("Accuracy: {}".format(score[1]\*100))

show\_predictions(img\_preprocessed)

def plotimage(path):

pred\_value = show\_predictions(img\_preprocessed)

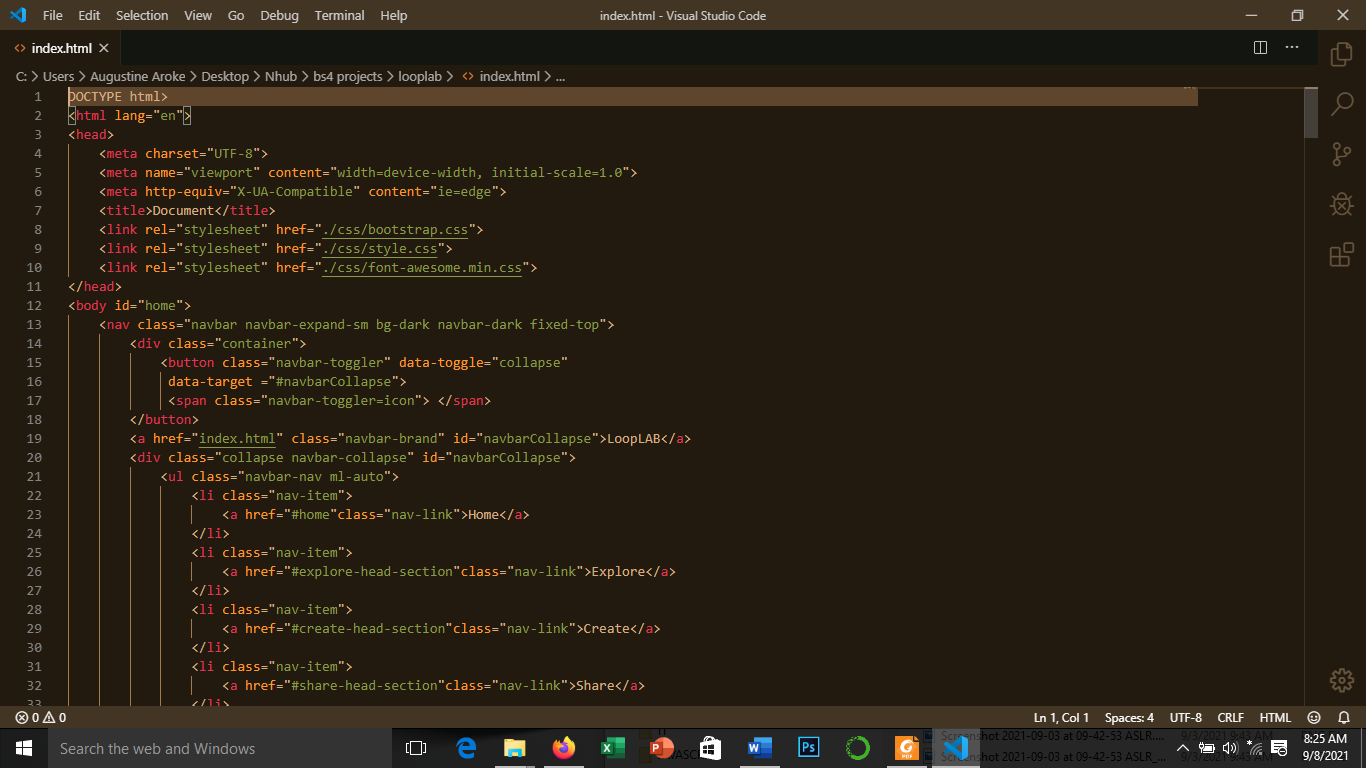
img = cv2.imread(path)

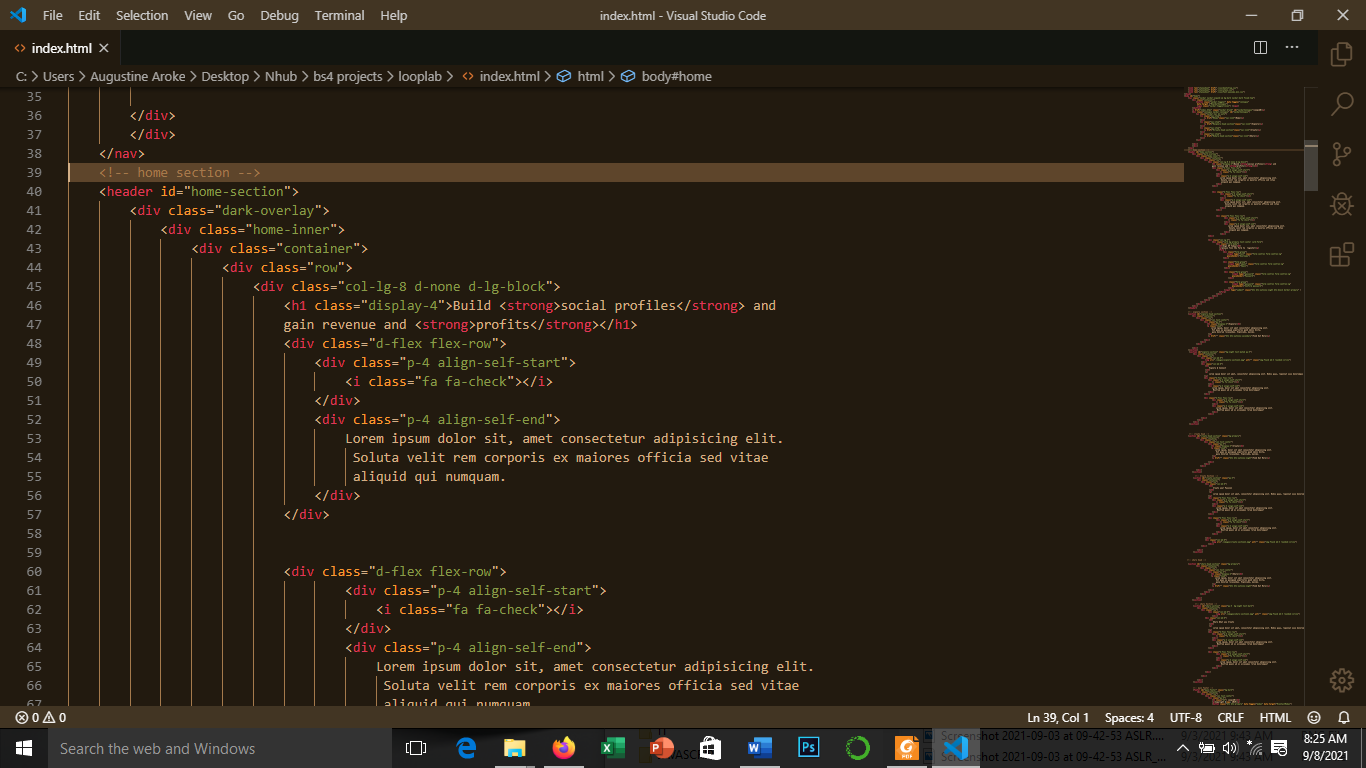
cv2.putText(img,text=pred\_value,fontScale = 10, color=(255,255,0), thickness = 10,org=(0,500),fontFace=5)

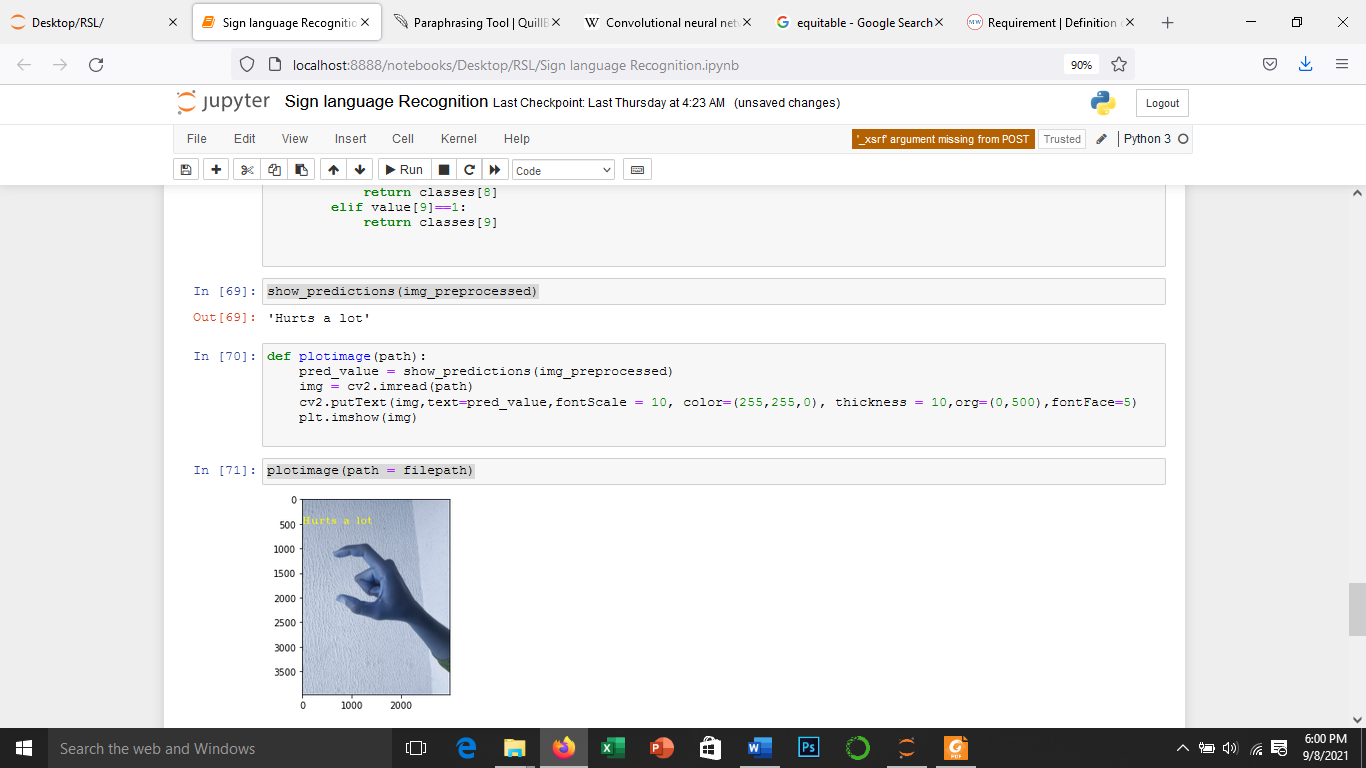
plt.imshow(img)

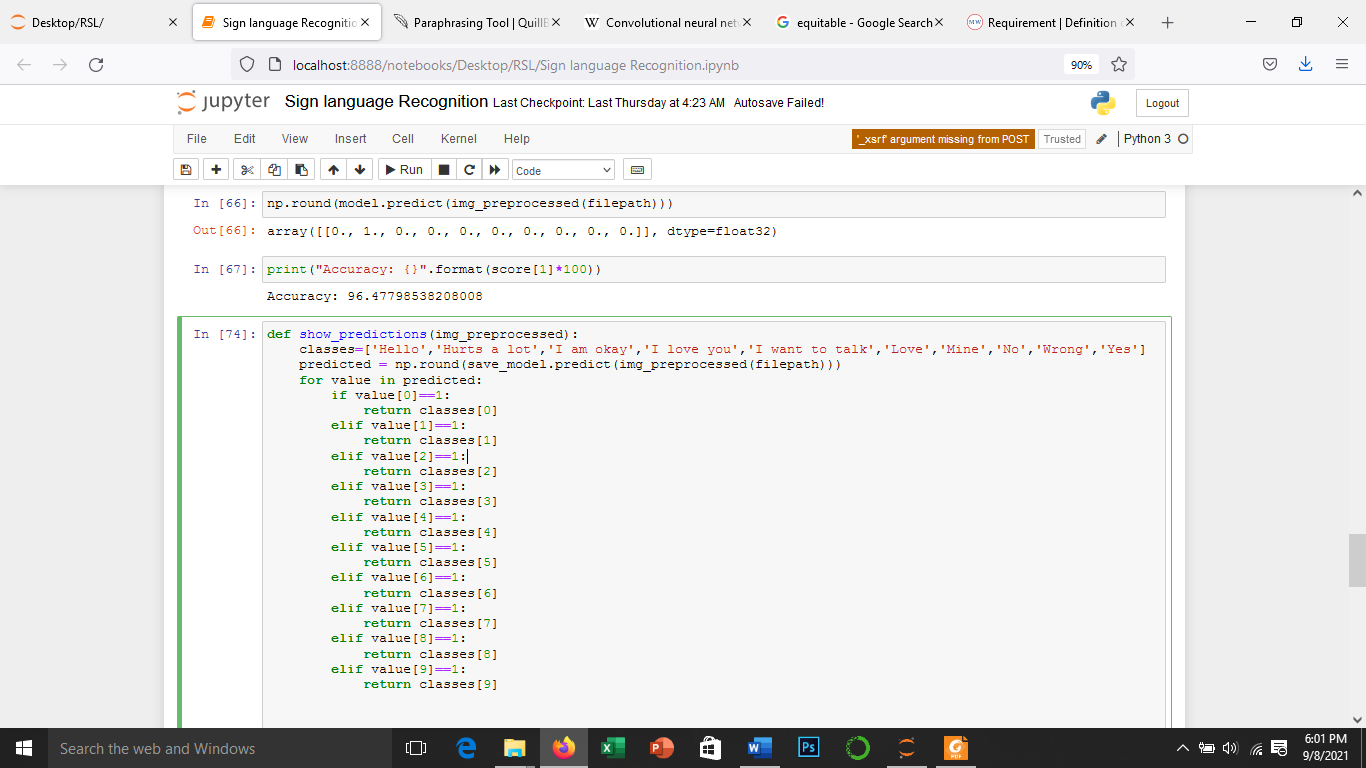
plotimage(path = filepath)

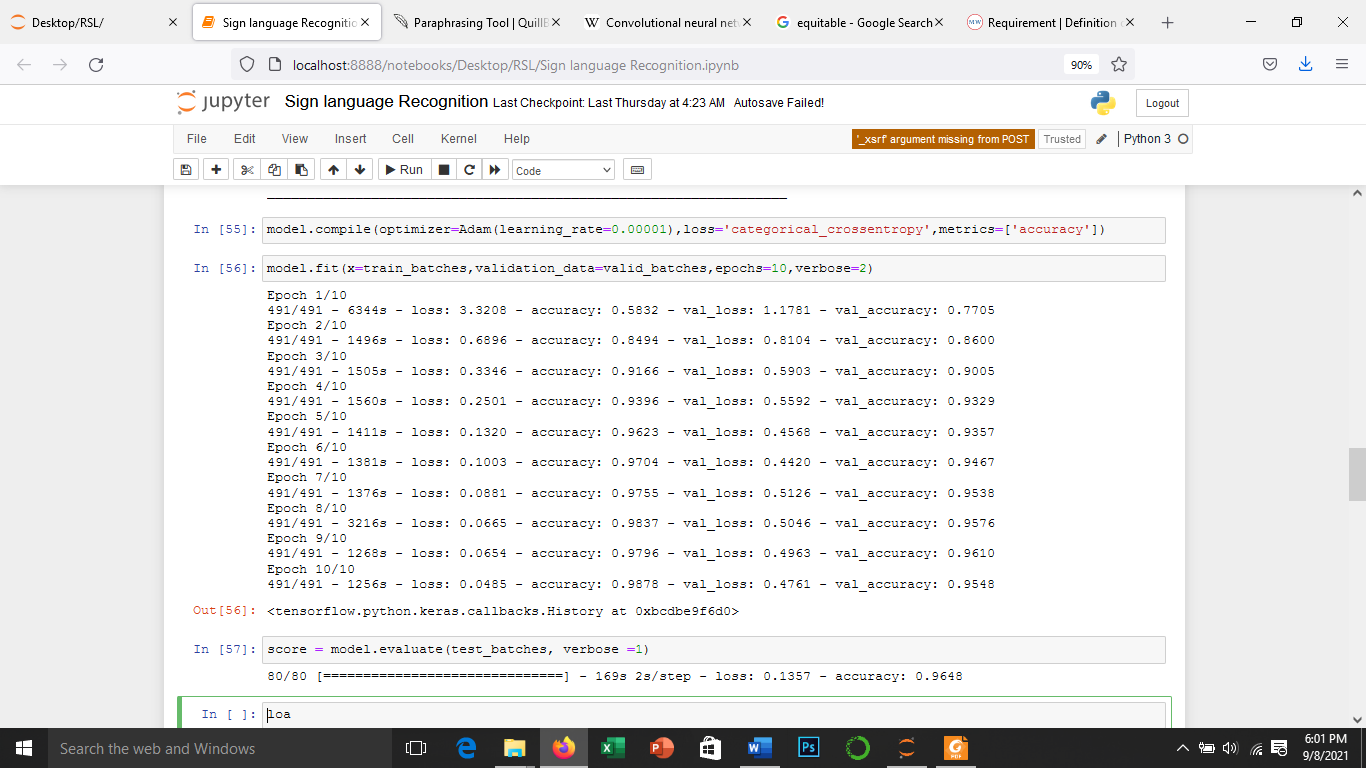
# **APPENDIX B**

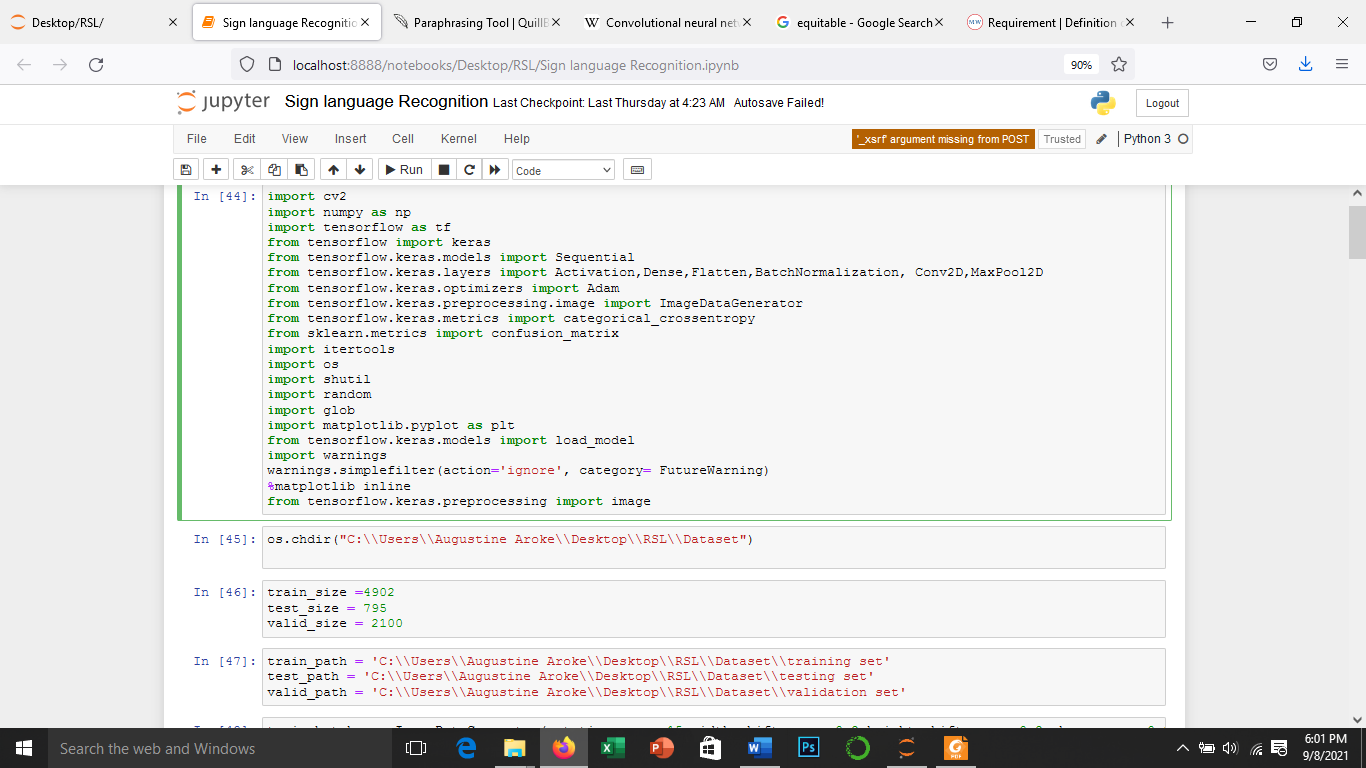
****

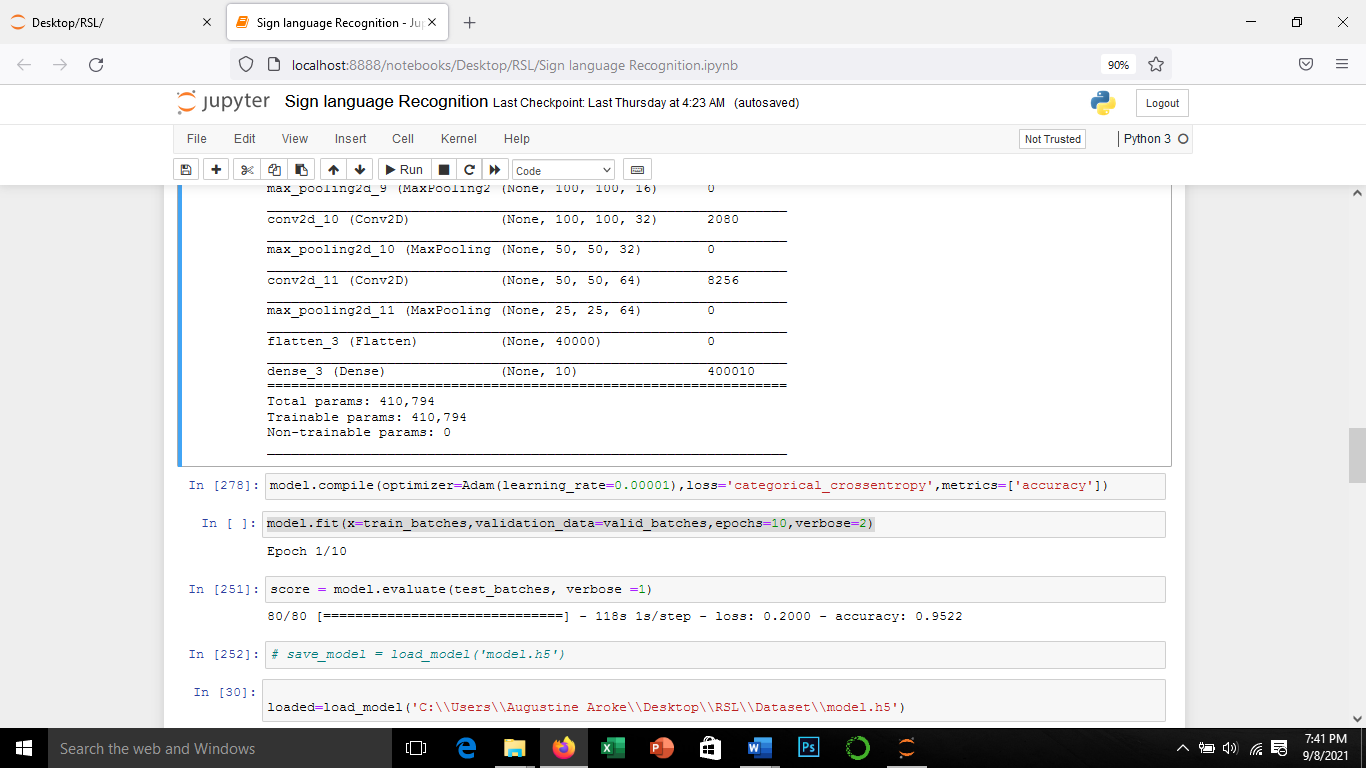
****

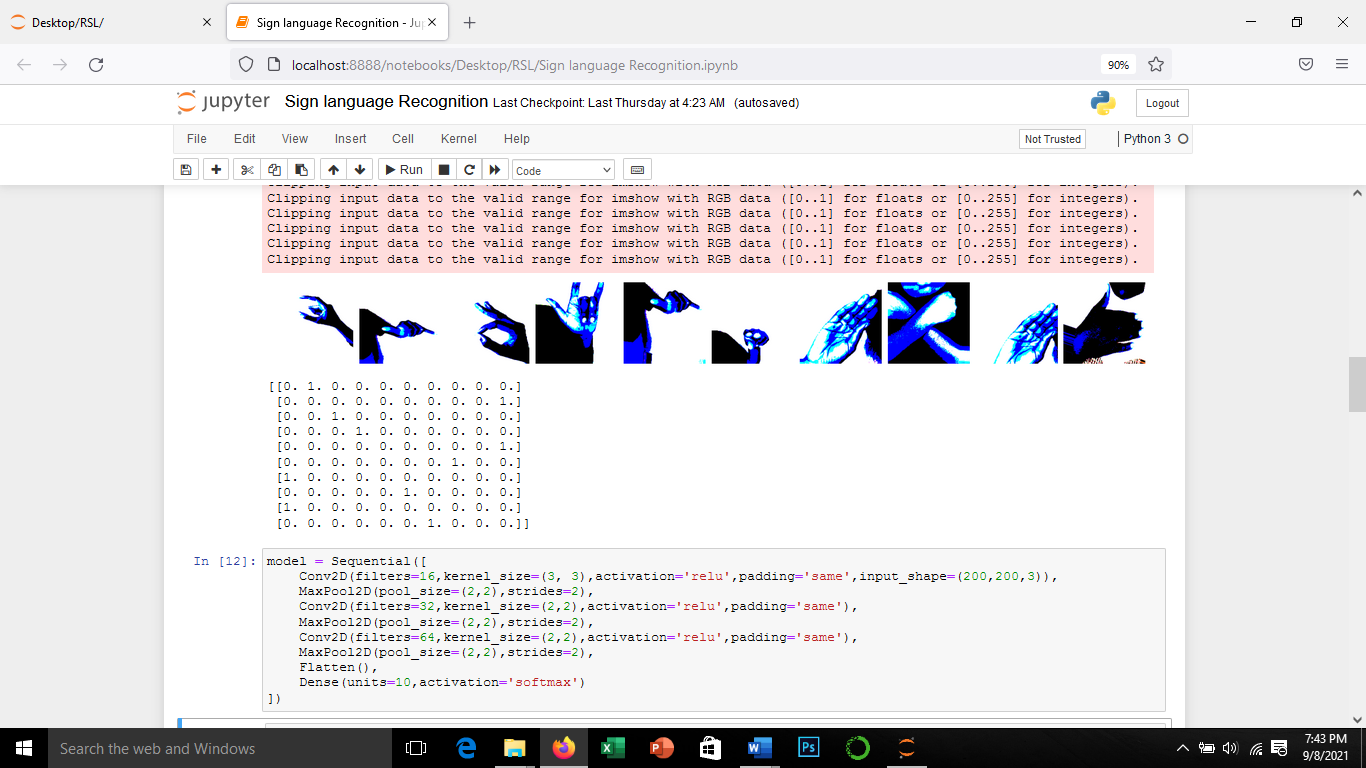
****

****

****

****

****

****