Project Id: 2023PJ-IT06

Project Report

On

SIGN LANGUAGE DETECTION USING COMPUTER VISION

A Project Report

Submitted

in Partial Fulfillment of the Requirements

for the Degree of

Bachelor of Technology

in

Information Technology

by

Kaustubh Ranjan (1902900130027)

Manak Rawal (1902900130034)

Under the Supervision of

Prof. Shivani Sharma

ABES INSTITUTE OF TECHNOLOGY
GHAZIABAD

AFFILIATED TO
Dr. A.P.J. ABDUL KALAM TECHNICAL UNIVERSITY,
LUCKNOW, UTTAR PRADESH.

JUNE-2023

DECLARATION

This is to certify that the project report entitled "**SIGN LANGUAGE DETECTION USING COMPUTER VISION**" is an authentic work carried out by us in the partial fulfillment of the requirements for the award of the degree of B. Tech in Information Technology under the guidance of Prof. Shivani Sharma. The matter embodied in this project work has not been submitted earlier for award of any degree or diploma to the best of our knowledge and belief.

Signature :

Name : Kaustubh Ranjan

Roll No. : 1902900130027

Signature :

Name : Manak Rawal

Roll No. : 1902900130034

Date:

Place:

CERTIFICATE

This is to certify that project entitled "**SIGN LANGUAGE DETECTION USING COMPUTER VISION**" submitted in partial fulfillment of the requirement for the award of degree of B. Tech in the Department of Information Technology, from ABES Institute of Technology Ghaziabad, affiliated to the Dr. A. P. J. ABDUL KALAM TECHNICAL UNIVERSITY, LUCKNOW, UTTAR PRADESH is an authentic work done by Mr. Kaustubh Ranjan (1902900130027) and Mr. Manak Rawal (1902900130034) under my guidance. The matter embodied in this project work has not been submitted earlier for award of any degree or diploma to the best of my knowledge and belief.

Project Supervisor	**Project Co-ordinator**
Prof. Shivani Sharma	Dr. Bipin Kumar Rai/ Prof. Shivani Sharma

Head of Department

Prof. Sumit Kumar

ACKNOWLEDGEMENT

It gives us a great sense of pleasure to present the Project Report undertaken during B. Tech. Final Year. We owe special debt of gratitude to Prof. Shivani Sharma, Professor, Department of Information Technology, ABES Institute of Technology, Ghaziabad for her constant support and guidance throughout the course of our work. Her sincerity, thoroughness and perseverance have been a constant source of inspiration for us. It is only her cognizant efforts that our endeavors have seen the light of the day.

We also take the opportunity to acknowledge the contribution of Prof. Sumit Kumar, Head of Department, Department of Information Technology, ABES Institute of Technology, Ghaziabad for his full support and assistance during the development of the project.

We also do not like to miss the opportunity to acknowledge the contribution of all faculty members of the department for their kind assistance and cooperation during the development of our project. Last but not the least, we acknowledge our friends for their contribution in the completion of the project.

Signature :

Name : Kaustubh Ranjan

Roll No. : 1902900130027

Signature :

Name : Manak Rawal

Roll No. : 1902900130034

ABSTRACT

Sign Language is a critical tool to aid the hearing-impaired population and allow them to meet the ends with the normal people. For the purpose of translating sign language, the camera is the primary element used in Sign Language Recognition (SLR). Mostly existing SLR through image processing uses high quality cameras whereas to reach larger audience, we need a better approachability, hence this paper proposes using of normal cameras like of smartphones and webcams. This paper depicts the significance of encompassing intelligent solution into the SLR systems and meets the requirement of a SLR web portal that is universally and easily available for every needful person. Altogether, it is anticipated that this study will promote the production of intelligent-based SLR, the accumulation of knowledge, and will give readers, researchers and practitioners a roadmap for future direction.

Keywords: Computer Vision, Machine Learning, CNN, Sign Language, Gesture Detection;

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

INTRODUCTION

Sign language is the mode of communication which uses visual ways like expressions, hand gestures, and body movements to convey meaning. Sign language is extremely helpful for people who face difficulty with hearing or speaking. Sign language recognition refers to the conversion of these gestures into words or alphabets of existing formally spoken languages. Thus, conversion of sign language into words by an algorithm or a model can help bridge the gap between people with hearing or speaking impairment and the rest of the world.

Vision-based hand gesture recognition is an area of active current research in computer vision and machine learning. Being a natural way of human interaction, it is an area where many researchers are working on, with the goal of making human computer interaction easier and natural, without the need for any extra devices. So, the primary goal of gesture recognition research is to create systems, which can identify specific human gestures and use them. For example, to convey information. For that, vision-based hand gesture interfaces require fast and extremely robust hand detection, and gesture recognition in real time. Hand gestures are a powerful human communication modality with lots of potential applications and in this context, we have sign language recognition, the communication method of deaf and mute people.

1.1 REVEALING STATISTICS ON SIGN LANUAGE

Fig. 1.1 depicts Between 1990 and 2019, the crude prevalence rate of all hearing loss increased by 27·8% (95% UI 26·6–29·0), from 15·9% (15·3–16·6) in 1990 to 20·3% (19·5–21·1) in 2019. By contrast, the global age-standardised prevalence rate of all hearing loss remained stable, changing from 19·1% (18·4–19·9) in 1990 to 19·3% (18·5–20·0) in 2019. The increase in hearing loss cases while age standardised rates remained stable indicates that increases in prevalent cases are driven by population growth and ageing.

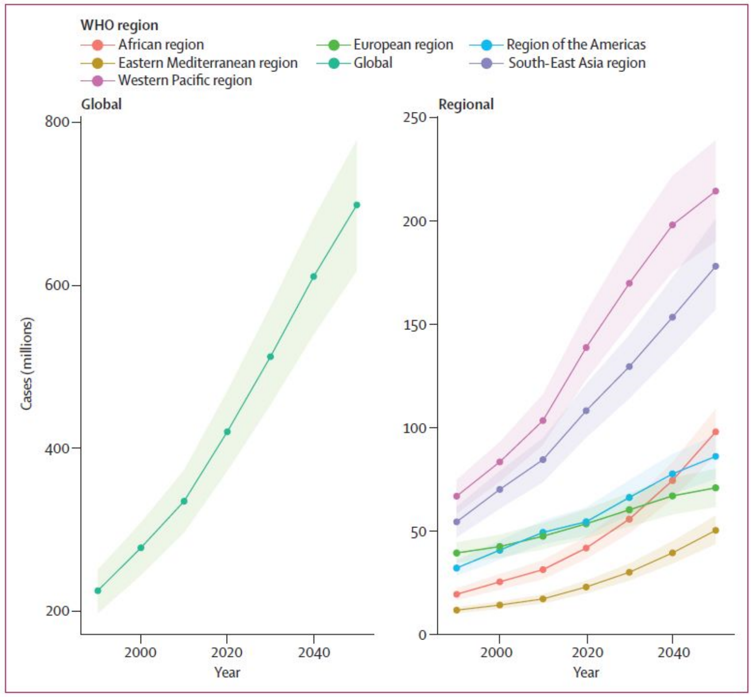

Fig. 1.1: Prevalence of hearing loss 35 dB or greater,1990-2019, with forecasts to 2050, by WHO region. Shading represents 95% UI. UI=uncertainty interval

Fig. 1.2 shows academic achievements of DHH (Deaf or Hard of Hearing) and normal hearing people. Only 18% of DHH students earn heir undergraduate degree compared to 33% of normal hearing individuals. Looking at this graph, we can see the achievement gap doubles after high school.

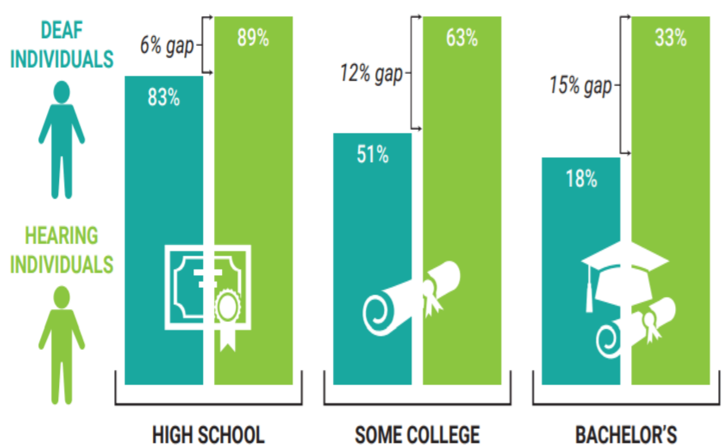

Fig. 1.2: Academic achievements analysis of DHH in US (2017)

1.1.1 INTERNATIONAL UNEMPLOYMENT OF DEAF AND DUMB

Fig.1.1.1 depicts employment of DHH (Deaf or Hard of Hearing) and normal hearing people. Additionally, only 48% of individuals who are DHH were employed, compared to 72% of hearing people. While these statistics may not be surprising to professionals who support DHH students, they continue to be alarming and motivating.

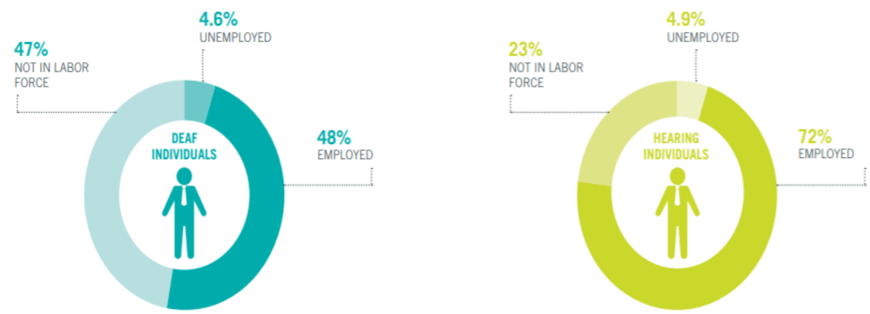

Fig. 1.1.1: Employment analysis of DHH in US (2016

1.1.2 UNEMPLOYMENT OF DEAF AND DUMB IN INDIA

Fig. 1.1.2 allows us to take a notice on the situation of D&D (Deaf and Dumb) in India. Not only are these statistics important to educators and policy makers, but also to the teens we work with. When they turn their graduation tassel from the right to the left, that is the last time they can fully rely on supportive adults in school to advocate for them.

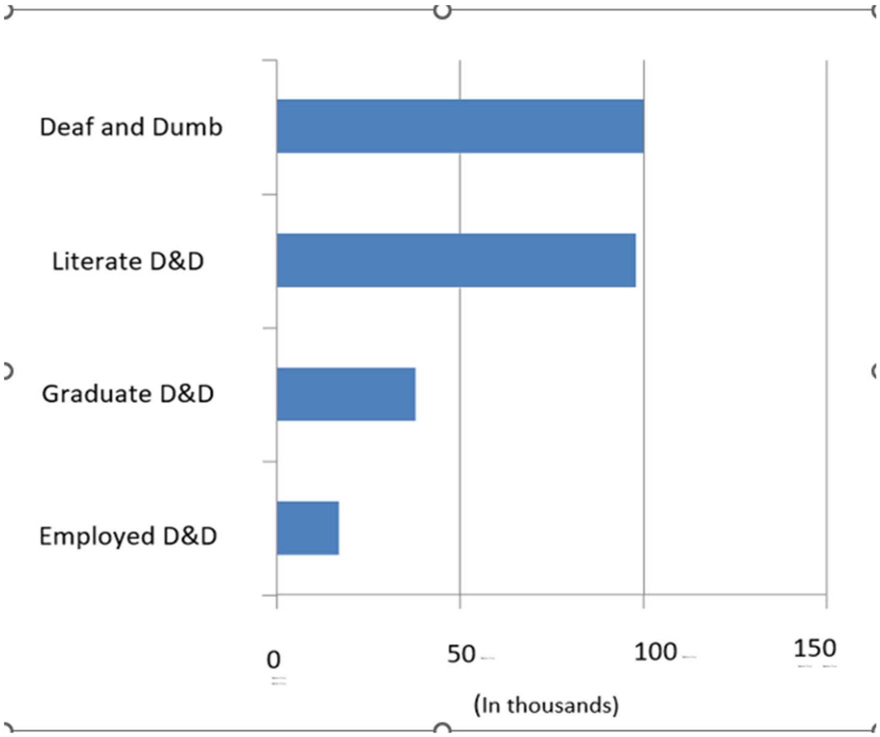

Fig. 1.1.2: Employment analysis of deaf and dumb population of India

CHAPTER 2

LITERATURE REVIEW

The purpose of our literature survey was to propose new research questions or hypotheses, evaluate existing ideas and techniques, and obtain a greater grasp of the subject field. The procedures followed in our work focuses on use of different methods mentioned below. Following observations were made after having a detailed study of previously acquired results in image processing.

2.1 IMAGE ACQUISITION

The process of acquiring digital images with various tools like cameras, scanners, or sensors is referred to as image acquisition. In computer vision and image processing applications, it is a crucial step. The subsequent analysis and interpretation of visual data are significantly influenced by the precision and quality of captured images. Depending on the specific needs and the type of the objects or sceneries being photographed, many techniques and technologies are used for image capturing.

2.1.1 CAMERA/WEBCAM

Cameras serve as essential tools for image acquisition in various domains, including photography, computer vision, and scientific research. On the other hand, webcams and smartphone cameras are designed for convenience, allowing users to capture images and videos quickly and easily. [2, 3]Camera/Webcam eliminates the requirement of equipping other devices and is easy to use, making it comparatively the best image acquisition method.

2.1.2 DATA GLOVE

A data glove is a cutting-edge and adaptable picture collection device that makes it possible to record hand and finger movements in three dimensions. It is a glove-like device with sensors and trackers that precisely tracks the wearer's hand and fingers' location and orientation.[4]Data glove is a feasible and easy feature extraction device as it prevents influence from outside environment. They are highly expensive with low convenience and fails to maintain originality of interaction.

2.1.3 KINETIC

Kinetic is another image acquisition device which has high usage in various applications involving human computer interaction with limited range depth detection. By capturing a subject's motion and dynamics, kinetic imaging is a novel and cutting-edge method for acquiring images. Kinetic imaging, as opposed to conventional static photography, focuses on capturing the motion, gestures, and changes in an item or scene through time.

2.1.4 LEAP MOTION CONTROLLER

The Leap Motion Controller was created specifically to capture and track hand and finger movements in three-dimensional space with extraordinary accuracy and speed, unlike conventional cameras or sensors.. Leap motion controller has elevated speed processing with high accuracy in recognition, that facilitates in detecting hand and fingers.

2.2 IMAGE ENHANCEMENT

Image enhancement is a fundamental process in the field of image processing aimed at improving the visual quality of an image. It involves a set of techniques and algorithms that are applied to digital images to make them more informative, or suitable for specific applications.

2.2.1 HISTOGRAM EQUALIZATION

[5]Using Histogram Equalization (HE) for images in grayscale gets best result and it has simple implementation. The downside is that image brightness gets modified due to which featured and noise become hard to distinguish.

2.2.2 ADAPTIVE HISTOGRAM EQUALIZATION

[6]Adaptive Histogram Equalization (AHE) has better performance than histogram equalization and is best suited to amplify edges and local contrast of image. By redistributing the pixel intensities, it is a potent image enhancement technique that seeks to boost the contrast and visibility of images.

2.2.3 CONTRAST LIMITED ADAPTIVE HISTOGRAM EQUALIZATION

It is a potent approach of image enhancement that overcomes the drawbacks of conventional histogram equalization techniques. By shifting pixel intensities based on the local image content, CLAHE improves the contrast and details in an image.[7]Original brightness is retained in Contrast Limited Adaptive Histogram Equalization, and reduced noise can be noticed when comparing to HE and AHE.

2.2.4 LOGARITHMIC TRANSFORMATION

[8]Logarithmic Transformation is useful when high intensity pixel values are to be reduced into lower intensities pixel values. It is a key method used in image processing to increase an image's visual quality and dynamic range.

2.3 IMAGE FILTERING

Using different filters or convolution processes, image filtering is a fundamental approach in image processing that tries to improve or modify images. The majority of the time, filters are created as matrices or masks that are convolved with the picture to carry out operations like edge detection, noise reduction, and image enhancement.

2.3.1 MEAN FILTER

[9]Mean filter implementation is easy, but shows significant impact in incorrect pixel value representation. In order to remove noise and smooth images, the mean filter is a popular and straightforward image filtering technique. It functions by substituting each pixel in an image with the average value of those closest to it.

2.3.2 MEDIAN FILTER

A prominent method in image processing for decreasing noise and maintaining significant image features is the median filter.[10]It counters problem of mean filter by retaining image sharpness and thin edges. The median filter, in contrast to linear filters like the mean filter, concentrates on the statistical order rather than the numerical average of the pixel values.

2.3.3 GAUSSIAN FILTER

[11]Gaussian type of noise is best removed using Gaussian filter. The Gaussian kernel, which is a matrix of values deriving from the Gaussian function, is convolved with an image in order for the Gaussian filter to function. This filter successfully blurs the image while keeping crucial structural information, making it mostly used to decrease noise and smooth images.

2.3.4 ADAPTIVE FILTER

Adaptive filters, as opposed to conventional filters with fixed coefficients, continuously adjust their coefficients or weights to reduce mistakes and enhance the desirable signal components. They are especially helpful in situations when the signal qualities change over time or when there is noise or interference because of their versatility.[12] Adaptive filter is better at preserving high frequency parts like edges than linear filter.

2.3.5 WIENER FILTER

[13]Wiener filter is a well-liked picture repair filter. Noise has no effect on it, so it is suitable for utilizing the image's statistical features. A signal or image's quality is improved by the Wiener filter by minimizing he mean square error between the original signal and the filtered version.

2.4 IMAGE SEGMENTATION

In computer vision, splitting an image into meaningful and separate regions or objects is a critical process known as image segmentation. Image segmentation aims to divide an image's pixels or regions into groups according to predetermined standards like color, texture, intensity, or other visual characteristics. By extracting and analyzing specific items or regions from a picture, or segmenting it, we can further process and comprehend the image's content.

2.4.1 THRESHOLDING

[14]Thresholding method as segmentation technique is an efficient and simple strategy. It can be used without any prior knowledge and requires less computation power. Thresholding is a widely used method for image segmentation, which seeks to distinguish objects or regions of interest from the background based on their pixel intensities.

2.4.2 EDGE BASED

[15]Edge based method is suitable for pictures with better object contrast. The goal of edge-based methods for image segmentation is to recognize and define boundaries between various objects or regions within an image.

2.4.3 REGION BASED

[16]Region based method is more useful and less sensitive to noise when the similarity criteria are simple to define. This technique concentrates on segmenting an image into useful areas or parts based on how similar the pixels are within those areas.

2.4.4 CLUSTERING

This technique divides the image into logical and visually meaningful portions by using clustering algorithms. Finding groups of pixels with similar properties, such as colour, texture, or intensity, is the fundamental idea.[17]Because of the usage of the fuzzy partial membership, Clustering method is more applicable to real-world problems.

2.4.5 ARTIFICIAL NEURAL NETWORK

[18]Artificial Neural-Network based method can function without a complicated program, it is less noisy in nature and have become effective tools for picture segmentation problems.

Table 2.1:Literature Survey of different methods of image acquisition and processing

Author	Proposed Work	Advantages	Limitations
[1](Adeyanju et al., 2021)	Using Camera/Webcam as Image Acquisition Technique	The device does not require wearing other external devices, and users only need to use their hands within the camera collection range. Low cost. Convenient and easy to use.	It is greatly affected by environmental factors such as light, skin colour, and occlusion. It requires several image processing techniques which might affect the recognition accuracy.

[1](Adeyanju et al., 2021)	Using Data Glove as Image Acquisition Technique	These devices are not Affected by the external environment when collecting data. It provides an improved recognition accuracy. Extraction of features with sensor based is relatively easier.	It reduces the naturalness of interaction. Inconvenient to use by user. Very expensive
[1](Adeyanju et al., 2021)	Using Kinetic as Image Acquisition Technique	It is useful for many human computer interaction applications. The depth of distance detection is limited.	It can be affected by lighting conditions, hand and face segmentation, complex background, and noise. Kinect not suitable for outdoor applications and sensitivity to sunlight.
[1](Adeyanju et al., 2021)	Using Leap motion controller as Image Acquisition Technique	It has high recognition accuracy and faster processing speed around 200 frames per second. It can detect and track hands, fingers, and finger-like objects.	Due to it highly sensitive, accuracy of the recognition might be affected with small movement in sign position.
[2](Verma et al., 2017)	Using Histogram Equalization(HE) as Image Enhancement Technique	It’s simple to implement and highly effective for grayscale images.	It may increase the contrast of background noise. It is difficult to distinguish between noise and the desired features. It changes the brightness of an image.

[3](Kamal et al., 2019)	Using Adaptive Histogram Equalization(AHE) as Image Enhancement Technique	It is suitable to enhance local contrast and edges in every region of an image. It outperforms the histogram equalization technique.	It has an adverse effect on desired output due to its noise-amplification behaviour. It fails to retain the Brightness on the input image.
[4](Suharjito et al., 2017)	Using Contrast Limited Adaptive Histogram Equalization (CLAHE) as Image Enhancement Technique	It has a reduced noise compared to AHE and HE. It provides local output response and avoids brightness saturation	It produces an unsatisfactory result when the input image has an unbalanced contrast ratio and increased brightness
[5](Chourasiya & Khare, 2019)	Using as Logarithmic transformation Image Enhancement Technique	It is used to reduce higher intensities pixel values into lower intensities pixel values	Applying the technique to a higher pixel value will enhance the image more and cause loss of actual information in the image. It does not apply to all kinds of images.
[6](Dhanushree et al., 2019)	Using Median Filter as an image filtering technique.	It preserves thin edges and sharpness from an input image. Both of the problems of the mean filter are tackled by the median filter.	It is relatively expensive and Complex to compute. It is good only for removing salt and Pepper noise. It is less effective at removing the gaussian type of noise from the image.
[7](Kasmin, 2020)	Using Mean Filter as an image filtering technique.	Easy to implement.	A single wrongly represented pixel value can significantly impact the mean value of all pixels

			in their immediate neighborhood. It blurs an edge when the filter neighborhood crosses a boundary.
[8](Basu, 2002)	Using Gaussian Filter as an image filtering technique.	It is effective for removing the gaussian type of noise. The weights give higher significance to pixels near the edge.	It has high computational time and sometimes removes edges details in an image.
[9], [10](Kaluri & Pradeep Reddy, 2016a, 2016b)	Using Adaptive Filter as an image filtering technique.	It preserves edges and other high-frequency parts better than a similar linear filter	It is computational complexity. There are still some visible distortions available in the image using an adaptive filter. .
[11], [12](Maru et al., 2017; Tania & Rowaida, 2016)	Using Wiener Filter as an image filtering technique.	It is a popular filter used for image restoration. It is not sensitive to noise. Suitable to exploit the statistical properties of the image. The small window size can be used to prevent blurring of edges.	Prior knowledge of the power spectral density of the original image is unavailable in practice. It is comparatively slow to apply because it works in the frequency domain. The output image is very blurred.
[13](Lee et al., 1990) [14](Cheng et al., 2002) [15](Xu et al., 2013) [16](Dong et al., 2008)	Using Thresholding Method as segmentation technique.	It is a fast and straightforward approach. It does not require prior information to operate. It has a low computation cost.	It is highly dependent on peaks, while spatial details are not considered. Sensitive to noise. Selection of an optimal threshold value is difficult.
[17], [18](Bhardwaj & Mittal, 2012; Rashmi et al., 2013)	Using Edge based Method as segmentation technique.	Suitable for images having better contrast between objects.	It is not suitable for images with too much noise or too many edges.

[19](Garcia-Lamont et al., 2018)	Using Region-based Method as segmentation technique.	It is less susceptible to noise and more useful when defining similarity criteria is easy.	It is quite expensive in terms of computation time and memory consumption.
[20], [21](Cebeci & Yildiz, 2015; Ghosh & Dubey, 2013)	Using Clustering Method as segmentation technique.	It's more useful for real-world challenges due to the fuzzy partial membership employed.	Determining membership functions is not easy.
[22](Khan, 2014)	Using Artificial Neural-Network based Method as segmentation technique.	It does not require a complex program to work. Less prone to noise.	Computational time in training is higher.

CHAPTER 3

METHODOLOGY

For the project, our domain analysis primarily focused on comprehending pattern recognition. In order to develop a model capable of recognizing gestures to assist and connect deaf and mute individuals with the world, we employ a comprehensive methodology that combines several techniques. We select the most appropriate machine learning model, considering architectures. The chosen model is then trained on a carefully split dataset, and recognition parameters are optimized. The trained model is evaluated using appropriate metrics such as accuracy and precision to assess its performance. This methodology ensures that the model is capable of recognizing gestures effectively and can serve as a valuable tool to connect deaf and mute individuals with the world around them. The following is a compilation of tools and techniques utilized:

3.1 MEDIAPIPE HOLISTIC

MediaPipe Holistic, developed by Google, is an advanced computer vision solution that enables real-time tracking and comprehension of human body movements and poses. By integrating multiple AI models and technologies, it offers a comprehensive approach that encompasses body tracking, hand tracking, and facial landmark detection capabilities.

The foundation of MediaPipe Holistic is a deep learning-based model that can estimate the 3D pose of a person from a 2D video stream. This model combines convolutional neural networks (CNNs) and geometric reasoning techniques to accurately track the positions and orientations of various body parts, including the head, torso, arms, and legs. It demonstrates robust and reliable tracking performance, even in challenging scenarios involving occlusions and varying camera viewpoints.In addition to body tracking, MediaPipe Holistic includes a dedicated module for hand tracking. This module excels at detecting and tracking the 3D positions of the user's hands in real-time. This functionality is particularly useful for applications such as sign language interpretation, virtual reality experiences, and augmented reality interactions.

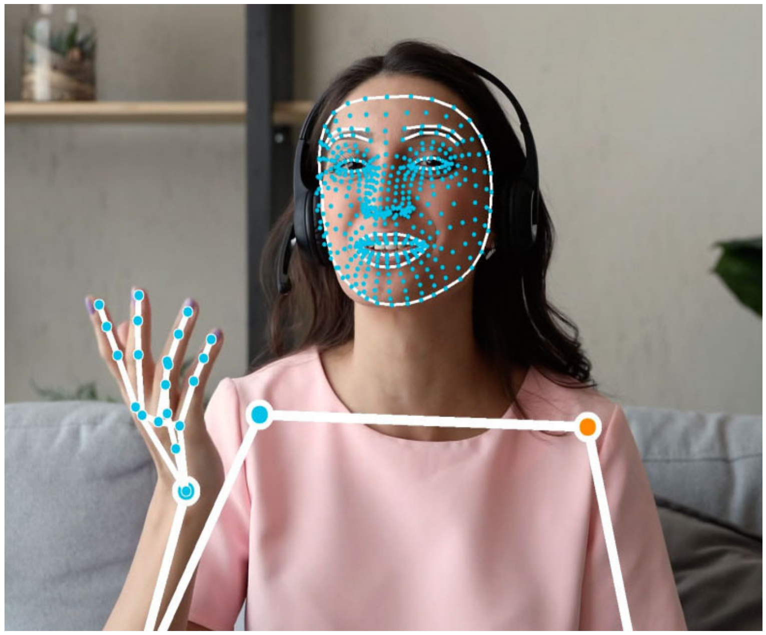

Fig. 3.1:Keypoints displayed using MediaPipe

Moreover, MediaPipe Holistic incorporates a facial landmark detection module that estimates the 2D positions of crucial facial landmarks, including the eyes, nose, and mouth. This capability opens up possibilities for applications requiring facial analysis, emotion recognition, and augmented reality effects on the user's face. By seamlessly integrating these three modules, MediaPipe Holistic offers a holistic understanding of human movements and interactions. The solution is optimized for real-time performance, making it ideal for a wide range of applications, including fitness tracking, gesture-based interfaces, virtual try-on experiences, and immersive gaming. MediaPipe Holistic is built upon the MediaPipe framework, which provides a flexible and scalable pipeline for developing and deploying real-time computer vision applications. The framework offers a diverse array of pre-built components and processing modules, empowering developers to easily customize and extend the functionality of MediaPipe Holistic to suit their specific requirements.

In summary, MediaPipe Holistic is an impressive computer vision solution that leverages body tracking, hand tracking, and facial landmark detection to achieve real-time understanding of human movements and poses. Its versatility, accuracy, and real-time performance make it a valuable tool in diverse fields, including entertainment, healthcare, and human-computer interaction.

3.2 LONG SHORT-TERM MEMORY (LSTM)

Long Short-Term Memory (LSTM) is an architecture of recurrent neural networks (RNNs) specifically designed to overcome the limitations encountered by traditional RNNs in capturing long-term dependencies and preserving information across lengthy sequences. LSTM networks excel in processing and predicting sequential data, such as time series, speech, and natural language.

The distinctive feature of LSTM lies in its ability to selectively retain or discard information from previous time steps using specialized memory cells and gates. These gates, namely the input gate, forget gate, and output gate, allow LSTM to regulate the flow of information within the network, effectively addressing challenges like the vanishing and exploding gradient issues often faced by traditional RNNs.The LSTM architecture comprises a series of memory cells responsible for storing and updating information over time. Each memory cell maintains a cell state, which ensures the preservation of information throughout the sequence. Additionally, three gates—input gate, forget gate, and output gate—govern the information flow. The input gate controls the amount of new information stored in the cell state, the forget gate determines the extent of discarding previous information, and the output gate governs the extraction of relevant information from the current cell state.

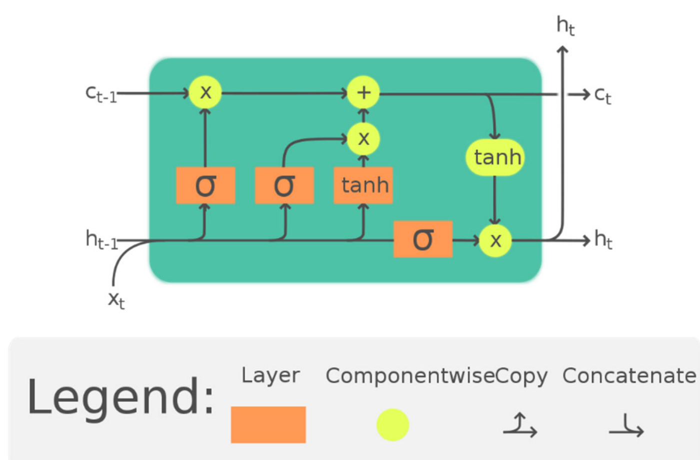

Fig.3.2: LSTM cell

During training, the parameters of an LSTM network are learned through backpropagation through time, which propagates gradients from the output to the input across the entire sequence. This enables the network to adjust its internal state and update the memory cells, facilitating the effective capture of dependencies and patterns within the sequential data. LSTMs have demonstrated remarkable effectiveness in various tasks. In natural language processing, LSTM models have been successfully applied to language modeling, machine translation, sentiment analysis, and text generation. In speech recognition, LSTM networks have achieved impressive results in converting spoken language to written text. Moreover, LSTMs have proven valuable in time series analysis, anomaly detection, and generating music and art.The versatility and power of LSTMs have led to their wide adoption and popularity within the machine learning community. They serve as a fundamental building block for modeling sequential data and have paved the way for more advanced variants, including bidirectional LSTMs, stacked LSTMs, and attention-based models.

In summary, Long Short-Term Memory (LSTM) represents a specialized architecture within recurrent neural networks that effectively addresses the challenges associated with capturing long-term dependencies in sequential data. By incorporating memory cells and gates, LSTM networks can selectively retain or discard information over time, enabling them to accurately model and predict complex sequences. LSTMs have exhibited significant success across various domains and continue to be a critical tool for tasks involving sequential data analysis and generation.

3.3 OPEN CV

OpenCV, an open-source computer vision and machine learning software library, offers a comprehensive range of tools and algorithms for image and video processing, analysis, and comprehension. Developed in C++, it provides interfaces for various programming languages, including Python and Java.

With its extensive collection of functions and modules, OpenCV facilitates numerous tasks such as capturing and preprocessing images and videos, detecting and extracting features, recognizing objects, calibrating cameras, and performing geometric transformations. It supports multiple image formats and delivers efficient algorithms for filtering, segmentation, and edge detection. Additionally, OpenCV includes methods for motion estimation, camera tracking, and optical flow analysis. OpenCV's greatest strengths lie in its versatility and adaptability. It finds applications in robotics, augmented reality, surveillance systems, facial recognition, and medical imaging. Its comprehensive documentation, vibrant community support, and cross-platform compatibility contribute to its popularity in computer vision projects.

Fig. 3.3: OpenCV logo

Furthermore, OpenCV seamlessly integrates with renowned machine learning frameworks like TensorFlow and PyTorch, allowing users to combine the power of deep learning with computer vision algorithms. This integration enables tasks such as object detection, image classification, and semantic segmentation.

Overall, OpenCV empowers developers and researchers to build advanced applications and systems in computer vision and machine learning domains. Its rich set of functions and algorithms make it a powerful tool for a wide range of computer vision tasks.

3.4 TENSORFLOW

TensorFlow, a machine learning framework developed by Google, is an open-source platform that offers a comprehensive suite of tools, libraries, and resources for the creation and deployment of machine learning models. It empowers developers to efficiently design, train, and deploy a diverse array of neural networks and machine learning algorithms.

At its core, TensorFlow adopts a computational graph paradigm, where computations are represented as interconnected nodes that denote mathematical operations, and the edges symbolize the flow of data between these operations. This graph-based approach enables TensorFlow to leverage parallelism and effectively distribute computations across multiple devices, such as CPUs, GPUs, and TPUs, thereby accelerating the training and inference processes. It encompasses a wide spectrum of applications, including deep learning, reinforcement learning, natural language processing, and computer vision. Simplifying the development and training of neural networks, TensorFlow provides a high-level API called Keras.

Fig. 3.4: TensorFlow Logo

Another compelling aspect of TensorFlow is its extensive collection of pre-built models and tools. Within the TensorFlow Model Zoo, developers can access a repository of pre-trained models that cover various tasks, enabling efficient transfer learning and adaptation to specific applications. Moreover, TensorFlow includes TensorFlow Extended (TFX), a production-ready platform for the deployment and management of machine learning pipelines at scale. Furthermore, TensorFlow seamlessly integrates with popular libraries and frameworks, such as OpenCV for computer vision tasks and NumPy for numerical computations. It boasts compatibility with multiple programming languages, including Python, C++, and JavaScript, ensuring accessibility and adaptability across diverse environments. The TensorFlow community is highly active and fosters continual growth of the framework. Abundant resources such as documentation, tutorials, and online forums are readily available, promoting knowledge sharing and collaborative development.

In summary, TensorFlow is an influential and flexible machine learning framework, effectively facilitating the construction and deployment of machine learning models. Its computational graph approach, rich selection of pre-built models, and compatibility with various languages and frameworks contribute to its popularity and make it a top choice for diverse machine learning tasks.

3.5 NUMPY

NumPy, also known as Numerical Python, is a crucial open-source library designed for numerical computations in the Python programming language. It offers efficient data structures and functions that facilitate the manipulation of large, multi-dimensional arrays and matrices, along with a diverse range of mathematical operations for data analysis and manipulation.

At its core, NumPy revolves around the ndarray (N-dimensional array) object, which enables optimized storage and manipulation of homogeneous data. The ndarray provides a robust interface for executing mathematical operations on entire arrays or individual elements. It supports various data types, exhibits efficient memory management, and boasts broadcasting capabilities that enable element-wise operations and mathematical computations across arrays with different shapes and sizes.NumPy encompasses an extensive collection of mathematical functions dedicated to array manipulation. These functions include arithmetic operations, linear algebra routines, Fourier transforms, random number generation, and statistical calculations. The performance of these functions has been optimized, making NumPy an indispensable tool for numerical computing tasks.

Fig. 3.5: NumPy Logo

A significant advantage of NumPy lies in its seamless integration with other scientific libraries within the Python ecosystem. It serves as the foundation for numerous high-level libraries such as Pandas, SciPy, and scikit-learn. These libraries build upon NumPy's functionality to provide advanced data manipulation, scientific computing, and machine learning capabilities. The simplicity and efficiency of NumPy make it an invaluable resource across a wide array of applications. It finds extensive use in scientific research, data analysis, machine learning, image processing, and signal processing. NumPy's intuitive syntax and comprehensive documentation cater to both novice and experienced programmers alike. The vibrant community surrounding NumPy ensures continuous development and support, with regular updates and contributions from a diverse group of users and developers. This active participation has solidified NumPy's popularity, stability, and high-performance, establishing it as a fundamental library within the Python ecosystem for numerical computations.

In conclusion, NumPy is a potent library that provides efficient data structures and functions tailored for numerical computations in Python. Its ndarray object and extensive range of mathematical operations enable efficient manipulation of large arrays and matrices. NumPy's integration with other scientific libraries positions it as a cornerstone for various applications, making it an indispensable component of the scientific Python ecosystem.

CHAPTER 4

IMPLEMENTATION

This section contains the diagrams and explanation of working of our model , as well as our approach to the defined objectives. We elaborate an architectural and sequence diagram , which depicts the functioning of our active work.

4.1 ARCHITECTURAL DIAGRAM

The architectural diagram in Fig. 4.1 describes working infrastructure of the proposed work. It depicts a user logging in to the application provided for the user interaction and after accepting requests for required permissions, it briefly introduces to the functioning of camera as the primary means of interplay between detection system and user. Fleeting operations including pre-processing, key points extraction and database accessing takes place. Data from database is used for training and testing, where testing data is also the feeded input data that is converted from gestures to signs and finally the user experience is received as feedback.

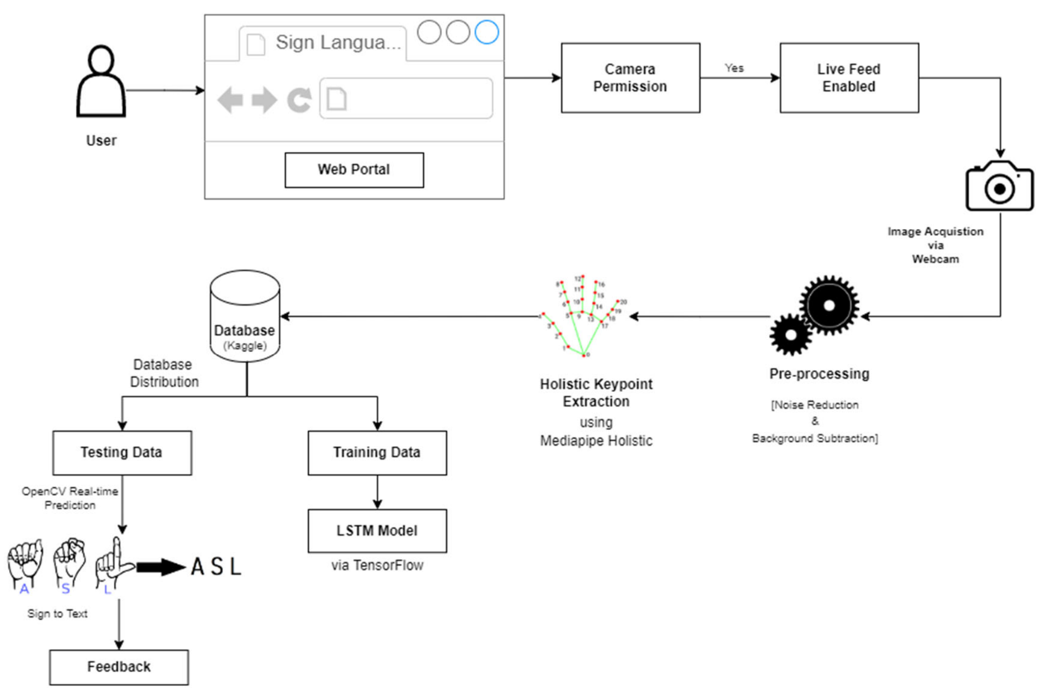

Fig.4.1: Architectural Diagram for Sign Language Detection

4.2 SEQUENTIAL DIAGRAM

Shown above in Fig. 4.2 is the sequence diagram which elaborates the working progression of the proposed work. Initially, the data as images is inputted by the user on the application. This data is stored in database and also used as training data. It is pre-processed and the model is trained. Trained model is saved and used on the same data to give output. The model is loaded and the inputted images after pre-processing is sent back to be classified. Finally, the output is received by the user in displayed text format.

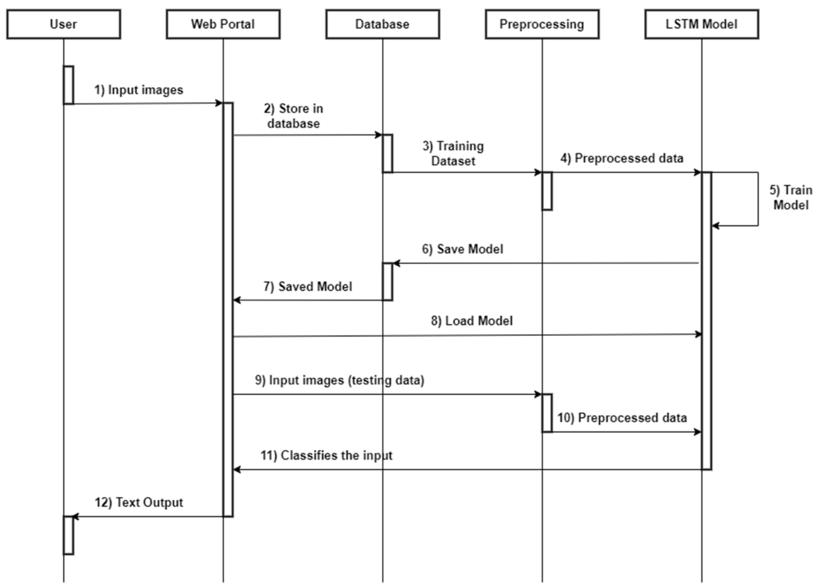

Fig. 4.2: Sequential Diagram for Sign Language Detection

4.3 RESULTS

In order to evaluate the accuracy of our trained gesture prediction model, we conducted a comprehensive test involving 10 participants. The objective was to compare the predicted gestures generated by our model with the actual gestures performed by the participants. By analyzing the results, we aimed to assess the effectiveness and reliability of our trained model.

Each participant was assigned a unique identifier, ranging from 1 to 10. During the testing phase, we captured the gestures using advanced motion-tracking. This data was then fed into our trained model, which utilized machine learning algorithms to predict the gestures based on the input signals. Following are the test results obtained.

Table 4.3(a): No. of times words predicted correctly

Word	Serial No. of Persons (Person abbreviated as "Per.")									
	Per.1	Per.2	Per.3	Per.4	Per.5	Per.6	Per.7	Per.8	Per.9	Per.10
Sometimes	9	10	9	10	10	10	10	10	10	10
Angry	6	7	7	7	5	7	7	6	7	6
Christmas	9	9	10	9	9	10	10	10	10	10
Disciple	9	8	9	8	8	10	9	10	9	10
House	10	10	8	10	8	10	6	9	8	8
Small	8	8	8	7	6	7	7	8	8	7
Narrow	8	9	9	10	10	9	8	10	7	10
Really	9	7	7	8	10	8	7	8	8	8
I(me)	8	7	7	9	8	8	8	7	7	8
Major	9	9	10	9	9	9	10	10	8	9
Love	8	7	9	8	7	8	9	9	6	7

During our evaluation of the trained gesture prediction model, we examined the accuracy on a person-by-person basis for 11 different words. This analysis allowed us to gain insights into the model's performance and its ability to correctly predict specific gestures for each participant.

For each participant, we recorded the accuracy percentage for each word they performed. The accuracy percentage represents the proportion of correct predictions made by the model for a particular word. Following mentioned are the results.

Table 4.3(b):Person wise accuracy for 11 words

Person No.	Accuracy (in %age)

Person 1	83.63
Person 2	82.70
Person 3	84.54
Person 4	86.30
Person 5	81.80
Person 6	87.27
Person 7	82.72
Person 8	88.18
Person 9	80.00
Person 10	84.54

CHAPTER 5

CONCLUSION AND FUTURE SCOPE

In this section, we conclude our discoveries, including comparison of our work with previously acquired results among different researches. We also determine the future scope of this domain of research along with possible forecasting applications.

5.1 ANALYSIS OF CONCLUDED RESUTLS

During our evaluation of word prediction accuracy, we tested our trained model on a diverse group of 10 individuals. The objective was to assess how well the model performed in predicting specific words based on the participants' gestures.

For each participant, we recorded the accuracy of word prediction by comparing the model's predicted word to the actual word performed by the participant. The accuracy was calculated as a percentage, representing the proportion of correct predictions made by the model. Results observed are as follows.

Table 5.1(a): Accuracy of word prediction when tested on 10 persons

Word	Accuracy (in %age)
Sometimes	98
Angry	65
Christmas	96
Disciple	90
House	91
Small	74
Narrow	90
Really	80
I(me)	77

Major	92
Love	78

The existing research methods have variable accuracy results and the findings of the accuracy of different approaches are mentioned in the table below.

Table 5.1(b): Comparison Result of Sign Language Recognition

Method or Application Systems	Accuracy Result (in %age)
Light-HMM	83.6
Multi Stream HMM	86.7
3D-CNN	78.8
EFD and ANN	91.5
Convolution Neural Network	83.0
Kohonen SOM	80.0
SimpSVM	98.9
Eigen Value	97
Tied-Density HMM	91.3
SOFM, SRN, HMM	91.3
SVM and HMM	85.14
SVM	97.5

5.2 CONCLUSION

Sign Language Recognition is an ongoing area of research that currently lacks widespread implementation. This paper presents a computer vision-based solution that offers improved accuracy and aligns with the relevant knowledge domain. The literature survey in this paper concentrates on exploring different approaches for gesture recognition, providing diverse perspectives on the utilization of various methods and techniques.

5.3 FUTURE SCOPE

In the future, further research will involve conducting a comparative analysis of this study in relation to other existing solutions and newly discovered approaches that incorporate emerging technologies. This analysis will consider the evolving nature of technology and its impact on sign language recognition.

REFERENCES

[1] I. A. Adeyanju, O. O. Bello, and M. A. Adegboye, “Machine learning methods for sign language recognition: A critical review and analysis,” *Intelligent Systems with Applications*, vol. 12, p. 56, 2021, doi: 10.1016/j.iswa.2021.20.

[2] N. Verma, M. D.-J. of Engineering, T. (IRJET), and undefined 2017, “Contrast enhancement techniques: A brief and concise review,” *academia.edu*, 2017, [Online]. Available: https://www.academia.edu/download/54173429/IRJET-V4I7375.pdf

[3] S. M. Kamal, Y. Chen, S. Li, X. Shi, and J. Zheng, “Technical Approaches to Chinese Sign Language Processing: A Review,” *IEEE Access*, vol. 7, pp. 96926–96935, 2019, doi: 10.1109/ACCESS.2019.2929174.

[4] Suharjito, R. Anderson, F. Wiryana, M. C. Ariesta, and G. P. Kusuma, “Sign Language Recognition Application Systems for Deaf-Mute People: A Review Based on Input-Process-Output,” in *Procedia Computer Science*, Elsevier B.V., 2017, pp. 441–448. doi: 10.1016/j.procs.2017.10.028.

[5] A. Chourasiya and N. Khare, “A Comprehensive Review Of Image Enhancement Techniques,” *International Journal of Innovative Research and Growth*, vol. 8, no. 6, May 2019, doi: 10.26671/ijirg.2019.6.8.101.

[6] M. Dhanushree, R. Priyadharsini, and T. S. Sharmila, “Acoustic image denoising using various spatial filtering techniques,” *International Journal of Information Technology 2019 11:4*, vol. 11, no. 4, pp. 659–665, Nov. 2019, doi: 10.1007/S41870-018-0272-3.

[7] F. Kasmin, “A Comparative Analysis of Filters towards Sign Language Recognition,” *International Journal of Advanced Trends in Computer Science and Engineering*, vol. 9, no. 4, pp. 4772–4782, Aug. 2020, doi: 10.30534/ijatcse/2020/84942020.

[8] M. Basu, “Gaussian-based edge-detection methods - A survey,” *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*, vol. 32, no. 3, pp. 252–260, Nov. 2002, doi: 10.1109/TSMCC.2002.804448.

[9] R. Kaluri and C. Pradeep Reddy, “A framework for sign gesture recognition using improved genetic algorithm and adaptive filter,” *Cogent Eng*, vol. 3, no. 1, p. 1251730, Oct. 2016, doi: 10.1080/23311916.2016.1251730.

[10] R. Kaluri and C. H. Pradeep Reddy, “Sign gesture recognition using modified region growing algorithm and Adaptive Genetic Fuzzy Classifier,” *International Journal of Intelligent Engineering and Systems*, vol. 9, no. 4, pp. 225–233, 2016, doi: 10.22266/ijies2016.1231.24.

[11] S. Tania and R. Rowaida, “A Comparative Study of Various Image Filtering Techniques for Removing Various Noisy Pixels in Aerial Image,” *International Journal of Signal Processing, Image Processing and Pattern Recognition*, vol. 9, no. 3, pp. 113–124, Mar. 2016, doi: 10.14257/ijsip.2016.9.3.10.

[12] M. Maru, P. G. Scholar, and M. C. Parikh, “Image Restoration Techniques: A Survey,” 2017.

[13] S. U. Lee, S. Y. Chung, and R. H. Park, “A comparative performance study of several global thresholding techniques for segmentation,” *Comput Vis Graph Image Process*, vol. 52, no. 2, pp. 171–190, Nov. 1990, doi: 10.1016/0734-189X(90)90053-X.

[14] H. D. Cheng, X. H. Jiang, and J. Wang, “Color image segmentation based on homogram thresholding and region merging,” *Pattern Recognit*, vol. 35, no. 2, pp. 373–393, Nov. 2002, doi: 10.1016/S0031-3203(01)00054-1.

[15] W. Xu, Q. Li, H. J. Feng, Z. H. Xu, and Y. T. Chen, “A novel star image thresholding method for effective segmentation and centroid statistics,” *Optik (Stuttg)*, vol. 124, no. 20, pp. 4673–4677, Nov. 2013, doi: 10.1016/J.IJLEO.2013.01.067.

[16] L. Dong, G. Yu, P. Ogunbona, and W. Li, “An efficient iterative algorithm for image thresholding,” *Pattern Recognit Lett*, vol. 29, no. 9, pp. 1311–1316, Nov. 2008, doi: 10.1016/J.PATREC.2008.02.001.

[17] Rashmi, M. Kumar, and R. Saxena, “Algorithm and Technique on Various Edge Detection : A Survey,” *Signal Image Process*, vol. 4, no. 3, pp. 65–75, Jun. 2013, doi: 10.5121/sipij.2013.4306.

[18] S. Bhardwaj and A. Mittal, “A Survey on Various Edge Detector Techniques,” *Procedia Technology*, vol. 4, pp. 220–226, 2012, doi: 10.1016/j.protcy.2012.05.033.

[19] F. Garcia-Lamont, J. Cervantes, A. López, and L. Rodriguez, “Segmentation of images by color features: A survey,” *Neurocomputing*, vol. 292, pp. 1–27, Nov. 2018, doi: 10.1016/J.NEUCOM.2018.01.091.

[20] Z. Cebeci and F. Yildiz, “Comparison of K-Means and Fuzzy C-Means Algorithms on Different Cluster Structures,” *Journal of Agricultural Informatics*, vol. 6, no. 3, Nov. 2015, doi: 10.17700/JAI.2015.6.3.196.

[21] S. Ghosh and S. K. Dubey, “Comparative Analysis of K-Means and Fuzzy C-Means Algorithms,” *International Journal of Advanced Computer Science and Applications*, vol. 4, no. 4, 2013, doi: 10.14569/IJACSA.2013.040406.

[22] W. Khan, “Image Segmentation Techniques: A Survey,” *Journal of Image and Graphics*, pp. 166–170, 2014, doi: 10.12720/JOIG.1.4.166-170.