Report Plag Check

by Rahul Jha

Submission date: 23-Apr-2023 12:53PM (UTC-0400)
Submission ID: 2072809935
File name: Plag_Check.pdf (882.43K)
Word count: 8864
Character count: 46923

1. INTRODUCTION

1.1 Problem Statement

Sign language is an essential tool for communication among deaf individuals. It is estimated that there are approximately 70 million deaf people worldwide, and many of them rely on sign language as their primary means of communication. However, not everyone understands sign language, which can lead to communication barriers and exclusion for deaf individuals. Additionally, many people are interested in learning sign language as a means of communicating with deaf individuals or as a new language skill.

Despite the significant demand for sign language learning resources, there are limited resources available, particularly for people who want to learn sign language in Hindi and English, two of the most widely spoken languages in India. Traditional methods of learning sign language can be tedious and uninspiring, which can discourage people from continuing to learn.

In response to this, the aim of this project is to develop a sign language learning application that will help people learn sign language in Hindi and English through fun and interactive games. This application will enable users to learn and practice sign language at their own pace while engaging in various games that make learning fun and enjoyable.

The application will include a vast library of signs and gestures in Hindi and English and detailed descriptions and examples of their usage. The learning process will be gamified, making it more engaging and interactive, and it will allow users to track their progress and set learning goals.

By developing this application, we can make sign language learning more accessible, interactive, and enjoyable for people in India and beyond. This will ultimately help break down communication barriers between hearing and deaf individuals.

The project will use computer vision techniques, specifically using the OpenCV library to capture images and process them. The dataset collected will include images of two different sign languages: American Sign Language (ASL) and Hindi Sign Language (HSL). ASL uses a manual

alphabet based on handshapes to represent letters and words, while HSL is used to communicate in the Hindi language through hand gestures and movements.

For each label (i.e., each letter or character in the sign language alphabet), a certain number of images has been collected. For ASL, we collected 3000 images per label, and for HSL, we collected 1000 images per label having a total of 78,000 (26 x 3000) images for ASL and 3000 (3 x 1000) images for HSL in our dataset. We kept the image size as 300x300 for both ASL and HSL. This means that all images in our dataset have the same dimensions of 300 pixels by 300 pixels.

By using these datasets, we can train the model to recognize the hand gestures and movements associated with each sign language character. The model can then be used to interpret sign language in real-time applications or to transcribe sign language into text or speech.

The application will include various features that will help users learn sign language effectively. One of the main features will be gamification, which will make learning sign language more engaging and enjoyable. The application will include various games that will test the user's knowledge of sign language and help them learn new signs and gestures.

Another essential feature of the application will be the ability to track progress and set learning goals. This feature will allow users to monitor their progress and identify areas where they need to improve. By setting learning goals, users will be motivated to continue learning sign language and achieve their objectives.

The application will also include a vast library of signs and gestures in Hindi and English. This library will be organized in a way that makes it easy for users to find the sign or gesture they are looking for quickly. Each sign or gesture will be accompanied by detailed descriptions and examples of its usage, making it easier for users to understand how to use each sign or gesture correctly.

By developing this application, we can make sign language learning more accessible, interactive, and enjoyable for people in India and beyond, which will ultimately help break down communication barriers between hearing and deaf individuals.

1.2 Project Overview

The idea for this project was born out of the need for better accessibility for deaf individuals and to promote sign language learning among the wider community. Currently, there is a significant communication gap between hearing and deaf individuals, which can lead to exclusion and discrimination. This project aims to address this issue by providing a tool that can help break down communication barriers and promote inclusion.

The application will be designed to be user-friendly and accessible to people of all ages and backgrounds. The user interface will be intuitive, and the application will be designed to accommodate different learning styles. The learning process will be structured in a way that makes it easy for users to progress at their own pace and track their progress.

The application's gamification aspect is particularly crucial as it makes learning more fun and engaging. Users will be able to learn sign language through various interactive games, making it more enjoyable and motivating. This approach also helps to reinforce learning and retention, as users are more likely to remember what they learn when they are actively engaged in the learning process.

The application will also allow users to set learning goals and track their progress. Users will be able to monitor their progress, and the application will provide feedback on areas that require improvement. This feature is essential as it helps users to stay motivated and track their progress, making the learning process more meaningful and rewarding.

The vast library of signs and gestures in Hindi and English will also be a critical feature of the application. The library will include all the essential signs and gestures necessary for effective communication in both languages. The descriptions and examples of usage will be detailed and informative, providing users with a comprehensive understanding of the language.

The application will also be designed to accommodate different levels of proficiency. Beginners will be able to start with the basics and gradually progress to more advanced levels. The more advanced levels will include complex signs and gestures and require more advanced skills. This approach ensures that users can progress at their own pace and build on their existing knowledge.

In conclusion, the Sign Language Learning Application project aims to develop an application that promotes accessibility, inclusion, and sign language learning. The application will be designed to be user-friendly, interactive, and gamified, making it more engaging and enjoyable. The vast library of signs and gestures in Hindi and English, along with detailed descriptions and examples of their usage, will provide users with a comprehensive understanding of the language. The ability to track progress and set learning goals will also make the learning process more meaningful and rewarding. Ultimately, this project has the potential to improve communication between hearing and deaf individuals and promote a more inclusive society.

The project is basically been divided into two stages:

1. Stage 1- Here we have trained the model in order to detect the sign languages based on ASL and HSL Dataset, and then convert the model to TensorFlow lite model to be integrated into the application.
2. Stage 2- Build the application which gives an interactive and fun way to learn sign languages, in form of games with the help of Android Studio which would be compatible with android devices.

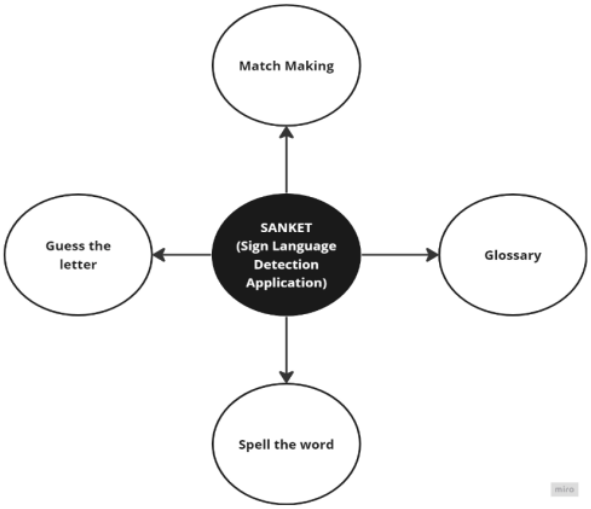

Fig. 1 Project Module

1.3 Expected Outcome

The project aims to achieve the following outcomes:

- Improved accessibility and communication: The application will help break down communication barriers and improve accessibility for deaf individuals, allowing them to communicate more easily with hearing individuals who have learned sign language.

- Increased sign language learning: The application will promote sign language learning among the wider community, helping more people to learn sign language in Hindi and English and, in turn, promoting social inclusion.

- Improved sign language skills: The gamified learning process and sign language detection feature will enable users to improve their signing skills and confidence in communicating with deaf individuals.

- Increased awareness and understanding: The application will increase awareness and understanding of the challenges faced by deaf individuals and promote empathy and inclusivity in society.

- Technological innovation: The application will utilize cutting-edge technologies such as computer vision, machine learning, and gamification to create an engaging and interactive learning experience.

Overall, the Sign Language Learning Application project's expected outcome is to make sign language learning more accessible, engaging, and enjoyable for people in India and beyond, helping break down communication barriers and promoting social inclusion.

1.4 Hardware & Software Specifications:

Software Requirements

- Operating System: Windows
- SDK: OpenCV, TensorFlow, Keras, Numpy
- Android Studio, Jupyter Notebook

Hardware Requirements

- The Hardware Interfaces Required are:
- Android Phone with Camera
- Camera:3MP (minimum)
- Ram: Minimum 4GB or higher Android Version: 10+

1.5 Other Non-Functional Requirements:

Safety Requirements

If there is extensive damage to a wide portion of the database due to catastrophic failure, such as a disk crash, the recovery method restores a past copy of the database that was backed up to archival storage (typically tape) and reconstructs a more current state by reapplying or redoing the operations of committed transactions from the backed-up log, up to the time offailure.

Security Requirements

Security systems need database storage just like many other applications. However, the special requirements of the security market mean that vendors must choose their database partner carefully so that the vendor should not be able to vendor lock in.

Software Quality Attributes

- Availability: The user should be able to use the application anytime for their learning,
- Correctness: The word detection should be proper and AI trainer should check for the gestures so that the user gets good experience on the app.
- Maintainability: The admin should maintain the updates of the app so that the users can try new games and keep learning, checking their status.

1.6 Report Outline:

Chapter 1: Introduction

- Briefly introduce the project and its purpose
- Describe the problem statement and the need for the project
- Provide an overview of the project
- List the expected outcomes
- Mention the hardware and software specifications
- Specify the other non-functional requirements
- Give an outline of the report

Chapter 2: Literature Survey

- Provide a review of the existing work in the field of sign language recognition and learning
- Describe the proposed system in detail
- Conduct a feasibility study of the proposed system

Chapter 3: System Design & Analysis

- Discuss the project perspective and the requirements
- List the performance requirements
- Describe the system features in detail
- Explain the methodology used in the project
- Discuss the testing process

Chapter 4: Results and Outputs

- Present the results and outputs of the project
- Provide snapshots of the outcomes

Chapter 5: Conclusion

- Summarize the main points of the report
- Discuss the success of the project in meeting the objectives
- Mention the limitations of the project
- Provide recommendations for future work

Chapter 6: References

- List all the references used in the report.

2. LITERATURE SURVEY

2.1 Existing Work

The purpose of this literature survey is to understand the current state of research in sign language detection and identify the most effective techniques for implementing a sign language learning application in Hindi and English.

Our research involved reviewing various papers that discussed sign language detection and recognition techniques for different languages, including American, Indian, Ukrainian, Arabic, Persian, Hindi, and Bangladeshi. We analyzed the different methods, technologies, and model architectures used in these studies and evaluated their effectiveness in terms of accuracy and efficiency.

Based on our analysis, we found that the most commonly used technology for sign language detection is the Convolutional Neural Network (CNN) model. This model has been shown to produce the highest accuracy and is also very efficient. However, we also found other architectures such as TensorFlow Object Detection, Hidden Markov Model (HMM), Radial Distance, and Fourier Transform, which have been used for implementation and have given decent accuracy results.

We also found that there is a lack of available data sets for Hindi Sign Language and that there have been very few implementations of sign language detection for Hindi. While some implementations have been done for building Android applications, the accuracy of these implementations in detecting sign gestures in real-time through the camera has been very low.

In Table 1, we have presented a comparative study of previous works that have been done on different sign languages using various datasets and model architectures to compare accuracy, advantages, and disadvantages. This study helped us understand the different types of implementations that have been done and decide which model would be best to achieve the highest accuracy for both Hindi and English languages.

Our analysis of the literature has informed our decision to use the CNN model architecture for our sign language learning application, given its proven accuracy and efficiency. We plan to develop a dataset for Hindi Sign Language and use the CNN model architecture to train our application to detect and recognize sign gestures accurately in real-time.

In conclusion, our literature survey has provided valuable insights into the state of research in sign language detection and recognition for various languages, including Hindi and English. The results of our analysis have informed our decision to use the CNN model architecture and develop a dataset for Hindi Sign Language to achieve the highest accuracy possible for our sign language learning application.

Table 1: Comparison of previous works on different languages.

Author Name	Technology Used	Accuracy	Languages	Advantage	Dis-advantage
[1] Aman Pathak, Avinash Kumar, Priyam, Priyanshu Gupta, Gunjan Chugh	CNN	88.7	American Sign Language	Have worked on 5 different words.	Gives accurate result only under control light and ensity
[2] Sharvani Srivastava, Amisha Gangwar, Richa Mishra, Sudhakar Singh	TensorFlow Object Detection	85.45%	Indian Sign Language	Despite the dataset being small, the system has achieved an average confidence rate.	It is not a real time SLR.
[3] Paranjoy Paul, Dr. G N Rathna	CNN	92.7%	Indian Sign Language	The offline accuracy is quite higher.	The training accuracy was low due to augmentation.
[4] Subhashini Yadav, Shreyashi Raj, Kashish Awasthi, Rohit Bidwan, Lokesh Jain	TensorFlow object detection	80.25%	American Sign Language	The model could be changed for better accuracy.	The training dataset used was very less.

Author Name	Technology Used	Accuracy	Languages	Advantage	Dis-advantage
[5] Pratibha Gupta, Priya Rajput, Priyanka Katiyar, Srishti Sharma, Shaivya Shukla, Dr. Umesh Dwivedi	CNN	74%	Hindi Sign Language	The model could be trained with more dataset and achieve better accuracy.	The model has trained only for 12 letters.
[6] Amrita Thakur, Pujan Budhathoki, Sarmila Upreti, Shirish Shrestha, Subarna Shakya	CNN	92.7%	American Sign Language	Text To Sign Language feature has been added.	Have trained the model only from letter A-H.
[7] M. Davydov, I. Nikolski, V. Pasichnyk	HMM	91.7%	Ukraine Sign Language	Use of hand shape recognition based on fingertip location.	Have be trained only for static signs.
[8] Shanableh T. and Assaleh K.	HMM	Sentence-75% Word-94%	Arabic Sign Language	The word recognition is quite accurate which could further be improved.	No grammar has been imposed.
[9] Bahare Jalilian and Abdolah Chalechale	Radial Distance and Fourier Transform	95.6%	Persian Sign Language	The approach needs no constraints.	The model works only for static image.
[10] Oishee Bintey Hoque, Mohammad Imrul Jubair , Md. Saiful Islam , Al-Farabi Akash , Alvin Sachie Paulson	CNN	98.2%	Bangladeshi Sign Language	The detection time is very fast.	Limitation while recognizing similar letter.

Sign language detection is a rapidly evolving field, with researchers developing new techniques and models to accurately detect and interpret hand gestures and movements. Here are some of the key techniques used for implementation:

Deep Learning-based Sign Language Recognition:

Deep learning algorithms have been shown to be highly effective in detecting and recognizing sign language gestures. One example of this is the work done by researchers at the University of Washington, who developed a deep learning-based sign language recognition system that achieved state-of-the-art results on the American Sign Language (ASL) dataset.

Hand Pose Estimation:

Hand pose estimation is a technique that involves estimating the position and orientation of the hand in 3D space. This is important for sign language detection because it can help improve the accuracy of hand gesture recognition. Researchers at the University of California, Berkeley developed a hand pose estimation system that achieved high accuracy on the ASL dataset.

Hybrid Approach:

A hybrid approach that combines multiple techniques, such as hand tracking and motion analysis, has also been used for sign language detection. One example of this is the work done by researchers at the University of Science and Technology in China, who developed a system that used hand tracking and motion analysis to recognize sign language gestures in real-time.

Convolutional Neural Networks:

Convolutional Neural Networks (CNNs) have also been used for sign language detection. CNNs are a type of deep learning algorithm that are highly effective at image classification and object recognition. Researchers at the Indian Institute of Technology developed a CNN-based sign language recognition system that achieved high accuracy on the Indian Sign Language dataset.

Transfer Learning:

Transfer learning is a technique that involves using a pre-trained model for a related task and then fine-tuning it for the target task. This has been used for sign language detection, with researchers at the University of Southern California developing a transfer learning-based sign language recognition system that achieved high accuracy on the ASL dataset.

2.2 Proposed System

The model is trained for the ASL (American Sign Language) and HSL (Hindi Sign Language), which is converted to the TensorFlow Lite model and integrated in the application for the recognition of the sign languages in order to learn in real time using the camera of the smartphone practicing through games using hand gestures, where the sign language detected as a gesture is converted in the form of text.

The application currently will be supported only in the android devices, and is built for the detection of the American Sign Languages which is the letters from A-Z with a great accuracy and is very fast. The app is designed in such a way where user have different games as a lesson which they could play in order to learn sign languages and see their status of learning through a dashboard.

The model we have selected to train is CNN, as based on the research a CNN with a 3 layer gives a higher accuracy, and very efficient and faster, which could be converted to the TensorFlow lite model to be integrated in the application for the recognition of sign language with the help of the camera.

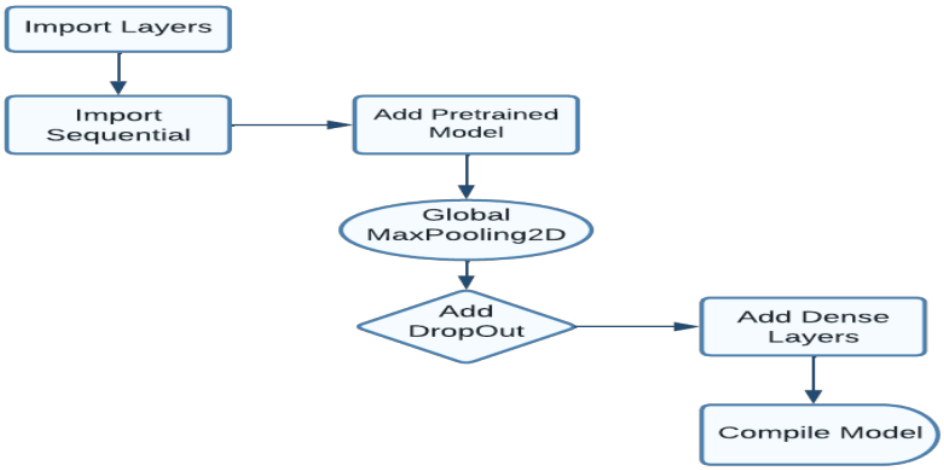

Fig.2 CNN Model

The Data Flow for the system: -

User Interface - this component includes the different screens and features of your sign language learning app, such as the letter guessing game, the letter matching game, the spelling game, and the glossary. Users interact with the app through the user interface, and their input is captured by the app and processed by the different components.

TensorFlow Lite Model - this component represents the machine learning model you trained to recognize American Sign Language gestures. It receives input from the user interface and generates output based on its predictions. The model is integrated into your app as a TensorFlow Lite model, which allows it to run on mobile devices with limited resources.

Progress Database - this component represents a database that stores the progress made by each user in the app. This includes the number of correct guesses, matches, and spellings, as well as the

current level of proficiency. The progress database is updated each time a user completes a game or a lesson, and is used to track the user's progress over time.

Glossary Database - this component represents a database that stores the sign language terms and their meanings. The glossary database is used to provide users with information about the signs they are learning and can be accessed through the glossary feature in the app.

User Account Database - this component represents a database that stores information about the users of the app, including their name, email, and password. The user account database is used to authenticate users when they log in to the app and to track their progress over time.

The data flow diagram shows the flow of data between these components. When a user interacts with the app through the user interface, their input is sent to the TensorFlow Lite Model for processing. The model generates an output based on its predictions, which is then stored in the Progress database to track the user's progress. The glossary database is used to provide users with information about the signs they are learning, and the user account database is used to authenticate users and track their progress over time.

2.3 Feasibility Study

1. **Technical Feasibility**

   The application would be able to run on all the android devices which has android version 10+, and a RAM of 4gb, as it requires a faster processing system for the detection and recognition of the sign language.

2. **Economic Feasibility**

   As the complete model is trained on the Jupyter Notebook platform and the application is build on Android Studio, which are available free of cost. Hence, no cost burden is there in term of developing the project.

3. **Data Feasibility**

When considering the American Sign Language Data, the data is available on Kaggle which is easier to do the operation on whereas for the Hindi Sign Language dataset, the data must be created manually based on different signs.

4. **Time Feasibility**

The time required to build the complete project would take 9 months, until when we can develop the outcome as a product.

5. **User Feasibility**

As the app is accessible in both Hindi and English languages, it can be used by most Indian users, and as the UI is quite simpler it is very convenient even for a person who doesn't know either of the languages.

3. SYSTEM DESIGN & ANALYSIS

3.1 Project Perspective

The development of a solution to help the deaf and mute communicate has been a challenging problem for a long time. With the advancement of technology, various solutions have been proposed, but most of them have been limited in terms of accessibility and affordability. However, with the increasing use of smartphones and mobile applications, there is an opportunity to create a cost-effective and easily accessible solution that can help the deaf and mute communicate more effectively.

The first step towards developing a solution is to understand the needs and requirements of the deaf and mute community. There are various communication methods used by the deaf and mute community, such as sign language, lip reading, and written communication. Among these, sign language is the most widely used and is considered as the primary language of the deaf community. Therefore, to develop a solution that is effective, it is essential to focus on sign language as the primary communication method.

To build a learning platform where any person can learn sign language at their comfort, the application should be designed to be user-friendly and interactive. One way to achieve this is by using gamification techniques to make the learning process fun and engaging. Gamification is the use of game design elements in non-game contexts to improve user engagement, motivation, and learning outcomes. By integrating gamification elements into the sign language learning application, learners can enjoy the process of learning sign language, and this can lead to better learning outcomes.

In conclusion, developing a solution to help the deaf and mute communicate and building a learning platform for sign language is an important step towards creating a more inclusive society. By using gamification techniques, the learning process can be made more engaging, fun, and interactive. Additionally, by designing the platform to be accessible and inclusive, learners can access the platform from anywhere and at any time, which can lead to better learning outcomes. With the increasing use of smartphones and mobile applications, there is a significant opportunity to create

a cost-effective and easily accessible solution that can make a positive impact on the lives of the deaf and mute community.

3.2 Performance Requirements

- To use the sign language learning application effectively, it is essential to have a smartphone that meets specific requirements. The first requirement is that the smartphone should have an Android version of 10 or above. This is because the sign language learning application is designed to be compatible with the latest version of Android, which provides better security, stability, and performance.

- Another crucial requirement is that the smartphone should have a RAM of at least 4GB. This is because the sign language learning application requires a lot of resources to function smoothly without any lag. With a 4GB RAM, the smartphone can handle multiple applications and processes simultaneously, which is essential for a seamless learning experience.

- The camera quality of the smartphone is also an important consideration. The ideal camera resolution for the sign language learning application is 8 megapixels or higher. This is because the application uses the smartphone camera to detect sign language gestures accurately. With a high-resolution camera, the application can capture the sign language gestures more precisely, which is essential for effective learning.

- Apart from these requirements, the smartphone should also have a good battery life to ensure that the application can be used for an extended period without interruption. Additionally, the smartphone should have a good internet connection to access the sign language learning platform and download new updates.

- It is also essential to consider the storage capacity of the smartphone. The sign language learning application requires a significant amount of storage space to store data and resources required for the application to function correctly. Therefore, the smartphone should have a minimum of 32GB of internal storage to ensure that the application can be installed and used effectively.

In conclusion, the choice of smartphone for the sign language learning application is critical to ensure that the application can be used effectively without any lag or interruption. The smartphone should have an Android version of 10 or above, a RAM of at least 4GB, a camera resolution of 8 megapixels or higher, a good battery life, and a minimum of 32GB of internal storage. By meeting these requirements, learners can have a seamless and effective learning experience using the sign language learning application.

3.3 System Features

The application has 3 major features:

1) Practice Games

We have given three types of games where we have tried to gamify the user experience on the application where user can see their stats.

Game 1:

Guess the letter

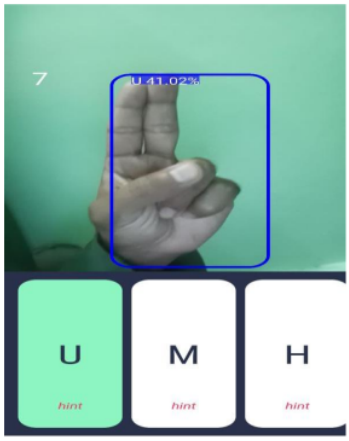

Fig.3

The application is designed with a user-friendly interface that allows the user to easily access the camera on their device. Once the user has access to the camera, they can then begin to use the application to learn sign language.

The application would have a screen where a letter would be displayed in a box. The user would then need to guess the correct letter by making the corresponding hand gesture for that letter in front of the camera. The application would use computer vision algorithms to detect the hand gesture and then compare it to the correct gesture for that letter.

If the user guesses the letter correctly, they would receive positive feedback such as a sound or visual cue. If they guess incorrectly, they would receive negative feedback and be prompted to try again. The application could also provide feedback on the correctness of the user's hand gesture, such as suggesting ways to improve their form.

To ensure that the user is learning sign language effectively, the application also provides additional resources such as video tutorials, a glossary of sign language terms, and quizzes to test the user's knowledge.

Game 2:

Spell the Word with Sign

Fig. 4

When the game starts, a word will appear on the screen and the player will have 30 seconds to guess the correct letters that make up the word using sign language.

To make a guess, the player can use the sign language alphabet to form each letter of the word. For example, if the word is "CAT", the player would use signs to represent the letters C, A, and T. If the player guesses a letter correctly, it will appear in the correct spot in the word on the screen. If the player guesses a letter incorrectly, nothing will happen and they will need to keep guessing until they get it right.

Game 3:

Make a match!

Fig.5

"Make a match!" is a game designed to help users test their knowledge of ASL (American Sign Language) alphabet sign gestures. In this game, the user will be presented with a set of letters in alphabetical order and a set of corresponding sign gestures. The goal of the game is for the user to match each letter with its respective sign gesture. To make a match, the user can tap on a letter and then tap on the corresponding sign gesture. If the match is correct, the letter and its corresponding sign gesture will be highlighted in green and stay on the screen. If the match is incorrect, the user can try again until they get it right.

The game will continue until the user has successfully matched all the letters with their respective sign gestures, or until the user chooses to end the game. Once the game is completed, the user can see their results and review any incorrect matches they made during the game.

"Make a match!" can be a fun and interactive way for users to practice and test their knowledge of ASL alphabet sign gestures. This game can be especially useful for those who are new to ASL and are learning the alphabet sign gestures for the first time.

2) Glossary

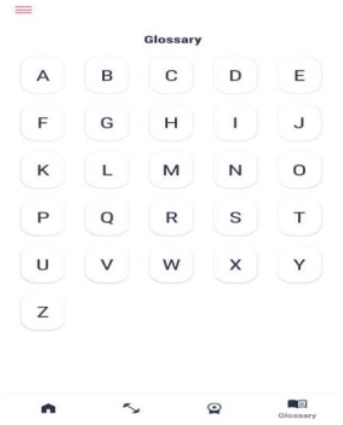

Fig.6

The ASL Alphabet glossary is a curated collection of sign gestures that correspond to each letter of the alphabet in ASL (American Sign Language). This glossary is designed to help beginners learn the ASL alphabet and its corresponding sign gestures.

The glossary is organized in alphabetical order, from A to Z, and each letter is accompanied by a video or image of the corresponding sign gesture. The user can click or tap on each letter to see the video or image of the sign gesture and can practice making the sign gesture themselves.

This glossary can be a valuable resource for beginners who are new to ASL and want to learn the alphabet sign gestures. By providing a visual representation of each sign gesture, users can better understand and practice the correct hand movements and positions for each letter.

Using this glossary can help users to build a strong foundation in ASL and improve their ability to communicate with the Deaf and hard-of-hearing community. With practice and dedication, users can become more proficient in ASL and improve their ability to understand and use the language in daily life.

3) User Stats

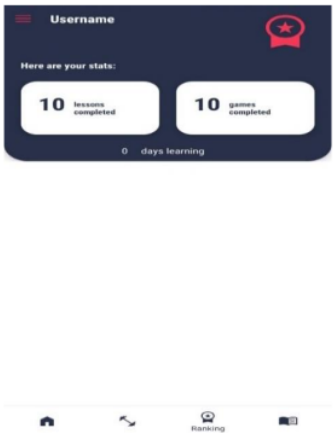

Fig.7

The statistics feature is designed to help users track their progress and measure their learning in the ASL (American Sign Language) app. This feature collects data on the user's learning activities, including games played, words learned, and time spent on the app, and displays this data in an easy-to-read format.

The user can access their statistics at any time from the app's menu. Once the user opens the statistics page, they will see a summary of their learning progress, including the number of words learned, the number of games played, and the total time spent on the app.

The statistics feature also provides detailed information on the user's performance in each game. For example, the user can see how many words they spelled correctly in the word-spelling game, or how many matches they made in the "Make a match!" game.

By tracking their progress over time, users can identify areas where they need to focus their learning efforts and see how they are improving in different aspects of ASL.

This feature can be a powerful motivator for users to continue practicing and learning, as they can see their progress and feel a sense of accomplishment as they reach their goals.

Overall, the statistics feature is a useful tool for users to track their learning progress and stay motivated as they continue to improve their ASL skills.

3.4 Methodology

- **Data Acquisition**

The sign language recognition model that we built is a computer vision system that uses machine learning techniques to recognize and interpret hand gestures and movements in real-time. Specifically, we are using the OpenCV library to capture images and process them.

We have collected a large dataset of images that represent different characters in two different sign languages: American Sign Language (ASL) and Hindi Sign Language (HSL). ASL is a manual alphabet-based sign language used to represent letters and words in English, while HSL is used to communicate in the Hindi language through hand gestures and movements.

For each label (i.e., each character in the sign language alphabet), our team has collected a certain number of images. For ASL, we have collected 3000 images per label, and for HSL, we have collected 1000 images per label, resulting in a total of 78,000 (26 x 3000) images for ASL and 3000 (3 x 1000) images for HSL in the dataset.

All of the images in the dataset are the same size, with dimensions of 300 pixels by 300 pixels. This consistency in image size is important because it allows your model to process the images efficiently and accurately.

To train the sign language recognition model, we used machine learning algorithms to analyze the dataset and identify patterns in the images that correspond to specific sign language characters. This will enable the model to recognize and interpret new hand gestures and movements in real-time.

Once the model is trained, it can be used in a variety of real-world applications, such as translating sign language into text or speech, or enabling communication between hearing and deaf or hard-of-hearing individuals. This has the potential to make a significant impact on the lives of people who rely on sign language as their primary means of communication.

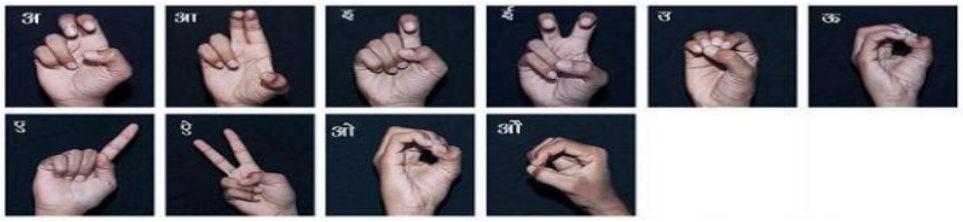

Fig. 8 Hindi Sign Language

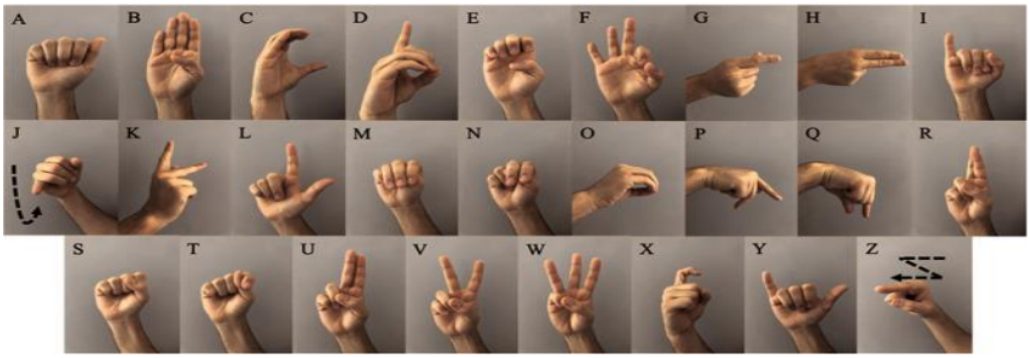

Fig.9 American Sign Language

- **Training The Model**

The first model is trained on the American Sign Language (ASL) dataset, which consists of 78,300 images of 29 different classes. The second model is trained on the Hindi Sign Language dataset, which consists of 5,000 images of 10 different classes.

In both cases, the images are resized to a resolution of 32x32 pixels, which reduces the computational complexity of the model and makes it easier to train. The images are then split into two arrays: one containing the image data and one containing the corresponding labels.

For the ASL dataset, the model is trained for 5 epochs using a batch size of 64. This means that the model is shown 64 images at a time during each training iteration, and the weights of the model are updated based on the error between the predicted labels and the true labels. The CNN

architecture used in this model consists of three layers, which are responsible for learning different features of the input images. The output of the final layer is passed through a SoftMax activation function, which normalizes the outputs to a probability distribution over the 29 possible classes. This allows the model to make predictions about which sign is being displayed in a given image.

The accuracy achieved by this model on the test set is 99%, which means that it correctly predicts the class of 99% of the images it is shown.

For the Hindi Sign Language dataset, the model is trained for 15 epochs using a batch size of 64. This dataset has fewer images and classes than the ASL dataset, but the same CNN architecture and training procedure is used. The accuracy achieved by this model on the test set is 93.7%, which is still quite high, but slightly lower than the ASL model.

- **Converting model to TensorFlow lite model**

When we convert a TensorFlow model into a TensorFlow Lite (TFLite) model, we are essentially creating a smaller and more efficient version of the original model that is optimized for deployment on mobile, embedded, and IoT devices with limited processing power and memory. The conversion process involves several steps, including:

Quantization: This is the process of reducing the precision of the weights and activations in the model from 32-bit floating-point numbers to 8-bit integers or even lower precision. This reduces the size of the model and makes it faster to run on devices with limited computational resources.

Optimization: This involves applying a set of optimizations to the model, such as pruning, folding, and fusion, to further reduce its size and improve its performance on device.

Conversion: This is the final step where the model is converted into the TFLite format, which can be easily deployed on mobile and embedded devices.

Once the model is converted into TFLite format, it can be easily integrated into mobile or web applications using the TFLite Interpreter API. The TFLite Interpreter allows us to run the model on device, making it faster and more efficient compared to sending the input data to a remote server for processing.

Using the TFLite model, we can create mobile applications that can recognize sign languages in real-time, even without an internet connection. This allows for a more seamless and responsive user experience, as well as improved privacy and security since the data does not need to be transmitted over the internet.

- **Building the android application**

The app we built using TensorFlow Lite model on Android Studio is an exciting and innovative way to help people learn sign language in a fun and interactive way. By integrating the model into games and activities, users can practice their sign language skills and improve their understanding of the language.

One of the key features of the app is its ability to detect and recognize the alphabet from A to Y using the TensorFlow Lite model. This means that users can sign the letters and have the app recognize them, which is a crucial aspect of learning sign language. By integrating this feature into games and activities, users can practice signing the letters in different contexts and scenarios, which can help with retention and recall.

Another impressive aspect of the app is its ability to recognize signs in any background. This is achieved through the use of a carefully curated dataset that includes images with different backgrounds. By training the model on a diverse set of images, it becomes more robust and able to recognize signs in a variety of contexts.

Overall, the app provides a valuable tool for people looking to learn sign language, and it showcases the power of machine learning and computer vision in making language learning more accessible and engaging.

3.5 Testing Process

Testing the application involves several steps to ensure that the application performs accurately and reliably. Here are some of the key steps we followed:

- **Unit Testing:**
  This involves testing individual components of the application, such as the TensorFlow Lite model and the user interface elements, to ensure that they are functioning as expected. Unit testing helps catch bugs early in the development process.

- **Functional Testing:**
  This involves testing the application, to ensure that it meets the functional requirements specified in the project scope. For a sign language detection application, this might include testing the accuracy of the model, the responsiveness of the user interface, and the overall performance of the application.

- **User Acceptance Testing:**
  This involves testing the application with real users to ensure that it meets their needs and expectations. This can be done through focus groups, surveys, or beta testing with a select group of users.

- **Regression Testing:**
  This involves testing the application after each update or change to ensure that it continues to perform as expected. Regression testing helps catch any issues that may have been introduced during the development process.

- **Performance Testing:**
  This involves testing the application under different load and stress conditions to ensure that it performs reliably in different scenarios. For example, you might test the application under different network speeds or device configurations to ensure that it can handle a variety of situations.

Overall, testing is a crucial part of the development process for any application. By following these steps, we ensured that the application is accurate, reliable, and user-friendly, and that it meets the needs of the target audience.

4. RESULTS AND OUTPUTS

4.1Results and Output

The output of the project is an android application that helps users learn sign language. The application provides an interactive and fun way for users to learn sign language and provides a variety of features to make the learning process more engaging.

One of the key features of the application is its ability to recognize sign language gestures using computer vision and machine learning algorithms. The application uses a CNN (Convolutional Neural Network) model to recognize different sign language gestures and convert them into text. This feature enables the user to practice signing in real-time, and get immediate feedback on their accuracy.

The application also includes a gamification element to make the learning process more engaging. It includes various games and challenges that users can complete to learn sign language. For example, the user can play games that test their ability to recognize signs or match signs with their corresponding letters. This feature keeps the user engaged and motivated to continue learning.

Another feature of the application is its ability to provide a glossary of sign language words. Users can use this feature to look up the sign language equivalent of a particular word or phrase. The application provides high-quality videos that demonstrate the sign language gesture, making it easy for the user to learn and practice.

The application also includes a dashboard feature that helps the user track their progress. The dashboard provides an overview of the user's performance, including their scores, levels, and achievements. This feature helps the user understand their strengths and weaknesses, and identify areas where they need to improve.

In summary, the output of the project is an android application that helps users learn sign language. It provides an interactive and fun way to learn sign language and includes a variety of features to make the learning process engaging and motivating. The application includes computer vision and

machine learning algorithms to recognize sign language gestures, gamification features to keep the user engaged, a glossary of sign language words, and a dashboard to track the user's progress.

4.2 Output Snapshots

1. Result for the American Sign Language Dataset

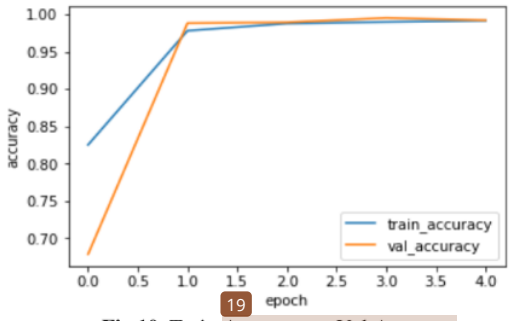

Fig 10. Train Accuracy vs Val Accuracy

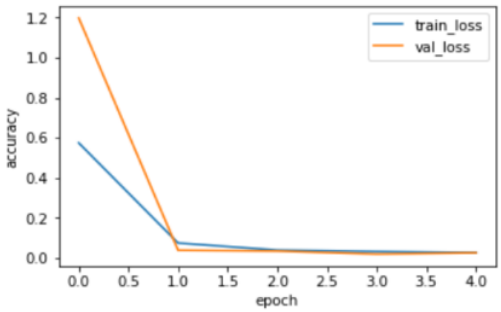

Fig 11. Train Loss vs Val Loss

```
Train time:  1159.551103591919
Test accuracy: 0.9945976734161377
Test loss: 0.01823984459042549
Test time:  8.337937831878662
```

2. Result for Hindi Sign Language

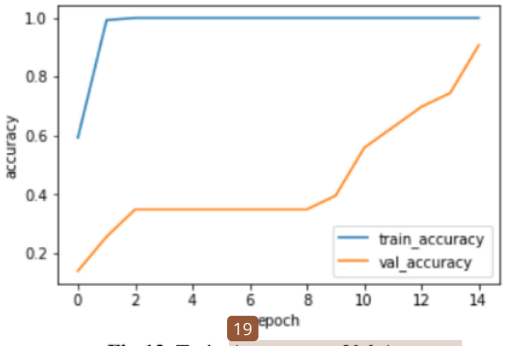

Fig 12. Train Accuracy vs Val Accuracy

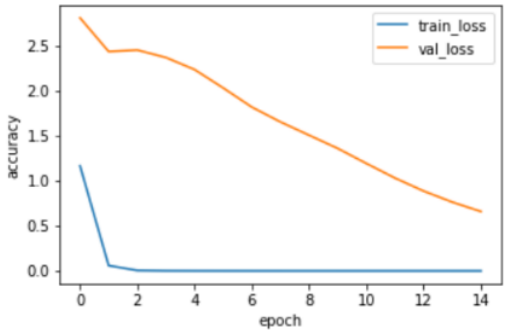

Fig 13. Train Loss vs Val Loss

```
Train time:  27.547500133514404
Test accuracy: 0.9375
Test loss: 0.6289675235748291
Test time:  0.1109311580657959
```

5. CONCLUSION

5.1 Conclusion

In conclusion, this project aimed to develop a solution to help the deaf and dumb communicate and provide a learning platform where anyone can learn sign language at their own pace. We decided to focus on Hindi and English languages and trained our model using the CNN model to achieve an accuracy of 99% for ASL and 93.7% for HSL.

We started by conducting extensive research on different models for different sign languages and selected the CNN model to train our models. This is because the CNN model is a deep learning model that has been proven to work well with image classification tasks, making it a suitable choice for our project.

We used a dataset of sign language images and trained our models to recognize the different signs accurately. We then evaluated our models' performance and achieved an accuracy of 99% for ASL and 93.7% for HSL, which is a good accuracy level for sign language recognition tasks.

After training our models, we converted the ASL model into a TensorFlow Lite model, which can be easily integrated into an application. The application will enable deaf and dumb individuals and the general public to learn sign language easily by gamifying the complete application. Gamification will make the learning process more fun, interactive, and engaging, keeping learners motivated throughout the process.

In summary, the project successfully developed a solution to help the deaf and dumb communicate and provide a learning platform for anyone to learn sign language at their own pace. We achieved a high level of accuracy with our models and gamified the application to make learning sign language more fun and interactive. The application's accessibility and usability aspects were also considered to ensure a seamless learning experience for all users. This project has the potential to make a significant impact on the lives of the deaf and dumb community and promote better communication and inclusion in society.

5.2 System Usability

User-friendly interface: The application should have a user-friendly interface that is easy to navigate, and the learners can quickly access the different features of the application.

Interactive learning experience: The application should provide an interactive learning experience that is engaging and keeps the learners motivated to continue learning sign language.

Feedback mechanism: The application should provide a feedback mechanism that helps the learners to assess their progress and identify areas that require improvement.

Multiple learning modes: The application should offer multiple learning modes, including videos, audio, images, and quizzes, to cater to different learning styles.

Customizable learning path: The application should allow learners to create their own learning path based on their individual needs and learning pace.

Accessible glossary: The application should provide an accessible glossary of sign language words and phrases to help learners to quickly look up the signs they want to learn.

Gamification: The application should incorporate gamification elements such as badges, rewards, and leaderboards to make the learning experience more fun and engaging.

Progress tracking: The application should provide a progress tracking system that helps learners to track their progress over time and monitor their improvement.

Multilingual support: The application should provide multilingual support to cater to learners who speak different languages.

Technical support: The application should provide technical support to learners who may encounter technical issues while using the application. This can include an FAQ section, email support, or a help desk feature.

By incorporating these system usability aspects, your sign language learning application can provide a seamless and effective learning experience for learners.

5.3Future Scope

- Train the model with all the alphabet in the Hindi Language, creating more datasets for each label to get higher accuracy.
- Use the trained model using Hindi Sign Language to be implemented on the application for giving a user to switch between languages.
- Add more games and lessons which would help people to learn faster and help communicate among themselves.
- We could implement a speech recognition module for users to learn things faster, where they could learn with the help of AI.
- There are different words for the sign language in both Hindi and English words which requires two hands, the model can be trained to detect both hands and help in learning words and sentences too, which could be utilized for the disabled person to write their thoughts using the sign language.

6. REFERENCES

[1] Pathak, Aman & Kumar|priyam|priyanshu, Avinash & Chugh, Gupta|gunjan & Ijmtst, Editor. (2022). Real Time Sign Language Detection. International Journal for Modern Trends in Science and Technology. 8. 32-37. 10.46501/IJMTST0801006.

[2] Srivastava, Sharvani & Gangwar, Amisha & Mishra, Richa & Singh, Sudhakar. (2022). Sign Language Recognition System using TensorFlow Object Detection API.

[3] Paranjoy Paul, Dr. G N Rathna. Real time Indian Sign language recognition.

[4] Subhashini Yadav, Shreyashi Raj, Kashish Awasthi, Rohit Bidwan, Lokesh Jain. International Journal for Research in Applied Science & Engineering Technology (IJRASET). ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538 Volume 10 Issue V May 2022

[5] Pratibha Gupta, Priya Rajput, Priyanka Katiyar, Srishti Sharma, Shaivya Shukla, Dr. Umesh Dwivedi. SIGN LANGUAGE RECOGNITION FOR HINDI VARNAMALA USING CNN. International Research Journal of Modernization in Engineering Technology and Science. Volume:04/Issue:05/May-2022

[6] Amrita Thakur, Pujan Budhathoki, Sarmila Upreti, Shirish Shrestha, Subarna Shakya(2020). Real Time Sign Language Recognition and Speech Generation. Journal of Innovative Image Processing (JIIP) (2020) Vol.02/ No. 02 Pages: 65-76

[7] Davydov M. V, Nikolski I.V. and Pasichnyk V. V. 2010. Real-Time Ukranian Sign Langauge Recognition System. Intelligent Computing and Intelligent Systems(ICIS). IEEE International Conference on 1:875-878, 29-31 October.

[8] Shanableh T. and Assaleh K. 2007. Arabic Sign Langauge Recognition in user-independent mode. Intelligent and Advanced Systems. ICIAS 2007.International Conference on. pp. 597-600.

[9] Bahare Jalilian and Abdolah Chalechale. Persian Sign Language Recognition Using Radial Distance and Fourier Transform. I.J. Image, Graphics and Signal Processing, 2014, 1, 40-46.

[10] O. B. Hoque, M. I. Jubair, M. S. Islam, A. -F. Akash and A. S. Paulson, "Real Time Bangladeshi Sign Language Detection using Faster R-CNN," *2018 International Conference on Innovation in Engineering and Technology (ICIET)*, 2018, pp. 1-6, doi: 10.1109/CIET.2018.8660780.

[11] Kishore, P. V. V., & Rajesh Kumar, P. (2012). A Video Based Indian Sign Language

Recognition System (INSLR) Using Wavelet Transform and Fuzzy Logic. International Journal of Engineering and Technology, 4(5), 537542.

[12] Kalidolda, N., &Sandygulova, A. (2018). Towards Interpreting Robotic System for Finger spelling Recognition in Real-Time. HRI Companion: 2018 ACM/IEEE International Conference on Human-Robot Interaction Companion.

[13] Lahoti, Sakshi & Kayal, Shaily & Kumbhare, Sakshi & Suradkar, Ishani & Pawar, Vikul. (2018). Android Based American Sign Language Recognition System with Skin Segmentation and SVM. 1-6. 10.1109/ICCCNT.2018.8493838.

[14] Cheok, Ming Jin & Omar, Zaid & Jaward, Mohamed. (2016). A mobile application of American sign language translation via image processing algorithms. 104-109. 10.1109/TENCONSpring.2016.7519386.

[15] Mr. Niraj A. Haval , Kedar V. Kulkarni ,Omkar G. Patil ,Aniket M. Mangalekar , Paul P. Panhalkar ,Prithviraj P. Badiger. An Android Application For Interpreting Sign Language Into Text Format Using Deep Learning. International Journal of Research in Advent Technology (IJRAT) Special Issue E-ISSN: 2321-9637.

[16] Preeti Karmore,Nutan Sonwane, Mayuri Gupta, Nainika Mendhe, Shruti Patel, Preet Gajbhiye. SIGN LANGUAGE RECOGNITION USING ANDROID. International Research Journal of Modernization in Engineering Technology and Science. Volume:04/Issue:04/April-2022

[17] A.W. Fitzgibbon, and J. Lockton, “Hand Gesture Recognition Using Computer Vision”, BSc. Graduation Project, Oxford University.

[18] A. Gayathri, A. Sasi Kumar. Sign Language Recognition for Deaf and Dumb People Using Android Environment. INTERNATIONAL JOURNAL OF CURRENT ENGINEERING AND SCIENTIFIC RESEARCH (IJCESR), VOLUME-4, ISSUE-5, 2017

[19] Mandeep Kaur Ahuja, Amardeep Singh, Hand Gesture Recognition Using PCA, International Journal of Computer Science Engineering and Technology (IJCSET), Volume 5, Issue 7, pp. 267-27, July 2015.

[20] Sunitha K. A, Anitha Saraswathi.P, Aarthi.M, Jayapriya. K, Lingam Sunny, Deaf Mute Communication Interpreter- A Review, International Journal of Applied Engineering Research, Volume 11, pp 290-296, 2016.

ANNEXURE I

Research Paper for the said project has been presented in ICCSAI 2022 International Conference On Communication, Security and Artificial Intelligence

Paper Title: CNN based bilingual sign language detection for Hindi and English.

Abstract:

People who are deaf or dumb can communicate with one another by using hand gestures to represent numbers, letters, words, and sentences. But when they have to interact with regular people who do not understand sign language, a difficulty occurs. To address this issue, numerous automation systems have been created, however, most of them require laptops and computers for the sign language detecting module to function. The creation of a mobile app to facilitate communication and learning is crucial to resolving the system's accessibility and portability issues. To make learning and communicating sign language easier, the software uses the camera on a mobile device to recognize signs and outputs them as text and speech. Agile modeling and a constructive research strategy were used for the project. This project focuses on the development of an application that would help people learn sign language in both Hindi and English in real-time through a glossary and practice through different games.

Authors: Rahul Kumar Jha, Aman Gupta, Dr. Shree Harsh Attri

GALGOTIAS UNIVERSITY NAAC GRADE A+ Accredited University ICCSAI IEEE UP SECTION (INDIA) | IEEE CONFERENCE RECORD NO. : #56668

International Conferenceon Communication, Security and Artificial Intelligence ICCSAI - 2022

23rd- 24th December 2022

Certificate of Presenter

This is to certify that Prof./Dr./Mr./Ms Rahul Kumar Jha has presented paper ID 698 entitled CNN based Bilingual Sign Language Detection for Hindi and English .during the **International Conference on communication, security and artificial intelligence, ICCSAI-2022,** (Technically Co-Sponsored by **IEEE UP SECTION**) held on **23rd-24th December 2022** organised by Galgotias University, Greater Noida, Uttar Pradesh, India.

Dr. Avadhesh Kumar
Conference Chair, ICCSAI-2022
PVC, Galgotias University (U.P.)

Dr. Nitin Kumar Gaur
Conference Chair, ICCSAI-2022
Registrar, Galgotias University (U.P.)

Dr. K. Mallikharjuna Babu
Patron, ICCSAI-2022
Vice Chancellor, Galgotias University (U.P.)

NBA | nirf | Institution's Innovation Council | Innovation Cell

GALGOTIAS UNIVERSITY NAAC GRADE A+ Accredited University ICCSAI IEEE UP SECTION (INDIA) | IEEE CONFERENCE RECORD NO. : #56668

International Conferenceon Communication, Security and Artificial Intelligence ICCSAI - 2022

23rd- 24th December 2022

Certificate of Presenter

This is to certify that Prof./Dr./Mr./Ms Aman Gupta has presented paper ID 698 entitled CNN based Bilingual Sign Language Detection for Hindi and English .during the **International Conference on communication, security and artificial intelligence, ICCSAI-2022,** (Technically Co-Sponsored by **IEEE UP SECTION**) held on **23rd-24th December 2022** organised by Galgotias University, Greater Noida, Uttar Pradesh, India.

Dr. Avadhesh Kumar
Conference Chair, ICCSAI-2022
PVC, Galgotias University (U.P.)

Dr. Nitin Kumar Gaur
Conference Chair, ICCSAI-2022
Registrar, Galgotias University (U.P.)

Dr. K. Mallikharjuna Babu
Patron, ICCSAI-2022
Vice Chancellor, Galgotias University (U.P.)

NBA | nirf | Institution's Innovation Council | Innovation Cell

ANNEXURE 2

Participated in the 48hr hackathon “REIGN'23”, event hosted by IGDTUW RESOURCE.

Link: https://unstop.com/hackathons/reign23-igdtuw-delhi-659747

Track- Smart Education

Qualified till the last round, in **Top 10** under 460 teams.

Report Plag Check

ORIGINALITY REPORT

13%	7%	6%	7%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	Submitted to Cleveland State University Student Paper	2%
2	Submitted to Istanbul Aydin University Student Paper	1%
3	www.researchgate.net Internet Source	1%
4	Submitted to Asia Pacific University College of Technology and Innovation (UCTI) Student Paper	1%
5	Submitted to Colorado Technical University Student Paper	1%
6	Uzma Farooq, Mohd Shafry Mohd Rahim, Nabeel Sabir, Amir Hussain, Adnan Abid. "Advances in machine translation for sign language: approaches, limitations, and challenges", Neural Computing and Applications, 2021 Publication	<1%
7	futuretechtrending.com Internet Source	<1%

8	Abdulla Eqab, Tamer Shanableh. "Android mobile app for real-time bilateral Arabic sign language translation using leap motion controller", 2017 International Conference on Electrical and Computing Technologies and Applications (ICECTA), 2017 Publication	<1 %
9	Submitted to Liberty University Student Paper	<1 %
10	"Soft Computing and Signal Processing", Springer Science and Business Media LLC, 2022 Publication	<1 %
11	Tzu-Yang Chen, Pai-Wen Ting, Min-Yu Wu, Li-Chen Fu. "Learning a deep network with spherical part model for 3D hand pose estimation", Pattern Recognition, 2018 Publication	<1 %
12	Submitted to Liverpool John Moores University Student Paper	<1 %
13	T J Siong, N R M Nasir, F H M Salleh. "A mobile learning application for Malaysian sign language education", Journal of Physics: Conference Series, 2021 Publication	<1 %
14	Submitted to Tower Hamlets College Student Paper	<1 %

15	"Intelligent Systems Design and Applications", Springer Science and Business Media LLC, 2018 Publication	<1 %
16	Shamsul Anuar Mokhtar, Siti Mashitah Shamsul Anuar. "Learning application for Malaysian sign language", Proceedings of the 9th International Conference on Ubiquitous Information Management and Communication - IMCOM '15, 2015 Publication	<1 %
17	wvvw.easychair.org Internet Source	<1 %
18	"Inventive Systems and Control", Springer Science and Business Media LLC, 2022 Publication	<1 %
19	Mohammad Khalid Imam Rahmani, Fahmina Taranum, Reshma Nikhat, Md. Rashid Farooqi, Mohammed Arshad Khan. "Automatic Real-Time Medical Mask Detection Using Deep Learning to Fight COVID-19", Computer Systems Science and Engineering, 2022 Publication	<1 %
20	Sumaya Siddique, Shafinul Islam, Emon Emtiyaz Neon, Tajnoor Sabbir, Intisar Tahmid Naheen, Riasat Khan. "Deep Learning-based Bangla Sign Language Detection with an Edge	<1 %

	Device", Intelligent Systems with Applications, 2023 Publication	
21	computerresearch.org Internet Source	<1 %
22	Submitted to Academy of Information Technology Student Paper	<1 %
23	"HCI International 2020 – Late Breaking Papers: Universal Access and Inclusive Design", Springer Science and Business Media LLC, 2020 Publication	<1 %
24	Submitted to Chester College of Higher Education Student Paper	<1 %
25	Submitted to ESCP-EAP Student Paper	<1 %
26	I.A. Adeyanju, O.O. Bello, M.A. Adegboye. "Machine learning methods for sign language recognition: A critical review and analysis", Intelligent Systems with Applications, 2021 Publication	<1 %
27	Submitted to Kennesaw State University Student Paper	<1 %
28	Sharvani Srivastava, Amisha Gangwar, Richa Mishra, Sudhakar Singh. "Chapter 48 Sign	<1 %

	Language Recognition System Using TensorFlow Object Detection API", Springer Science and Business Media LLC, 2022 Publication	
29	Submitted to University of Glasgow Student Paper	<1 %
30	Submitted to Tarumanagara University Student Paper	<1 %
31	Submitted to University of Essex Student Paper	<1 %
32	www.icphs2019.org Internet Source	<1 %
33	link.springer.com Internet Source	<1 %
34	dev.to Internet Source	<1 %
35	www.irojournals.com Internet Source	<1 %
36	Linjie He, Yumin Chen, Keshou Wu. "Fuzzy granular deep convolutional network with residual structures", Knowledge-Based Systems, 2022 Publication	<1 %
37	Meghana Pai Kane, Sherwin Fernandes, Ricky Fonseca, Shika Desai, Akhil Shetye, Ananya Sharma. "Sign Language Apprehension using	<1 %

Convolution Neural Networks", 2022 13th International Conference on Computing Communication and Networking Technologies (ICCCNT), 2022
Publication

38	ijarcce.com Internet Source	<1%
39	www.ijert.org Internet Source	<1%
40	www.ijraset.com Internet Source	<1%
41	"Emergent Converging Technologies and Biomedical Systems", Springer Science and Business Media LLC, 2022 Publication	<1%
42	D Tresnawati, R Setiawan, N Alawiah, D Heryanto, S Rahayu. "The development of sign language learning media using multimedia systems", IOP Conference Series: Materials Science and Engineering, 2021 Publication	<1%
43	Progya Paromita Urmee, Md. Abdullah Al Mashud, Jasmin Akter, Abu Shafin Mohammad Mahdee Jameel, Salekul Islam. "Real-time Bangla Sign Language Detection using Xception Model with Augmented Dataset", 2019 IEEE International WIE	<1%

Conference on Electrical and Computer Engineering (WIECON-ECE), 2019
Publication

44	Sherly Noel. "Human computer interaction(HCI) based Smart Voice Email (Vmail) Application - Assistant for Visually Impaired Users (VIU)", 2020 Third International Conference on Smart Systems and Inventive Technology (ICSSIT), 2020 Publication	<1 %
45	arxiv.org Internet Source	<1 %
46	deepai.org Internet Source	<1 %
47	dspace.bracu.ac.bd:8080 Internet Source	<1 %
48	prr.hec.gov.pk Internet Source	<1 %
49	www.sciencegate.app Internet Source	<1 %
50	www.slideshare.net Internet Source	<1 %
51	Ankita Wadhawan, Parteek Kumar. "Deep learning-based sign language recognition system for static signs", Neural Computing and Applications, 2020	<1 %

Publication

52	Benfano Soewito, Alexander Khrisna Satyadhana, Raymond, Sarah Maria. "Communicaton On Mobile Phone For The Deaf Using Image Recognition", 2020 International Conference on Information Management and Technology (ICIMTech), 2020 Publication	<1 %
53	S.A.M.A.S Senanayaka, R.A.D.B.S Perera, W. Rankothge, S.S. Usgalhewa, H.D Hettihewa, P.K.W. Abeygunawardhana. "Continuous American Sign Language Recognition Using Computer Vision And Deep Learning Technologies", 2022 IEEE Region 10 Symposium (TENSYMP), 2022 Publication	<1 %
54	"Intelligent Computing & Optimization", Springer Science and Business Media LLC, 2022 Publication	<1 %

Exclude quotes	On	Exclude matches	Off
Exclude bibliography	On		

Report Plag Check

PAGE 1

PAGE 2

PAGE 3

PAGE 4

PAGE 5

PAGE 6

PAGE 7

PAGE 8

PAGE 9

PAGE 10

PAGE 11

PAGE 12

PAGE 13

PAGE 14

PAGE 15

PAGE 16

PAGE 17

PAGE 18

PAGE 19

PAGE 20

PAGE 21

PAGE 22

PAGE 23

PAGE 24

PAGE 25

PAGE 26

PAGE 27

PAGE 28

PAGE 29

PAGE 30

PAGE 31

PAGE 32

PAGE 33

PAGE 34

PAGE 35

PAGE 36

PAGE 37

PAGE 38

PAGE 39

PAGE 40

PAGE 41