Project Report - ECE 285

Rijul Sherathia
Data Science
A59020686

Neha Mittal
Data Science
A59020456

Abstract

The project focuses on developing a deep learning-based system for sign language detection, with a specific emphasis on American Sign Language (ASL). By segmenting the hand region using skin masking and employing Canny edge detection, we aim to accurately identify ASL gestures in images. Training our models using a variety of algorithms, including Convolutional Neural Networks (CNN), will enable us to recognize and interpret different hand shapes and movements. The ultimate goal is to create a real-time system that can facilitate communication between non-sign language speakers and individuals who rely on ASL, enhancing their ability to understand and interact effectively. This project has the potential to significantly improve communication and quality of life for deaf and mute individuals by providing a means for non-sign language speakers to comprehend and engage with them.

1 Introduction

Our project focuses on developing a system for sign language detection using deep learning techniques, specifically targeting American Sign Language (ASL). ASL is a complete and complex visual language used by many deaf and hard-of-hearing individuals in the United States and Canada. Unlike spoken languages, ASL relies on hand shapes, facial expressions, body movements, and other visual cues to convey meaning. However, since sign language is not universally understood, our goal is to create a system that can facilitate communication for non-sign language speakers.

We will begin by segmenting the hand region using skin masking, which will allow us to filter out unwanted portions of the video sequence. This will be followed by Canny edge detection, which will help us to further refine the hand region and identify the gesture being made.

To train our model, we plan to use a variety of algorithms including SVM, K-NN, neural networks, and CNN. However, our primary focus will be on model training using CNN, as it has shown great strength in image classification tasks due to its ability to learn spatial features directly from images.

We observe that our CNN model achieves the highest testing accuracy of 97%.

2 Related Work

ASL recognition has been a well-studied problem in computer vision for the past two decades. Researchers have explored various classifier categories, including linear classifiers, neural networks, and Bayesian networks [2-11]. Linear classifiers, while relatively simple, require sophisticated feature extraction and preprocessing techniques to achieve success [2, 3, 4]. Singha and Das achieved a 96% accuracy for recognizing gestures of one hand across ten classes by using Karhunen-Loeve Transforms in a forward neural network [9]. Their approach involved extensive image preprocessing steps such as image size normalization, background subtraction, contrast adjustment, and image segmentation. Admasu and Raimond utilized Gabor Filters and Principal Component Analysis for feature extraction.

One notable work in this field is L. Pigou et al.'s application of Convolutional Neural Networks (CNNs) to classify 20 Italian gestures from the ChaLearn 2014 Looking at People gesture spotting competition [11]. By using a Microsoft Kinect to capture full-body images of individuals performing the gestures, they achieved a cross-validation accuracy of 91.7%. The use of depth features captured by the Kinect proved to be beneficial in accurately classifying ASL signs. These studies and others demonstrate the ongoing efforts to address the ASL recognition problem, employing diverse methodologies and technologies to improve accuracy and performance.

3 Method

We began by segmenting the hand region using skin masking, which allows us to filter out unwanted portions of the images. This was followed by Canny edge detection, which helped us to further refine the hand region and identify the gestures being made.

Skin masking is a technique used to extract the hand region from the images by creating a binary mask that identifies pixels that belong to the skin color. This technique helps to remove unwanted portions of the image and extract only the relevant hand regions.

Canny edge detection is an image processing technique used to detect edges in the image. This technique uses a multi-stage algorithm to detect a wide range of edges in images. By applying canny edge detection after skin masking, we can further refine the hand region and extract the edges of the hand gesture.

To train our model, we used a variety of algorithms including SVM, K-NN, neural networks, and CNN. However, our primary focus was on model training using CNN, as it has shown great strength in image classification tasks due to its ability to learn spatial features directly from images.

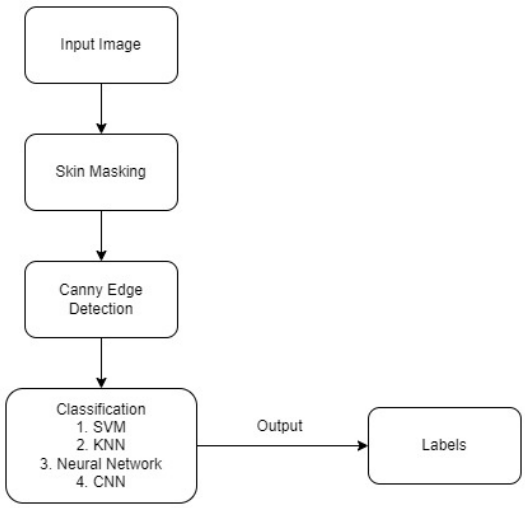


Figure 1: Workflow Diagram

CNN Architecture:

The given CNN architecture consists of multiple layers that are sequentially applied to process the input data. Here is a stepwise explanation of the architecture:

1. **Input Layer:** The input data is specified to have a shape of [None, 89, 100, 1], where None represents the batch size, 89 and 100 are the height and width of the input image, and 1 indicates a single channel (grayscale image).

2. **Convolutional Layers:** The architecture begins with a convolutional layer with 32 filters and a kernel size of 2x2. ReLU activation function is applied to introduce non-linearity. Subsequently, a max pooling layer with a pool size of 2x2 is employed to downsample the feature maps.

3. **Repeat Convolutional and Pooling Layers**: The above convolutional and max pooling operations are repeated five more times with varying filter numbers (64, 128, 256, 256, 128) and the same kernel and pool size. Each convolutional layer is followed by a ReLU activation and a max pooling layer.

4. **Fully Connected Layers**: After the last max pooling layer, the feature maps are flattened, and a fully connected layer with 1000 units and ReLU activation is applied. Dropout regularization with a rate of 0.75 is then employed to prevent overfitting. Finally, another fully connected layer with 24 units and a softmax activation function is used for multi-class classification.

The architecture aims to extract relevant features from the input images through the convolutional layers and downsampling operations. The fully connected layers perform high-level feature aggregation and map the learned features to the output classes. The ReLU activation introduces non-linearity, and dropout helps in regularization to enhance generalization capabilities. The softmax activation in the last layer provides probability scores for each class, allowing the model to predict the ASL alphabet represented in the input image.

Other Algorithms:

1. K-Nearest Neighbour
2. Gaussian Naive Bayes
3. Support Vector Classifier
4. XGBoost
5. Random Forest Classifier

4 Experiments

The dataset used in this project is a custom dataset created specifically for training a model to recognize American Sign Language (ASL) alphabets. It consists of approximately 11,500 images representing different ASL alphabet gestures.

The dataset was generated by capturing images of both a male and a female person's hands. Images were collected for each hand, resulting in a total of four perspectives: male left hand, male right hand, female left hand, and female right hand. Each image in the dataset represents a specific ASL alphabet gesture, capturing the hand shape and position of that alphabet. The dataset covers all 24 alphabets of the English language in ASL.

The images are stored in JPEG format, a widely used and compressed image format that maintains good image quality while reducing file size. The JPEG format allows for efficient storage and processing of the large number of images in the dataset.

During the experiments, the dataset was divided into training and testing sets. The training set was used to train the ASL alphabet recognition model. The testing set served as an independent evaluation set to assess the final model's accuracy and generalization ability.

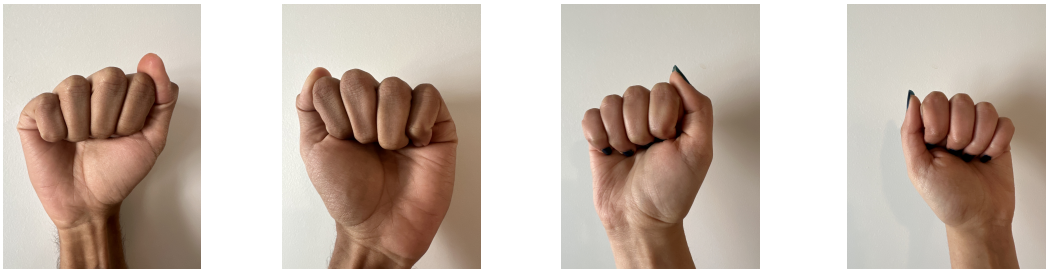


Figure 2: ASL Alphabet A

Preprocessing techniques such as resizing, normalization, and augmentation have been applied to the images to enhance the model's performance and generalization. These techniques help ensure consistent input dimensions, reduce the impact of lighting and background variations, and increase the diversity of the dataset. Before training the model, we performed two preprocessing steps on the dataset: skin masking and canny edge detection. Skin masking helped isolate the hand regions in the

images, while canny edge detection enhanced the edges of the hand gestures, providing additional features for recognition.

Figure 3: ASL Alphabet A - Preprocessed

We conducted experiments using various machine learning models, including Support Vector Machines (SVM), K-Nearest Neighbors (K-NN), Neural Networks, and Convolutional Neural Networks (CNN). These experiments aimed to assess the performance of each model and compare their accuracy in recognizing hand gestures from a live video sequence. Our primary focus was on training the model using CNN and comparing its performance with the other algorithms.

We divide our experiments into 2 parts, the first part involves training and testing using several machine learning models and the second part involves using Convolutional Neural Networks as our model.

We ran the experiments several times on the following models performing hyperparameter tuning for some models to reduce overfitting on training data:
1. K Nearest Neighbour - (k = 3)
2. Gaussian Naive Bayes
3. Support Vector Classifier
4. XGBoost - (n_estimators = 50, max_depth = 7, min_child_weight = 3)
5. Random Forest Classifier - (n_estimators = 150, min_samples_split = 40, min_samples_leaf = 15, max_depth = 15)

5 Results

The following table lists the training and testing accuracy of all models:

Model	Training Accuracy	Testing Accuracy
KNN	90.98%	79.47%
Gaussian Naive Bayes	46.3%	40.4%
SVM	99.5%	91.37%
XGBoost	100%	83.79%
Random Forest	94.03%	78.72%
CNN	99.34%	97.12%

Table 1: Results Table

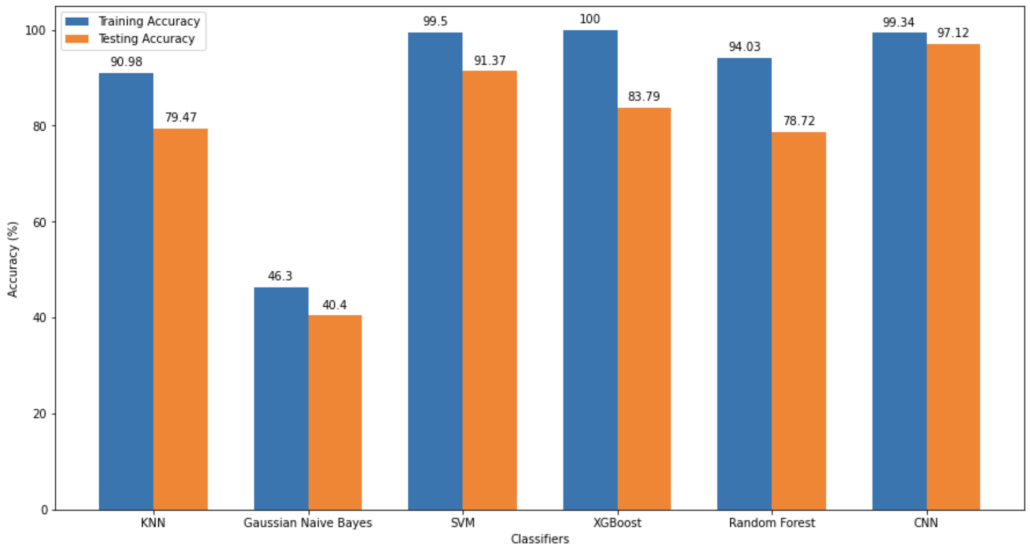
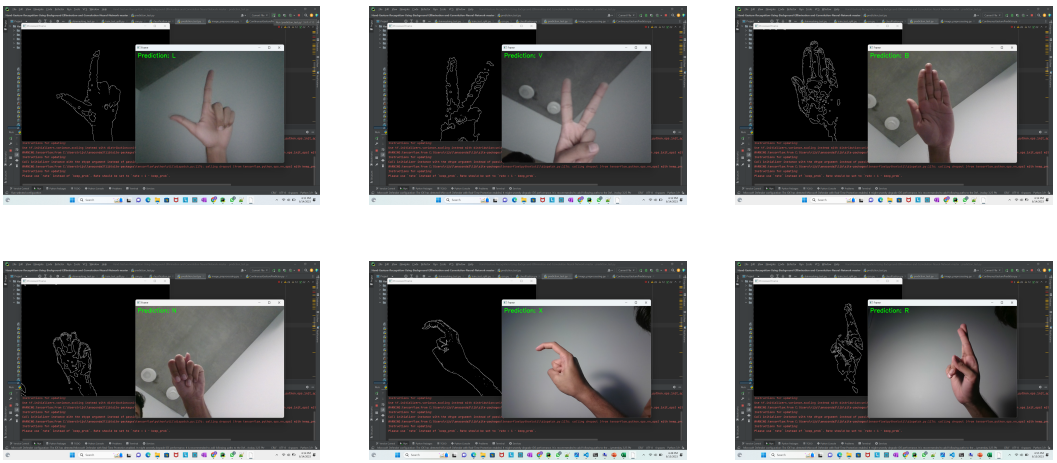


Figure 4: Model Training Accuracy vs. Testing Accuracy

The reported results are obtained by computing the average performance across multiple iterations of the model. By conducting multiple runs, we ensure a comprehensive assessment of the model's performance and establish a reliable measure of its effectiveness.

Among these models, the CNN demonstrates the highest performance, achieving an impressive training accuracy of 99.34% and a remarkable testing accuracy of 97.12%. These results indicate that the CNN model effectively captures the complex patterns and relationships present in the data, enabling it to accurately classify and recognize the ASL sign language gestures. The CNN's ability to learn and extract meaningful features from the image data, coupled with its deep hierarchical architecture, contributes to its superior performance compared to the other models evaluated. These findings highlight the suitability of CNNs for the task of ASL gesture recognition and support their potential for real-world applications in facilitating communication and interaction with individuals who rely on sign language.

Real Time Prediction



References

1. https://gogulilango.com/software/hand-gesture-recognition-p1

2. Singha, J. and Das, K. “Hand Gesture Recognition Based on Karhunen-Loeve Transform”, Mobile and Embedded Technology International Conference (MECON), January 17-18, 2013, India. 365-371.
3. D. Aryanie, Y. Heryadi. American Sign Language-Based Finger-spelling Recognition using k-Nearest Neighbors Classifier. 3rd International Conference on Information and Communication Technology (2015) 533-536
4. R. Sharma et al. Recognition of Single Handed Sign Language Gestures using Contour Tracing descriptor. Proceedings of the World Congress on Engineering 2013 Vol. II, WCE 2013, July 3 - 5, 2013, London, U.K.
5. K. Bantupalli and Y. Xie, "American Sign Language Recognition using Deep Learning and Computer Vision," 2018 IEEE International Conference on Big Data (Big Data), Seattle, WA, USA, 2018, pp. 4896-4899, doi: 10.1109/BigData.2018.8622141.
6. Garcia, B., and Viesca, S.A. Real-time American Sign Language Recognition with Convolutional Neural Networks
7. Nandy, A., Prasad, J. S., Mondal, S., Chakraborty, P., and Nandi, G. C. (2010). Recognition of isolated indian sign language gesture in real time. Information Processing and Management, 102-107
8. Neha Titarmare, Chandu Vaidya, Rohit Meshram, Abhay Dongre, Prathmesh Jawale, Nihal Bambale, Ojas Awachaat, "Hand Sign Language Detection - Using Deep Neural Network", 2023 IEEE International Students’ Conference on Electrical, Electronics and Computer Science (SCEECS), pp.1-4, 2023.
9. Y.F. Admasu, and K. Raimond, Ethiopian Sign Language Recognition Using Artificial Neural Network. 10th International Conference on Intelligent Systems Design and Applications, 2010. 995-1000.
10. Meghana Pai Kane, Sherwin Fernandes, Ricky Fonseca, Shika Desai, Akhil Shetye, Ananya Sharma, "Sign Language Apprehension using Convolution Neural Networks", 2022 13th International Conference on Computing Communication and Networking Technologies (ICCCNT), pp.1-7, 2022.
11. L. Pigou et al. Sign Language Recognition Using Convolutional Neural Networks. European Conference on Computer Vision 6-12 September 2014
12. Arya Sanil, Hella Santhosh Lal, Rohit Krishnan, Syam M, Seena R, Aseena A, "Smart American Sign Language Recognition For Deaf", 2022 International Conference on Innovations in Science and Technology for Sustainable Development (ICISTSD), pp.209-214, 2022.
13. Divyansh Mahida, Divyansh Jain, Hansal Shah, Jainil Patel, Rajeev Kumar Gupta, Ashutosh Sharma, "Hand Gesture to Character Recognition using Convolutional Neural Network", 2022 Seventh International Conference on Parallel, Distributed and Grid Computing (PDGC), pp.346-351, 2022.
14. Vedant Puranik, Varun Gawande, Jash Gujarathi, Ayushi Patani, Tushar Rane, "Video-based Sign Language Recognition using Recurrent Neural Networks", 2022 2nd Asian Conference on Innovation in Technology (ASIANCON), pp.1-6, 2022.
15. 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), pp.1-6, 2022.