A Deep Learning–Based Framework for Correcting Erroneous Character-Level Bengali Sign Images

by

Sameen Islam
21201662
Mohammed Ayman
21201263
S.M. Tawsif Islam
24141182

A thesis submitted to the Department of Computer Science and Engineering
in partial fulfillment of the requirements for the degree of
B.Sc. in Computer Science and Engineering

Department of Computer Science and Engineering
Brac University
October 2025

Declaration

It is hereby declared that

1. The thesis submitted is our own original work while completing degree at Brac University.
2. The thesis does not contain material previously published or written by a third party, except where this is appropriately cited through full and accurate referencing.
3. The thesis does not contain material which has been accepted, or submitted, for any other degree or diploma at a university or other institution.
4. We have acknowledged all main sources of help.

Student's Full Name & Signature:

Sameen Islam

21201662

Mohammed Ayman

21201263

S.M. Tawsif Islam

24141182

Approval

The thesis/project titled " A Deep Learning–Based Framework for Correcting Erroneous Character-Level Bengali Sign Images " submitted by

1. Sameen Islam (21201662)
2. Mohammed Ayman (21201263)
3. S.M. Tawsif Islam (24141182)

of [Semester], [Year] has been accepted as satisfactory in partial fulfillment of the requirement for the degree of B.Sc. in Computer Science in [Current Semester] Year.

Examining Committee:

Supervisor:
(Member)

Md. Tanzim Reza

Senior Lecturer
Department of Computer Science and Engineering
Brac University

Thesis Coordinator:
(Member)

Dr. Md. Golam Rabiul Alam

Professor
Department of Computer Science and Engineering
Brac University

Head of Department:
(Chair)

Dr. Sadia Hamid Kazi

Associate Professor & Chairperson
Department of Computer Science and Engineering
Brac University

Ethics Statement

Our research confirms the claim made, and all cited and referenced papers and sources are accurate. The work has never been submitted for a degree to any other college or academic organization. The four co-authors acknowledge and accept any violations of the thesis rule. In addition, we would like to take this chance to thank everyone who has supported us through this process. We finished our thesis without committing to any illegal techniques. Our work complies with the ethical standards of Brac University

Abstract

People who cannot hear or speak rely on sign language as the primary source of communication. Sign language is generally expressed at a fast pace, therefore errors and miscommunication may happen frequently. Our thesis introduces a deep learning–based framework for correcting erroneous character-level Bengali sign language images. The main focus of our research is correctly mapping incorrect bengali character level sign gestures to their closest semantically accurate signs. A dataset consisting of 36 classes for both correct and potential incorrect hand signs were generated for bengali characters. The proposed framework utilizes convolutional neural networks(CNNs) along with triplet loss to extract discriminative embeddings. These embeddings are later represented on a vector space where metric distance from incorrect images are used to map them to the correct gestures. Our primary goal is to develop a system where the intended meaning is correctly delivered even in case of improper hand gestures. Overall, 9 models were used to assess the proficiency of the idea for correcting erroneous bengali sign characters. ResNet50 delivered the most excellent results with an accuracy of 97.6% on a 200 epoch experiment. Further ablation studies also resulted in other models performing well. VGG16 had an accuracy of 94.4% and EfficientNetB0 delivered an accuracy score of 96.4%. The research concluded that longer epochs provided significant increases in accuracy for both black and normal backgrounds. Both KNN and centroid based distance were used as similarity based mapping approaches in order to compare the accuracy of the models under various changes in hyper-parameters. The centroid performed better throughout all the experiments, Thus, the overall results highlight the successful mapping of incorrect sign gestures to their appropriate correct classes.

Keywords: Triplet Loss, Convolutional Neural Networks, Metric Distance

Dedication

This thesis is dedicated to our families who have supported us throughout this trip through their unconditional support and constant encouragement. Our profoundest thanks also go to our esteemed supervisor, Md. Tanzim Reza, whose advice, patience and insightful commentaries on this study have been utter invaluable. Lastly, this work is dedicated to each other as teammates and friends due to the common effort, the sleepless nights and the perseverance that helped to achieve the given accomplishment.

Acknowledgement

We would like to thank the Almighty, in the first place, for providing us this opportunity. Secondly, we would like to thank our thesis supervisor, Md. Tanzim Reza for his intermittent support, coaching, and counseling as we navigated through our work. Thirdly, we also would like to take this occasion to show our thanks to all members of the faculty in their aid and support when we were at Brac University. And lastly, we would like to acknowledge our beloved parents with their unwavering support, encouragement and prayers.

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

Sign language continues to be a vital form of communication for people who have difficulty hearing or speaking. Using hand gestures and body movements , these people express their ideas, emotions and thoughts. Just like any spoken language, sign language has very strict grammatical structures that may vary according to the culture and geography. For the Bengali speaking community, character level sign language acts as a base for communication. But due to rapid hand gestures, muscle fatigue and limited execution time, proper delivery of the intended message is frequently not exercised. Thus, all these can add upto reduced confidence and barriers to social and professional environments.
In order to combat these challenges, our research introduces a translation model that can correct Bengali character level sign language. Our object is to map incorrect sign gestures to their most semnatically accurate ones and thus provide better communication between the deaf and Non-Verbal community. The system uses convolutional neural networks (CNNs), along with metric learning and triplet loss to learn discriminative representations of sign images. Traditional classifications like logistic regression, often just detect sign gestures and classify them. Meanwhile, our approach is to construct a vector space by the extracted embeddings from our images where similar signs will clutter closer together and dissimilar images will position further apart.
Our proposed framework utilizes multiple pre-trained CNN backbones like ConvNeXt-Tiny, DenseNet121, ResNet50, EfficientNetB0, MobileNetV2, VGG16, Xception, CustomCNN, ConvNet in order to extract features from the images. The classification head from all the backbone is removed in order to generate a 128 dimensional embedding. These embeddings will later be optimized through triplet-loss training. Before training begins, the images are resized to 224x224 pixels and are preprocessed according to individual backbone specific requirements. Pixelated, blurred or inaccurate images are rejected from the dataset in order for our dataset to be accurate and maintain integrity. Augmentation techniques such as flipping, gray scaling and other methods are used to properly balance the dataset for both the correct and incorrect samples.
After training, the embeddings of correct signs are used to generate centroids. The centroids highlight the mean position of each image specific to their class. The evaluation process involves computing distance from incorrect test images to the centroids and analyzed by means of distance metrics and k-NN. This helps to quantify similarity and dissimilarity between the incorrect images and the correct image centroids. Utilizing dual methods of evaluation helps to analyze the effectiveness of models and cement how effectively each gesture can map incorrect gestures to their correct ones in the vector space.

1.1 Motivation

Sign language remains to be an important medium for bridging the communication gap between the hearing and speech impaired community. Although there exists sufficient work on Bengali Sign Language, most of the focus is biased towards more widely studied languages like American Sign Language (ASL). Existing research on Bengali Sign Language is predominantly focused on error detection and classification rather than correction. The limited attention for the Bengali Sign Language community encouraged us to deliver our own contribution towards helping these unfortunate individuals. Our research aims to help these people by developing a translational model for Bengali character-level sign language error correction. Our work will hopefully lay a foundation for further development and attention towards helping the speech and hearing impaired community.

1.2 Problem Statement

People with hearing or speech difficulty need Bengali Sign Language to communicate. This is particularly essential at character level, where motion is often conveyed at fast pace. The smallest deviations in hand gestures can result in misinterpreted communications. There has been growing research on sign language recognition, yet majority of the work has been done on classifying sign language rather than correcting them.
Currently, there are no methods explicitly developed for rectifying errors in Bengali character-level sign language. Significant advancements have occurred in the study of American Sign Language (ASL); yet, research on Bengali Character Level Sign Language remains relatively underappreciated. The majority of currently available models are limited to performing classification tasks. They lack the capability to convert erroneous motions into their accurate visual representations. This discrepancy complicates the development of improved communication technologies enabling Bengali deaf and Non-Verbal individuals to interact with one another.

This study helps to fulfil the gap by creating a system which is trained to correct character level errors of Bengali sign language. The model takes in an incorrect sign image and maps it to the proper correct class. Our aim is to improve accuracy and provide a system that can help people correct their sign gestures.

1.3 Objective

The main research objectives of this study are as follows:

1. **Design and Implementation Phase:** To develop a framework capable of correcting erroneous Bengali character-level sign languages into their correct versions. A Convolutional Neural Network (CNN)-based embedding model will be utilized to ensure high-quality translation and accurate feature representation.

2. **Enhancing Communication Precision:** To deliver an accurate translation system for Bengali character-level sign language. The focus will be on processing images containing postural or hand orientation errors and mapping them effectively to their correct sign gestures.

3. **Measurement of System Effectiveness:** To evaluate the proposed system using class-wise and overall accuracy. Additional performance metrics such as *F1-score*, *precision*, *recall*, and the *confusion matrix* will be employed to comprehensively assess model performance.

4. **Ablation Study:** To conduct hyperparameter tuning in order to identify the best-performing model configuration. The resulting variations in accuracy will be analyzed to derive valuable insights and comparative performance outcomes.

1.4 Methodology in Brief

Our proposed methodology involves a deep learning approach for correction of Bengali character level sign language. Pre trained CNN backbones are utilized to generate embeddings in a vector space for the purpose of correction of Bangla error sign language. Our methodology involves a number of steps:

- **Data Collection & Preprocessing:** The dataset was generated for correct and incorrect character level Bengali character level gestures, It consisted of accurate representations of characters and also deliberate potential error sign images. The dataset was further split into a ratio of 70-15-15 for training testing and validation. Furthermore, before loading them into the embedding models, pre-processing such as grayscaling, resizing and augmentations were executed.

- **Architecture:** Pre-trained CNN backbones along with a custom CNN without any pre-trained weights were used as the feature extractors. The models utilized the triplet loss function whose responsibility was trying to reduce the metric distance between the anchors and the positives.

- **Training & Analysis:** Training had been conducted for multiple models, varying hyper parameters. Instead of traditional classifiers, Centroid Distance and KNN were encompassed with metric distance in order to correctly map incorrect gestures . Performance was assessed through multiple evaluation metrics such as classwise accuracy, F1 score, etc. T-SNE plots, heatmaps and other visualizations were used to demonstrate the model performance.

In conclusion, the system starts with data pre-processing and moves up to embedding extraction, model refinement and classwise accuracy for the proper correction of Bengali character level sign language correction.

1.5 Scopes and Challenges

Scope

The system enables broad application in assisting improvement of correct sign language delivery through error detection. Educational institutions, NGOs and other edtechs can utilize this research and develop tools for helping deliver accurate sign language gestures. In the communication sector, it can be used as an assistive agent for properly delivering correct sign gestures in environments such as online meetings and interviews, where

accuracy and expeditions are a concern. Additionally, the research can be further developed to include broader categories of sign language such as words and regional specific gestures.
Apart from communication, the system can be utilized as a rehabilitating tool for people that are suffering from motor impairments which can affect the gesture accuracy. The scalability and light weight models proposed in this paper also makes it suitable for mobile devices that focus on real-time environments.

Challenges

Despite all the potential scopes, there exists several challenges that hindered our research and development of our system. The Bengali Sign language in itself is neglected compared to other sign languages like ASL. Moreover, in order to correct sign gestures, potential erroneous images of gestures are also required to train the models. Although there exists a number of papers about detection, very few exist in the realm of sign language correction. Due to the lack of dataset, future works branching off this research will need to develop their own versions of incorrect sign dataset. Additionally, real-time correction of sign language continues to be a challenge as real-time detection would often involve inconsistent lighting conditions or non-uniform backgrounds. Thus, deployment on mobile devices not only depends on the models, but also the device specifications and the environment. Finally, properly delivering the accurate meaning or idea goes beyond individual sign representation. Grammars and contextual nuances provide a complex issue relating to errors that traditional hand gestures cannot properly reflect.

1.6 Key Learnings and Insights

This research demonstrated that triplet-loss-based deep metric learning can effectively extract discriminative embeddings from Bengali sign gestures and accurately map incorrect gestures to their proper representations. By leveraging pretrained CNN backbones, optimized embeddings, and robust preprocessing strategies, the system achieved high accuracy and interpretability.
The dual dataset approach—using both black and natural backgrounds—provided valuable insights into how contextual variations influence embedding separability. Both datasets performed exceptionally well, confirming that pretrained architectures capture key visual features critical to hand gesture recognition.
Overall, this framework establishes a foundational direction for future research on embedding-based gesture correction and extends the boundaries of conventional sign language recognition. It holds significant potential for developing assistive technologies that reduce communication barriers and enhance inclusivity for the speech and hearing-impaired communities.

Chapter 2

Literature Review

2.1 Preliminaries

Sign language is an essential medium of communication for individuals with hearing or speech impairments, enabling them to express complex linguistic concepts through manual and non-manual gestures. In the context of Bengali Sign Language (BdSL), character-level recognition presents particular challenges due to high intra-class variability in hand shape, finger placement, and orientation.
Most existing works in BdSL focus on gesture detection or classification, aiming to identify the sign being performed. However, these approaches rarely address error correction, which are crucial for improving fluency and intelligibility in real-world communication.
Recent advances in deep learning, especially convolutional neural networks (CNNs) and transformer-based architectures, have significantly improved the extraction of spatial and contextual features from sign images. Furthermore, embedding-based methods trained with metric learning objectives such as triplet loss or contrastive loss have demonstrated strong capability in learning discriminative feature spaces—where correct signs form compact clusters and incorrect variants are distinctly separated. This property makes them particularly suitable for automatic error correction, where an erroneous sign can be mapped to its nearest valid representation in the learned embedding space.

2.2 Review of Existing Research

A work on an article named "Bengali-Sign: A Machine Learning-Based Bengali Sign Language Interpretation for Deaf and Non-Verbal People" (2024) [24] was published in Sensors. This introduced a model that translated BdSL gestures to vocal expressions and text. The purpose of this research was to pinpoint the shortcomings in sign language assistive technology. The research idea used a comprehensive dataset of BdSL. Using this dataset, they trained a deep learning model and ensured natural language processing features and computer vision architecture. By changing various conditions, durability and performance was tested. The system delivered very good accuracy under multiple test conditions. As most of the research focuses on popular sign languages such as ASL, there exists a gap in regional and unique sign languages. The system utilizes a diverse and optimal dataset but it still faces issues in real world conditions where there are non-uniform lighting conditions and noise. This research has built a good platform for BdSL interpretation. Additional research may need to be conducted for the results to be

generalized. Moreover, improvements in technology can further help these marginalized communities.

The research on "Techniques for Detecting the Start and End Points of Sign Language Utterances to Enhance Recognition Performance in Mobile Environments" investigates the problem of authenticating utterance boundaries of sign language for mobile recognition systems. Moreover, there exists challenges when using sign language recognition systems in mobile environments due to poor computation resources and variable frame rates. For mobile devices, the author utilized critical start and end points for the performance of sign language recognition. The system has a hard time understanding signs accurately when the frame rate is low. This causes a decrease in performance. On the other hand, the proposed methods precisely highlight the boundaries of hand signs. This ensures that the recognition model only receives the meaningful frames. With the help of proper identification, computer resources are optimized efficiently. So, it allows for a better and more efficient sign language recognition system for real-time applications. According to the authors, the methods resulted in promising performance in mobile devices as they frequently have limited processing power (MDPI, 2024) [23]. Due to the limited performance, it can lead to imprecise gesture recognition. Thus, the proposition of the authors resulted in systems implementing start and end point actions within the recognition stage. This crucial step leads to a better-optimized recognition process that enhances accuracy for mobile-based sign language identification capabilities.

Another study called E^{2}GAN: Efficient Training of Efficient GANs for Image-to-Image Translation (2023) focuses on the computation challenges (Yifan, 2023) [22]. It talks about the computational difficulty in training GANs for image-to-image translation, specifically for mobile devices. The performance of E^{2}GAN surpasses random initialization performance levels without adding any considerable training duration expenditures. The E^{2}GAN framework leverages data distillation through diffusion algorithms and LoRA model adaptation to minimize both storage requirements and operational computing costs during the pretrained model fine-tuning duration. Complex data distillation processes transform universal pre-trained architectures into specialized GANs while achieving more efficient parameter-sensitive model training procedures. E^{2}GAN exhibits superior performance in visualization and system efficiency based on benchmark tests of Cityscapes and Facades compared to standard GAN algorithms. This system achieves maximum operational and visual preservation while minimizing resource usage to run efficiently on limited mobile environments. Real-time hybrid architecture deployment systems must address performance problems that emerge from demanding resource restrictions in platforms. The framework requires multiple image-to-image translation experiments to prove generalized scalability once performance changes on distinct datasets pass validation tests. E^{2}GAN promotes future practical GAN applications in development work. It does this by enhancing development of essential system components and GAN optimization capabilities of E^{2}GAN.

Grammatical Error Correction (GEC) is an important Natural Language Processing (NLP) challenge for grammar and fluency improvement in text. Rule-based early work leveraged hand-engineered grammar but could not scale (Sakaguchi, 2018) [11]. With advances in statistical approaches in the 1990s and 2000s, learner corpus training with probabilities helped improve error correction; however, with no uniform datasets and

evaluation metrics in use, development stalled. The creation of shared tasks such as Helping Our Own (HOO) and CoNLL (2013, 2014) helped enable systematic benchmarking of systems in the community. Early evaluations focused on grammaticality—namely, closed-class errors such as determiners, prepositions, and verb forms. Despite that, studies subsequently acknowledged that fluency—naturalness in native speech—was no less important. Character-level GEC leveraged psycholinguistic observations, such as the Cambridge University effect, in creating strong correctors for spelling. Token-level techniques integrate with error-repair operations and dependency parsing for concurrent fixing and parsing of a text. Whole-sentence correctors have emerged with neural architectures, such as with reinforcement learning for optimizing objectives for fluency. Evaluation metrics, even with development, have continued to pose a challenge. Traditional token-level metrics cannot detect improvements in fluency, and new benchmark sets such as the JFLEG corpus have become a necessity. The field is shifting towards overall rewriting of sentences rather than individual grammar fixes, opening doors for GEC models with fluency awareness. The future lies in combining contextual embeddings, semi-supervised training, and complex reinforcement techniques for adaptability in fixing texts.

The study Readiness of Novel Sign Language (BdSL) Words using Deep Convolutional Neural Networks (DCNNs) investigates the development of automatic word recognition tools for speech and hearing-impaired people. Through advanced deep learning techniques, the authors use DenseNet201 and ResNet50-V2 modeling architectures to classify ten mainstream BdSL words, which exhibit high training accuracy rates of 99% and testing accuracy of 93%. A curated picture set containing 992 images of ten BdSL words was designed specifically to strengthen model training effectiveness and validation accuracy [14]. The most definitive part of this research comes from focusing on BdSL as it is one of the least appreciated sign languages. This would ensure critical accessibility opportunities for improvement. Using DCNNs, the system allows accurate and scalable recognition of sign language for people who have hearing and speech impairments. In order to optimize the model, hand signs were varied instead of environment which allowed for a more robust application for real world users. Further work on the sign language recognition system was enhanced due to the influence of the authors' BdSL work. The relatively small size of the available data poses an obstacle, and the authors highlight how the inclusion of machine learning applied to this context brings new levels of achievement to the creation of a more integrated environment for communication access by minority groups.

Lin (2023) [17] conducted a comparative study of two important frameworks—Pix2Pix and CycleGAN—on image-to-image translation, a very important task in computer vision aimed at translating images among different domains. In Pix2Pix, cGAN is utilized to effect changes in pictures with paired datasets. This means that the input photographs and the generated outputs remain highly correlated. CycleGAN processes unpaired datasets through a cyclic consistency approach that enables its operation. The cycle loss algorithm enables the preservation of original text outputs while style or domain properties change in the output. The models use PatchGAN as their discriminator component. The models operate optimally because they concentrate on discrete visual sections rather than processing complete images. The U-Net generator in Pix2Pix creates exact structured changes, yet CycleGAN generates structured changes through ResNet-based generators employing skip connections for gradient stability. This study finds Pix2Pix shows superior performance for transformations demanding exact output-input relational correspon-

dences. Unpaired data enables the training of CycleGAN, so this model matches diverse applications that include domain adaptation and style transfer tasks. The different approaches of these models present distinct implementation barriers. For example, Pix2Pix requires high-quality datasets, and CycleGAN does not have encoding that may limit the model's ability to extract domain-specific features. The complementary nature of these frameworks has been indicated in this study, opening the door for further development of approaches on image-to-image translation.

A revolutionary open dataset was created by Jim et al. [15], called "KU-BdSL." The authors created this open dataset to promote further research on BdSL recognition. The BdSL dataset was developed to support the speech and hearing-impaired community. Approximately 2.4 million people depend on this dataset. The dataset was produced using a static single-hand technique, and it consists of 1,500 high-resolution pictures. Overall, 38 Bengali consonants in 30 different hand motions exist within this collection. Fifty images in each category ensure that any machine learning work performed with the dataset remains balanced. To make the photographs long-lasting, the shooting was done under different lighting conditions to increase longevity. The images were labeled into DarkNet-compatible formats for easy integration into computer vision models. The dataset was created to assist in developing deep learning algorithms that could correctly identify BdSL with high reliability. This is a critical task, considering the complexities of the Bengali script, which contains more consonants compared to many other alphabets in different languages. The authors created three extra variants of the dataset: (1) the raw Multi-scale Sign Language Dataset (MSLD), (2) the Uni-scale Sign Language Dataset (USLD) with standardized dimensions, and (3) the Annotated Multi-scale Sign Language Dataset (AMSLD), which includes bounding boxes for gestures. Despite the many advantages of the dataset, it is currently limited to consonants. Vowels and sentence-level indicators are not included; however, the authors plan to improve these aspects in future studies. Therefore, the dataset has important applications in machine translation, human-computer interaction, and assistive technologies, with great potential to foster inclusivity and accessibility among the deaf and Non-Verbal community.

A continuous word-level sign language recognition system was introduced by the Sreemathy et al. research team (2023) [19]. Their objective was to lessen the communication gap between hearing-impaired individuals and members of mainstream society. The authors utilized the YOLOv4 architecture along with MediaPipe to detect hand gestures and classify them using a support vector machine (SVM). A dataset of 676 images was generated using 80 signs from Indian Sign Language, augmented with possible manipulations such as rotations, brightness changes, etc., for training purposes. The integration of YOLOv4—including its backbone, neck, and head components—with SVM resulted in a classification accuracy of 98.8% for gestures related to preventive care of physical infirmities, while SVM achieved 98.62% accuracy after analyzing hand keypoint data using MediaPipe. The study also devised a decision algorithm that integrated predictions from both models based on their prediction accuracy scores. This method enabled a favorable combination of real-time performance execution and high accuracy, with the expert system's finalization yielding results superior to earlier designs using CNNs, YOLOv3, and LSTMs.

An improvement in performance on the paradigms for Pix2Pix image-to-image translation

was introduced by Zhao et al. (2023) [21] through the use of the U-Net++ architecture. The improved model addresses two major weaknesses of older models: it eradicates data loss that typically occurs during encoding-decoding processes and resolves constraints related to model generation vulnerabilities. The Reading.U-Net++ implementation (Runtime 2023) boosts performance with its expanded skip connections to create superior outputs, while its differential image discriminator extends model capabilities by analyzing target-image–input pairings. Tests conducted on the Facades and CUHK Sketch Portrait datasets revealed substantial improvements in key metrics such as Inception Score (IS), Fréchet Inception Distance (FID), and Structural Similarity Index (SSIM). In comparison to CycleGAN and Pix2Pix models, the system generates better images with greater clarity and depth. However, certain metrics showed only limited progress, indicating that research obstacles still remain in developing an effective combination of enhanced models. This study demonstrates that the cumulative effect of all proposed modifications leads to improved quality in various image translation tasks (Zhao et al., 2023) [21].

Stamoulis et al. (2022) [13] designed a novel approach toward the real-time recognition of Greek Sign Language (GSL) to reduce communication barriers among deaf individuals. The authors' approach was based on the use of the Google MediaPipe library for extracting landmarks and key points, combined with the LSTM neural network for accurate interpretation of motion sequences. Laboratory scale databases of GSL words andl exicons were built in order to facilitate the building of emergency alert systems which demand such algorithms. The combination of contemporary machine learning platforms and, specifically, TensorFlow, allowed the development of a competent model that can observe dynamic and stable scenarios effectively. The correct identification of sequential gestures was possible with LSTM layers, which is good at processing time-related relationships. But the study was limited because of the limited data, which prevents the generalisation of the framework to other sign languages. The overlapping movements also had an impact on the performance of the framework by bringing delays and audio artifacts of video quality and hardware dependencies. Although the study demonstrated the potential of deep learning methods in solving social inequality, it did not fail to experience performance-related limitations.

Natarajan and colleagues (2022) [12] created a real-time sign language recognition system based on their end-to-end hybrid deep neural architecture (H-DNA). The system is based on the CNN and Bi-LSTM, Neural Machine Translation (NMT) and Dynamic GAN, to solve the major problems of sign language processing, including noise and motion variability. The CNN+Bi-LSTM model allows gesture recognition that is reliable since it allows various gestural inputs to be processed to form correct gesture patterns. The NMT translation module translates the spoken sentences into sign language representation. The production of gestures is processed through Dynamic GAN and MediaPipe and results in improved precision of output creation. The system recorded finer test results, gesture recognition accuracy of more than 95 percent and BLEU score of 38.56. Other evaluation measurements, such as SSIM (0.921) and FID2Vid (3.46), demonstrated that the test video quality was better in comparison with video quality when using various language datasets. The researchers presented the methods of overcoming communication barriers with hearing- and speech-impaired people through the H-DNA framework that involved a series of low-cost and scalable solutions (Natarajan et al., 2022).

The fundamental problem of the real-time American Sign Language (ASL) gesture recognition is revealed in one of the papers named Spelling Correction Real-Time American Sign Language Alphabet Translation System Based on YOLO Network and LSTM (Rivera-Acosta et al., 2021) [10]. The researchers suggested a new approach to combine the YOLO network to identify the handshape with a spelling correction system that is driven by Bidirectional Long Short-Term Memory (BiLSTM) networks. The datasets were 24,000 images of ASL signs, 24 different letters and 5,200 other images of 26 fingerspelled letters. It was analyzed that the system had an average correct identification of gestures of 81.74 percent. It has also been tested to be e!ective in dynamic environments with a high rate of 61.35 frames per second when performing calculations with real-time data of 10 test subjects. The BiLSTM network was trained using a dictionary of 370 words and this model achieved a training accuracy of 98.07 which helped in enhancing the robustness of the system. The integrated spelling correction system, through which the predicted sequence of letters were narrowed down to, resulted in an increased accuracy in translation. This work presents a thorough and methodical approach to implementing real-time ASL translation while addressing key limitations, such as weak performance on unfamiliar inputs and spelling errors. While the initial results are promising, further evaluation is needed to assess the model's scalability to larger vocabulary sets and more diverse datasets. Comprehensive technical assessments—especially against alternative ASL systems—are necessary to evaluate usability features and real-world deployment readiness with lasting effectiveness.

The article "Sign Language Recognition and Translation: A Multidisciplinary Approach From the Field of Artificial Intelligence" gives a broad review related to the development of artificial intelligence in sign language recognition technologies while underlining the interdisciplinary nature of the field. The article explores sign language recognition technology development through a chronological review that includes the original "Ralph" robotics system which was created for finger-spelling recognition (Parton, 2005) [1]. Throughout its evolution, the field achieved enhanced solutions incorporating virtual reality (VR) technologies. The CyberGlove system stands as a prime example of advanced sign language recognition by detecting VR sensor-based isolated and continuous hand movements. This research examines camera-based systems through an analysis of the CopyCat interactive system that enhances sign language detection by using visual information in natural dynamic environments. The scientific advancement process requires multiple professional fields to work together. Thus robotic systems are integrated with virtual reality headsets and computer vision systems to create advanced recognition capabilities. All these technological progressions work together to develop gesture recognition within the sign language as well as to go beyond the constraints of the conventional pre recorded motions and equipment specifically designed to ensure signers. The study deals with the existing issues of construction of large datasets and constraints of real-time processing, yet it shows that there are many significant AI applications in enhancing accessibility to Deaf and Hard of Hearing communities. The paper is relevant in a sense that it aids to comprehend the interrelation of the different technological fields to address certain issues in the interpretation of sign language.

Stoll et al. (2020) presented Sign Language Production (SLP) system, which was based on Neural Machine Translation (NMT) and Generative Adversarial Network (GAN) in their paper [9], to generate sign language poses of a verbal sentence in a written form.

In addition to that, this design has minimum input and standard labels, which are very monotonous to label. The system consists of two phases: First, the text is translated to gloss with the help of the NMT, followed by Motion Graph (MG) which processes frame sequences to generate photo-realistic sign language videos with the help of a pose-conditioned GAN; it achieves the best model found on PHOENIX14T with a BLEU-4 score of 16.34. The findings indicate that facial keypoints along with pose positions and hand configurations are better parameters to use when producing HD continuous signs. The effectiveness of the system might help them get reasonably good sign language video production but will also be put in a situation where character representation is constrained and reliance on low-resolution datasets. However, it may be an effective method of video recording in sign language (Stoll et al., 2020) [9].

Cherian and Sullivan (2019) introduce a novel method of semantic consistency in unpaired image-to-image translation, called Sem-GAN [8]. By doing so the framework removes CycleGAN-like framework constraints in ambiguous situations that do not violate semantic accuracy. Sem-GAN framework combines semantic segmentation with translation, so as to preserve original semantic values of the source domain, as well as be able to tailor each translation to the style of the target domain. It does so by having a segmentation loss function and a new semantic dropout method which randomly removes some semantic classes of input images in order to improve the resilience of the classification. It had demonstrated impressive results on Cityscapes and VIPER by providing a single architecture with adversarial, cycle-consistency, and segmentation losses. The findings mentioned that the mean Intersection over Union (mIoU) increased by 20 percent compared to CycleGAN as well as significant improvement in high-level tasks like semantic segmentation. The Sem-GAN-adapted data was used to segmentation models that were able to increase mIoU by 6 percent thus strengthening the bright future of Sem-GAN. It will have practical application values, such as in autonomous driving. Sem-GAN enhances further unpaired image-to-image translation and it maintains semantic consistency and more preserved content.

Grundkiewicz (2017) [7] considers automated grammatical error correction, focusing on the errors committed by ESL learners. The paper treats GEC as a machine translation problem; it converts bad English into grammatically correct English by phrase-based SMT. Probably the most interesting outcome of this work is the development of the WikEd Error Corpus, which is the largest GEC dataset available publicly that is derived from Wikipedia document edit histories. This corpus overcomes the data sparsity problem, as it provides real examples of errors, thereby being very useful for training GEC models. Furthermore, this thesis conducts the first large-scale human evaluation of GEC systems to examine the correlation between automated metrics and human judgments and, in turn, establishes the M2 metric as a valid benchmark. The SMT framework takes advantage of state-of-the-art optimization techniques combined with task-specific dense and sparse features that reach state-of-the-art performance in the CoNLL-2014 benchmark. In addition, Grundkiewicz introduces improvements to generative SMT systems by adding a discriminative component. It does that using two novel approaches, including the use of a discriminative classifier as a feature function, and by combining sparse features generated out of error profiles. These contribute considerably to system performance, reflecting their strengths in bringing both generative and discriminative methods together. Yet, there are still some deficiencies: the training used in WikEd Error Corpus

may not entail every type of error that generally characterizes ESL contexts. SMT-based systems are inherently limited to their phrase-based architecture, which prevents the ability to keep track of long-range relationships and, therefore, limits their contextual understanding. These systems also rely heavily on the quality and quantity of parallel training data, which may be hard to get for all error types. Regulating specific error corrections proves difficult through untargeted training initiatives because text-based GEC systems neglect the requirements for real-time and spoken language error management. The work by Grundkiewicz represents a strong foundation for GEC by demonstrating how SMT approaches with discriminative methods perform effectively regardless of their limitations. Neural machine translation (NMT) and long-range dependency improvements together with real-time grammatical error correction (GEC) application expansion to spoken language become essential directions for upcoming research studies (Grundkiewicz, 2017).

Rahaman et al. (2014) [5] introduced real-time computer vision-based recognition of Bengali Sign Language (BdSL) through a system which identified 6 Bengali vowels and 30 consonants. The processing system accomplishes vital achievements by implementing step-by-step operations beginning with visual hand detection, then advancing to segmentation until reaching precise identification. The detection system uses Haar-like feature-based cascaded classifiers which identify open and closed hand positions within image frames. Skin color analysis in the HSV color model, together with Gaussian smoothing, dilation, and erosion, produces noise reduction to achieve hand shape isolation. Binary image processing converts hand images to reveal geometric features, including positions of fingers and hand shapes. A K-Nearest Neighbors (KNN) algorithm operates for classification activities but splits its work into distinctive models for vowel and consonant detection. Testing against both familiar and unfamiliar participants using 3600 images (36 signs $\times$ 10 participants $\times$ 10 repetitions) enabled an accuracy rate of 98.17% for vowels and 94.75% for consonants as the system found its terms during training. The system demonstrates difficulties with distinguishing similar sign appearances ("ro" and "loh") and, in certain cases, with light changes and objects that share similar color patterns with skin. The system provides better recognition capabilities than previous BdSL models which adopted Principal Component Analysis (PCA) together with neural networks methods. Edge detection techniques are recommended by the authors as enhancements to boost the system's resistance, according to Rahaman et al. (2014).

The paper Ensemble Classifiers for Biomedical Data: Elshazly et al. (2013) [4] evaluated ensemble learning methods to boost biomedical dataset classification precision in Performance Evaluation by Elshazly et al. (2013). The authors focus on two prominent ensemble classifiers: The authors tested two ensemble learning techniques known as Random Forest (RF) and Rotation Forest (ROT) on five different medical datasets. Three diverse feature selection techniques were used to acquire the vital features from each dataset for quality data enhancement. Rotation Forest produced the best accuracy performance in most of the considered situations. It is presented in this study that the ensemble classifiers are crucial resources in case of biomedical data since they are the ones that indirectly enable the enhancement of predictive capabilities that are vital in medical diagnosis. Ensemble analysis with machine learning analysis of medical data is an ensemble method because the presence of several models in combination makes it possible to properly analyze intricate patterns in patients that consequently result in a more

accurate prediction. The new physical diagnosis tools include applying ensemble classifiers, especially that of ROT on the results of the output and the systematic selection of features is a key factor at every level of classification. The leading features of expertise are useful in the enhancement of ensemble classifiers, optimal solutions and it enhances efficiency in the running operations. The studies aimed at improving the potential of healthcare machine learning have found that data entry is significant to the effectiveness of prediction models. The findings demonstrate that the ensemble methodologies applied to the task with the help of the appropriate feature selection methods provide higher classification rates and positive advances in the development of the medical diagnostic devices.

Error detection and correction, as one of the features presented by Tolba et al. (2022)[3] proposed a system which enhanced the recognition of the Arabic Sign Language (ArSL). The existing sign language recognition systems fail in noisy environments, so this model overcomes this and maintains constant recognition rates. This approach is better than the traditional techniques as it enables detection of semantic as well as lexical errors beyond simple lexical processes. In the case of real time video stream, the recognition engine processes the high-definition video which monitors the newly identified moving gestures and then reconstruction of the sentence is done. A correction algorithm helps in solving most common mistakes in classification as well as overlapping hand gestures. Post-processing module executes the semantic oriented statistical algorithms concurrently in order to augment the quality of the final product. Precision and system dependability (to overcome noisiness in the real world) are obtained using error correction at an iterative step compared to their current machine learning equivalents. Studies have demonstrated that the integration of such knowledge of the domain in sign language recognition systems results in the improvement of performance of the system. The findings of this research establish principle bases in the development of software that automatically deciphers real time sign language irrespective of natural variations.

John Sie Yuen Lee's thesis titled Automatic Correction of Grammatical Errors in Non-Native English Text deals with grammar problems found specifically in essay writing by non-native speakers and employs sophisticated techniques to correct these issues. The research analyzes linguistic errors related to article, preposition, and verb misuse, identifies the source of such errors, and offers an explanation in a particular native context. Statistical models are combined with linguistic rules based on syntax to enhance the error detection and feature extraction capabilities. With dual detection, a model can achieve very powerful detection of subtle errors and highly accurate correction of the errors (Lee, 2009) [2]. According to the description, the system applies a decision-making algorithm that extracts the strategies from the evaluation of the input text quality and the user's domain error pattern base. The system makes general corrective changes tailored to the particular language competency of each student. The results of the personalization approach demonstrate better performance in correction. This research analyzes the smart ways of calculation which achieve an accuracy level suitable for actual application. The thesis showcases the large-scale applicability of optimization approaches by utilizing different large-scale data sets. The system has advanced grammatical detection capabilities which excel at identifying.

Temkar et al. (2023) [20] proposed an Indian Sign Language (ISL) recognition framework

that is lightweight and real-time, using MobileNetV2 and transfer learning to address the lack of scalable solutions for hearing-impaired people in India. The model is particularly designed to be used on mobile and edge devices, where there are limited computational resources. By freezing the base layers of MobileNetV2 and incorporating custom ReLU and Softmax activation dense layers, the model was optimized to high accuracy with minimal processing overhead. It was trained on a publicly available ISL dataset with gestures for alphabets, numbers, and common signs and with rich data augmentation (rotation, zoom, shift, flip) for high robustness. The model was evaluated using categorical cross-entropy loss and accuracy metrics to reach 97% accuracy at a loss of just 0.09, and real-time capability at 60ms/frame latency. The authors concluded that their architecture presents a real-world assistive solution and referred to its platform flexibility, with potential in facilitating facial expression, YOLO-based gesture tracking, and educational deployment in inclusive environments.

In their work, Lata et al. (2022) [16] developed an end-to-end Thai Sign Language (TSL) recognition system aimed at translating sentences and words from video input using deep learning. The system employed YOLOv5 for the detection of human and region-of-interest isolation and then three state-of-the-art CNN architectures – VGG16, ResNet50, and DenseNet121 – for the extraction of features. To facilitate model generalization, data was augmented by 100-fold expansion of video frame samples with each sample spanning 32 frames. The features were then passed through 1D convolutional layers as well as through LSTM networks, and the final sentence prediction was decoded using the Connectionist Temporal Classification (CTC) algorithm. Experimental trials conducted in TSLS dataset, which comprises 500 synthesized videos from 23 words placed in various grammatical positions, have shown that DenseNet121 with 4 Conv1D and BiLSTM (128 size) gave the best Word Error Rate (WER) of 0.4286. Other trials conducted with optical flow and GRU-based RNNs yielded poorer performance compared to LSTM-based models. The research shows the effectiveness of using CNN, Conv1D, LSTM, and CTC to create an efficient, real-time TSL recognition system that would be utilized to translate challenging gestures as well as sentence-level translation.

Verma et al. (2024) [25] in their work proposed a real-time American Sign Language (ASL) recognition system integrating MediaPipe and Convolutional Neural Networks (CNN) to precisely detect and interpret hand gestures from webcam feed. The system uses MediaPipe for accurate 3D hand landmark detection and CNN for classification with an unexpected 99.12% accuracy on a 29-class ASL dataset. The authors used a large set of ASL alphabets with more than 87,000 samples and undertook data pre-processing, such as normalization, landmark reshaping, and data augmentation to improve the generalizability of its models. The CNN model framework consists of three convolution layers, pooling layers, full connected layer and dropout to prevent overfitting. The effectiveness of the system was authenticated by real-time testing of the system by OpenCV, which identified the live hand gestures

In conclusion, different works have beenn addressed related to the issue of sign language recognition and correction. Some of the major highlights have been the performance of deep learning based systems like CNNS and generative adversarial networks (GANS) for accurate gesture recognition. Some of the studies has been very informational for addressing Bengali Sign Language.

2.3 Summary of Key Findings

Through our literature review, several research gaps and limitations in existing sign language recognition and translation systems have been identified. First, most prior studies focus on isolated gestures or limited alphabets, such as ASL or ISL, and rarely address sentence-level translation or regional languages like Bengali Sign Language (BdSL). While models like CNNs, Bi-LSTMs, and hybrid architectures (e.g., H-DNA, Rivera-Acosta et al., 2021; Natarajan et al., 2022) achieve high accuracy on curated datasets, they often struggle in real-world environments due to background noise, overlapping gestures, and varying lighting conditions.

Image-to-image translation approaches, such as Pix2Pix, CycleGAN, and U-Net++, have demonstrated potential in enhancing gesture clarity and generating synthetic data. However, unpaired or semantically inconsistent translations remain a challenge, limiting their applicability for dynamic sign recognition tasks. Similarly, error correction modules and spelling correction systems have been explored in ASL, but scalable, real-time implementations for BdSL remain largely unexplored.

Another notable gap is the lack of lightweight, mobile-friendly models. While MobileNetV2 and YOLO-based frameworks enable low-latency inference, most high-accuracy models rely on computationally expensive architectures, limiting real-world deployment. Furthermore, semantic retention and sequential gesture understanding critical for accurate sentence translation are often neglected in existing pipelines.

Chapter 3

Requirements, Impacts and Constraints

3.1 Specifications and Requirements

3.1.1 Hardware Requirements

To support the training and evaluation of the proposed image-to-image Bengali Sign Language (BdSL) correction model, the experiments were conducted on a high-performance local workstation configured as follows:

- **Operating System:** Windows 11 (64-bit) and Ubuntu 24.04 LTS (64-bit)
- **Processor:** Intel Core i7-12700 with 24 threads
- **Memory:** 32 GB RAM
- **Motherboard:** ASUS ROG Strix
- **GPU:** NVIDIA RTX 4070 Ti with 12 GB GDDR6 memory
- **GPU Driver:** Version 531.18

GPU acceleration was essential for processing high-resolution images (224×224 pixels) and training multiple CNN backbones efficiently. During peak training, GPU utilization reached nearly 100%, enabling full parallelism and reducing training time without compromising model convergence.

3.1.2 Software Requirements

All experiments were implemented in Python 3.11, utilizing the following libraries:

- **TensorFlow 2.14** and **Keras** for deep learning model construction, training, and evaluation
- **scikit-learn** for preprocessing, evaluation metrics, and embedding analysis
- **CUDA Toolkit 12.2** and **cuDNN 8.9** for GPU acceleration

Additional preprocessing steps—including data augmentation, normalization, and image resizing—were applied to all images to improve model robustness. Experiments used batch sizes of 16–32 and embedding sizes of 128–256, optimized to balance memory usage and training speed. Triplet-loss-based training ensured effective metric learning across correct and incorrect gesture classes.

3.2 Social Impact

Several potential benefits exist for BdSL correction systems, some of which includes:
Accessibility: Capable of delivering accurate representation of erroneous gestures which will facilitate in helping people with hearing or speech impairments.
Education: Will motivate unfortunate individuals who have difficulty in hearing or speaking to learn Bangla sign language properly.
Empowerment: It will uplift confidence and independence for users by proper feedback and error correction.

3.3 Environmental Impact

This research mainly is dependent on computation resources but environmental concerns may include:
Consumption of mass amount of electricity for running High-power GPU.
Optimized batch sizes and minimal high powered computational time was used to reduce redundant computations.
Deployment of cloud-based systems and future development can reduce environmental footprint.

3.4 Ethical Issues

Ethical considerations were incorporated throughout the study:
All images in the BdSL dataset were collected with informed consent, ensuring participants' privacy and data protection.
The model is intended solely for educational and accessibility purposes, avoiding misuse in surveillance or unauthorized monitoring.
Bias mitigation strategies, including data augmentation and balanced class sampling, were applied to reduce model bias across gesture variations.

3.5 Standards

Where applicable, the study followed the following standards:
Use of **IEEE-compliant coding practices** for reproducibility and modularity.
Adoption of standard **image Pre-Processing conventions** compatible with ImageNet-pretrained CNN backbones.
Alignment with best practices in **deep learning evaluation metrics**, including accuracy, F1-score, and confusion matrices.

3.6 Risk Management

Key risks and mitigation strategies included:
Hardware Failure: Ensured frequent backups and checkpoints during GPU-intensive training.
Data Loss or Corruption: Maintained multiple dataset copies and version control.
Model Overfitting: Applied regularization and dropout.
Computational Bottlenecks: Optimized batch sizes, embedding dimensions, and utilized GPU acceleration efficiently.

3.7 Economic Analysis

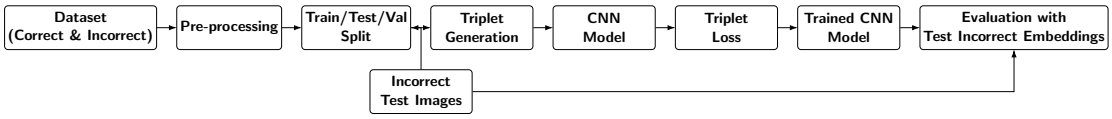
The economic considerations of this study include:
High-performance GPU hardware required significant initial investment, justified by the speedup and efficiency gains during training.
Use of open-source software (TensorFlow, Keras, scikit-learn) reduced software licensing costs.

Chapter 4

Proposed Methodology

4.1 Methodology Overview

Our system on Deep Learning Based Framework for Correcting Erroneous Character-Level Bengali Sign Images learns to separate the embeddings for Bengali character gestures. A triplet network architecture along with pre-trained CNN backbones are used to train the model in properly mapping erroneous sign gestures to their proper classes. Our workflow follows a step-by-step procedure which includes:
Data Preprocessing and Organization: The input images, both correct and incorrect, are resized and pre-processed using grayscaling, CLAHE and augmentation. Then it is split into a 70-15-15 train-test-valid split in order for the model to have enough images for training and testing.
Embedding Model Construction: The pre-trained CNN backbones included ResNet50, MobileNetV2, EfficientNetB0, Xception, VGG16, ConvNeXtTiny, DenseNet121 and ConvNet. Additionally, to compute overall performance of the pre-trained backbones, a custom CNN model of 7 layers was also used. These backbones were used to extract embeddings from the images. Moreover, using these embeddings and triplet loss, the model was trained to separate classes and bring corresponding embeddings of incorrect images to their correct class in a vector space.
Metric-Based Evaluation: Learned embeddings are analyzed using k-nearest neighbors (k-NN) and centroid-based distance metrics to assess intra-class similarity and inter-class separability.



The design emphasizes a metric learning paradigm rather than a conventional classification pipeline. Instead of predicting class labels directly, the model learns to project gesture images into an embedding space where distances reflect semantic similarity. This allows incorrect gestures to be mapped to their correct counterparts using geometric relationships in the embedding space.

4.2 Preliminary Design

This study adopts a quantitative experimental methodology, supported by comparative analysis across multiple CNN architectures. Each model backbone is trained under con-

trolled conditions with identical preprocessing, enabling a systematic ablation study to identify the most effective feature extractor for BdSL gesture representation.

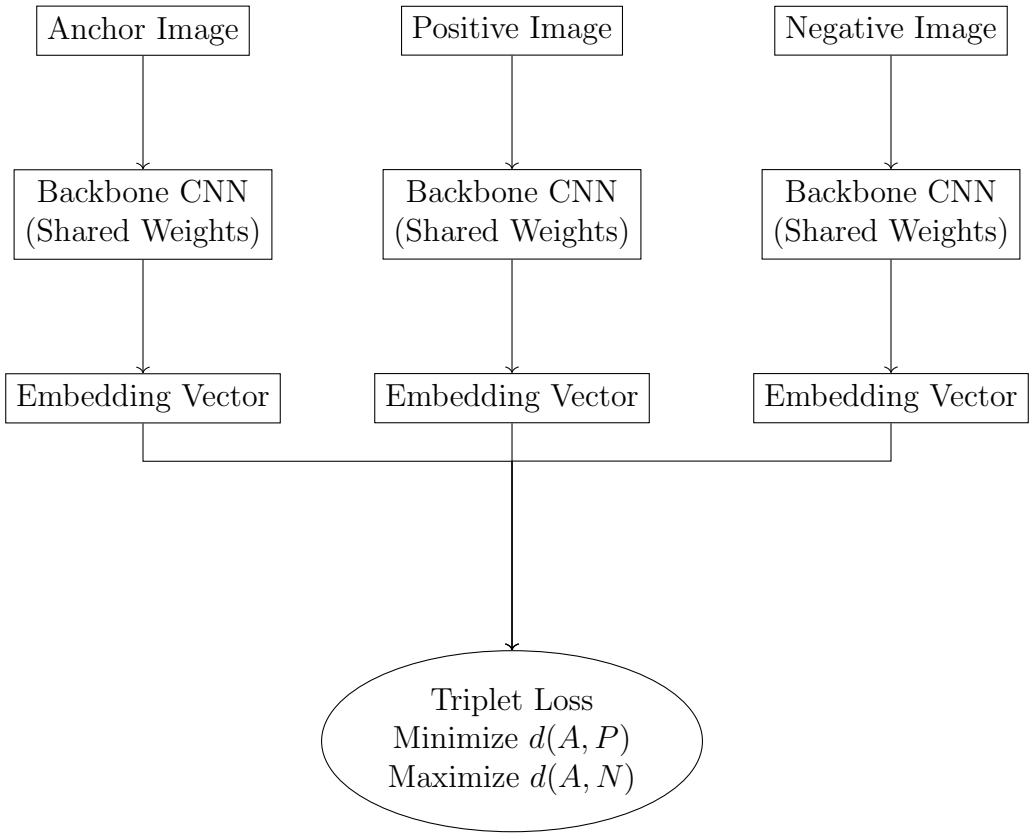


Figure 4.1: Triplet Network Architecture: Anchor, Positive, Negative inputs and Embedding Output

The triplet-loss framework is central to the learning process. Each training triplet consists of an anchor (incorrect gesture), a positive sample (correct gesture of the same class), and a negative sample (correct gesture from a different class). By minimizing intra-class distances and maximizing inter-class distances, the model learns a robust embedding space that generalizes to unseen gestures.

4.3 Data Collection

The data used for this study is branched into two parts: a correct set and an incorrect set of Bengali character level sign language gestures. Each set contained a total of 36 classes which highlighted hand signs corresponding to Bengali characters. All the images were taken under a consistent background with sufficient lighting so that the focus would on accurate hand gestures. This was done in order to reduce external factors and to enable to the model to only focus on hand signs.
The images were taken in RGB format which was crucial inorder to distinguish subtle variations in hand shape. The whole dataset maintained a consistent resolution across all the images in order to enable a uniform pre-processing function and feature extraction on the entire dataset. The controlled data setup assured that it contained gestures of uniform visual quality but diverse in sign representations.

The dataset had a total of 36 classes for each of the correct and incorrect sets. The 36 classes would represent the 36 Bengali characters used in this research. Each of the characters were then manually mapped to a specific class label in order to produce a detailed comparison of classwise accuracy.

0 অ	1 আ	2 ই	3 উ	4 ঊ	5 ঋ	6 ক
7 খ	8 গ	9 ঘ	10 চ	11 ছ	12 জ	13 ঝ
14 ট	15 ঠ	16 ড	17 ঢ	18 ত	19 থ	20 দ
21 ধ	22 প	23 ফ	24 ব	25 ভ	26 ম	27 য
28 র	29 ল	30 ন	31 স	32 হ	33 ড়	34 ং
35 ঃ						

Figure 4.2: Bengali Characters with Class Labels

In order to execute our research we had to rely on Bengali Character Level Sign Language Dataset. Unlike popular dataset like ASL which had a broad works done on it, accurate Bangla Sign Language dataset was difficult to acquire. Moreover, negligible work has been done on correcting bangla single language, so there existed no dataset on deliberate incorrect hand gestures of correct Bangla sign characters. So, in order to utilize metric learning, we had to generate deliberate incorrect versions of existing characterwise sign gestures. Additionally, to keep our dataset consistent, we also created our own correct version of characterwise Bengali sign gestures. The correct images had accurate representation of the bengali character sign gestures. Meanwhile, incorrect classes contained deliberate incorrect images such as handshape errors, finger errors, positioning errors, etc.
The Bengali characters were labeled according to the number 0-35. Majority of the pictures were taken of the correct classes as they would later be used to generate centroids. The distribution of the correct classes are as follows.

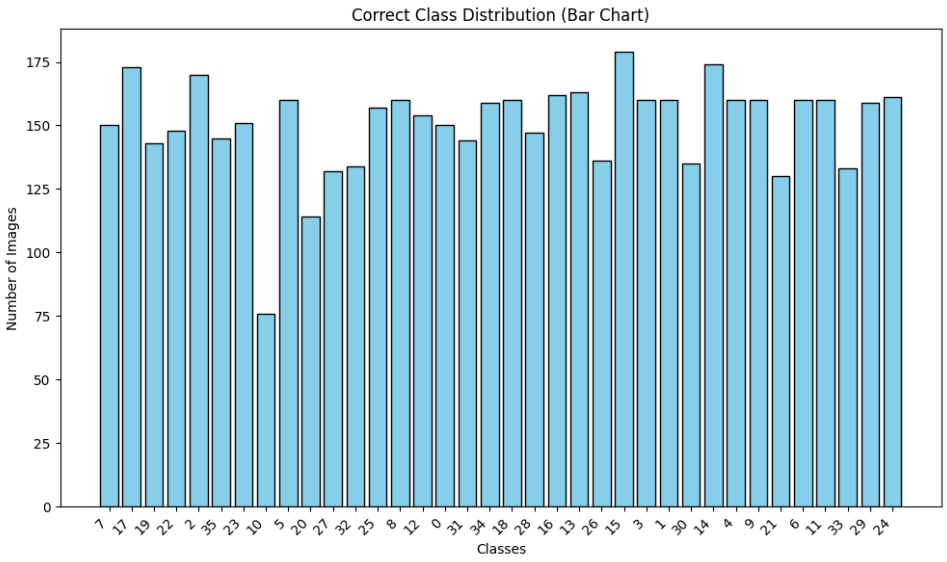


Figure 4.3: Correct Class Distribution (Bar Chart)

In contrast, the number of images belonging to the incorrect character set was comparatively moderate. However, a range of variations and intentional inaccuracies were included to enhance the model's robustness and ability to correct hard inaccuracies more effectively.

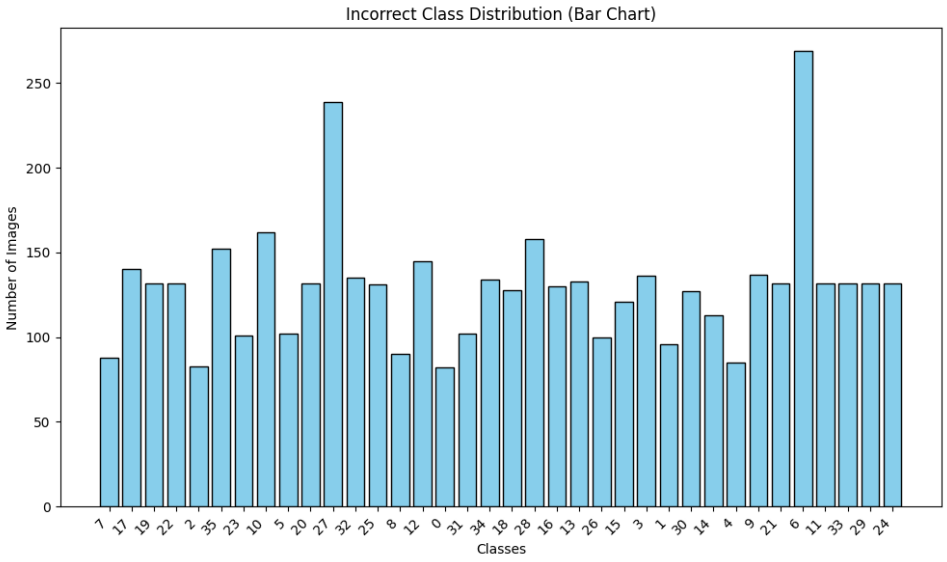
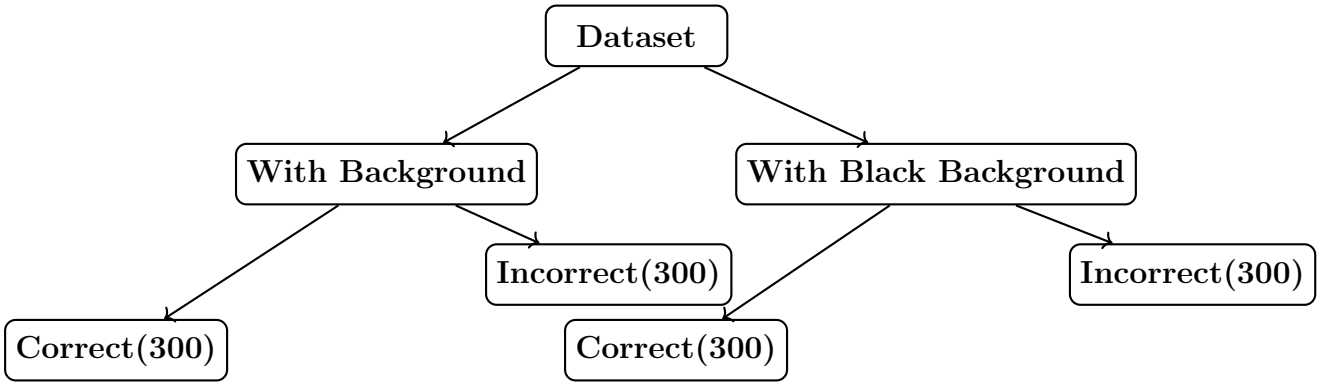


Figure 4.4: Incorrect Class Distribution (Bar Chart)

4.3.1 Data Cleaning

The data cleaning process focused on converting the image in such a way that the models could more accurately discriminate and correct the incorrect images. The images were skimmed through to remove any images that displayed excess pixelations, blurriness or gestures where certain parts of the hands were out of frame. Additionally, images that had completely overlapped with signs from other classes were removed in order to maintain the integrity of the dataset.

There also existed classwise inaccuracies between the number of pictures existing in both the correct and incorrect sets. In order to dissipate these classwise non-uniformity, which could introduce bias during training, augmentation of both sets were implemented so that all the neighboring classes of each set had the same number of images. Furthermore, images with poor resolutions and unwanted background artifacts were also rejected from the dataset.
To make our research more robust and for greater comparison, a separate dataset with completely black background was generated. This not only forced the model to focus on hand signs but added an extra feature for greater model analyzation and comparison.
So, cleaning all the data produced a more coherent and refined dataset that focused solely on high quality gestures which resulted in better training.



4.3.2 Data Transformation

In order for data to be consistent and have proper quality before model training, a comprehensive pre-processing and transformation pipeline was executed. All the images were resized to a dimension of 224x224 pixels. This provided consistent input for all models and were compatible with CNN architectures. To reduce any effect of color during model training, the images were converted to grayscale which were then again reverted back to a three channel format.
Additionally, to further enhance local contrast and emphasize on fine-grained details, Contrast Limited Adaptive Histogram Equalization(CLAHE) was implemented. This step provided better feature visibility especially for those images that had uneven lighting conditions or poor contrast overshadowed hand-signs. Finally, the images went under extensive class-wise balancing to address disparity in class distributions. The correct samples were augmented to 300 images per class and the incorrect samples were adjusted to 200 images per class.

Figure 4.5: Image representation before and after pre-processing

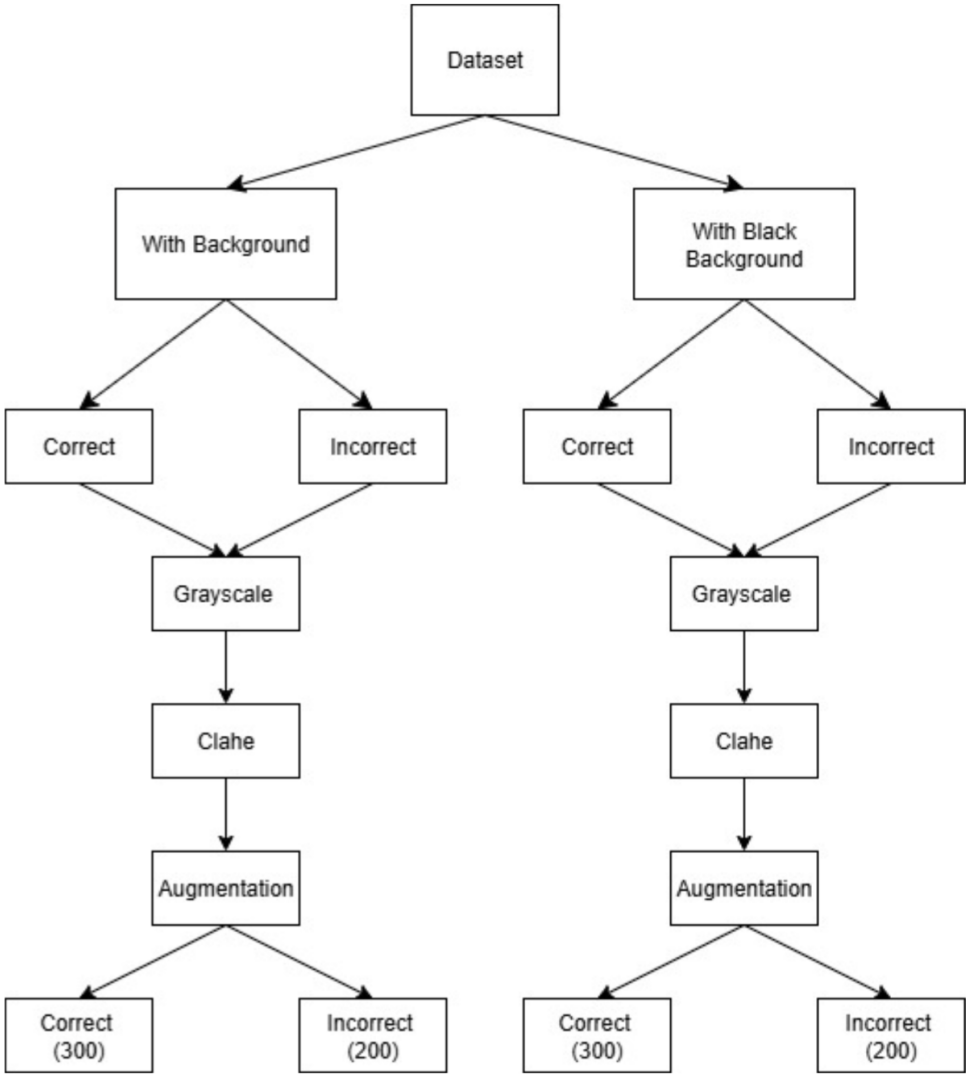


Figure 4.6: Pre-Process Flowchart

During the pre-processing, when the samples were lower than the target, augmentation techniques such as random rotations, height and width shifts, zooming and horizontal flip-

ping were used to meet the proper requirements. In contrast, the classes which contained excess images were randomly down-sampled to meet the target.

4.3.3 Summary of Preprocessed Data

In summary our Bengali Character Level Sign Language dataset were branched into 2 parts, the correct set and incorrect set. Both of these contained 36 classes each and were labeled from 0-35. The images were allocated according to the folder structure of correct and incorrect. Then each of these folders consisted of 36 subfolders representing each sign gesture. The correct sign images were the accurate hand gestures while the incorrect images were deliberate accuracies implemented upon classwise analysis. The dataset was again split into 2, one with normal background and the other with black background.
The images went under further pre-processing which included grayscaling, resizing, augmentation to ensure each correct class had 300 images and each incorrect class had 200 images.
Finally, the images were normalized according to CNN backbone requirements for the best feature extraction. Thus, the overall summary highlights how the pre-processing and transformation produced a clean, balanced and structured dataset ready for embedded metric learning and accurate evaluation.

4.4 Implementation of Selected Design

4.4.1 Dataset Preparation

The dataset was organized using Python's `pathlib` and `os` modules. A custom train–validation–test split ensured balanced representation. Images were resized, converted to NumPy arrays, normalized, and fed into a triplet generator that continuously produced anchor-positive-negative triplets during training.

4.4.2 Model Architecture and Training

Each backbone was loaded with ImageNet weights (excluding top layers) and connected to a dense projection layer of 128 units, followed by batch normalization, dropout (0.3), and L2 normalization. Triplet loss optimized embedding distances. Model training used a batch size of 16 triplets with checkpointing for weight saving on validation loss.

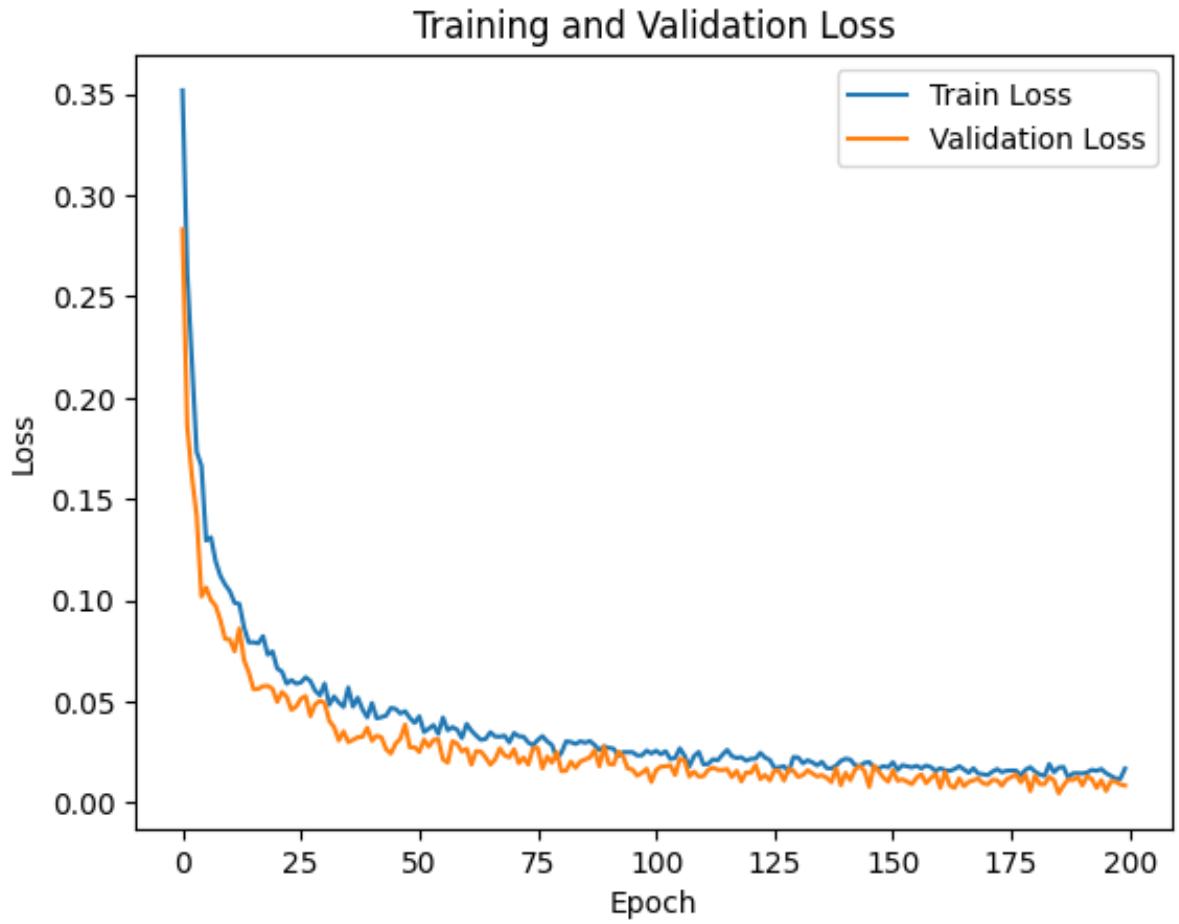


Figure 4.7: Training and Validation Loss of ResNet50

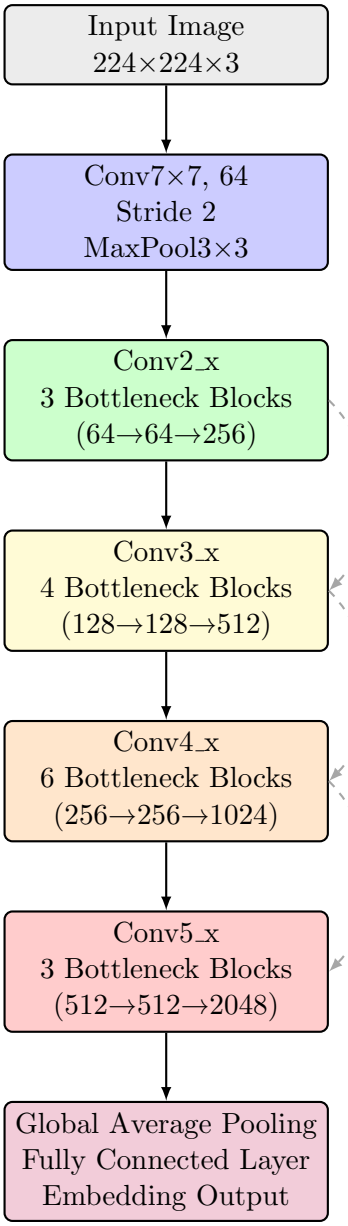


Figure 4.8: ResNet-50 architecture used for Bengali Sign Language gesture correction.

4.4.3 Triplet Architecture

The triplet network architecture was utilized in order to learn the relationship between images. Additionally, using triplet loss, it enabled to deliver the embeddings in such a way that the correct and incorrect gestures of the same class would be closer together while distance from the other correct class gestures would be further apart. It allows us to learn the discriminative embedding space unlike traditional classifiers.
Thus, in our thesis, the Triplet Network architecture learns the discriminative embedding space for Bengali Sign characters. Normal classifiers use categorical decision boundaries which cannot properly distinguish subtle intra-class and inter-class similarities. These similarities could arise from slight differences in hand shape, finger position and hand structure. So in our research, the triplet generator creates the following which are the anchor(correct image of a class), positive(incorrect image of the same class) and negative(correct image from the rest of the class). Then the images are delivered to the CNN

model to generate the embeddings. The embeddings are computed using the triplet loss function and depending upon a certain margin the weights of the model are updated. So, the triplet loss is configured in such a way that it tries to minimize the distance between anchor and positive, while also increasing the distance from the negative. This allows the inaccurate sign gestures to be mapped to their semantically most accurate representation. So, the model's weights are updated accordingly and the model learns an embedding space where each cluster of the correct classes are distinct. So, this embedding space allows now to properly perform error correction on incorrect sign gestures.
The Triplet Network was used by FaceNet (Schroff, Kalenichenko, & Philbin, 2015)[6], and it showed that embeddings from similar looking pictures has promising performance in clustering and recognition tasks. Mopidevi et al. (2023)[18] presents a triplet network framework in his paper that shows how this architecture, utilizing triplet loss, learns the embeddings space between anchors, negative and positive. This distance between anchor to positive and anchor to negative was used to ensure embeddings from similar gestures are closer and to different gestures are further apart. So the triplet loss was used to train the model to map anchor and positive closer while pushing the negatives apart. In their work of sign language recognition, the triplet network architecture allowed for proper sign language error detection. Thus, using the triplet architecture was essential for Bengali character sign correction tasks as well. The architecture had excellent synergy with metric distance which was primarily used in our thesis for mapping incorrect sign gestures to their proper classes.

4.4.4 Embedding Generation

In our thesis, embedding generation is a crucial part of our architecture for properly mapping incorrect sign gestures to the correct class. Unlike typical classification, the system learns vector representations(embeddings). The distance between these embeddings represent the visual similarity and semantic between sign images. Each of the embeddings highlight the orientation, curvature and finger position of the sign gesture. Moreover, using the embeddings, a continuous feature space is formed where embeddings of the same class are tightly clustered, while embeddings from different classes remain distant. Each CNN backbone was employed as a feature extractor with its classification head removed. The extracted features were projected into a 128-dimensional embedding space via a fully connected dense layer. Batch normalization and dropout were applied to improve generalization, and L2 normalization ensured embeddings had unit length.
Training used triplets of anchor, positive, and negative samples with the triplet loss function:

$$L = \max(d(a, p) - d(a, n) + m, 0) \tag{4.1}$$

where $d(a, p)$ and $d(a, n)$ are squared Euclidean distances between anchor–positive and anchor–negative embeddings, respectively, and m is a fixed margin (0.5).

The embedding based approach has various advantages over typical classification. First of all, in normal classification, the model learns strict class boundaries. Meanwhile, for metric learning, it learns a space where it can bring similar gesture embeddings close together and dissimilar ones further apart. Additionally, the embeddings can be used for error correction by comparing the embeddings of an error sign with all the class centroids of the correct class. The class centroids are the mean vector positions of all the correct

classes. For our research, we used 36 bengali characters, so there will be 36 centroids. So by comparing the distance between a test incorrect embedding and all of the centroids, it can quantifiably measure by how much a sign is similar to another or dissimilar. Thus, for an incorrect image, our system can map it to its proper class label by measuring the distance between the centroids and the incorrect image embedding in a vector space.
Embedding generation is commonly used in computer vision where fine-grained similarities matter. The paper "FaceNet: A Unified Embedding for Face Recognition and Clustering" uses a 128 dimensional embedding for face images (Schroff, Kalenichenko, & Philbin, 2015)[6]. Using these embeddings they measure the distances to find similarities between different faces to see whether the person is the same or not. Euclidean distance was used to measure the distance between the embeddings and once the embedding space was learned, verification tasks like same face or not were easily done by metric distance. As previous works used metric distance and embeddings for recognition tasks, it provided a baseline for our research. Alternatively, research on bengali characters using metric learning and vector embeddings were limited. Thus, for our research on correcting erroneous bengali character level sign language, we were motivated by past works that used metric learning to evaluate the distance between embeddings.

4.4.5 Evaluation Methodology

Two evaluation strategies were applied:
k-NN Evaluation: A 3-NN classifier using euclidean similarity assessed local consistency of embeddings.
Centroid-Based Evaluation: Class centroids were computed from training embeddings. Each test embedding was assigned to the nearest centroid based on distance, evaluating global class separation.
Performance was quantified using accuracy, per-class metrics, and confusion matrices.

4.4.6 Visualization and Analysis

Embedding quality was visualized using:
Confusion matrices via `seaborn.heatmap()`.
t-SNE for 2D projection of embeddings.
Bar plots of centroid distances for selected samples.
All visualizations and quantitative metrics were organized per backbone for comparative analysis.

4.4.7 Machine Learning Techniques

The implementation integrates:
Transfer Learning: Pre-trained CNNs for feature extraction.
Metric Learning: Triplet loss for embedding optimization.
Normalization and Regularization: Batch normalization, dropout, and L2 normalization.
Distance-Based Evaluation: k-NN and centroid analysis for both local and global similarity measures.

4.4.8 Testing and Validation

Model performance was validated quantitatively (accuracy, per-class scores, confusion matrices) and qualitatively (t-SNE visual inspection). Fixed random seeds ensured reproducibility.

Chapter 5

Result Analysis

5.1 Performance Evaluation

For the sake of performing effective and distinguishable evaluation, the CNN model architectures, along with the dataset were analyzed by various performance metrics. These metrics highlight both the quality of the embeddings from the selected models and their mapping abilities.

5.1.1 k-NN Accuracy

The K Nearest Neighbors (k-NN) classification was utilized during the mapping of the incorrect test images. A $k = 3$ nearest neighbors with euclidean distance was utilized for all the models throughout all the experiments. For a test sample i, let $\hat{y}_i$ denote its predicted class and y_i its true class. The k-NN accuracy is calculated by:

$$\text{Accuracy}_{\text{kNN}} = \frac{1}{N} \sum_{i=1}^{N} \mathbf{1}(\hat{y}_i = y_i) \tag{5.1}$$

For our research, the distance from test incorrect embeddings to all the correct embeddings are computed to calculate the classwise and overall accuracies.

5.1.2 Centroid Accuracy

Centroid accuracy is another method by which the embeddings were properly mapped to their correct class. The formula for centroid accuracy is as follows:

$$\mathbf{c}_k = \frac{1}{n_k} \sum_{i=1}^{n_k} \mathbf{e}_i \tag{5.2}$$

where n_k is the number of training samples in class k, and $\mathbf{e}_i$ represents the embedding of a sample. The distance from the sample incorrect image to the class centroids were computed to find the overall accuracy. This approach highlights how well embeddings cluster around their true class centers, which is crucial for mapping incorrect signs to their correct forms.

5.1.3 Macro Precision, Recall, and F1-Score

To account for class imbalance and evaluate the model performance across all classes uniformly, macro-averaged precision, recall, and F1-score were computed. For class k, the metrics are defined as:

$$\text{Precision}_k = \frac{TP_k}{TP_k + FP_k}, \quad \text{Recall}_k = \frac{TP_k}{TP_k + FN_k}, \quad F1_k = \frac{2 \cdot \text{Precision}_k \cdot \text{Recall}_k}{\text{Precision}_k + \text{Recall}_k} \tag{5.3}$$

where TP_k, FP_k, and FN_k represent true positives, false positives, and false negatives for class k, respectively. The macro-averaged F1-score is:

$$\text{Macro-F1} = \frac{1}{K} \sum_{k=1}^{K} F1_k \tag{5.4}$$

These metrics provide a balanced view of performance across all 36 Bengali character classes, preventing dominance by more frequent classes.

5.1.4 Intra-class and Inter-class Distances

The embedding space was further analyzed by computing intra-class and inter-class distances to assess cluster compactness and class separability. For class k with n_k embeddings $\mathbf{e}_i$, the average intra-class distance is:

$$\text{Intra-class distance}_k = \frac{2}{n_k(n_k - 1)} \sum_{i<j} \|\mathbf{e}_i - \mathbf{e}_j\|_2 \tag{5.5}$$

The inter-class distance between class k and class l centroids is:

$$\text{Inter-class distance}_{k,l} = \|\mathbf{c}_k - \mathbf{c}_l\|_2 \tag{5.6}$$

Small intra-class distances indicate tight clustering of embeddings belonging to the same class, while large inter-class distances ensure clear separation between different classes. Visualization techniques such as t-SNE and UMAP were employed to illustrate these relationships, providing qualitative confirmation of embedding quality.

5.1.5 Confusion Matrix

The confusion matrix is a visualization table that can accurately evaluate how well a model is performing on a set of images. The vertical axis represents the true values and the horizontal axis represent the predicted values It shows how many test samples were correctly mapped to the proper class label.

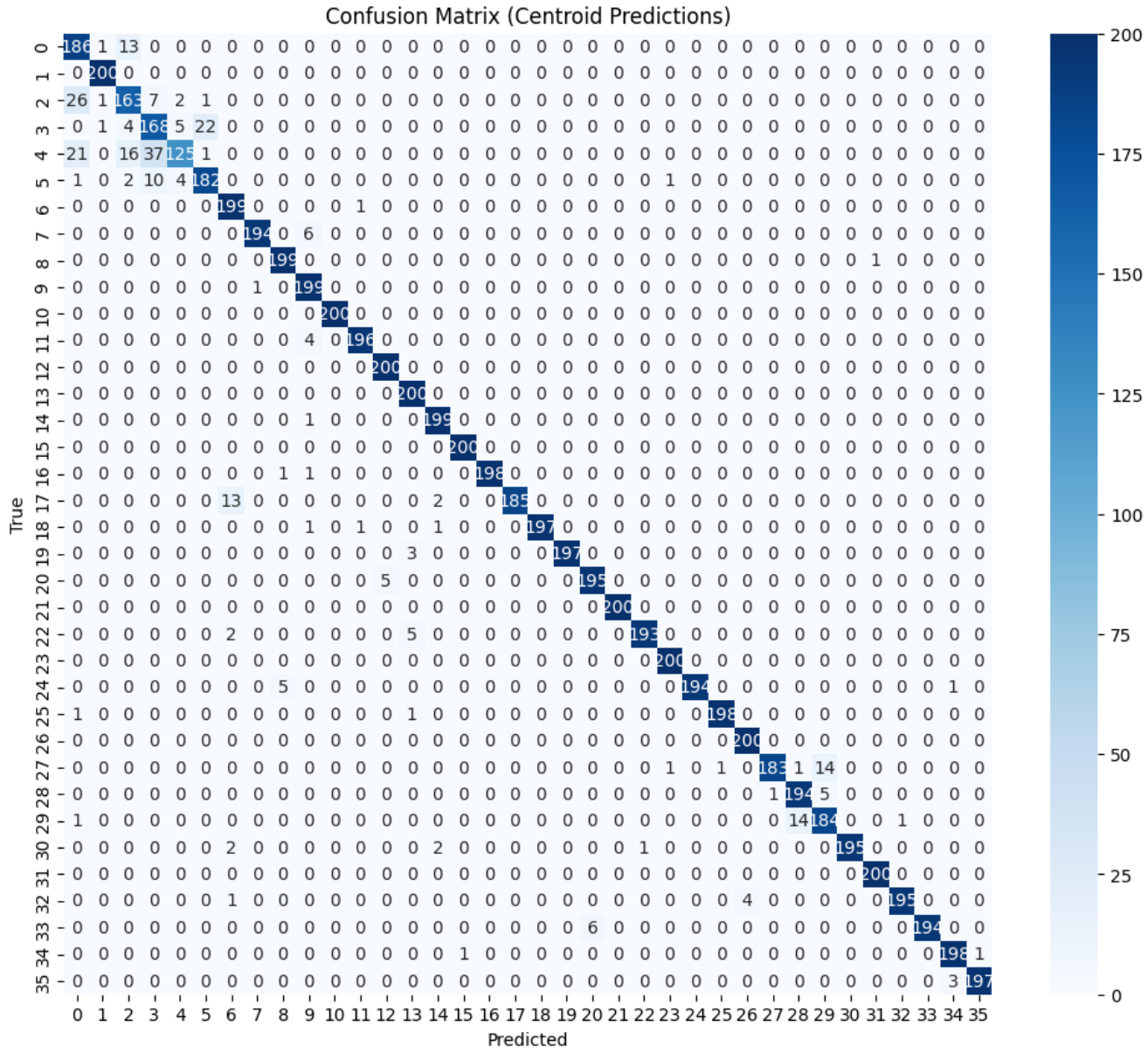


Figure 5.1: Confusion Matrix of ResNet50.

In the above confusion matrix, centroid based predictions on the incorrect test images for ResNet50 were illustrated. Along the diagonal, most of the values are concentrated, this shows that the majority of the incorrect images were mapped properly to their semantically accurate labels. Some instances of wrong class mapping are seen as represented by few off-diagonal entries. This strong diagonal pattern shows that the centroid based model achieved good accuracy and can properly distinguish the 36 Bengali characters.

5.2 Class-wise Accuracy

In order to analyze the performance of the models, classwise accuracy was generated for both KNN and Centroid based distancing. Each test incorrect images were placed into the trained embedding model and the correct class according to either centroid mapping or KNN were delivered. This predicted class and the true class were compared in order to

quantifiable assess whether the incorrect image was properly mapped to its correct class.

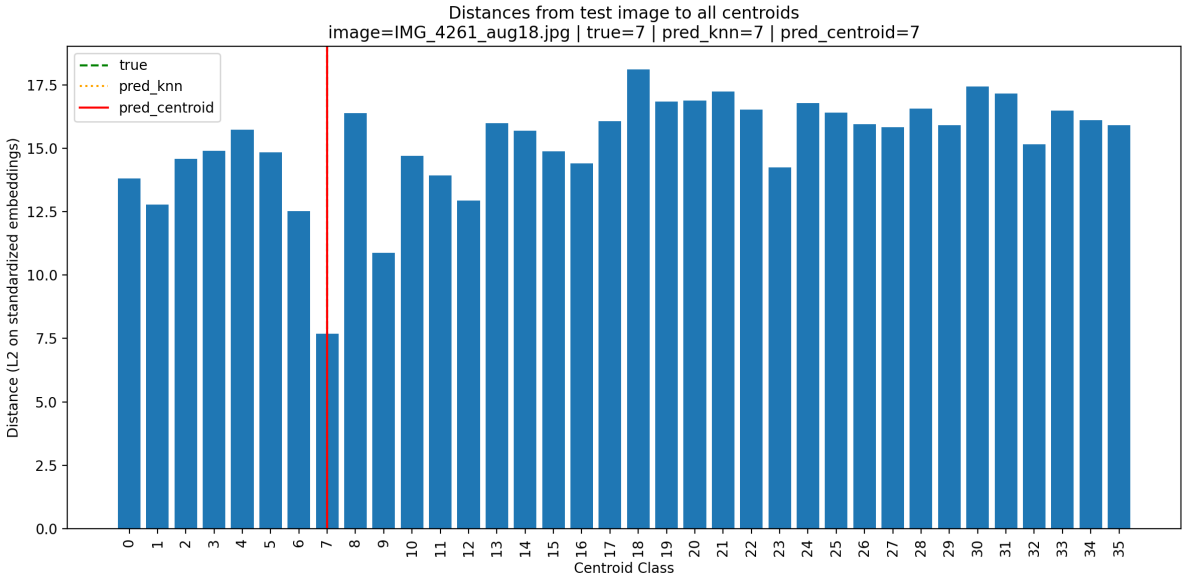


Figure 5.2: Class-wise accuracy of ResNet50 embeddings on the test set.

As we can see from the above example, an incorrect test image was passed into the trained embedding model. The embeddings from the test incorrect image were generated and plotted in an embedding space. The distance between the test embedding and all the centroids were calculated. According to the above figure, it shows that the distance between class 7 centroid and the incorrect image embedding is the smallest. Thus, the predicted class is computed as 7 which is compared against the true value. As both the ground truth and the predicted class is both 7, the model could properly map an incorrect image to its corresponding class for both centroid and KNN methods. So, all the incorrect images are tested within the same criterias and classwise accuracy is computed.

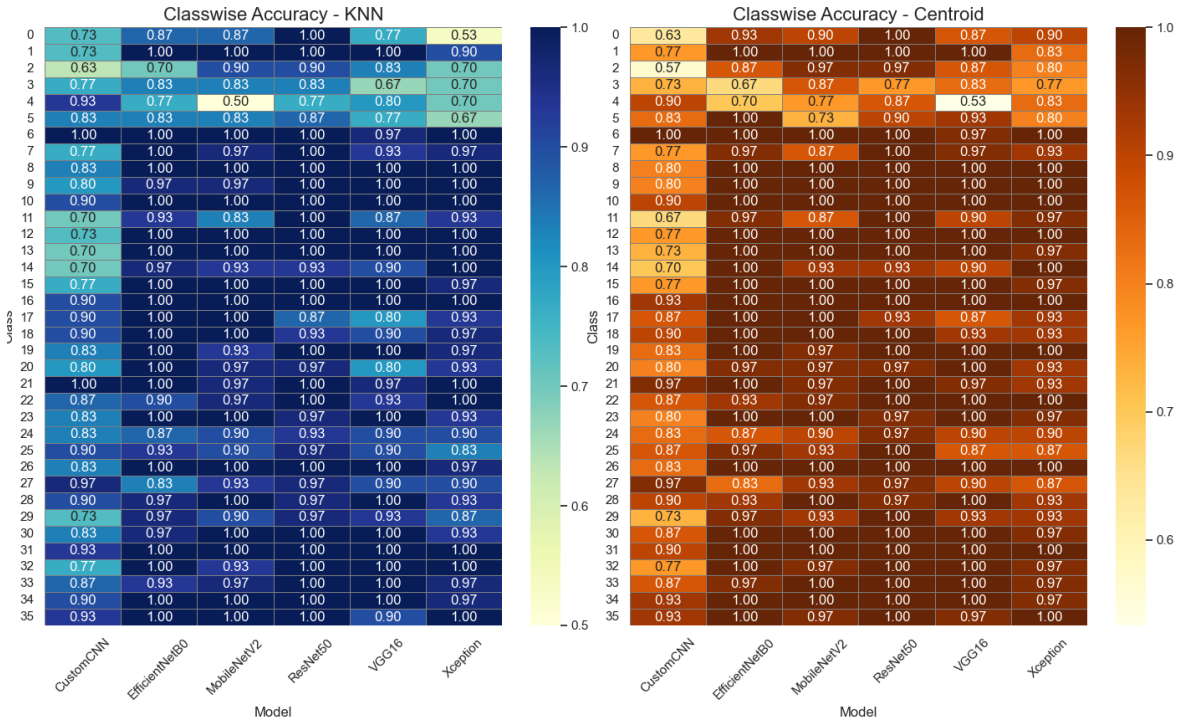


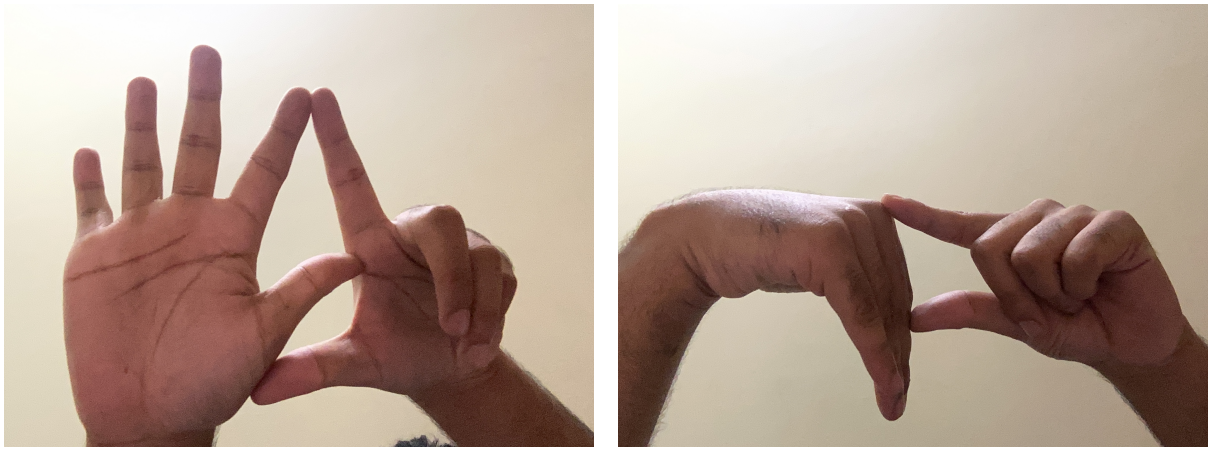
Figure 5.3: Classwise accuracy of best 6 model

5.2.1 Class-wise Accuracy Insights

In order to have a better understanding about the best-performing models, the class-wise accuracy was investigated using KNN and Centroid based evaluation. These two approaches were implemented to understand the structure and separability of the embeddings in the vector space. The KNN method used the local neighborhood relationships between samples, meanwhile the centroids used global classwise compactness for each. Together, they provide a comparative view on how each class is represented and quantitatively demonstrate the relationship between intra class sign gestures.

Although both methods of evaluation provided optimal results, but overall, the centroid heatmap showed the most uniform and higher accuries compared to the KNN heatmap. This indicated that the feature extractors were successful in generating compact clusters for each classes although visual overlaps were seen from similar gestures

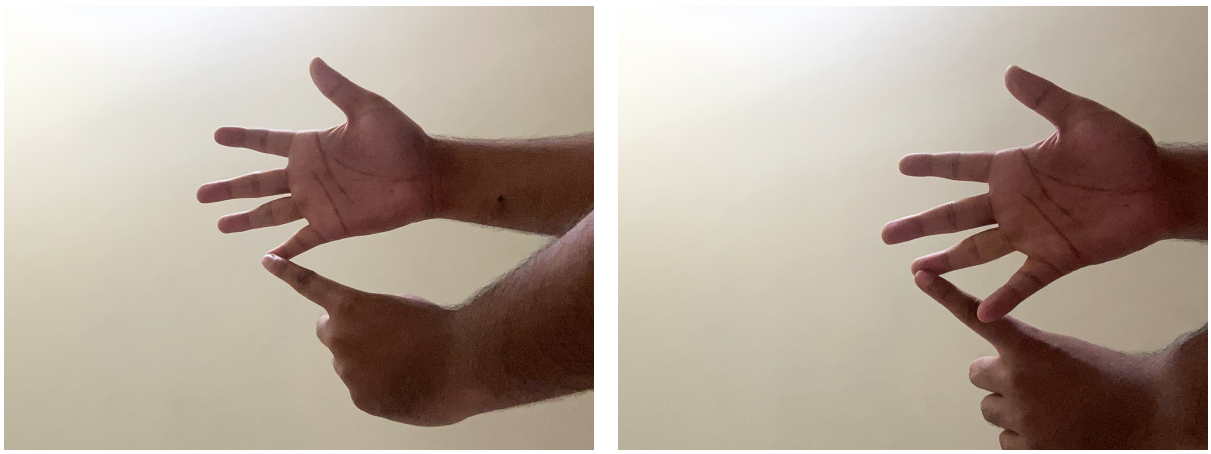
Some classes that had better performance from both methods included 6,9,10,15,18,21,28,34 and 35. This showcase that features of these classes included unique structure patters like finger separation and hand orientaion which were more effecitvely captured by the CNN backbones. Thus, as these form tight clusters, it becomes easier for both KNN and Centroid method to yield the best accuracy among these classes



(a) Hand 6 (b) Hand 10

Figure 5.4: Better Hand Distinction

The above sample images show how class 9 and class 10 have distinct hand gestures. Thus proving that it became easier for the CNN models to form tight clusters on classes of separable hand signs.
On the other hand, certain classes like 0,2,4,11 showed fluctuating accuries especially for the KNN method. These variations in accuricies show how these classes contain similarities between themselves such as similar finger positions and overlapping hand structures. Thus, due to the overall tendencies of these classes, the KNN method suffered the worst, resulting in a decline in accuracy. But, in general, the centroid based method accuracy still demonstrated better accuracy which highlights the fact that although local there may exist local confusions or outliers, the global representation of these clusters in the vector space is still meaningful. This suggests that the pattern and the embeddings learning by the models are robust, although some fine grained gestures posed slight challenges.



(a) Hand 2 (b) Hand 4

Figure 5.5: Similar Sign Language

The sample image of class 2 and 4 showcase an overallaping structure between the 2 datasets. These posed slight challenges during local individual embeddings in the vectors space which resulted in lower accuries for the KNN evaluation. But the Centroid based

evaluation still demonstrated better performance, explaining tight clusters and excellent feature extraction from the models.

5.2.2 Classwise Model Evaluation Insights

Analyzing the centroid based heatmap, EfficientNetB0, Resnet50 and Xception demonstrated the most stable class wise accuracies in both the KNN and Centroid analysis. The models pre-trained ImageNet features could effectively generalize the dataset and resulted in high separable embeddings. ResNet50 has residual (skip) connections which would enable the network to train without vanishing gradients. Moreover, it can learn hierarchical and fine grained spatial features which is perfect for sign gestures where subtle finger positions matter. On the other hand, EfficientNetB0 has squeeze and excitation blocks which can help focus on hand regions. Finally, Xception, although with the lowest parameters, factorizes convolutions into spatial and depthwise components which can learn finer finger gestures.
Meanwhile, MobileNetV2 has a light architecture which could mean it could not accurately differentiate subtle sign language gestures. VGG16 has the largest parameters but lacks skip connections which might have resulted in slower convergence. Finally, the custom CNN did not inherit a pretrained backbone and so struggled to separate the classes which resulted in the worst clustering.

5.3 Statistical Analysis

In order to understand the clustering and separation of embeddings, intra-class distance and inter nearest centroid distances were computed. Boxplots were chosen in order to visualize and assess these distributions more effectively.

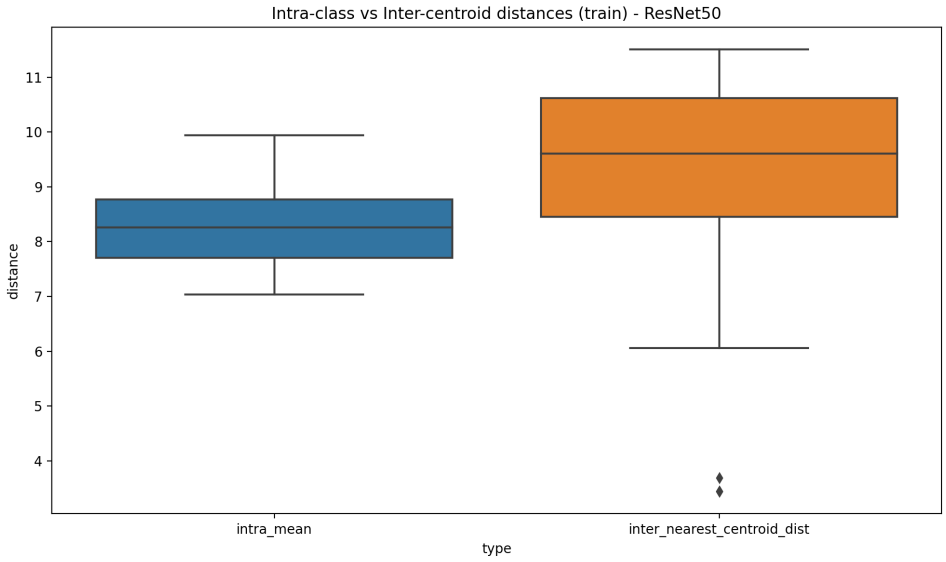


Figure 5.6: Intra-class vs. nearest Inter-class distances for ResNet50 embeddings.

- **Compact Clusters:** The mean intra-class distance indicates how embeddings of the same class cluster near each other in the vector space. Meanwhile, the intra nearest centroid distance represents the average distance between centroids of

different classes. Smaller mean intra-class distances show that clusters are tightly compressed, and when the intra nearest centroid distance is greater than the mean intra-class distance, it demonstrates that the triplet models effectively separate different clusters in the embedding space.

- **Consistency:** Low variance or standard deviation indicates that intra-class embeddings are tightly grouped, reflecting consistent embedding behavior within the same class.
- **Variability:** Outliers are present, as suggested by the diagram. This indicates that additional computation or strategies may be required to handle these exceptional embeddings in the vector space.

5.3.1 Comparison and Relationships

t-SNE were used to visualize embeddings:

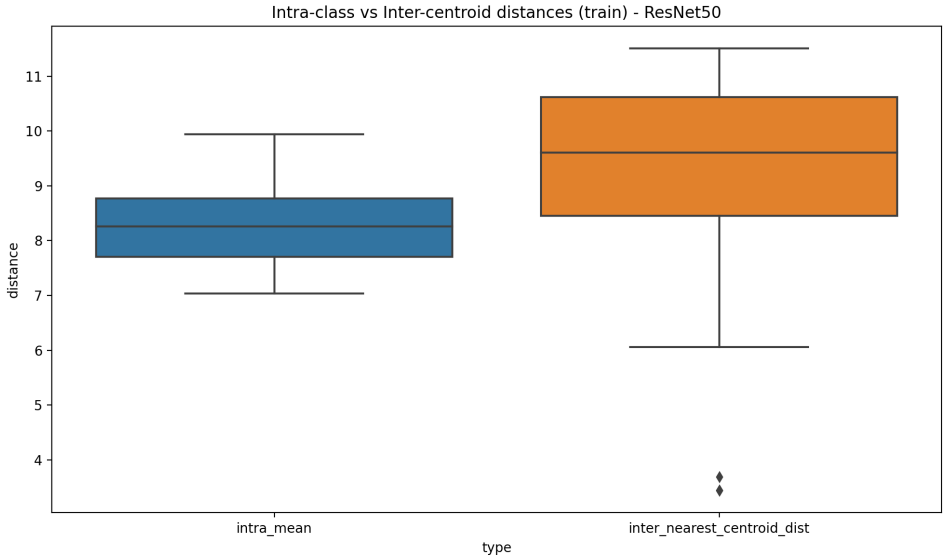


Figure 5.7: t-SNE visualization of test embeddings colored by class.

As we can see from the above diagram. The incorrect test embeddings of the same classes are clustering around a fixed position, with the exception of certain outliers. This shows that the model has done an exceptional job in correctly mapping incorrect gestures to their corresponding correct images.

5.4 Ablation Study: Hyperparameter Analysis with Results

In order to properly compute how different hyperparameters affect the performance of the triplet network, we conducted seven separate experiments with 9 different models. Each experiment had different parameters such as number of epochs, validation steps, steps per epoch, triplet loss margin, embedding size. Parameters such as input image size, random seed, batch size, neighbors for KNN evaluation were kept constant.

- **Effect of Training Duration (Epochs and Steps per Epoch):** Increasing the number of epochs and steps per epoch consistently improved accuracy for both KNN and centroid-based methods across all backbones. For example, Test 3 achieved the best performance with 200 epochs and 100 steps per epoch, indicating that longer training iterations are required to extract meaningful embeddings.
- **Effect of Triplet Loss Margin:** A triplet loss margin of 0.5 generated the most promising separation between classes. Smaller margins, such as 0.3 in Test 5, resulted in reduced intra-class separation, diminishing the discriminative power of the model. Larger margins, like 0.7 in Test 6, led to overly strict separation, negatively affecting generalization.

- **Effect of Embedding Size:** Increasing the embedding size from 128 to 256 slightly improved class separability for some models. Shallow models such as CustomCNN and ConvNet showed very limited improvement, suggesting that higher-dimensional embeddings require larger model capacity.
- **Dataset Background Impact:** A slight increase in performance was observed for models trained on natural backgrounds compared to black backgrounds. This indicates that pretrained networks can leverage background information for better feature extraction. Nevertheless, the best performing models remained equally accurate on both datasets.
- **Backbone Performance:** Pretrained models such as ResNet50, MobileNetV2, EfficientNetB0, and DenseNet121 consistently outperformed CustomCNN and other shallow models. ResNet50 achieved the highest overall accuracy, followed closely by MobileNetV2. Shallow networks underperformed, highlighting the importance of using a robust pretrained backbone for effective feature extraction.
- **KNN vs. Centroid Evaluation:** Centroid-based evaluation consistently outperformed KNN across all experiments. This indicates that centroid distance is a superior metric, particularly for error correction tasks.
- **General Trends:** The best results were obtained when embedding size, training duration, and margin were carefully tuned. This emphasizes that proper hyperparameter selection is crucial for effectively mapping incorrect sign gestures.

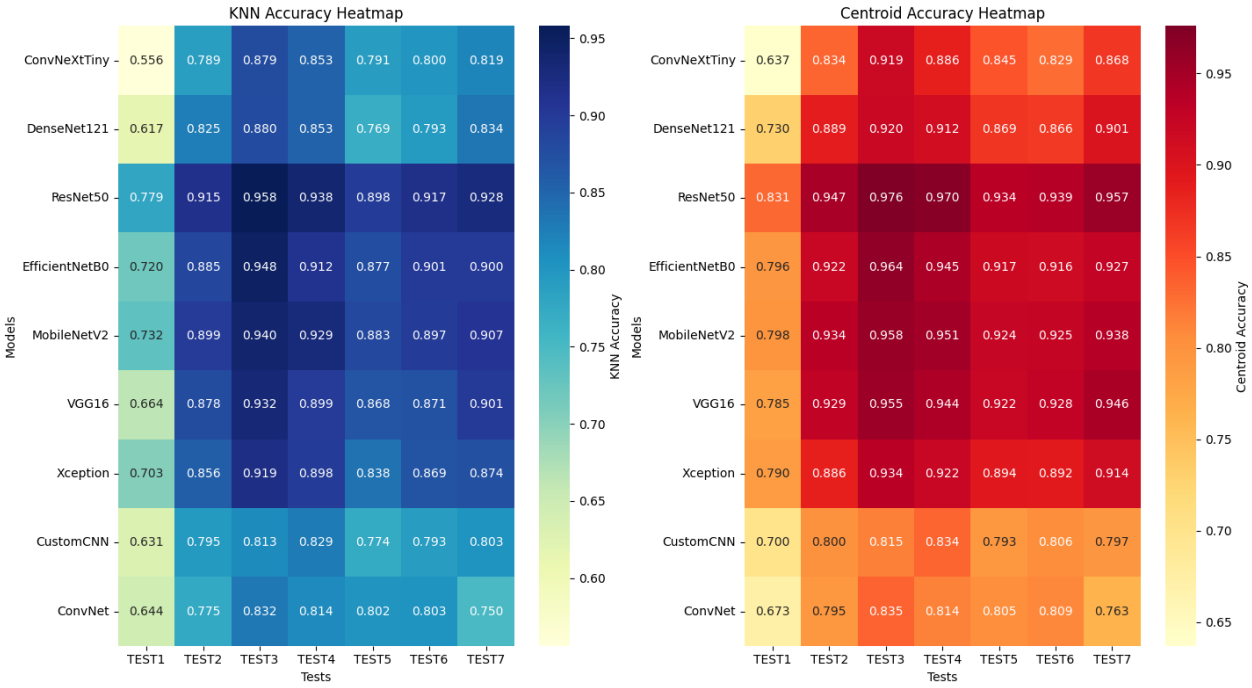


Figure 5.8: Heatmap of Background Dataset

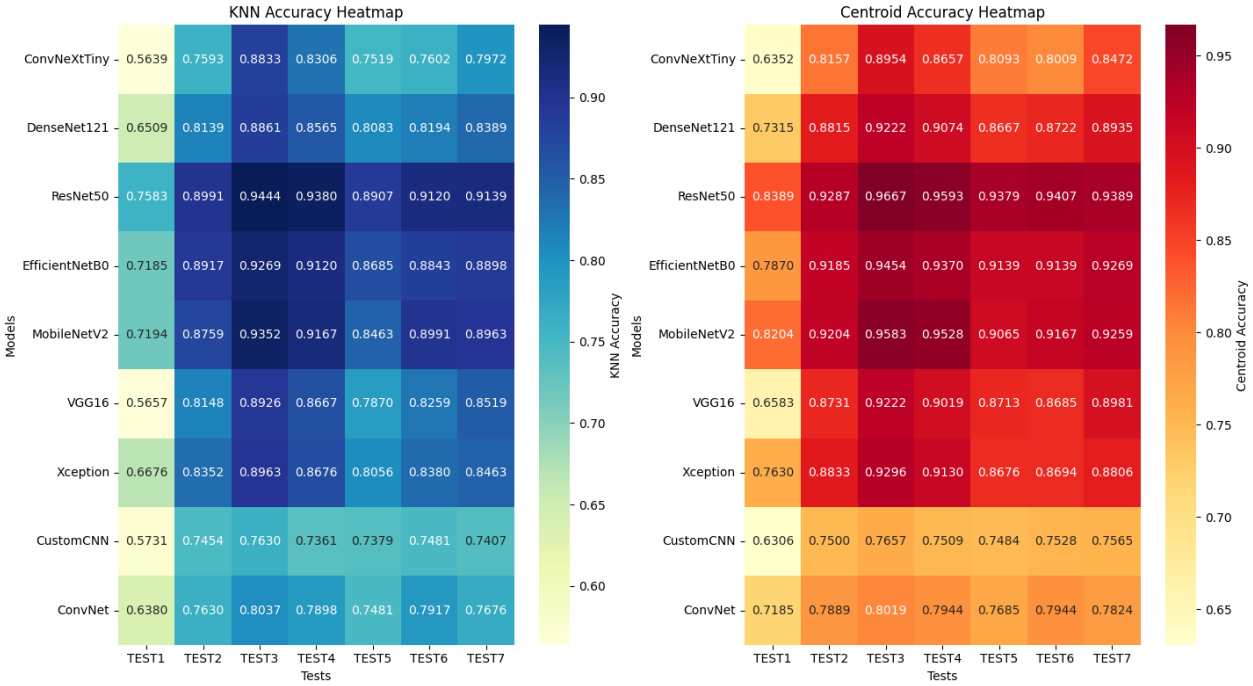


Figure 5.9: Heatmap of Black Background Dataset.

5.5 Discussions

Our analysis confirms the following:

- The triplet network successfully learns discriminative embeddings for correct mapping of incorrect images.
- Pre-trained models perform comparatively better compared to custom CNN that lacks pretrained weights. Thus, MobileNetV2 stands out to be the best model as it balances both accuracy and computational costs.
- Class-wise evaluation explains how similar signs are less easily discriminative in the vector space. So, they provide the most varied accuracy between all the models.
- Hyperparameter tuning is essential in order to provide better inter class separation and compact cluster quality. Thus, as discussed in the ablation study, cautious tuning of these parameters result in more refined accuracy.

Overall, our proposed system gives an effective and robust system for correcting Bengali characterlevel sign language gestures, as evident from multiple model insights, classwise analsyis and evaluation metrics.

Test	Backbone	With Background		Black Background		Hyperparameters			
		KNN	Centroid	KNN	Centroid	EPOCHS	Steps/EP	VAL	MARGIN
1	ConvNeXtTiny	0.556	0.637	0.5639	0.6352	100	30	10	0.5
1	DenseNet121	0.617	0.730	0.6509	0.7315	100	30	10	0.5
1	ResNet50	0.779	0.831	0.7583	0.8389	100	30	10	0.5
1	EfficientNetB0	0.720	0.796	0.7185	0.7870	100	30	10	0.5
1	MobileNetV2	0.732	0.798	0.7194	0.8204	100	30	10	0.5
1	VGG16	0.664	0.785	0.5657	0.6583	100	30	10	0.5
1	Xception	0.703	0.790	0.6676	0.7630	100	30	10	0.5
1	CustomCNN	0.631	0.700	0.5731	0.6306	100	30	10	0.5
1	ConvNet	0.644	0.673	0.6380	0.7185	100	30	10	0.5
2	ConvNeXtTiny	0.789	0.834	0.7593	0.8157	100	100	50	0.5
2	DenseNet121	0.825	0.889	0.8139	0.8815	100	100	50	0.5
2	ResNet50	0.915	0.947	0.8991	0.9287	100	100	50	0.5
2	EfficientNetB0	0.885	0.922	0.8917	0.9185	100	100	50	0.5
2	MobileNetV2	0.899	0.934	0.8759	0.9204	100	100	50	0.5
2	VGG16	0.878	0.929	0.8148	0.8731	100	100	50	0.5
2	Xception	0.856	0.886	0.8352	0.8833	100	100	50	0.5
2	CustomCNN	0.795	0.800	0.7454	0.7500	100	100	50	0.5
2	ConvNet	0.775	0.795	0.7630	0.7889	100	100	50	0.5
3	ConvNeXtTiny	0.879	0.919	0.8833	0.8954	200	100	50	0.5
3	DenseNet121	0.880	0.920	0.8861	0.9222	200	100	50	0.5
3	ResNet50	0.958	0.976	0.9444	0.9667	200	100	50	0.5
3	EfficientNetB0	0.948	0.964	0.9269	0.9454	200	100	50	0.5
3	MobileNetV2	0.940	0.958	0.9352	0.9583	200	100	50	0.5
3	VGG16	0.932	0.955	0.8926	0.9222	200	100	50	0.5
3	Xception	0.919	0.934	0.8963	0.9296	200	100	50	0.5
3	CustomCNN	0.813	0.815	0.7630	0.7657	200	100	50	0.5
3	ConvNet	0.832	0.835	0.8037	0.8019	200	100	50	0.5
4	ResNet50	0.938	0.970	0.93796	0.95926	150	100	10	0.5
4	EfficientNetB0	0.912	0.945	0.91204	0.93704	150	100	10	0.5
4	MobileNetV2	0.929	0.951	0.91667	0.95278	150	100	10	0.5
4	Xception	0.898	0.922	0.86759	0.91296	150	100	10	0.5
4	VGG16	0.899	0.944	0.86667	0.90185	150	100	10	0.5
4	DenseNet121	0.853	0.912	0.85648	0.90741	150	100	10	0.5
4	ConvNeXtTiny	0.853	0.886	0.83056	0.86574	150	100	10	0.5
4	ConvNet	0.814	0.814	0.78981	0.79444	150	100	10	0.5
4	CustomCNN	0.829	0.834	0.73611	0.75093	150	100	10	0.5
5	ResNet50	0.8981	0.9343	0.8907	0.9379	100	100	10	0.3
5	EfficientNetB0	0.8769	0.9167	0.8685	0.9139	100	100	10	0.3
5	MobileNetV2	0.8833	0.9241	0.8463	0.9065	100	100	10	0.3
5	DenseNet121	0.7694	0.8694	0.8083	0.8667	100	100	10	0.3
5	Xception	0.8380	0.8944	0.8056	0.8676	100	100	10	0.3
5	VGG16	0.8676	0.9222	0.7870	0.8713	100	100	10	0.3
5	ConvNeXtTiny	0.7907	0.8454	0.7519	0.8093	100	100	10	0.3
5	ConvNet	0.8019	0.8046	0.7481	0.7685	100	100	10	0.3
5	CustomCNN	0.7741	0.7935	0.7379	0.7484	100	100	10	0.3
6	ResNet50	0.9167	0.9389	0.9120	0.9407	100	100	10	0.7
6	MobileNetV2	0.8972	0.9250	0.8991	0.9167	100	100	10	0.7
6	EfficientNetB0	0.9009	0.9157	0.8843	0.9139	100	100	10	0.7
6	VGG16	0.8713	0.9278	0.8259	0.8685	100	100	10	0.7
6	Xception	0.8694	0.8917	0.8380	0.8694	100	100	10	0.7
6	DenseNet121	0.7935	0.8657	0.8194	0.8722	100	100	10	0.7
6	ConvNeXtTiny	0.8000	0.8287	0.7602	0.8009	100	100	10	0.7
6	ConvNet	0.8028	0.8093	0.7917	0.7944	100	100	10	0.7
6	CustomCNN	0.7926	0.8056	0.7481	0.7528	100	100	10	0.7
7	ResNet50	0.9278	0.9565	0.9139	0.9389	100	100	10	0.5
7	MobileNetV2	0.9074	0.9380	0.8963	0.9259	100	100	10	0.5
7	EfficientNetB0	0.9000	0.9269	0.8898	0.9269	100	100	10	0.5
7	VGG16	0.9009	0.9463	0.8519	0.8981	100	100	10	0.5
7	Xception	0.8741	0.9139	0.8463	0.8806	100	100	10	0.5
7	DenseNet121	0.8343	0.9009	0.8389	0.8935	100	100	10	0.5
7	ConvNeXtTiny	0.8194	0.8676	0.7972	0.8472	100	100	10	0.5
7	ConvNet	0.7500	0.7630	0.7676	0.7824	100	100	10	0.5
7	CustomCNN	0.8028	0.7972	0.7407	0.7565	100	100	10	0.5

Table 5.1: Ablation study results showing KNN and centroid accuracies for all models on both datasets across seven tests.

Chapter 6

Conclusion

6.1 Summary of Findings

Multiple experiments were conducted across nine deep learning backbones, each yielding promising results. The black background dataset demonstrated high intra-class compactness and clear separability between gesture classes. Increasing the number of epochs to 200 and improving the number of steps per iteration further enhanced embedding quality and classification accuracy.
Between the two evaluation strategies, the Centroid-based method proved more stable and effective than kNN, offering better inter-class discrimination and overall accuracy. Among the backbones, ResNet50 consistently achieved over 90% accuracy across experiments, demonstrating strong generalization and stable embedding structures. Although other pretrained models such as MobileNetV2 and EfficientNetB0 also performed well, ResNet50 exhibited the most reliable classwise accuracy. In contrast, models like CustomCNN and ConvNet, which were trained from scratch, showed lower performance due to their inability to extract high-quality embeddings.
Overall, these findings confirm that the triplet-loss-based framework effectively learns a discriminative embedding space capable of mapping incorrect gestures to their correct representations. The system achieved compact intra-class embeddings and strong inter-class separations, validating the proposed method as a robust approach for gesture correction.

6.2 Contributions to the Field

Our thesis has provided several novel approaches to the research of Bengali character level erroneous sign correction:

- **Metric-based Correction:** Previous researches have shown to devote most of their resources on classification using functions like logistic regression or softmax. Correcting character level bengali sign gestures via metric learning is highly overlooked. In our research, we have shown that distance based correction can be a more quantifiable way to map incorrect gestures to their correct forms.
- **Dual Evaluation Strategy:** Using both Centroid based and KNN allowed us to provide a dual perspective on evaluation. It allowed us to gain key insights onto which would perform the best on sign gestures, local neighbors or classwise

centroids. Thus, it allowed to contribute further information on the pros and cons of KNN and centroid based accuracy method.

- **Pretrained Triplet Embeddings:** : The research demonstrated that pretrained CNN backbones significantly improve embedding discriminability, offering a scalable solution for real-world gesture correction systems.
- **Dataset Contribution:** As little work has been done on rectification of Bengali character level sign gesture, we generated our own correct and incorrect character-level Bengali gesture dataset. This will enable other researchers to work on error correction and address the critical gap of correction in BdSL research.

These contributions advance the understanding of how embedding-based metric learning can extend beyond classification to enable error correction, feedback, and adaptive learning systems in sign language research.

6.3 Recommendations for Future Work

Future research could explore several directions and applications based on this framework:

- **Larger and More Diverse Datasets:** Expanding the dataset with more signers, environments, and gestures could improve model generalization and enable cross-domain learning.
- **Hybrid Pre-processing:** Combining black background segmentation with adaptive context preservation may enhance embedding robustness and handle real-world background variations.
- **Real-time Feedback Systems:** Integrating this framework into real-time systems could help learners practice and correct their signs instantly, bridging communication gaps for the deaf and hearing-impaired communities.
- **Media-Pipe Integration:** Incorporating pose or bounding box information using tools like Media-Pipe could further enhance embedding accuracy by isolating gesture regions.
- **Advanced Metric Learning:** Future work could experiment with contrastive learning, quadruplet loss, or self-supervised embeddings to improve robustness and scalability.

These directions aim to translate the theoretical findings of this study into practical, real-world solutions for education, accessibility, and assistive technologies.

Bibliography

[1] B. S. Parton, "Sign language recognition and translation: A multidisciplined approach from the field of artificial intelligence," *The Journal of Deaf Studies and Deaf Education*, vol. 11, no. 1, pp. 94–101, 2006. DOI: 10.1093/deafed/enj003.

[2] J. S. Y. Lee, "Automatic correction of grammatical errors in non-native english text," Ph.D. dissertation, Massachusetts Institute of Technology, 2009. [Online]. Available: https://sls.csail.mit.edu/publications/2009/Thesis_Lee.pdf.

[3] A. Samir and M. Aboul-Ela, "Error detection and correction approach for arabic sign language recognition," in *IEEE 7th International Conference on Computer Engineering & Systems (ICCES)*, Cairo, Egypt, 2012. DOI: 10.1109/ICCES.2012.6408496.

[4] H. I. Elshazly, A. M. Elkorany, A. E. Hassanien, and A. T. Azar, "Ensemble classifiers for biomedical data: Performance evaluation," in *8th International Conference on Computer Engineering & Systems (ICCES)*, Cairo, Egypt, 2013, pp. 184–189. DOI: 10.1109/ICCES.2013.6707198.

[5] M. A. Rahaman, M. Jasim, M. H. Ali, and M. Hasanuzzaman, "Real-time computer vision-based bengali sign language recognition," in *17th International Conference on Computer and Information Technology (ICCIT)*, 2014, pp. 192–197.

[6] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, pp. 815–823. DOI: 10.1109/CVPR.2015.7298682. [Online]. Available: https://www.cv-foundation.org/openaccess/content_cvpr_2015/papers/Schroff_FaceNet_A_Unified_2015_CVPR_paper.pdf.

[7] R. Grundkiewicz, "Algorithms for automatic grammatical error correction," Ph.D. dissertation, Adam Mickiewicz University in Poznań, Faculty of Mathematics and Computer Science, 2017.

[8] A. Cherian and A. Sullivan, "Sem-gan: Semantically-consistent image-to-image translation," in *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2019, pp. 1796–1806. DOI: 10.1109/WACV.2019.00196.

[9] S. Stoll, N. C. Camgoz, S. Hadfield, and R. Bowden, "Text2sign: Towards sign language production using neural machine translation and generative adversarial networks," *International Journal of Computer Vision*, vol. 128, no. 4, pp. 891–908, 2020. DOI: 10.1007/s11263-019-01281-2.

[10] M. Rivera-Acosta, J. M. Ruiz-Varela, S. Ortega-Cisneros, J. Rivera, R. Parra-Michel, and P. Mejia-Alvarez, "Spelling correction real-time american sign language alphabet translation system based on yolo network and lstm," *Electronics*, vol. 10, no. 9, p. 1035, Apr. 2021. DOI: 10.3390/electronics10091035.

[11] M. S. Islam, M. S. Hossain, M. A. Hossain, and M. A. Hossain, "Kunet—an optimized ai based bengali sign language translator for hearing impaired and non verbal people," *IEEE Access*, vol. 10, pp. 1–10, 2022. DOI: 10.1109/ACCESS.2022.10705308.

[12] B. Natarajan et al., "Development of an end-to-end deep learning framework for sign language recognition, translation, and video generation," *IEEE Access*, vol. 10, pp. 104 358–104 371, 2022. DOI: 10.1109/ACCESS.2022.3210543.

[13] S.-I. Stamoulis, "Sign language detection," Ph.D. dissertation, University of West Attica, 2022. [Online]. Available: https://example.com.

[14] A. Haque, R. A. Pulok, M. M. Rahman, S. Akter, N. Khan, and S. Haque, "Recognition of bangladeshi sign language (bdsl) words using deep convolutional neural networks (dcnns)," *Emerging Science Journal*, vol. 7, no. 6, pp. 2183–2201, Dec. 2023. DOI: 10.28991/ESJ-2023-07-06-019.

[15] A. A. J. Jim, I. Rafi, M. Z. Akon, U. Biswas, and A.-A. Nahid, "Ku-bdsl: An open dataset for bengali sign language recognition," *Data in Brief*, vol. 51, p. 109 797, 2023. DOI: 10.1016/j.dib.2023.109797.

[16] S. Lata, "The automated sign language translation system using deep learning," Ph.D. dissertation, Mahasarakham University, Dept. of Information Technology, Mahasarakham, Thailand, 2023. [Online]. Available: http://202.28.34.124/dspace/handle/123456789/2165.

[17] E. Lin, "Comparative analysis of pix2pix and cyclegan for image-to-image translation," *Highlights in Science, Engineering and Technology*, vol. 39, pp. 915–925, 2023.

[18] S. Mopidevi, M. V. D. Prasad, and P. V. V. Kishore, "Multiview meta-metric learning for sign language recognition using triplet loss embeddings," *Pattern Analysis and Applications*, vol. 26, no. 3, pp. 1125–1141, 2023. DOI: 10.1007/s10044-023-01134-2. [Online]. Available: https://doi.org/10.1007/s10044-023-01134-2.

[19] R. Sreemathy et al., "Continuous word-level sign language recognition using an expert system based on machine learning," *International Journal of Cognitive Computing in Engineering*, vol. 4, pp. 170–178, 2023.

[20] R. Temkar, S. Jagtap, K. Jadhav, and M. Deshmukh, "Real-time sign language recognition using mobilenetv2 and transfer learning," *arXiv preprint*, Dec. 2023. [Online]. Available: https://arxiv.org/abs/2412.07486.

[21] X. Zhao, H. Yu, and H. Bian, "Image-to-image translation based on differential image pix2pix model," *Computers, Materials & Continua*, vol. 77, no. 1, pp. 182–197, 2023. DOI: 10.32604/cmc.2023.041479.

[22] Y. Gong et al., "E2gan: Efficient training of efficient gans for image-to-image translation," in *Proceedings of the 41st International Conference on Machine Learning (ICML)*, Honolulu, HI, USA, May 2024. [Online]. Available: https://openreview.net/forum?id=lrPrkWXqzd.

[23] T. Kim and B. Kim, "Techniques for detecting the start and end points of sign language utterances to enhance recognition performance in mobile environments," *Applied Sciences*, vol. 14, no. 20, p. 9199, 2024. DOI: 10.3390/app14209199.

[24] M. J. Raihan, M. I. Labib, A. A. J. Jim, J. J. Tiang, U. Biswas, and A.-A. Nahid, "Bengali-sign: A machine learning-based bengali sign language interpretation for deaf and non-verbal people," *Sensors*, vol. 24, no. 16, p. 5351, Aug. 2024. DOI: 10.3390/s24165351.

[25] A. R. Verma, G. Singh, B. RamJi, K. Meghwal, and P. K. Dadheech, *Enhancing sign language detection through mediapipe and convolutional neural networks (cnn)*, ResearchGate, Jun. 2024. [Online]. Available: https://www.researchgate.net/publication/381190776.