Ramakrishna Mission Residential College
Narendrapur, Kolkata-700103
An Autonomous College affiliated to the University of Calcutta

INTERNSHIP REPORT 2025

On

Computer Vision

Submitted by
Jisnoo Dev Pal
College Roll Number: CSUG/079/23
Registration Number: 4R2ACMSA2005
Programme: B.Sc. Honours [4 Years]
Semester: V

© **All Rights Reserved**

Internship Organization: Ramakrishna Mission Vivekananda Educational and Research Institute (RKMVERI)
Internship Duration: **June 2025**
Total Hours Completed: **100+** Hours
Date of Submission:

Name of the Internship Supervisor: Swami Punyeshwarananda

Affiliation of the Supervisor: HOD of Computer Science, RKMVERI, Belur

Remarks on the Internship Report by the Internship Supervisor::

Jisnoo Dev Pal has worked on three distinct problems of increasing complexity during his internship. He has worked on an image collager, digit recognition and 3D object classification based on point cloud data. Working on these problems has given him good exposure to developing computer vision and deep learning based applications. He has worked diligently on these projects and in the process has gained solid knowledge and skills. Overall, he has demonstrated strong potential for further academic development and scholarly pursuits.

Signature of the Supervisor
with official seal:

Signature of the Student

Head, Dept. of Computer Science
Ramakrishna Mission Vivekananda Educational and Research Institute
PO Belur Math, Dist Howrah, West Bengal 711202, India

Ramakrishna Mission Vivekananda Educational and Research Institute
(Deemed-to-be-University as declared by the Ministry of Education, Government of India under Section 3 of UGC Act, 1956)
Accredited by NAAC with A++ Grade in the 1st cycle
Department of Computer Science
PO Belur Math, Howrah, West Bengal - 711202

Ref.no. CSD-INT/003/2025

5 July, 2025

CERTIFICATE OF INTERNSHIP

This is to certify that **Shriman Jisnoo Dev Pal,** a student of Ramakrishna Mission Residential College, Narendrapur, Kolkata-700103, has successfully completed an internship programme in the Department of Computer Science at the Ramakrishna Mission Vivekananda Educational and Research Institute, Belur during the month of June, 2025, completing over 100 hours of work.

Sincerely,

Punyeshwarananda

Swami Punyeshwarananda

Head, Dept. of Computer Science
Ramakrishna Mission Vivekananda Educational and Research Institute
PO Belur Math, Dist Howrah, West Bengal 711202, India

1 Declaration

I hereby declare that this internship work on Computer Vision submitted to Ramakrishna Mission Vivekananda Educational and Research Institute (RKMVERI) is a record of my original work carried out under the supervision of Sw. Punyeshwarananda, Head of Department, Department of Computer Science, RKMVERI. I further declare that the work presented has not been submitted earlier for any degree or diploma in any other university or institution, and that there are no conflicts of interest in the conduct or reporting of this research work. All sources of information and materials used have been duly acknowledged through proper citations and references.

2 Acknowledgement

This one-month research internship at RKMVERI has been an extremely rewarding experience. I consider myself fortunate to have had the opportunity to learn from brilliant individuals—both within academia and from online resources. I would like to express my heartfelt gratitude to my advisors, teachers, collaborators, and friends whose support was invaluable throughout this journey.

First and foremost, I am fortunate to pursue my summer internship under the guidance of Swami Punyeshwarananda. I sincerely thank Maharaj for his insightful suggestion on the collage project, which improved the quality of the collaged images.

I would also like to thank my brother, Jimut Bahan Pal, for his guidance throughout my internship. His mentorship was instrumental in my professional development, as he consistently provided essential resources for learning and challenged me to exceed my capabilities in achieving project objectives. Given the substantial GPU computational requirements for training 3D data in batches, his generous provision of GPU resources was crucial to the successful completion of my model training and overall project outcomes. His strategic recommendation to pursue 3D computer vision during my internship has had a profound impact on my academic trajectory. I am immensely thankful for this direction, as working in this domain has not only expanded my technical expertise but also ignited a lasting interest in 3D Computer Vision that I plan to continue exploring in my future research.

I extend my gratitude to my college for introducing this internship opportunity in the third year of our curriculum, enabling us to gain early exposure to research and practical work.

Finally, I would like to thank my parents for their continuous and undaunted support, patience, and encouragement throughout the internship as I worked from home.

Table of Contents

3 Introduction

My prime objective in this internship is to develop expertise in 3D object classification, a critical task within the rapidly evolving field of 3D Computer Vision. This domain gained significant momentum following the groundbreaking publication of **PointNet: Deep Learning on Point Sets for 3D Classification and Segmentation**[2] seven years ago.

The growing importance of 3D Deep Learning stems from several converging factors: the exponential increase in available data, the proliferation of autonomous vehicles, and the expanding applications of augmented reality technologies. These developments have created an urgent demand for sophisticated 3D understanding capabilities. Current research in 3D Computer Vision focuses heavily on real-world applications, with robotics serving as a prime example where spatial understanding and object recognition are fundamental requirements.

In addition to my core work in 3D classification, I aim to explore complementary areas in image processing. Specifically, I plan to develop a user-friendly application that performs innovative image manipulations, making advanced image processing techniques accessible to a broader audience.

4 Objectives

1. **Collager:** To turn images and videos into dataset mosaics
2. **MNIST Classification:** 2D Handwritten digits classification using Deep Neural Networks
3. **PointNet:** 3D Object Classifcation

5 Work Done

5.1 Collager

It is an open source software to convert images and videos into dataset collages. When looked at the resulting image/video from a distance, one would see the shadow of the original image/video, which is replaced by nearest dataset images.

5.1.1 Milestones

- Dataset used are ImageNet10, MNIST, SVHN, Anime Image Dataset. Processed the datasets to pickled versions for better memory efficiency.

- Process the image or video frames to be broken into grids as specified by the user
- Replace these patches with the nearest neighbour of the patch from the dataset.
- A Gradio UI to facilitate naive users
- Deployment into Github to make it open source

5.1.2 Algorithm

The algorithm for collage is below.

Algorithm 1 Collager Algorithm

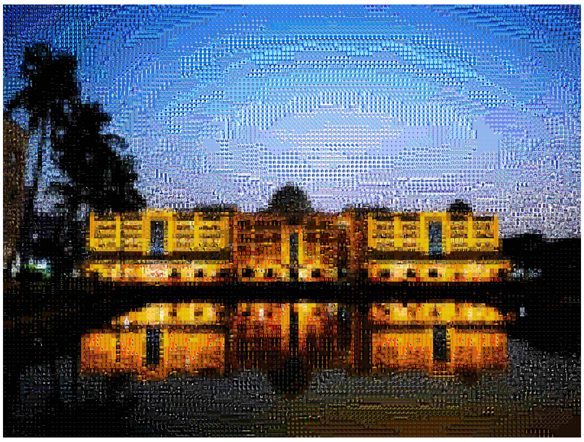
procedure COLLAGER(img, $dataset$, m, n)
  Process the dataset
  $rb\_dataset \leftarrow \emptyset$ ▷ Will contain mean pixel values of each channel of the processed dataset
  for $i = 1$ **to** $|dataset|$ **do**
    $resized\_dataset\_img \leftarrow \text{RESIZE}(dataset[i], 10\text{px}, 10\text{px})$
    $blurred\_dataset\_img \leftarrow \text{GAUSSIAN\_BLUR}(resized\_dataset\_img, \text{kernel\_size} = 3)$
    $rb\_dataset \leftarrow rb\_dataset \cup \{\text{MEAN}(blurred\_dataset\_img)\}$
  end for
  Process the original image
  $blur\_big\_img \leftarrow \text{GAUSSIAN\_BLUR}(img, \text{kernel\_size} = 1)$ ▷ Initial Blurring
  $grids \leftarrow \text{BREAK\_INTO\_GRIDS}(blur\_big\_img)$ ▷ Break into $m \times n$ grids and store in list
  Core computation
  for each $grid$ **in** $grids$ **do**
    $blurred\_patch \leftarrow \text{GAUSSIAN\_BLUR}(grid, \text{kernel\_size} = 3)$
    $mean\_patch \leftarrow \text{MEAN}(blurred\_patch)$
    $dataset\_img\_index \leftarrow \arg\min_i(mean\_patch - rb\_dataset[i])$
    $Dataset\_img \leftarrow dataset[dataset\_img\_index]$
    Replace $grid$ with $Dataset\_img$
  end for
  Save the image
end procedure

5.1.3 Results

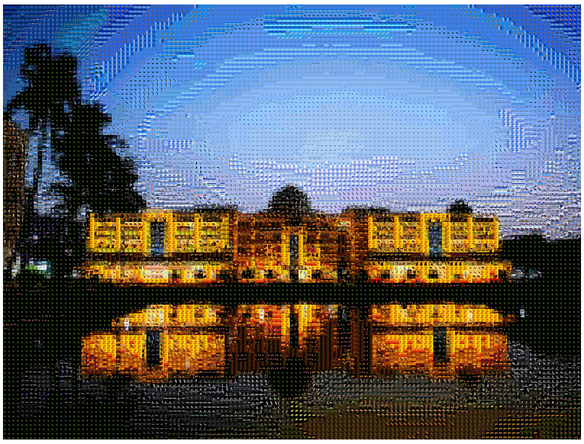
When number of grids are greater than 100, the rendered images are stunning collages of high resolution.

(a) Original Image of RKMVERI

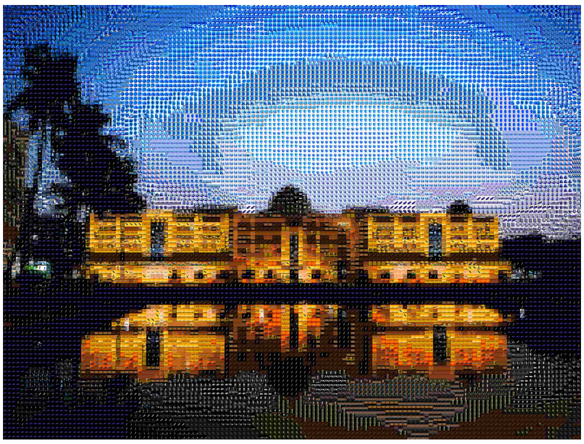
(b) Collaged with SVHN



(c) Collaged with Anime



(d) Collaged with CIFAR_10



(e) Collaged with ImageNet10

Figure 1: Results of collage on a beautiful image of RKMVERI

5.1.4 User Interface

A UI made out of gradio to generate collages easily!

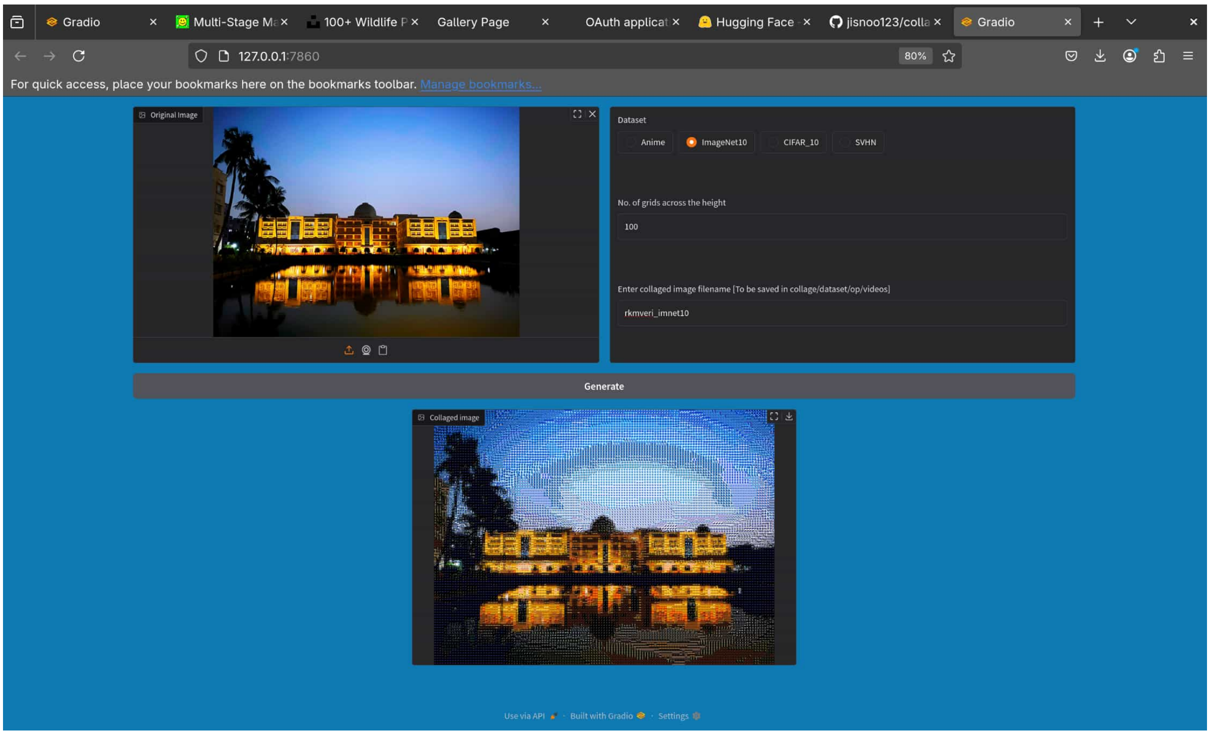


Figure 2: Gradio UI

5.1.5 Deployment

Collage can be used anytime and anywhere with Hugging Face Spaces. Collage is deployed in GitHub as an open source software. The code to download and generate the processed datasets is provided along with detailed documentation of the usage of this sofware.

5.2 MNIST Classification

The MNIST database (Modified National Institute of Standards and Technology database) is a large database of handwritten digits that is commonly used for training various image processing systems. Classification is a supervised machine learning method where the model tries to predict the correct label of a given input data. Use of Convolutional Neural Networks [1] has tremendously improved the classification task. We use CNNs to perform 2D classification.

5.2.1 Milestones

- Splitting the MNIST dataset into three fold train, validation and test sets.
- Creating the dataloader in PyTorch to process the images in batches to fully utilize GPUs.
- Build the Deep Neural Network Model to classify MNIST images.
- Code out the train, validation and test epochs.
- Test the model on the test set.
- Compute and plot the metrics commonly used for classification to see the model performance.

5.2.2 Model

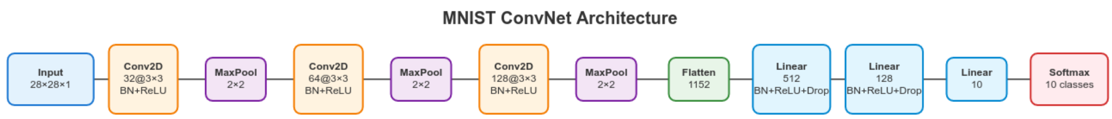
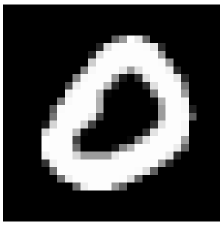
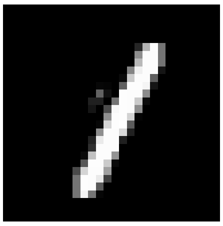


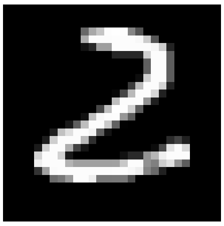
Figure 3: Model Architecture

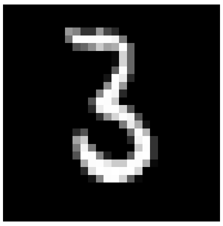
5.2.3 Results

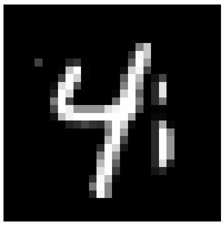
We divided the MNIST dataset into three fold train (40K images), validation (20K images) and test (10K images) dataset to work on. After training the model for 50 epochs, we achieved an accuracy, recall and precision score of 98.22% which is impressive and shows that the model is performing quite well in classifying the images. Some correctly classified and misclassified images are shown below.

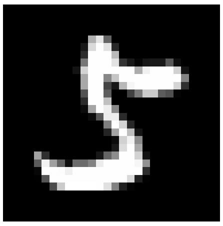

(a) GT:0 Pred:0

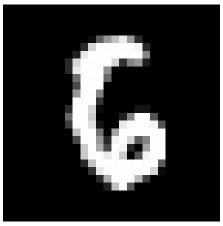

(b) GT:1 Pred:1

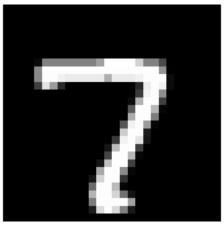

(c) GT:2 Pred:2

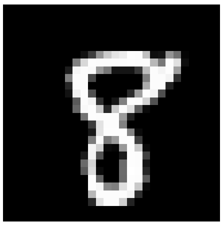

(d) GT:3 Pred:3


(e) GT:4 Pred:4


(f) GT:5 Pred:5


(g) GT:6 Pred:6


(h) GT:7 Pred:7


(i) GT:8 Pred:8

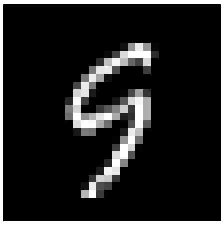
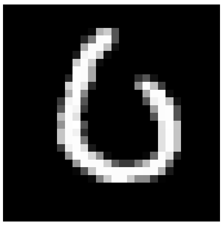
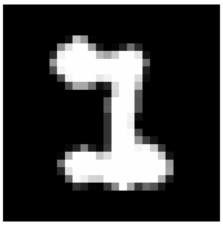
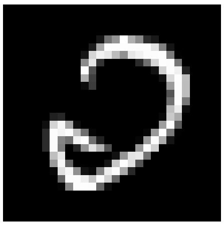
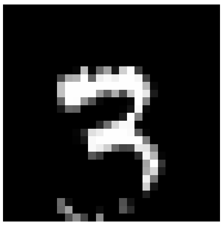

(j) GT:9 Pred:9

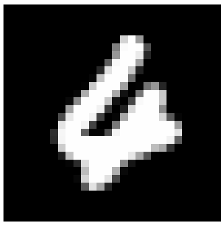
Figure 4: Some correctly classified images from each class

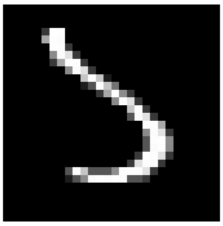

(a) GT:0 Pred:6

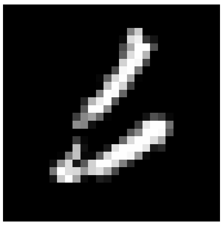

(b) GT:1 Pred:2

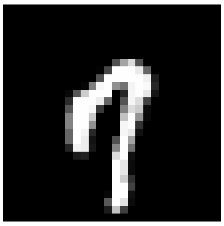

(c) GT:2 Pred:0

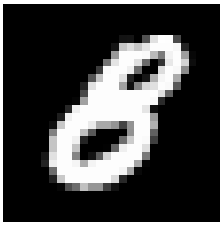

(d) GT:3 Pred:7


(e) GT:4 Pred:6


(f) GT:5 Pred:3


(g) GT:6 Pred:1


(h) GT:7 Pred:9


(i) GT:8 Pred:0

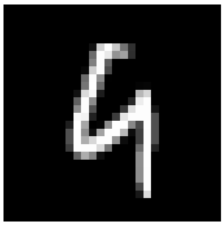

(j) GT:9 Pred:4

Figure 5: Some misclassified images from each class

5.2.4 Visualized Metrics

Figure 6: Loss plot: train vs. val

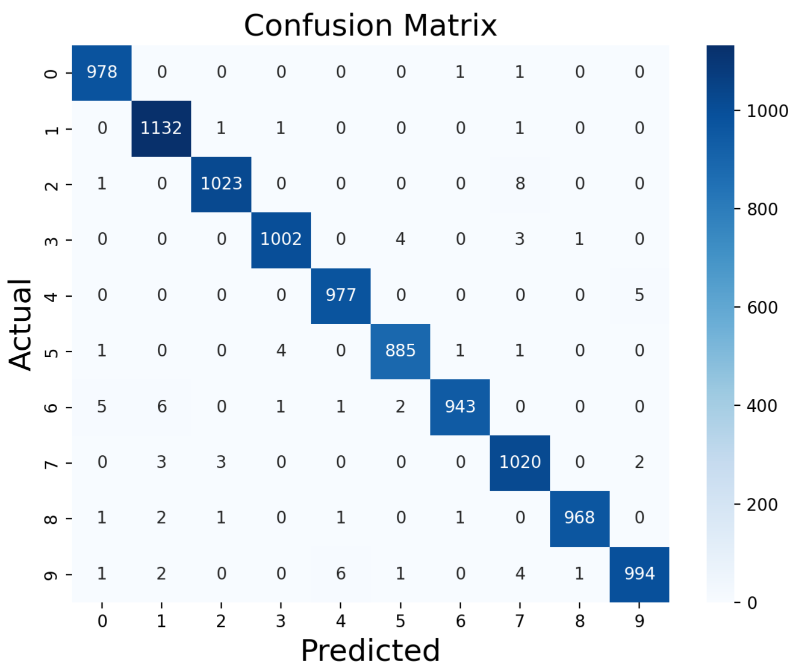


Figure 7: Confusion Matrix

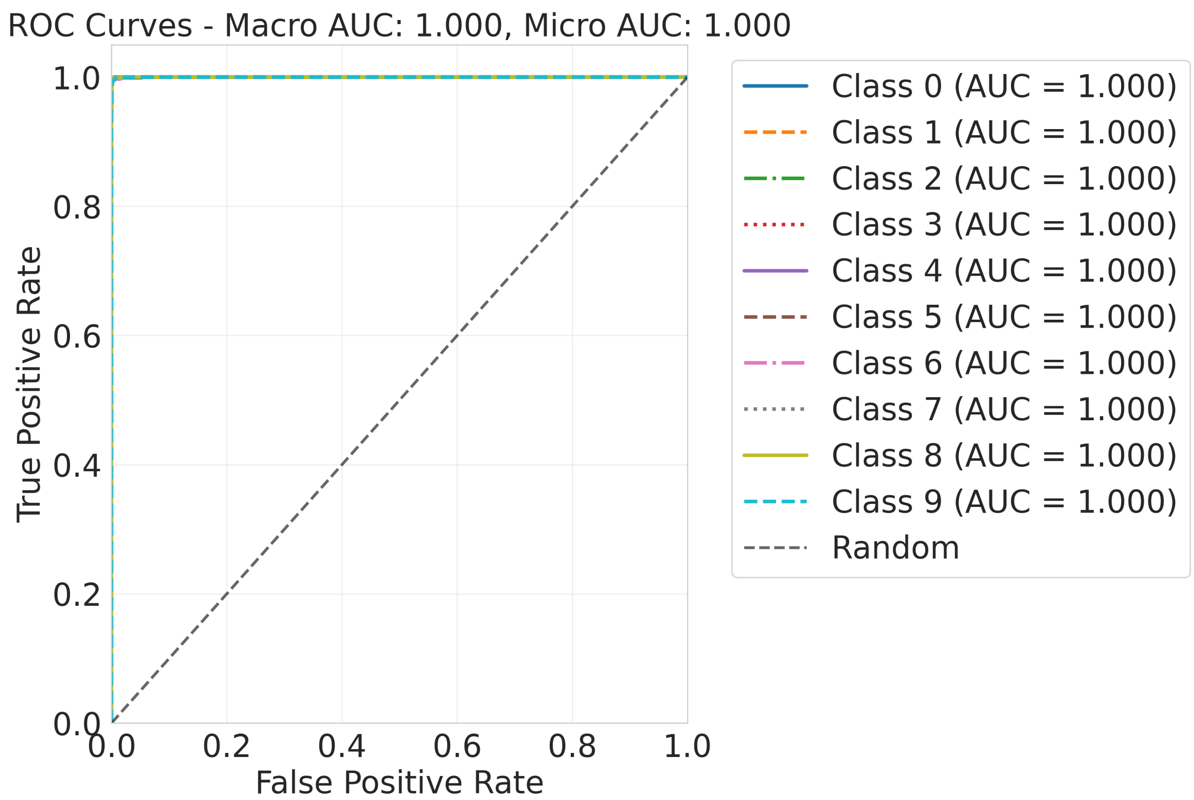


Figure 8: Multiclass ROC Curve and AUC

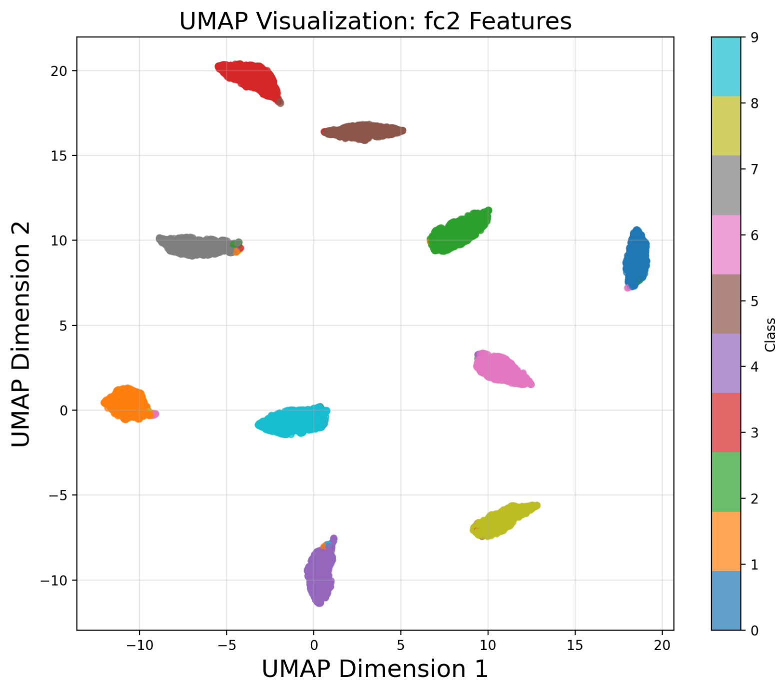


Figure 9: UMap plot of features of final FC layer

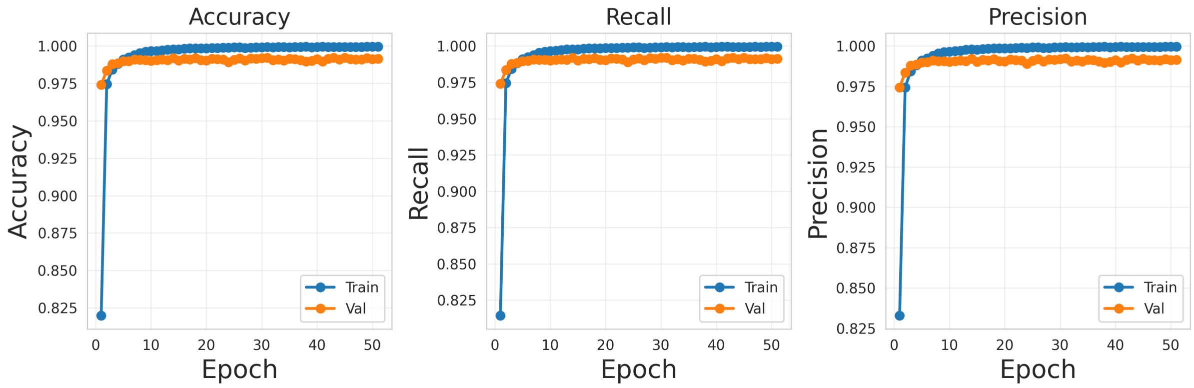


Figure 10: Accuracy, Precision and Recall plot

5.2.5 Code

My implementation of the MNIST Classification is available here on GitHub.

5.3 PointNet: 3D Classification

PointNet [2] is a Deep Neural Network that directly consumes point cloud (a representation of 3D objects) for 3D Classification, Part Segmentation and Semantic Segmentation.

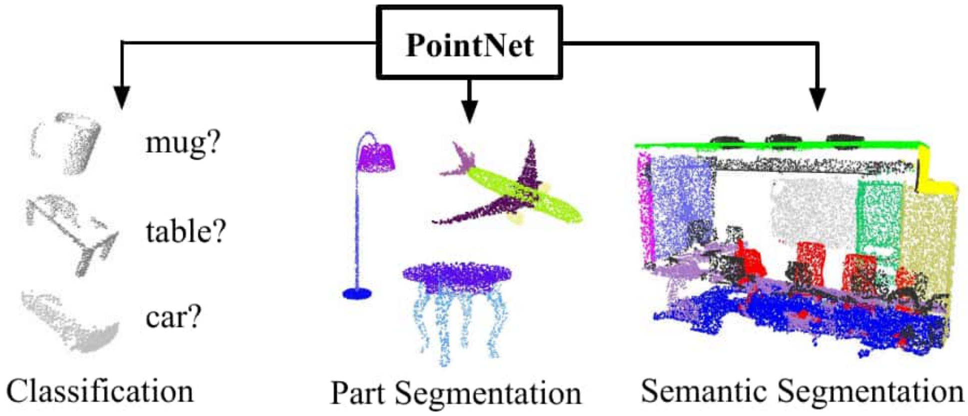


Figure 11: Applications of PointNet

5.3.1 Model

The classification network takes n points as input, applies input and feature transformations, and then aggregates point features by max pooling. The output is classification scores for k classes. The segmentation network is an extension to the classification net. It concatenates global and local features and outputs per point scores. "mlp" stands for multi-layer perceptron, numbers in bracket are layer sizes. Batchnorm is used for all layers with ReLU. Dropout layers are used for the last mlp in classification net.

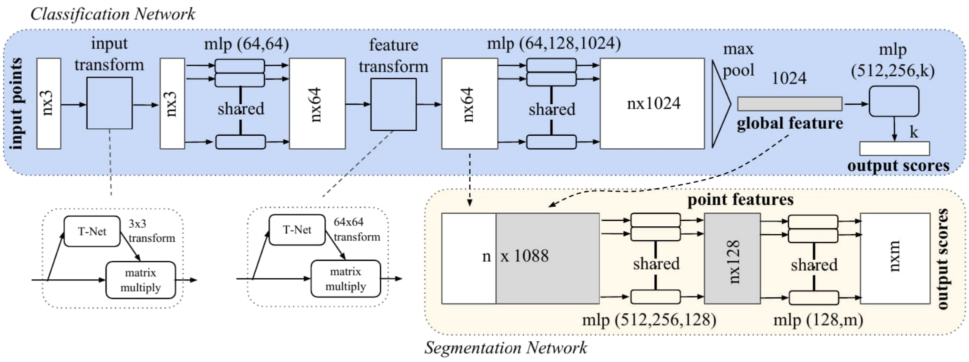


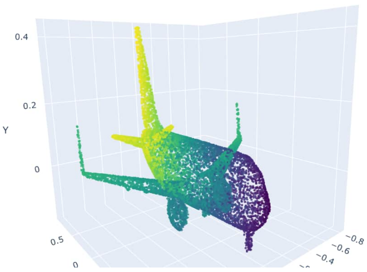
Figure 12: PointNet: Classification and Segmentation Model

5.3.2 Milestones

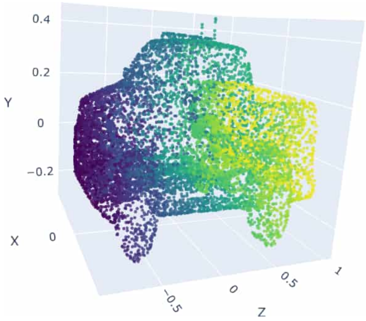
- Studying the paper on PointNet.
- Exploring the official code and PyTorch implementation of PointNet on GitHub [4].
- Fork the pre-exisiting repository for classification.
- Train and test the model.
- Compute and plot the metrics to see the model's performance.

5.3.3 Results

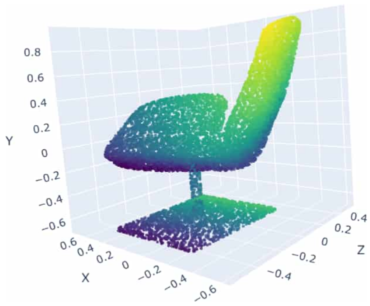
We used the ModelNet40 Dataset [3] for our classification task, the train data consists of 9843 and test consists of 2468. We used the test dataset as the validation dataset. We achieved a test accuracy, precision and recall of 91.693% after training the model for 90 epochs. Some correctly and incorrectly classified objects are shown below.



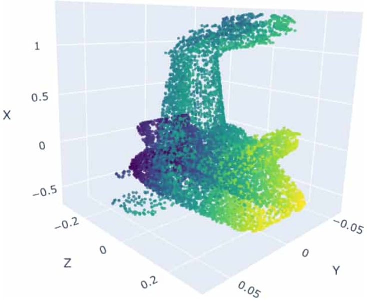
(a) GT, Pred: Airplane



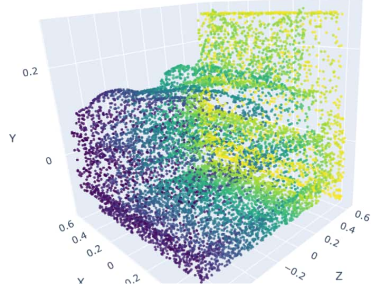
(b) GT, Pred: Car



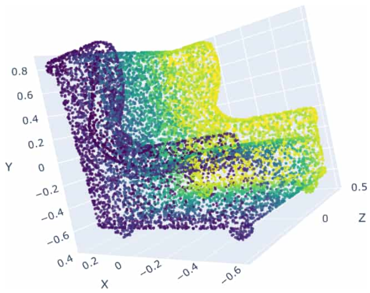
(c) GT, Pred: Chair



(d) GT, Pred: Guitar

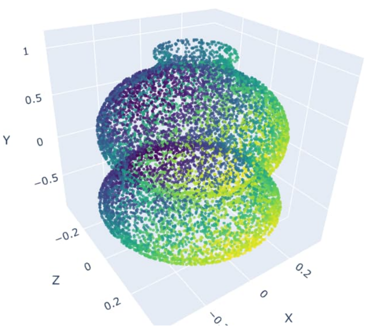


(e) GT, Pred: Bed

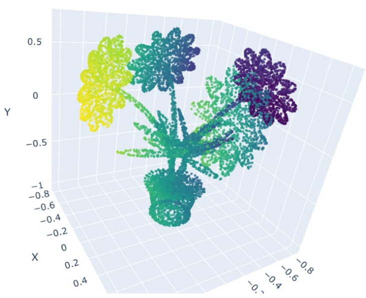


(f) GT, Pred: Sofa

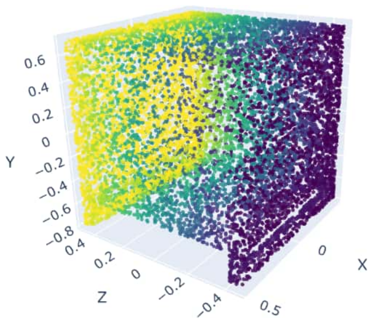
Figure 13: Some correctly classified Point Sets



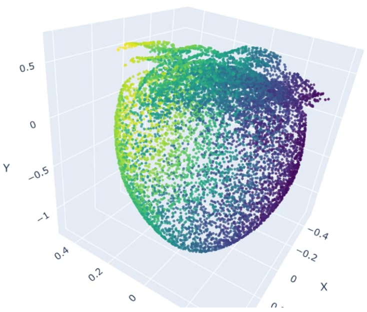
(a) GT: Bottle, Pred: Vase



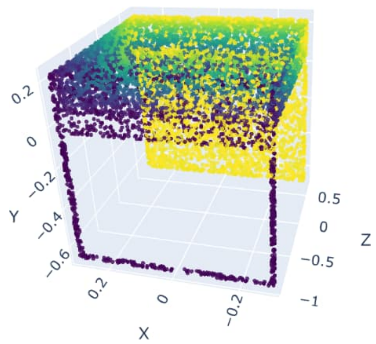
(b) GT: Flowerpot, Pred: Plant



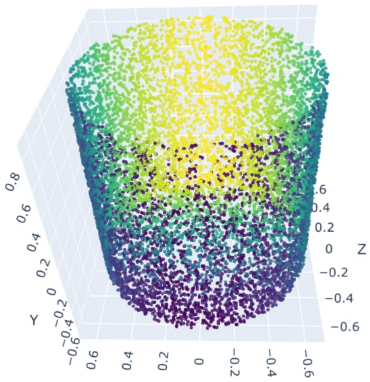
(c) GT: Night Stand, Pred: Dresser



(d) GT: Plant, Pred: Flowerpot



(e) GT: Table Pred: Desk



(f) GT: Vase, Pred: Cup

Figure 14: Some highly misclassified Point Sets

5.3.4 Visualized Metrics

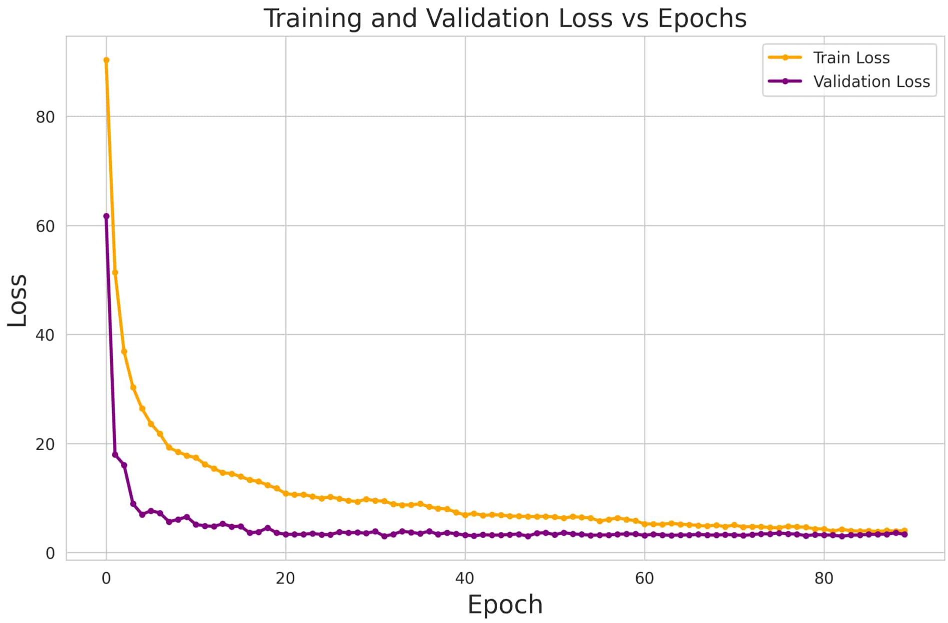


Figure 15: Loss: Train vs. Validation

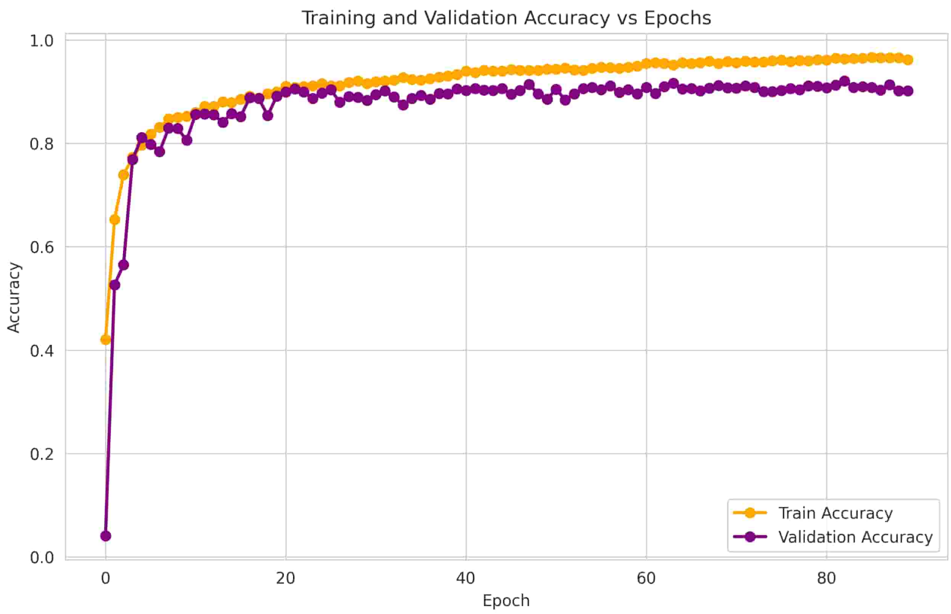


Figure 16: Accuracy

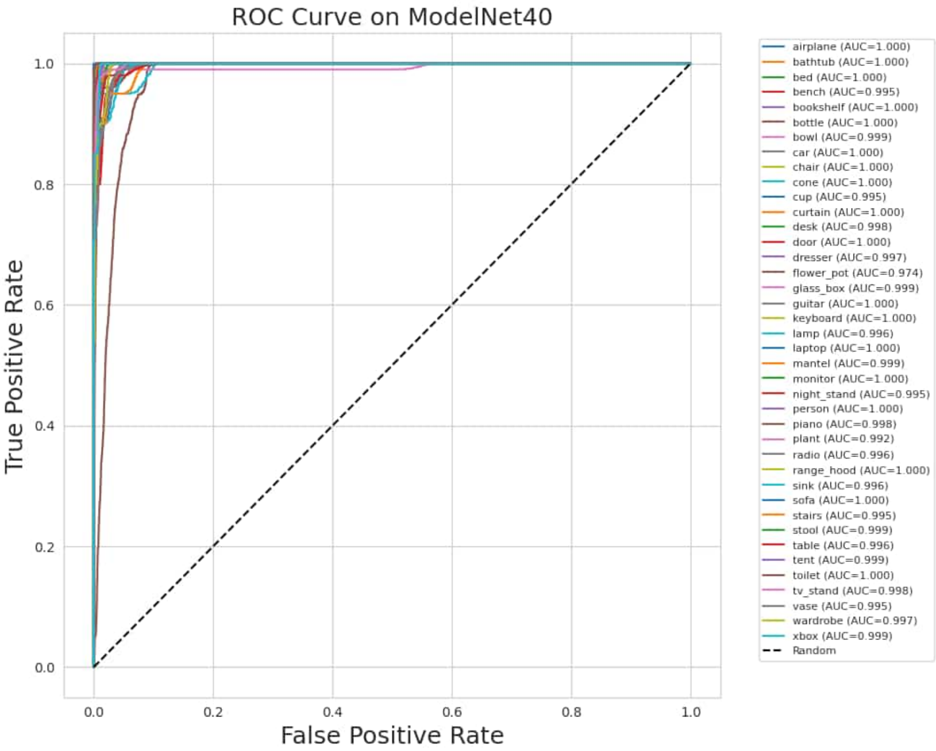


Figure 17: ROC

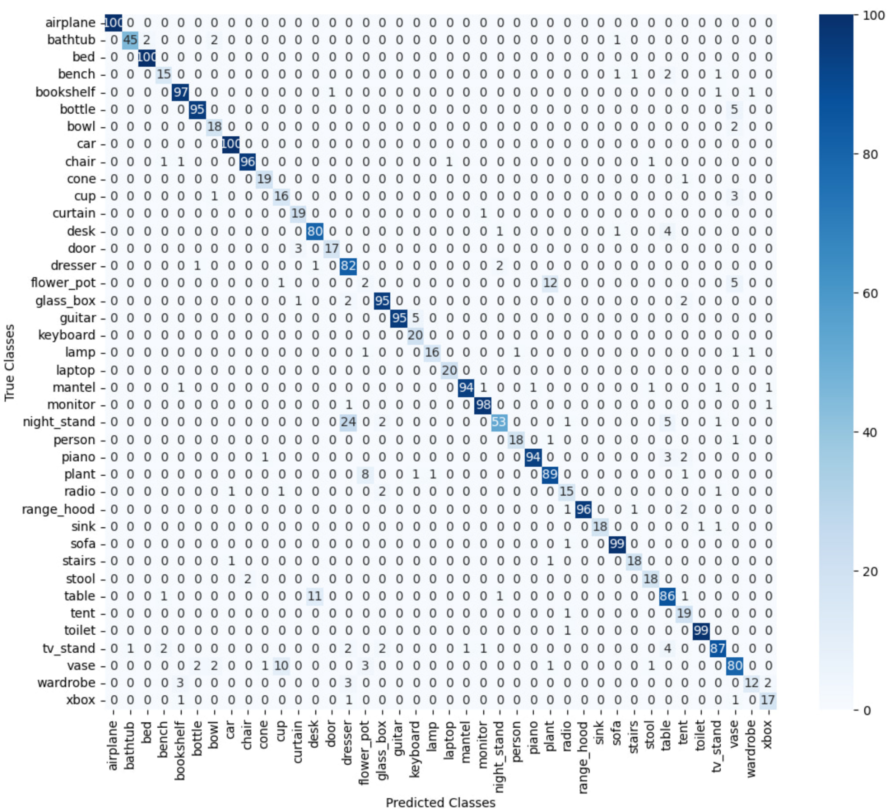


Figure 18: Confusion Matrix

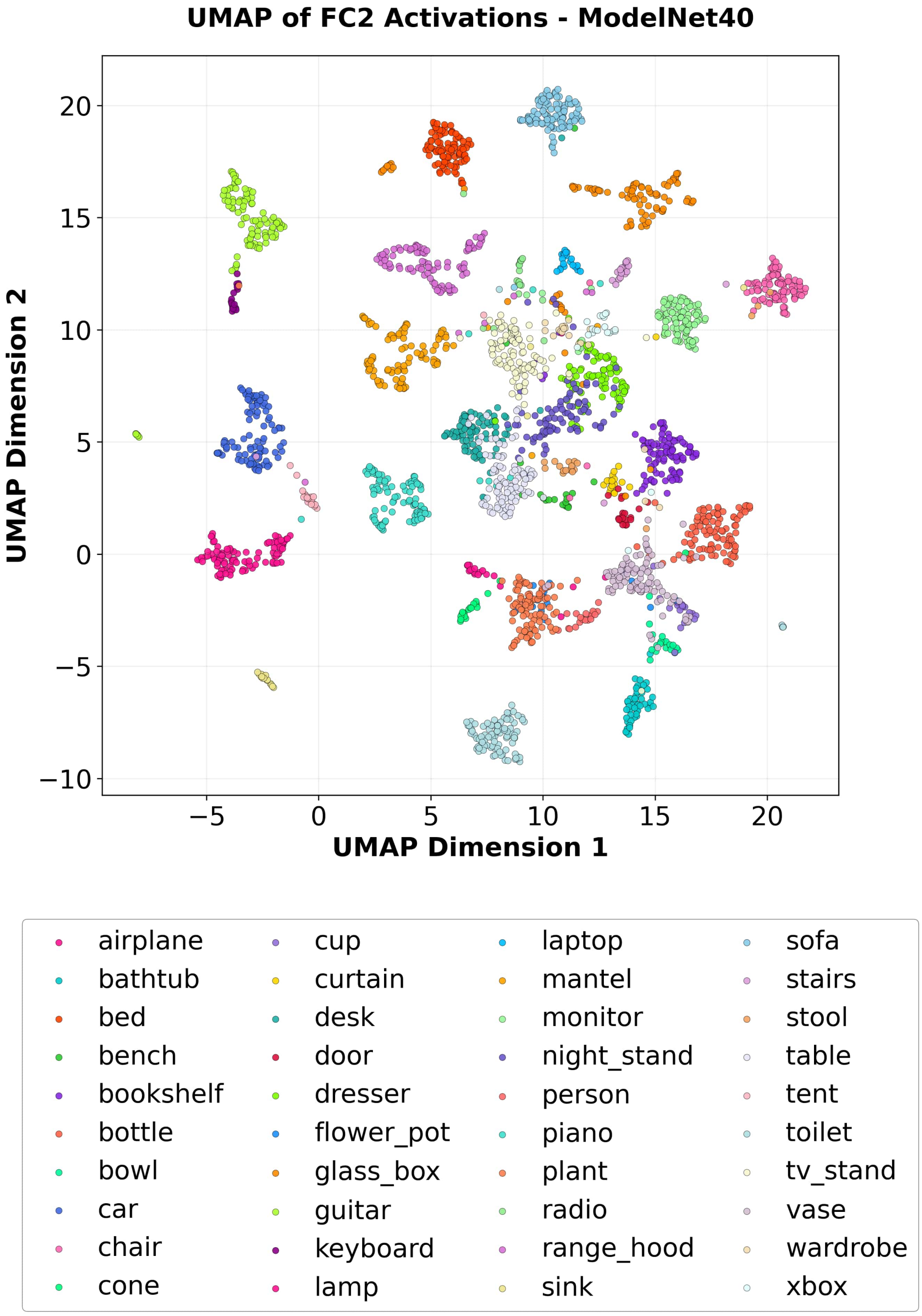


Figure 19: UMAP Plot of Fully Connected Layer 2

5.3.5 Code

My forked repository and additions made to the current pre existing repository is here

6 Key Learnings

- In my collage project, I learnt to use the OpenCV library in Python to manipulate images and videos, and the FFMPEG library to manipulate videos and audio. I also learnt how to deploy the Gradio App on Hugging Face Spaces.
- In my MNIST Classification project, I learnt PyTorch framework for Deep Learning. I got to train my model on multiple GPUs for the first time. I was introduced to the concept of processing large datasets in batches for faster training and testing. In addition to that, I learnt about the most commonly used metrics for classification to evaluate the model's performance.
- In my PointNet project, I got to know about the recent advancements in 3D Computer Vision. I learnt about the different representations of 3D objects such as PointClouds, Meshes and Voxel Grids. I learnt the architecture of PointNet which is the first of its kind to apply Deep Learning on Point Sets. I learnt about some of the mathematics behind PointNet, where the model must be robust to different spatial placements (permutation invariance of Point Sets). All these foundations on 3D Computer Vision would be of much help in the future.

7 Conclusion and Future Work

My prime interest in this internship was on 3D classification. However, I started with the basics first. I learnt and as I was progressing, I got to know what might be of great importance in 3D Computer Vision. The 3D classification is my first stepping stone to the world of 3D Computer Vision. I hope to extend my 3D Classification task to Part Segmentation and Semantic Segmentation to have a deep understanding of PointNet. I have developed quite a bit of interest in 3D, and as a research work in the future, I want to combine 3D CV with Diffusion Models.

References

[1] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60(6):84–90, 2017.

[2] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 652–660, 2017.

[3] Zhirong Wu, Shuran Song, Aditya Khosla, Fisher Yu, Linguang Zhang, Xiaoou Tang, and Jianxiong Xiao. 3d shapenets: A deep representation for volumetric shapes. In

Proceedings of the IEEE conference on computer vision and pattern recognition, pages 1912–1920, 2015.

[4] Xu Yan. Pointnet/pointnet++ pytorch.