NATIONAL UNIVERSITY OF COMPUTER AND EMERGING SCIENCES

GENERATIVE AI PROJECT REPORT

ON

PASCAL CHALLENGE ON CLASSIFICATION

SUBMITTED BY

MAHAD HAMEED
20K-0338

ALISHBA SUBHANI
20K-0351
MANNAHIL MIFTAH
20K-0234

SUBMITTED TO

Dr. Muhammad Atif Tahir

Contents

0.1 PROBLEM STATEMENT

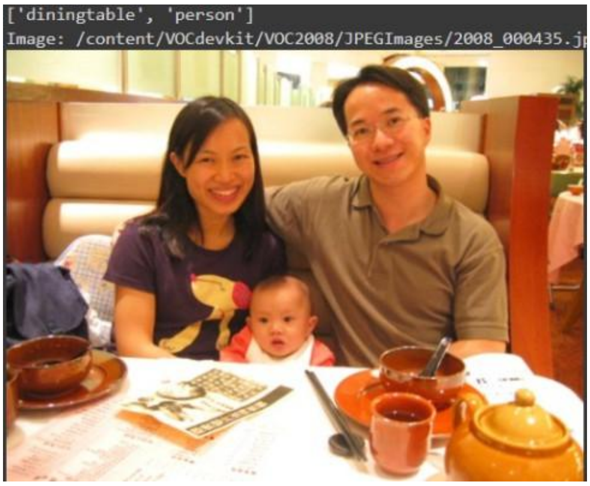
The Pascal VOC2008 challenge focuses on the task of object detection and classification. The problem is to classify the objects in the dataset of images into the following twenty classes.

1. person, bird, cat, cow, dog, horse, sheep, aeroplane, bicycle, boat
2. bus, car, motorbike, train, bottle, chair, dining table, potted plant, sofa, tv/monitor

0.2 DATA

The VOC2008 database contains 10,057 annotated images, with 2111 samples in the training set and 2221 samples in the validation set. Each image has an annotation file containing labels for multiple objects.

Following is an example of an image with the labels of ‘dining table’, and ‘person’ signifying their presence in the image.



```
['diningtable', 'person']
Image: /content/VOCdevkit/VOC2008/JPEGImages/2008_000435.jp
```

Figure 1: Sample Image

0.3 EVALUATION

A precision/recall (PR) curve evaluates the trade-off between precision and recall, with the average precision (AP) being the principal quantitative measure. AP summarizes the PR curve as the area under it, commonly used in image classification and detection tasks.

0.4 CLASSIFICATION

0.4.1 Pseudocode

```
Download and extract PASCAL VOC 2008 dataset
Identify labels
Read training and validation file lists
Define function to extract labels from XML annotations
Define VGG16 model with pre-trained weights, add layers, and compile
Define data generator for training and validation
Iterate over each label, create and train model, save trained model
```

0.4.2 Precision Recall Curve

For each class, a binary VGG model was trained.For each class, the Precision-Recall curve, and average precision is given below:

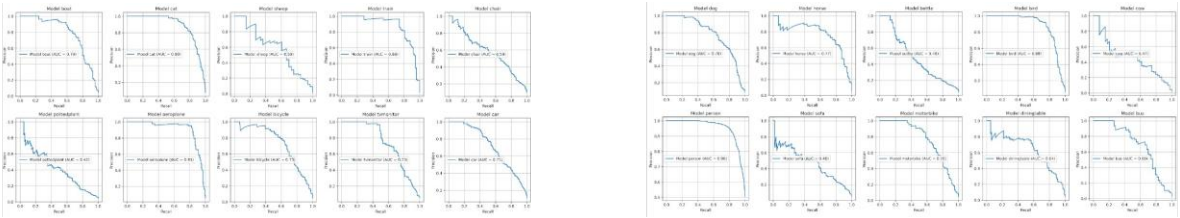


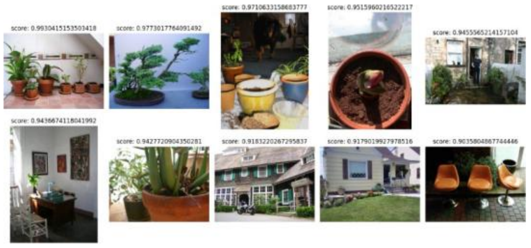
Figure 2: Precision-Recall Curve for each class

0.4.3 Highest Ranked Images

Best & Worst Model

The highest performing model, for the ”Person” class with higher support, achieved an AP of 0.96. Conversely, the lowest performing model, for the ”Potted Plant” class with low support, achieved an AP of 0.42.

(a) Class Person


score: 0.9930415153503418
score: 0.9773017764091492
score: 0.9710633158683777
score: 0.9515960216522217
score: 0.9455565214157104
score: 0.9436674118041992
score: 0.9427720904350281
score: 0.9183220267295837
score: 0.9179019927978516
score: 0.9035804867744446

(b) Class Potted Plant

0.4.4 Mean Average Precision

The mean average precision is 0.706. There is a clear correlation between class support and average precision: classes with higher support, like ”Person” with 1022 examples, achieve better AP, whereas classes with lower support, like ”Sheep,” ”Cow,” and ”Potted Plant” with less than 100 examples, show lower AP.

0.5 DATA AUGMENTATION

The hypothesis is that data augmentation, through both Variational Autoencoder (VAE) and Deep Convolutional Generative Adversarial Network (DCGAN), can improve model performance for classes with low support by generating more samples and mitigating class imbalance.

0.6 VARIATIONAL AUTOENCODERS

0.6.1 Pseudocode & Training Process

```
Define VAE model class
Create dictionaries for models, optimizers, and statistics per label
Iterate over labels and train loaders
Initialize model, optimizer, and stats
Iterate over epochs, batches, compute loss, update model
Print label, epoch, and average loss
```

Training
During the training process of the Variational Autoencoder (VAE), the model progressively improved in reconstructing the displayed images over time, showing evidence of learning.

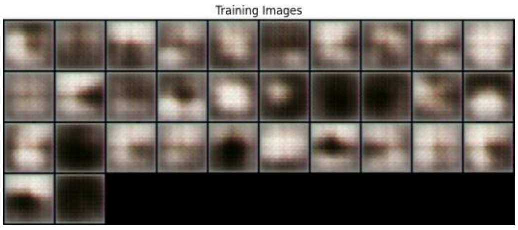
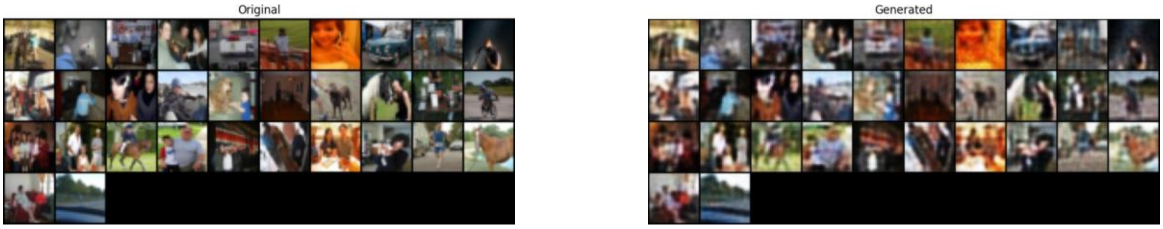


Figure 5: Images during training process

0.6.2 Results

Showing results of VAE for person class. Total 32 images have been reconstructed for each class.



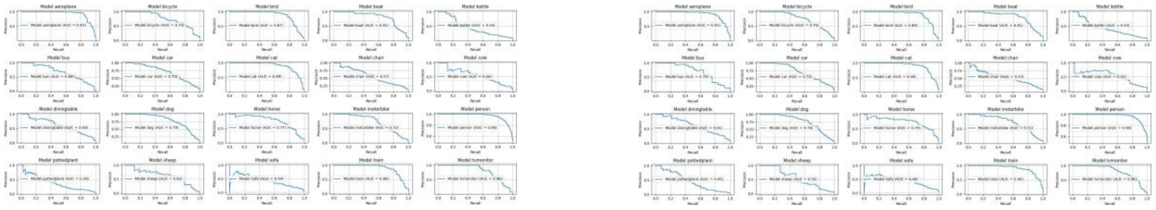
(a) Original Images **(b)** Reconstructed Images

Figure 6: VAE Results

0.6.3 Classification

Before training each class-specific binary VGG model, reconstructed images were augmented to the original training set for that class. Experiments were conducted with augmenting 10, 25, and 32 samples, respectively, and Precision-Recall curves and average precision were analyzed.

10 & 25 samples per class



(a) Precision-Recall Curve of 10 samples **(b)** Precision-Recall Curve of 25 samples

32 samples per class

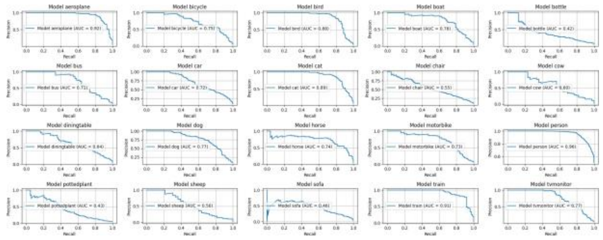


Figure 8: Precision-Recall Curve of 32 samples

0.6.4 Mean Average Precision

10, 25 & 32 samples per class

For 10 samples, the mean average precision is 0.70854, with the ”Person” class achieving the highest AP of 0.96, and the ”Potted Plant” class the lowest at 0.38. For 25 samples, the mAP is 0.70596, with the ”Person” class still having the highest AP of 0.96, and the ”Bottle” class the lowest at 0.43. For 32 samples, the mAP is 0.70975, with the ”Person” class maintaining the highest AP of 0.96, and the ”Bottle” class the lowest at 0.42.

Analysis

The initial VGG training achieved an mAP of 0.706 on Pascal VOC dataset. Augmenting with 10 reconstructed images per class improved mAP to 0.7085, but increasing to 25 resulted in a minor decrease to 0.7059, suggesting diminishing returns. Adding 32 reconstructed images improved mAP to 0.7097. This highlights the balance needed between augmentation and dataset quality for optimal results.

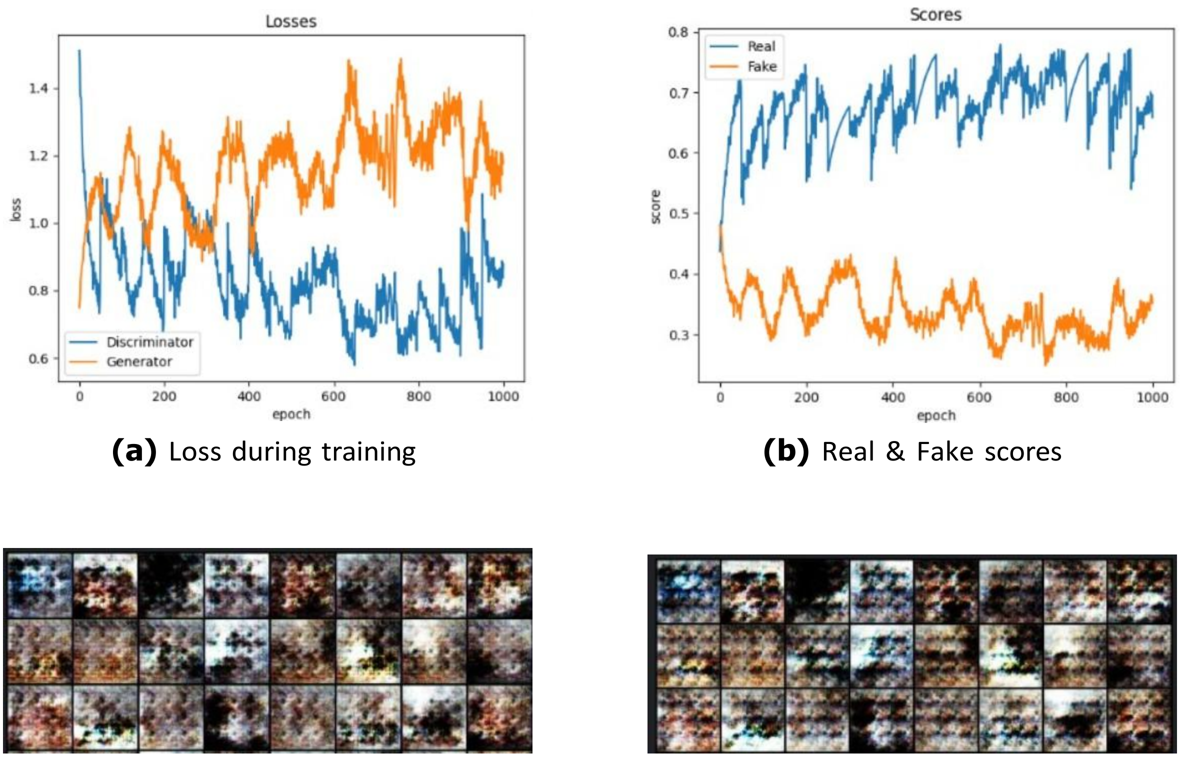
0.7 DC GENERATIVE ADVERSIAL NETWORK

0.7.1 Pseudocode

```
Define discriminator and generator architectures
Train models with Adam optimizers per class
Iterate over epochs and batches, train discriminator and generator
Save models, return losses and scores
```

0.7.2 Training Process

Initially, the model faced issues with increasing generator loss and fluctuations in losses during training for 25, 100, and 500 epochs. By training for 25 epochs and using the output as a pretrained model for subsequent runs, significant improvements were observed in image quality through 3 to 4 iterations, suggesting further improvement with more iterations.



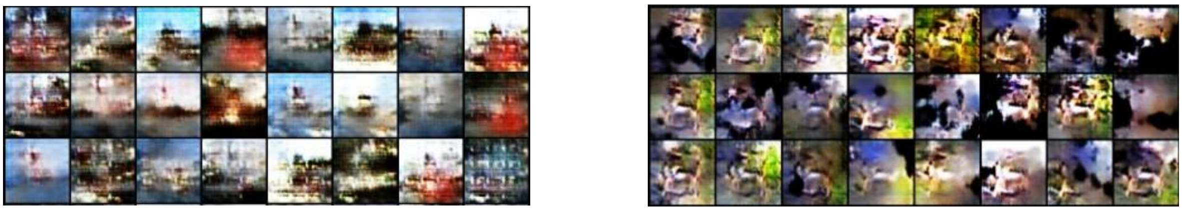
(a) Loss during training

(b) Real & Fake scores

Figure 10: Images during training process

0.7.3 Results

Showing results of DCGAN for person class. Total 25 images have been reconstructed for each class.



(a) Reconstructed Images of Class Boat

(b) Reconstructed Images of Class Dog

Figure 11: DCGAN Results

0.7.4 Classification

Before training each class-specific binary VGG model, reconstructed images were added to the original training set for that class, augmenting with 25 samples. Precision-Recall curve and average precision are presented below.

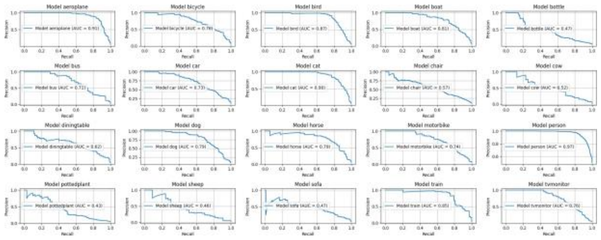


Figure 12: Precision-Recall Curve of 25 samples

0.7.5 Mean Average Precision

25 samples per class

The mean average precision is 0.70619. The person class has the best AP of 0.91, whereas, the potted plant has the lowest AP of 0.43.

Analysis

The consistent mAP value of 0.70619 after adding DCGAN-generated images suggests that while DCGAN helped address class imbalance, the image quality may not have been sufficient for significant classification improvement. Despite adding 25 sample images per class, noise likely caused confusion, resulting in minimal mAP improvement. Further refinement of the DCGAN model for more realistic images is needed, requiring additional resources.

0.8 CONCLUSION

This project addressed class imbalance in a 20-class dataset by augmenting data using VAE and DCGAN. Additional images generated by VAE and DCGAN were incorporated to enhance classification performance, despite containing noise.

0.9 REFERENCES

Goodfellow, Ian, et al. ”Generative adversarial nets.” In Proceedings of the 27th International Conference on Neural Information Processing Systems, pp. 2672-2680. 2014.
Kingma, Diederik P., and Max Welling. ”Auto-encoding variational bayes.” In Proceedings of the 2nd International Conference on Learning Representations, San Diego, CA, USA, 2014.
Radford, Alec, Luke Metz, and Soumith Chintala. ”Unsupervised representation learning with deep convolutional generative adversarial networks.” Proceedings of the 4th International Conference on Learning Representations (ICLR), San Juan, Puerto Rico, 2016.