Science and Technology Council, Indian Institute of Technology Kanpur

Model Zoo

Project Report

July 18, 2020

Contents

1 Introduction

Deep Learning is a subfield of machine learning concerned with algorithms inspired by the structure and function of the brain called artificial neural networks.Deep learning excels on problem domains where the inputs (and even output) are analog. Meaning, they are not a few quantities in a tabular format but instead are images of pixel data, documents of text data or files of audio data.In deep learning, each level learns to transform its input data into a slightly more abstract and composite representation. In an image recognition application, the raw input may be a matrix of pixels; the first representational layer may abstract the pixels and encode edges; the second layer may compose and encode arrangements of edges; the third layer may encode a nose and eyes; and the fourth layer may recognize that the image contains a face. Importantly, a deep learning process can learn which features to optimally place in which level on its own. (Of course, this does not completely eliminate the need for hand-tuning; for example, varying numbers of layers and layer sizes can provide different degrees of abstraction.)

This project focuses on making a collection of Deep Learning models that can be applied to various fields ranging from image classification to auto completion of sentences.

Figure 1 below demonstrates a simple Neural Network.

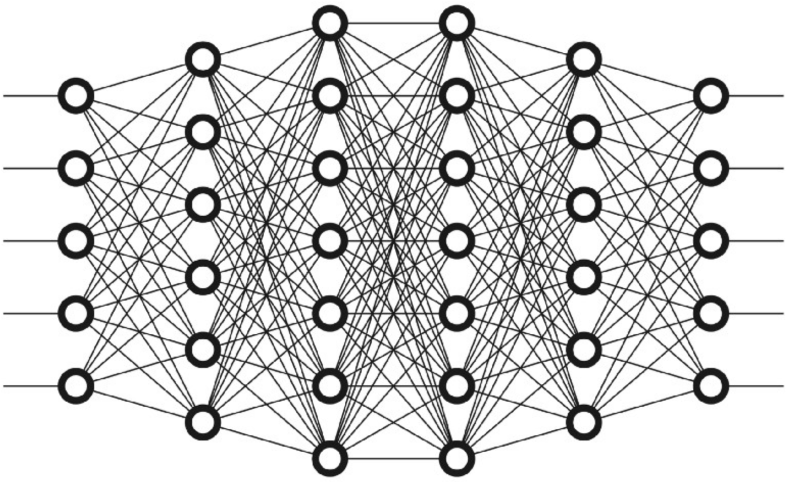


Figure 1: A simple neural network

2 Overview

We implemented 31 models in the Model-Zoo spanning many categories. We used both PyTorch and Tensorflow which are the most popular Deep Learning frameworks for implementation of the models. The project repository can be found on GitHub (`https://github.com/pclubiitk/model-zoo`). Each model is in a separate directory with a README including the usage and architecture details. All the models are additionally listed in the tables below :

Model	**PyTorch**	**Tensorflow**
Generative Models		
AAE	✓	✗
AC-GAN	✓	✗
CycleGAN	✓	✗
DCGAN	✓	✓
DiscoGAN	✓	✗
InfoGAN	✓	✓
MoCoGAN	✓	✗
StarGAN	✓	✗
VAEGAN	✓	✗
Vanilla GAN	✓	✓
WGAN	✗	✓
cGAN	✓	✓
SS-GAN	✗	✓
Natural Language Processing		
BERT	✓	✓
Bi-LSTM	✗	✓
GloVe	✓	✓
Transformer	✗	✓
Word2Vec	✓	✗

Model	**PyTorch**	**Tensorflow**
Classification		
ResNet	✓	✗
T3D ConvNets	✗	✓
Multimodal Models		
StackGAN	✓	✗
VQA	✗	✓
Image Captioning	✓	✓
Audio Generation		
WaveGAN	✗	✓
Object Detection		
YOLOv2	✗	✓
YOLOv3	✗	✓
Super Resolution		
SRCNN	✗	✓
SRGAN	✓	✓
VDSR	✓	✗
Image Inpainting		
ContextEncoder	✓	✓
3D Vision		
3DGAN	✓	✗

3 Categories

3.1 Generative Models

Generative modeling is an unsupervised learning task in machine learning that involves automatically discovering and learning the regularities or patterns in input data in such a way that the model can be used to generate or output new examples that plausibly could have been drawn from the original dataset.

GANs are a clever way of training a generative model by framing the problem as a supervised learning problem with two sub-models: the generator model that we train to generate new examples, and the discriminator model that tries to classify examples as either real (from the domain) or fake (generated). We have implemented GANs that use a vareity of techniques to acheive this task, such as :

- **Minimax GANs**
  This technique uses a game theoretic approach to train both the discriminator and the generator. The discriminator and generator play a two-player, zero-sum, minimax game, which ends if a Nash equilibrium is achieved. Sometimes, extra conditioning can be provided to the inputs to the generator,to provide outputs which are targeted. Models like **AC-GAN**, **InfoGAN**, **DCGAN**, **cGAN**, **Vanilla GAN** fall in this category.

- **Encoder-Decoder GANs**
  One of the main drawbacks of variational autoencoders is that the integral of the KL divergence term does not have a closed form analytical solution except for a handful of distributions. Adversarial autoencoders avoid using the KL divergence altogether by using adversarial learning. In this architecture, a new network is trained to discriminatively predict whether a sample comes from the hidden code of the autoencoder or from the prior distribution $p(z)$ determined by the user. The loss of the encoder is now composed by the reconstruction loss plus the loss given by the discriminator network. Models like **VAE-GAN**, **AAE-GAN** fall in this category.

- **Image to Image translation** These models give us the ability to learn the mapping between one image domain (say A and B) and another using an unsupervised approach. These often include the usage of two generators and two discriminators, one pair used for learning the mapping from domain A to domain B, and the other for learning the mapping from B to A. Models like **Cycle GAN** and **Disco GAN** fall in this domain.

Figure 2 shows a result from CycleGAN [1].

3.2 3D Vision

To address the problem of 3D object generation, a 3D Generative Adversarial Network (**3DGAN**) likewise to 2D GANs , generates 3D objects from a Probabilistic Latent Space of Object Shapes by leveraging volumetric convolutional networks .
The use of an adversarial criterion enables the generator to capture object structure implicitly and to synthesize high-quality 3D objects , and the discriminator to provide a powerful 3D shape descriptor which, learned without supervision, has wide applications in 3D object recognition
The algorithm follows the use of minimax function and binary cross entropy loss for adverserial classification and the Architecture uses 3D Convolutions and Transpose 3D Convolutions .

Figure 3 shows objects synthesized by 3D-GAN.

[1] All the Figures are in the Results section, image numbers are hyperlinks.

3.3 Natural Language Processing

In this project, we studied computation in **Language Modeling** and implemented extensions of:
- skip-gram model, LSTMs, GRUs and Bidirectional RNNs for sequence tagging on temporal data .
- Word vector representations and embeddings in sentiment analysis .
- Attention mechanisms for Neural Machine Translation .

The implemented models find their application in speech recognition, text synthesis, audio synthesis, chatbots , machine translation etc.

Sequence Tagging

- **Bi-LSTM-CRF -** In sequence tagging task, we have access to both past and future input features for a given time, thus we utilize a Bi-LSTM network to memorize long distance features. The CRF model also makes use of neighbor tag information and thus By combining a Bi-LSTM and CRF network , the state transition matrix of CRF layer is able to learn the past and future tags better proving higher accuracy in predicting current tags. In terms of robusteness, the implemented model beats most other models like Conv-CRF etc.
  Figure 4 shows resulting entity tags encoded in BIO annotation scheme .

Word Vectorization

- **GOOGLE word2vec -** Word2Vec model is trained to represent each word in the text corpus in form of a vector embedding which can further be used to perform many NLP tasks . Our implementation of the Continuous Bag Of Words **(CBOW)** model utilizes the improvements suggested by Google inc. namely Hierarchical Softmax, Negative Sampling, and Subsampling of Frequent Words.

- **GloVe -** The semantics of the word2vec model are only affected by a word's local surroundings. The GloVe model improves this by using global statistics. The semantic relationships between a corpus of V words are derived from a VxV co-occurrence matrix making our implementation a powerful word vector learning tool.
  Figure 5 shows the TSNE plot for word embeddings of top 300 words.

Machine Translation

- **Transformer -** Transformer uses the attention mechanism for transforming one sequence into another one with the help of two parts (Encoder and Decoder). The striking feature is that it handles variable-sized input using stacks of self-attention layers instead of RNNs or CNNs. We translated Portuguese text to English.

  Portuguese sentence : *este é o primeiro livro que eu fiz.*
  Real English translation : *this is the first book i've ever done.*
  Predicted English translation : *this is the first book that i did .*

- **BERT -** BERT improvises on the Transformer model. This multi-layer bidirectional Transformer encoder alleviates the unidirectionality constraint by using a "masked language model" (MLM) as a pre-training objective enabling fusion of left and right context in an unlabelled text and also the "next sentence prediction" task. There are two steps in BERT framework: pre-training (on unlabeled data) and fine-tuning of pre-trained parameters. The Implementation involves code related to pretraining the BERT model from scratch which can be used for NLP tasks such as Classification, Entailment, and Similarities.

3.4 Classification

Classification has played a immense role in the field of deep learning. Deep Convolutional Neural Networks have led to a series of breakthroughs for image classification.Deep networks naturally integrate low/mid/high level features and classifiers in an end-to-end fashion. We have implemented two main models in this area :

- **ResNet** The ”levels” of features in a Deep-CNN can be enriched with the number of stacked layers(depth). The problem lies in the fact that models that are too deep have a hard time optimizing their parameters. These problems are addressed by introducing a *Deep Residual learning* framework, in which the layers are made to fit a residual mapping, which is easier to optimize. We were able to get an accuracy of 91% on the CIFAR-10 Dataset.

- **Temporal 3D ConvNets** The T3D model is used in video classification.Instead of just using 3D kernels, the authors introduce a VGG-Net inspired *Temporal Transition Layer (TTL)*. This model introduces a new temporal layer that models variable temporal convolution depths, thus allowing it to capture short, mid and long term temporal information.

Figure 6 shows the architecture of a Temporal 3D ConvNet.

3.5 Multimodal Models

Modality refers to the way in which something happens. Our experience of the world is multimodal — we see objects, hear sounds, feel the texture, smell odors, and taste flavors. In order for Artificial Intelligence to make progress in understanding it’s surroundings, it needs to be able to interpret such multimodal signals together.

Deep Learning has made the field of Multimodal Learning efficient to explore for many reseachers and as a result several multimodal models are emerging dealing with a vast variety of Modals some of them which we implemented are Image generation from text (**StackGAN**), answering questions based on a visual content (**VQA**) and Image captioning (**Deep Visual Semantic**).

- **StackGAN** – This is a **GAN** based network having 2 GANs stacked to produce Photo-Realistic Images from their text descriptions. Such a model can be very productive in the fields of editing, CAD, image augmentation and cognitive research.

- **Visual Question Answering(VQA)** – involves answering short questions about provided visual content. This model uses **Transfer Learning** to get input from 2 Modals and a **LSTM** based network to answer the asked questions. It can be highly benificial to several fields of research and medical recovery.

- **Image Captioning** – Earlier models rely on some hard-coded visual concepts and sentence templates, limiting the scope of model. Deep Visual-Semantic Alignments for Generating Image Descriptions model uses a Deep Neural Network (**DNN**) and a multimodal Recurrent Neural Network (**RNN**) architecture to take an input image and generates its description in text. This implementation would be a great help in CCTV cameras, blind aid or even search engines.

Figures 7 shows results from StackGAN and Image Captioning.

3.6 Audio Generation

Goal – Generating convincing raw audio such as speech, music, etc..

Since audio signals are generally sampled and stored at a very high temporal frequency, synthesising them requires capturing a pattern across a range of timescales. **GANs** have proven to be very effective for this task. **WaveGAN** was the first model to use a **GAN** to generate audio. In this model we use a network similar to DCGAN with small changes to make it compatible with 1D data. Like any GAN it has a Generator model and a Discriminator model, where the Generator model generates an audio sample and the Discriminator model identifies then as real/fake and it's trained using WGAN-GP loss
Audio generation by giving emotion associated with them as an input can be very helpful in various psychological researches and content creation industry.

3.7 Object Detection

Goal – Detecting and Classifying any object present in a given image.

Object detection is has applications in many areas of computer vision, including image retrieval and video surveillance.
This problem itself dates as back as 1960s. Several approaches are proposed to solve this but a very special network known as **You Only Look Once (YOLO)** makes real time object detection possible over large classes using Deep Neural Networks. So far there's been 4 versions of YOLO with the latest one(v4) released this year. We have successfully implemented v2 and v3. Some key points of them are as follows:

- **YOLOv2** – It uses a custom deep **Darknet-19** architecture of 30 layers with batch normalisation and bounds the location using logistic activation and uses softmax for class prediction.
- **YOLOv3** – It uses an even deeper **Darknet-53** network ditching softmax as it is not the most suitable choice. Instead, Independent logistic classifiers are used and binary cross-entropy loss is used and bounding boxes are predicted on 3 different scales for detection on different scales

Figure 8 shows Object Detection with YOLOv3.

3.8 Image Inpainting

Goal – Given an image with some portion of it been cut out, fill in the missing part to complete the image.

The problem has been around for quite a while and has been addressed after GANs had been introduced in 2014 through **Context Encoders**.

The architecture consists of a Generator and a Discriminator like any GAN.

- The Generator itself consists of an encoder-decoder network from a Variational Auto-Encoder (VAE). A masked image is fed into the Generator and it predicts the missing portion of the image. The reconstruction loss L_{rec} is the Mean Squared Error with the ground truth.
- The Discriminator identifies the generated images as Real/Fake. The adversarial loss L_{adv} is the simple Binary Cross Entropy (BCE) Loss.

Figure 9 shows results from a Context Encoder.

3.9 Super Resolution

Goal – Reconstructing a high resolution photo-realistic image from its counterpart low resolution image.

The very earliest solution for this task was the method of interpolation in image processing. The resultant image after a Bilinear / Bicubic interpolation is blurred and very unrealistic.
Enter Deep Learning, a preliminary application of FCNN (Fully Convolutional Neural Network) is **SRCNN**. Further improvements to SRCNN are **VDSR** and **SRGAN**.
We implemented all these architectures in our project. Here LR, HR and SR refer to the Low Resolution input image, High Resolution ground truth and Super Resolution predicted image respectively.

- **SRCNN** – The image is upsampled using bicubic interpolation and then fed into a simple FCNN. There are no pooling operations involved since the output shape is same as the input shape. The loss is computed by measuring the Mean Squared Error between the SR image and the HR image.

- **VDSR** – Very Deep Super Resolution was proposed to address some drawbacks of SRCNN. It takes inspiration from VGG-net for Classification, it is basically a deeper SRCNN.

- **SRGAN** – SRGAN is a GAN based network, where the generator learns to generates SR images from LR images as close as possible to HR. The discriminator learns to distinguish generated SR images from real images.
  SRGAN also uses ResNet instead of Simple FCNNs, it introduces a new Perceptual Loss where instead of computing pixel-wise differences we look at higher level features to get more realistic SR images.

Figure 10 shows a result from SRGAN.

4 Results

This section contains some of the results of the Models.

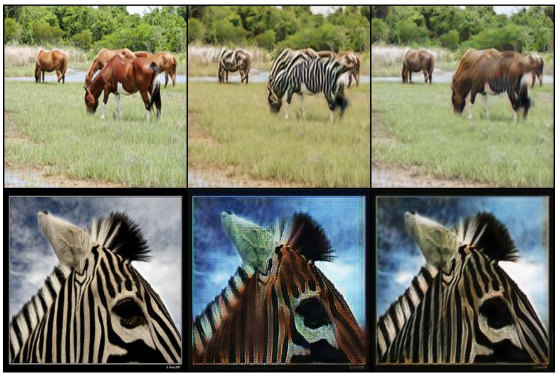


Figure 2: CycleGAN (Generative Models)

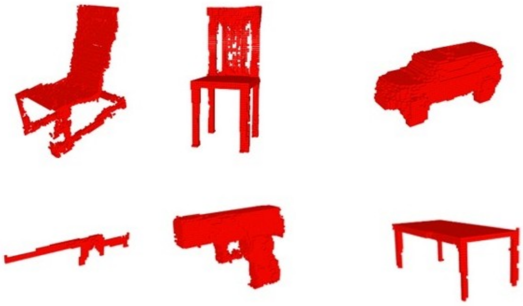
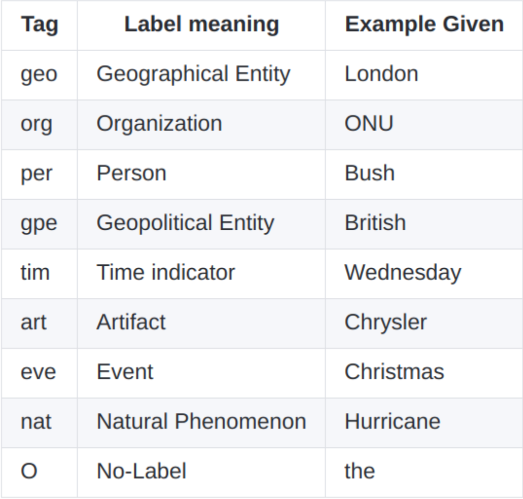


Figure 3: 3DGAN (3D Vision)



Tag	Label meaning	Example Given
geo	Geographical Entity	London
org	Organization	ONU
per	Person	Bush
gpe	Geopolitical Entity	British
tim	Time indicator	Wednesday
art	Artifact	Chrysler
eve	Event	Christmas
nat	Natural Phenomenon	Hurricane
O	No-Label	the

Figure 4: Bi-LSTM-CRF (NLP)

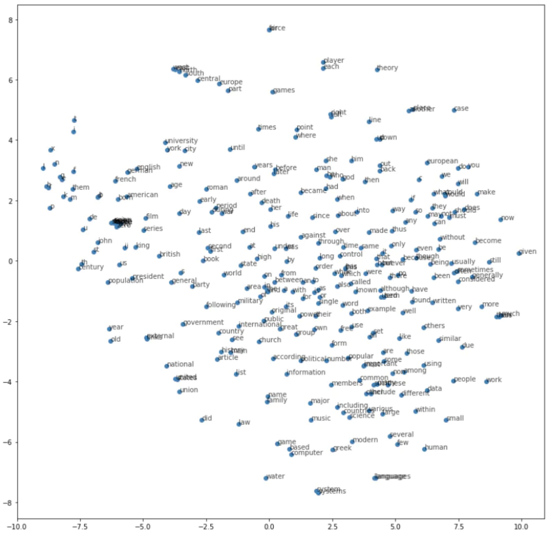


Figure 5: GloVe (NLP)

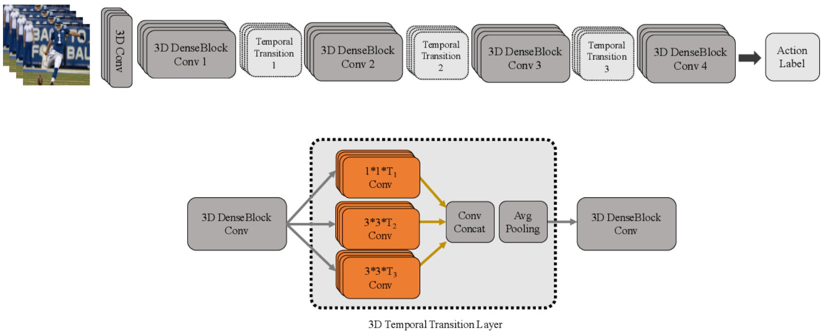
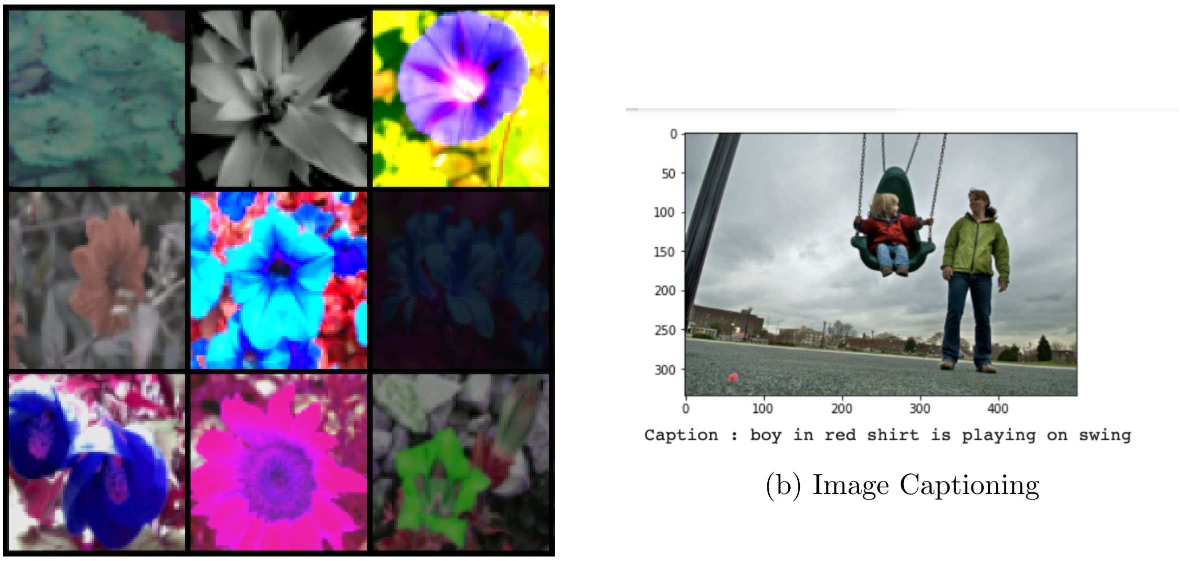


Figure 6: T3D (Classification)



```
Caption : boy in red shirt is playing on swing
```

(a) StackGAN

(b) Image Captioning

Figure 7: Multimodal Models

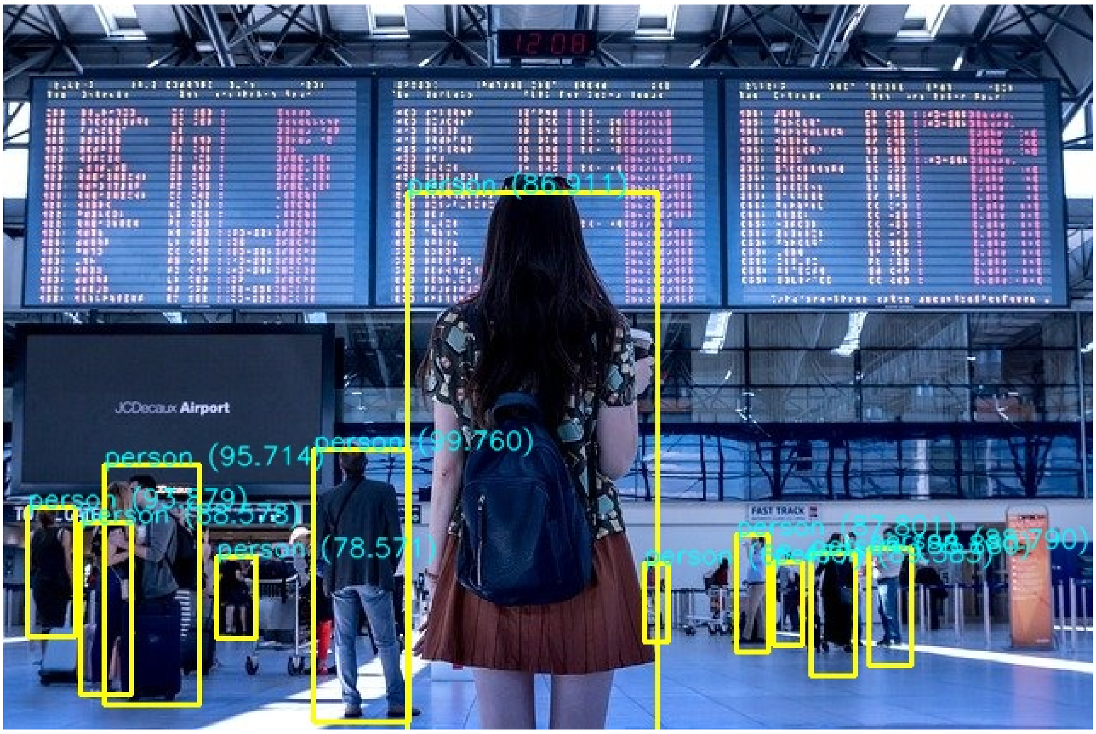
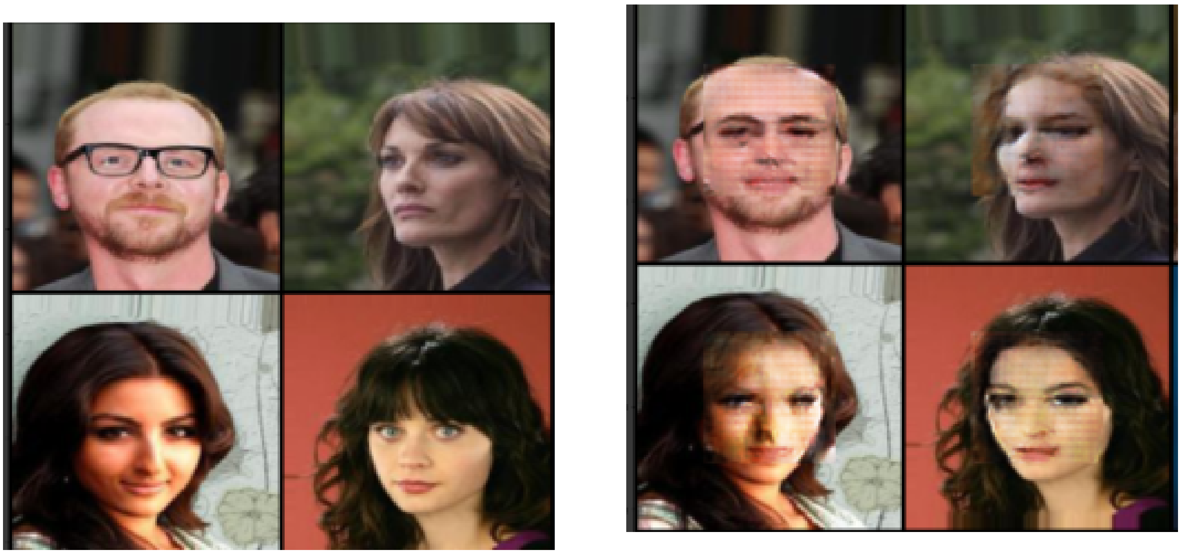


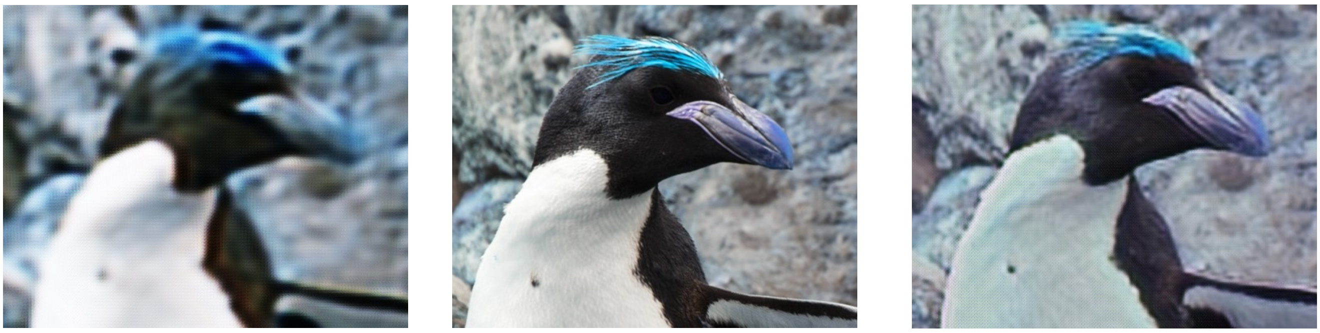
Figure 8: YOLOv3 (Object Detection)



(a) Original Images

(b) Generated Images after masking some part of the faces

Figure 9: Context Encoder (Image Inpainting)



(a) Bicubic Interpolation

(b) High Resolution

(c) Super Resolution

Figure 10: SRGAN (Super Resolution)

5 Team

5.1 Contributors

Akshay Gupta
Antreev Brar
Ashish P Murali
Atharv Singh Patlan
Ayush Gupta
Gurbaaz Singh Nandra
Mridul Dubey
Nakul Jindal
Naman Gupta
Rishabh Dugaye
Shivamshree Gupta
Shivanshu Tyagi
Som Tambe
V Pramodh Gopalan
Vansh Bansal

5.2 Mentors

Dev Chauhan
Deepankur Kansal
Naman Biyani
Nirmal Suthar