**15EC401M-MULTIDISCIPLINARY DESIGN REPORT**

***on***

**Super-Resolution Using GANs**

***by***

**RA1711004010411 SATYAM ROUT**

**RA1711004010427 DIVYA GUPTA**



**COLLEGE OF ENGINEERING AND TECHNOLOGY**

**DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING**

**SRM INSTITUTE OF SCIENCE AND TECHNOLOGY**

**SRM Nagar, Kattankulathur-603203,**

**Kancheepuram District, Tamilnadu**

**Dec 2020**



**SRM INSTITUTE OF SCIENCE AND TECHNOLOGY**

**COLLEGE OF ENGINEERING AND TECHNOLOGY**

**DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING**

**SRM Nagar, Kattankulathur-603203.**

**Kancheepuram District, Tamilnadu**

**BONAFIDE CERTIFICATE**

**Register No.**

Certified to be the bonafide record of work done by **RA1711004010411, RA1711004010427**, of **ECE B.Tech** Degree course in **15EC401M Multidisciplinary Design** in **SRM Institute of Science and Technology**, Kattankulathur during the Academic year **2020-2021 (ODD Semester)**

**Course Incharge**

**Date: 4/12/2020**

**Table of Contents**

|  |  |  |
| --- | --- | --- |
| **S-No** | **Content** | **Page No** |
| I | Abstract | Ⅰ |
| II | List of Figures | II |
| III | List of Tables | III |
| 1 | Introduction | 1 |
| 2 | Literature Survey | 2 |
| 3 | System Design and Description | 5 |
| 4 | Methodology Adopted | 9 |
| 5 | Software requirements and cost modelling | 11 |
| 6 | Realistic constraints and deliverables | 12 |
| 7 | Conclusions | 13 |
| 8 | References | 14 |

**Abstract**

Super-Resolution using deep-learning techniques is one of the most researched fields now. Super-resolution is an important aspect in various real world applications. Previously various photoshop tools were used to achieve this scenario but now due to extensive research, it is possible to achieve this phenomena using artificial intelligence. Although there is lots of research in achieving super resolution with good accuracy, speed is always a constraint to it. Achieving a light weighted model with good accuracy is a challenging part of this proposal. So here in this proposal the main aim is to create such an adversarial model which can fine tune the minimal textures of the image to super resolute it. So previously extensive research has been done to reduce the mean squared error of the high Signal to Noise ratio of the distorted images but the speed became a constraint to it. So here we have proposed a more optimized model using Generator and discriminator to generate a more realistic image with good accuracy and speed. The loss function here is optimized to calculate the loss and compare with the previous loss for better estimation and less computation. Despite the transfer learning approach i.e, using heavy weighted models like ResNet-50, we have achieved much lighter models due to layer optimization and due to batch normalization techniques. The model is capable of generating accurate super resolute images using GANs with a less time and space complexity.

**List of Tables**

3.1. Generator Model ……………………………………………………………………………..5

3.2. Discriminator Model …………………………………………………………………………7

4.1. Various modules/Apis used…………………………………………………………………..9

**List of Figures**

1.1. Diagram for basic Architecture Generative Adversarial Network(GANs) …………………………..1

2.1. Spatial GAN network…………………………………………………………………………3

2.2. Wasserstein GAN Framework………………………………………………………………..3

2.3. TecoGAN super-resolution …………………………………………………………………..4

3.1 Basic ResNet-50 architecture………………………………………………………………….6

3.2 Diagram representation of ping pong loss function…………………………………………...8

4.1 Model Architecture…………………………………………………………………………..10

**Chapter 1**

**Introduction**

Multimedia components like images, videos, graphics etc play a vital role in our daily lives .We use augmented videos/images from various resources such as webpages/youtube and many more for purposes like learning, entertainment and many more. These augmented images/videos are processed in order to make them computer understandable. As a result of which, its property starts to degrade due to addition of various noises or due to loss of pixels. As a result we are unable to get proper images/videos. So we use various computer applications to enhance the video/image qualities, Which requires some prerequisites.

In order to overcome this, researchers have come up with an AI solution, which can regenerate the blurred/distorted images. They proposed the idea of Generative Adversarial Networks also known as GANs. As the name suggests it is a Deep learning framework which can regenerate images similar to that of the given one. GANs have also proved to be useful for Supervised and Reinforcement learning.

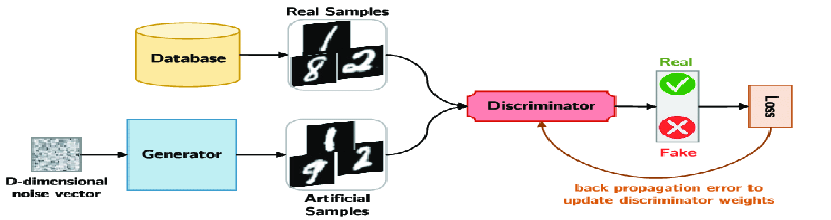


Fig.1.1: Diagram for basic Architecture Generative Adversarial Network(GANs)

In this proposal we aim to create super-resolution images and videos from the blurred low end images/videos.The high Dpi images such as satellite images and long range images which cannot be distinguished by human eyes can also be viewed clearly with the help of this AI based approach. In this approach the distorted images/videos are fed through neural networks in order to get super resolution images. This proposal can be then further extended to creation of fast video streaming even in low bandwidth of internet, by applying the SRGAN in the backend to get a regenerated super resolution video streaming.

**Chapter 2**

**Literature survey**

There have been many researches done for the high resolution of static images but there are very few in the case of videos as compared to images. This is because videos contain additional temporal dynamics. The behaviour of temporal variations is way different than that of spatial variations. Spatial variations can be referred to as the object whereas the temporal variations can be referred to as the dynamics of the object[1]. There arises computational and storage problems with video based algorithms. In videos , due to action recognition problems, the static images are not sufficient for prediction even if the best frame is chosen[1]. There are various ways found in which this can be done using generative models such as auto-encoders, auto-regressive models and generative adversarial networks(GANs) [1].

Mean squared error loss is used in the process of optimization as it takes into consideration the difference between the pixels of highly resoluted output and the upscaled low-resolution input thus minimizing the error to produce a smooth output[2].The major issue that GAN suffered all these years was its instability while training and failure to converge, this has been greatly improved with the help of more developed loss functions,architecture and training models. While designing models for generation of videos we need to consider the spatial as well as the temporal aspects of the video which infer the objects and the dynamics of the objects respectively. One of the methods to do this is to demonstrate the spatio-temporal coherence which supervises the short-term temporal coherence and self-supervision for long term temporal coherence[1]. For this modelling of temporal information for regeneration of video has become very important for recognition of action. Despite training the existing dataset of videos it is very difficult to determine whether the model understands the motion or not. Frame selection while subsampling proves to be an important aspect for improvising the prediction of motion by the model[4].

For adversarial training, we use two networks namely the generator and the discriminator.The loss function is designed so that the generator's goal is to generate samples that fool the discriminator. The discriminator tries to distinguish whether the generated sample is from a given distribution or not. The discriminator’s goal is designed to avoid being fooled by the generator[5].

For spatio-temporal self-supervised adversarial training the output of the generator is again fed into the generator as input so as to use the input as well as the previous generated output of the generator this improves the perceptual quality [5]. This method is known as spatio-Temporal adversarial learning. The frame triplets provide the network with gradient information regarding the realism of spatial structures as well as short-term temporal information, such as first and second-order time derivatives.

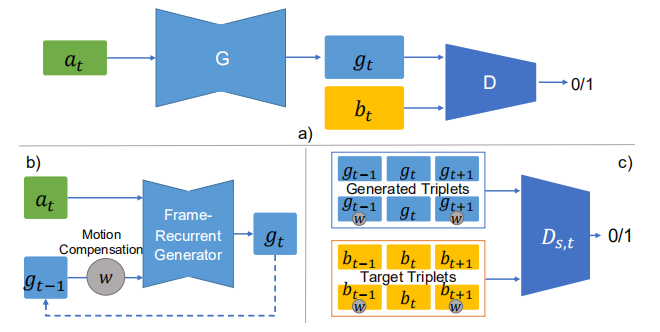


Fig.2.1:(a)Spatial GAN network(b)A spatio-temporal Generator(c)A spatio-temporal Discriminator.

In early years, discriminators were trained with sigmoid cross entropy loss as they were trained as classifiers for real and generated data. However, this led to vanishing gradient problems. For this least square GANs which used least square loss were proposed[1][2]. This too had problems related to lack of convergence. To reduce this the Wasserstein GAN also known as WGAN was introduced[1] .It is an extension of the GAN that seeks an alternate way of training the generator model to better approximate the distribution of data observed in a given training dataset.

Instead of using a discriminator to classify or predict the probability of generated images as being real or fake, the WGAN changes or replaces the discriminator model with a critic that scores the realness or fakeness of a given image.

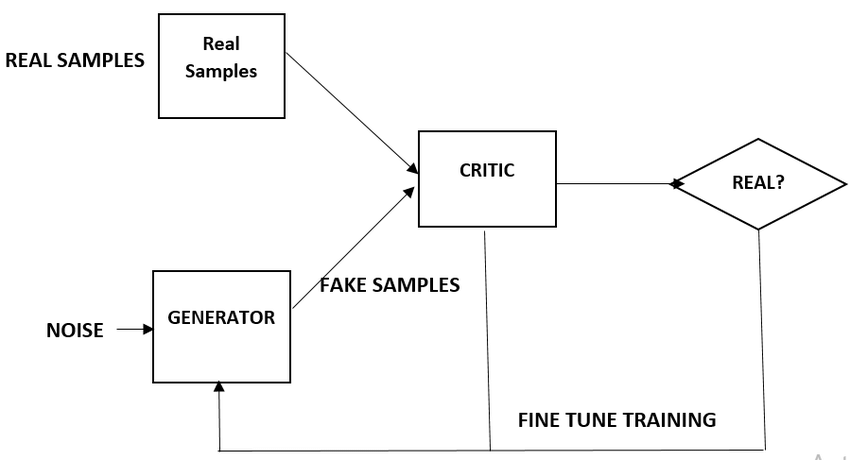


Fig.2.2:Wasserstein GAN Framework

Similarly, another loss function which has proved to be useful is the ping-pong loss function[3]. It is a linear loss function which is used in Self-Supervision for Long-term Temporal Consistency. The main purpose behind using this is to reduce the computational complexity of the neural network while finding out the loss.

The original size of the video is 64X64 which has been upscaled upto 256X256 in the previous research[1]. There are various adversarial networks implemented until now, some of which are ENet, DsOnly, DsDt, DsDt+PP, TecoGAN⊝and TecoGAN. Using two discriminators requires 70% more GPU memory, and leads to a reduced training performance by 20%. The TecoGAN⊝model yields similar perceptual and temporal quality with a significantly faster and more stable training. Since the TecoGAN⊝model requires less training resources, it can also train a generator with 50% more weights.



Fig.2.3: TecoGAN super-resolution

In our proposed model we will upscale the 64X64 video to 320X320. The main constraints in generating high resolution is generating smooth video which is only possible if the computation time and complexity is decreased. Here, we are going to use linear loss function instead of non linear loss function. To achieve accurate prediction results only usage of linear loss functions is not enough as vanishing gradient problems will arise. For this we are going to use Transfer learning . A pretrained model which is smaller in size as compared to that of the original dataset is used so as to get better prediction.

**Chapter3**

**System Design and Description**

The system is divided into three parts:

1. Generator
2. Discriminator
3. Loss

*Generator:*

The generator is the most important part of our system as it generates the realistic super-resolution images. The generator is built using a pretrained model(ResNet-50) on top of a base model. As shown in the table below.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Layer name** | **Input size** | **Conv\_matrix, stride** | **Pooling, stride** | **Activation function** | **Batch Normalization** | **Output size**  **(** |
| Conv\_1 | 112x112 | 7x7/64, 2 | \_ | Relu | \_ | 56x56 |
| Conv\_2x | 56x56 | 1x1/64  3x3/64 X 3  1x1/256 | Max-pooling(3x3), 2 | PRelu | Yes | 28x28 |
| Conv\_3x | 28x28 | 1x1/128  3x3/128 X 4  1x1/512 | \_ | PRelu | Yes | 14x14 |
| Conv\_4x | 14x14 | 1x1/256  3x3/256 X 6  1x1/1024 | Avg-pooling(3x3), 2 | PRelu | Yes | 7x7 |
| Conv\_5x | 7x7 | 1x1/512  3x3/512 X 3  1x1/2048 | \_ | PRelu | Yes | 1x1 |

Table.3.1: Generator Model

Here ResNet-50 is used with some modification in its layers like adding Batch-Normalization in the middle of each convolutional layer which reduces the time complexity and also applying PRelu activation function in each conv\_x layers to increase the accuracy of the model by parameterizing the instead of fixing it to The output layer is calculated using the equation where,

Mi - size of input matrix

Ki - size of kernel/filter matrix

P - padding, S - stride

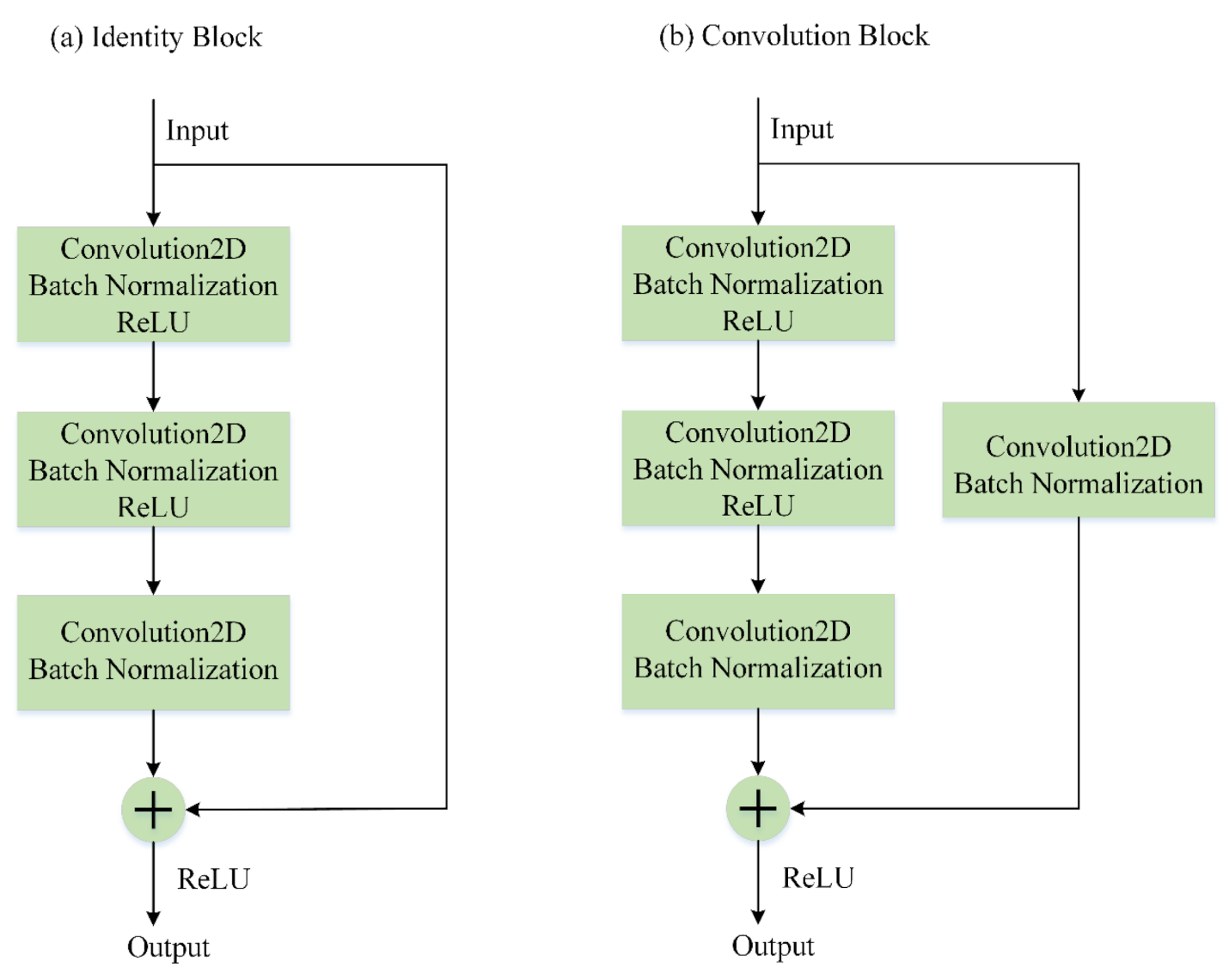


Fig.3.1: Basic ResNet-50 architecture(a. Identity block, b. Convolution Block)

*Discriminator:*

This is a basic architecture used to determine that the generated image is real(1) or fake(0), compared with the real image present in the database. The following architecture is shown below table.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Layer name** | **Input size** | **Conv\_matrix, stride** | **Pooling, stride** | **Activation function** | **Output size**  **(** |
| Conv\_1 | 112x112 | 3x3/64, 1 | \_ | Relu | 110x110 |
| Conv\_2 | 110x110 | 3x3/64, 1 | Max-pooling(3x3), 2 | Relu | 108x108 |
| Dropout | 108x108 | \_ | \_ | \_ | 108x108 |
| Conv\_4x | 108x108 | 3x3/64, 1 | Max-pooling(3x3), 2 | Relu | 106x106 |
| Dropout | 106x106 | \_ | \_ | Relu | 106x106 |

Table.3.2: Discriminator Model

*Loss:*

Here we have used two loss functions i.e, cross\_entropy loss and the ping-pong loss in discriminator and generator respectively.

1. Corss\_entropy\_loss function works on the principle of entropy where the measure of uncertainty in a system is measured. Here in discriminator it is used to measure the uncertainty of the generated images from the real image and predict the output that it is a fake or real image.

Equation :

Where,

q(x), p(x) are the probabilities of occurrences

- if target is 1(real)

- if target is 0(fake)

1. Ping-Pong loss function considers two sequences one in forward direction and other in counter direction like x1 ,......., xt-1 , xt , xt+1 , ....... ,xn and xn ,.......,xt+1,xt, xt-1 , ......., x1 considering the fact that the generated frame gt is identical irrespective of the order of the sequence .

For, Forward result: gt = G(xt,gt-1)

For, Reverse result: gt' = G(xt,g't+1)

Equation: Lpp = ||g1-g1'||2 + ||g2-g2'||2 + ...................... + ||gn-1 - gn-1'||2

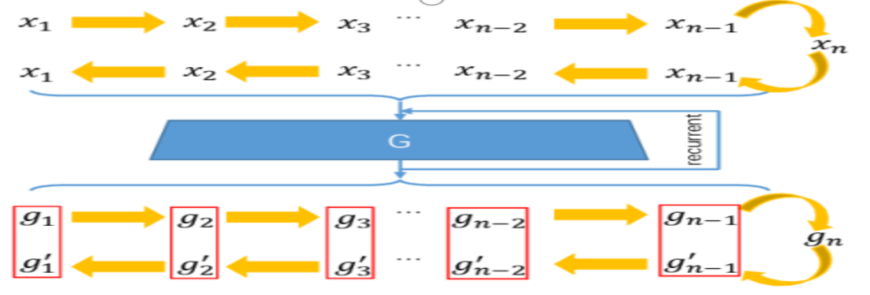


Fig.3.2:Diagram representation of ping pong loss function

**Chapter4**

**Methodology Adopted**

To get the super-resolution images/videos we have used the deep-learning approach using a special kind of architecture i.e, GANs(general adversarial networks). Here we have used the python programming language and its various modules to develop the architecture. The below table gives the complete methodology of the modules used to develop the proposal.

|  |  |  |
| --- | --- | --- |
| Python modules/Apis | description | Used for |
| Pillow/OpenCV | This libraries are used for image-processing | In our project this library is used to resize the images and read/write the images and to manipulate the images like upsampling and downsampling the images |
| Numpy | Library is used for complex matrix calculations | As mentioned earlier in our proposal we have used various matrix convolution of images so this library is used to do complex matrix calculation like matrix multiplication, finding vectors from matrix etc. |
| Pandas | Library used for dining various analysis of the features of the data | This library is used to do k-fold analysis and various testing of models like AUC/ROC curve and many more for finding out the best models among all. This is also used to create accuracy, recall, sensitivity .csv files to get the analysis of models and compare it efficiently. |
| Scikit-learn | This is most used library for various machine learning algorithms | This library is used here to do hyper-parameter optimizations using functions like grid\_searchCV/random\_searchCV for finding out the best hyperparameters like learning rate, momentum, optimization ratio etc. |
| Keras | This is a High level Api solely used for deep learning applications. | This is the important Api used for developing Generator and Discriminator architectures and also used for implementing the ResNet-50 model on top of our base model. It is solely used to create the neural network and assign the various functionalities to it. |
| Tensorflow | An Api by google developed using C++. | This api is used for creating the links/ graphs for neural networks among the perceptrons. |

Table 4.1: various modules/Apis used.

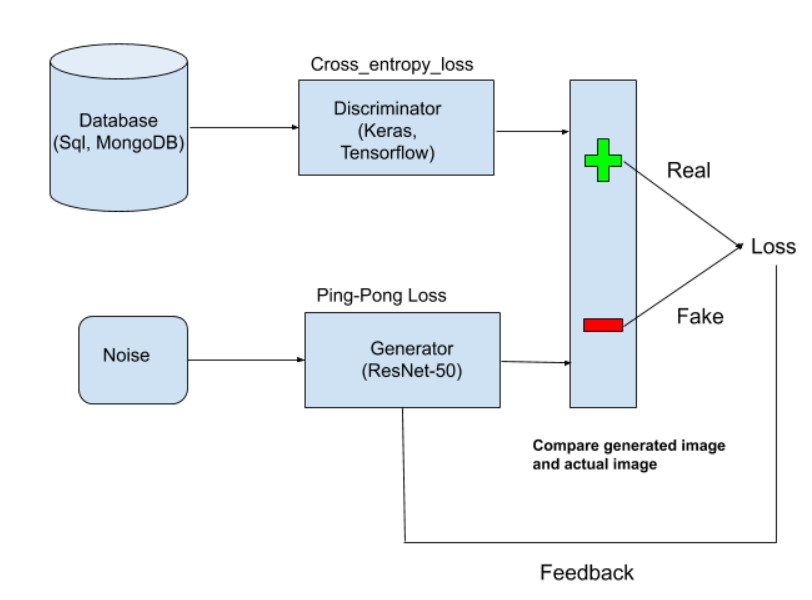


Fig 4.1: Model Architecture

In the above figure the complete methodology of the proposal is shown we can observe from the above figure that how the real images are feed to the discriminator which act as a decision maker and the noise is feed to the generator which tries to create the super-resolution real images when the discriminator calculates the loss in accordance with the real or fake images.

**Chapter 5**

**Software requirements and cost modelling**

*Softwares used:*

1. Python (its various modules)
2. AWS EC2 instance cloud platform
3. GPU/TPU based system(for model creation)

*Cost modelling:*

1. GPU/TPU system from AWS(p3.16xlarge with V100 tensor GPU) - $8.39/hour

**Chapter 6**

**Realistic constraints and deliverables**

*Realistic Constraints:*

1. Requires high computational resources for model creation.
2. Minimum 12-core GPUs required for compilation of the model
3. Heavy weight model usage for better accuracy
4. May over generate the images i.e, overfitting problem
5. Make the images blurry due to more Dpi compression
6. Requires resources having prior knowledge of building models in case of system crash.
7. May generate wrong textures in the images which can lead to some other images than original one.
8. Some Latency will be present if run in the backend of a live video streaming.
9. May over-sharpen the images/videos.

*Deliverables:*

This proposal can be used for various applications in the real world from defence to paper scanning.

Some of the deliverables of this proposal are:

1. This can be deployed in mobile scanning applications such as cam-scanner for super resolution of blurry images.
2. Can be used for security purposes like security surveillance for monitoring the crime scenes which are not visible by normal cameras.
3. Can be used to modify the normal camera videos for clearer footages in night or darker areas, instead of using night-vision cameras.
4. This proposal can be used to replace costly optical cameras.
5. This can be deployed in military drones for surveillance
6. This can also be used in various manufacturing industries for tracking the machines and its complete mechanisms.
7. Can be used for studying the high Dpi satellite and underwater images.

**Chapter 7**

**Conclusion**

In this project we have implemented image/video super resolution also known as VSR using Generative Adversarial Networks(GANS). We gained a good amount of practice with hyperparameter tuning. We have hit and trail with various pre-trained model and weights. We have aslo optimized the model using various above mentioned practices to achieve a model with good accuracy and speed of video streaming. This proposal is implemented using various research works and taking various review papers in account, and came up with an optimized model. Still this model can be further improved and optimized for better super-resolution using various other efficient techniques like auto-encoders.

**Chapter 8**

**References**

[1] Dinesh Acharya. Towards High Resolution Video Generation with Progressive Growing of Sliced Wasserstein GANs. arXiv:1810.02419v2 [cs.CV] 6 Dec 2018.

[2] Boris Kovalenko. Super resolution with Generative Adversarial Networks. SUID 06201315.

[3] MENGYU CHU, YOU XIE , JONAS MAYER, LAURA LEAL-TAIXÉ, and NILS THUEREY. Learning Temporal Coherence via Self-Supervision for GAN-based

Video Generation. arXiv:1811.09393v4 [cs.CV] 21 May 2020.

[4] De-An Huang , Vignesh Ramanathan , Dhruv Mahajan, Lorenzo Torresani, Manohar Paluri , Li Fei-Fei , and Juan Carlos Niebles. What Makes a Video a Video: Analyzing Temporal Information in Video Understanding Models and Datasets. 10.1109/CPVR.2018.00769.

[5] Leonardo Galteri, Lorenzo Seidenari, Marco Bertini, Tiberio Uricchio, Alberto Del Bimbo. Fast Video Quality Enhancement using GANs. MM ’19, October 21–25, 2019, Nice, France .