Project Report

Entitled

Wireless Capsule Endoscopy using Deep Learning

Submitted to the Department of Electronics Engineering in Partial Fulfilment for the Requirements for the Degree of

Bachelor of Technology
(Electronics and Communication)

: Presented & Submitted By :

Anand Gupta, Bhavay Savaliya, Aishwary Mehta, Harshit Bhardwaj

Roll No. (U20EC001, U20EC096, U20EC105, U20EC107)

B. TECH. IV(EC), 8^{th} Semester

: Guided By :

Dr. Kishor P. Upla
Assistant Professor, SVNIT

(Year: 2023-24)

DEPARTMENT OF ELECTRONICS ENGINEERING

SARDAR VALLABHBHAI NATIONAL INSTITUTE OF TECHNOLOGY

Surat-395007, Gujarat, INDIA.

Sardar Vallabhbhai National Institute Of Technology

Surat - 395 007, Gujarat, India

DEPARTMENT OF ELECTRONICS ENGINEERING

CERTIFICATE

This is to certify that the Project Report entitled "**Wireless Capsule Endoscopy using Deep Learning**" is presented & submitted by Anand Gupta, Bhavay Savaliya, Aishwary Mehta, Harshit Bhardwaj, bearing Roll No. U20EC001, U20EC096, U20EC105, U20EC107, of B.Tech. IV, 7^{th} Semester in the partial fulfillment of the requirement for the award of B.Tech. Degree in Electronics & Communication Engineering for academic year 2023-24.

They have successfully and satisfactorily completed their **Project Exam** in all respects. We, certify that the work is comprehensive, complete and fit for evaluation.

Dr. Kishor P. Upla
Assistant Professor & Project Guide

PROJECT EXAMINERS:

Name of Examiners	Signature with Date
1. _____	_____
2. _____	_____
3. _____	_____
4. _____	_____

Dr. Rasika N. Dhavse
Head, DoECE, SVNIT

Seal of The Department
(May-2023)

Acknowledgements

We would like to express my profound gratitude and deep regards to our guide Dr. Kishor Upla Prof. Kiran Raja for their guidance. We are heartily thankful for suggestion and the clarity of the concepts of the topic that helped us a lot for this work. We would also like to thank Dr. Rasika N. Dhavse, Head of the Electronics Engineering Department, SVNIT and all the faculties of ECED for their co-operation and suggestions. We are very much grateful to all our classmates for their support.

Student Name
Sardar Vallabhbhai National Institute of Technology
Surat

May 10, 2023

Abstract

Medical imaging is a crucial component of medical diagnosis and treatment process. Due to limitations of imaging technology and varying imaging conditions, many medical images have low spatial resolution. This low resolution of medical images makes it difficult to detect minor anatomical features and abnormalities. Super Resolution (SR) is a class of software based techniques that are used to enhance the resolution of an image. Enhanced-resolution endoscopy has shown to improve adenoma detection rate for conventional endoscopy and is likely to do the same for capsule endoscopy.
In this work, we have done an extensive literature review of already existing state-of-art models for image super resolution (SRGAN [**?**], DenseNet [**?**], RCAN [**?**] etc.) In addition, we have developed a model architecture Densenet with Channel Attention Block(DCAN) which is able outperform existing state of art models in terms of perceptual quality as well as providing better values for the different image quality assessment metrices such as PSNR, SSIM and LPIPS. We aslo performed statistical analysis on the predicted images to ensure the consistency of the model.

Table of Contents

List of Figures

List of Tables

List of Abbreviations

CycleGAN	Cycle Generative Adversarial Networks
GAN	Generative Adversarial Networks
GT	Ground Truth
HR	High Resolution
LR	Low Resolution
MSE	Mean Squared Error
PSNR	Peak Signal to Noise Ratio
SR	Super Resolution
SRCNN	Super Resolution Convolutional Neural Network
SRGAN	Super Resolution Generative Adversarial Networks
SSIM	Structural Similarity Index
WCE	Wireless Capsule Endoscopy

Chapter 1
Introduction

Single image super resolution using deep learning is one of the key area of research in recent times. Conventionally the resolution of images was increased using various interpolation techniques like bicubic interpolation, Nearest neighbour interpolation, bilinear interpolation etc. However, these techniques are not very efficient and looses minute features from images. These minute features are of great importance in medical images. Hence Deep learning based approaches explored to overcome this problem. Deep learning algorithms in Wireless Capsule Endoscopy (WCE) image analysis are a promising research area to overcome the pitfalls presented in the hand-crafted feature selection method. Medical Images contains a lot of complex data, which is easily missed while dealing with hand crafted features.

1.1 Capsule Endoscopy

Endoscopy is a common imaging procedure that may be utilised for diagnostic as well as minimally invasive surgical treatment. Video Capsule Endoscopy (VCE) is a groundbreaking imaging technology that allows for unparalleled direct vision of the gastrointestinal tract with little patient discomfort. A normal colon VCE test generates around 8 hours of RGB video data. Wireless Capsule Endoscopy (WCE) is a non-invasive procedure for detecting irregularities in the Gastro intestinal tract of humans. It consists of an optical dome, illuminator, imaging sensor, battery, and RF transmitter in a capsule-shaped item with a length of 26 mm and a diameter of 11 mm. During the examination, the patient swallows the WCE and then slides it slowly down the small intestine, taking photos of the whole gastro intestinal tract while doing so. As shown in Fig. **??** Finally, these images are transferred wirelessly to a data-recording device so that doctors can study the photos later for diagnosis. While moving through the gastro intestinal tract, an average of 50k to 60k images are taken. It is a time-consuming and arduous operation for the medico to verify all 60k images in order to find any abnormalities in the gastro intestinal tract.

The small bowel is located in the middle of the gastrointestinal (GI) tract, between the stomach and the large intestine. It is three to four metres long and has a surface area of roughly 30 m^2 ,including the villi's surface area, and plays an important role in nutrient absorption. As a result, small bowel problems can cause significant development retardation in children as well as nutritional deficiencies in both children and adults. Chronic illnesses such as Crohn's disease, celiac disease, and angiectasis, as well as malignant diseases such as lymphoma and adenocarcinoma, can damage this organ.

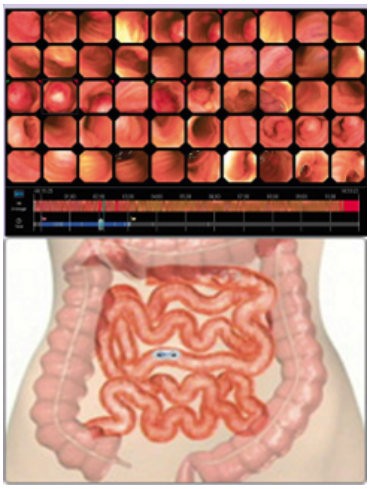


Figure 1.1: The capsule movement in the gastro-intestinal track and also provides theglimpse of images captured by the capsule camera during its journey ingastro-intestinal track. [**?**]

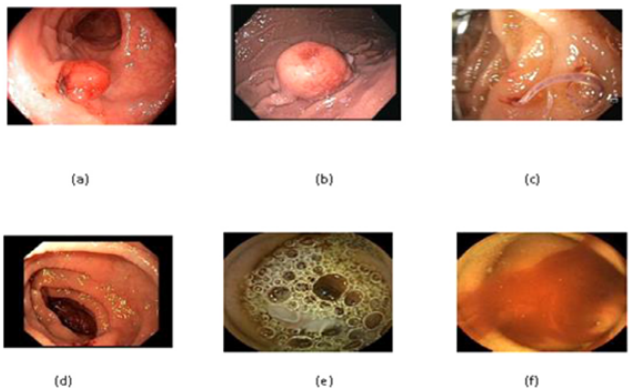


Figure 1.2: (a) Polyp, (b) Tumour (c) Hookworm (d) Celiac (e) Bubbles (f) Bleeding. [**?**]

These illnesses can pose a significant health risk to individuals and society, and diagnosing and treating them often necessitates a comprehensive inspection of the lumen. The small intestine, on the other hand, is less accessible for inspection by flexible endoscopes frequently employed for the upper GI tract and the large bowel due to its anatomical position. VCE has been utilised as a supplementary diagnostic for patients with GI bleeding since the early 2000s. A VCE is a tiny capsule that houses a wide-angle camera, as well as light sources, batteries, and other electronics. The patient eats the capsule, which records video as it travels through the GI tract passively. The glimpse of images taken by capsule is provided in Fig. **??**

1.2 Image Super Resolution

The process of converting one or more low-resolution observations of the same scene into high-resolution photographs is known as Super-Resolution (SR). The SR can be divided into Single Image Super Resolution (SISR) and Multi-Image Super-Resolution (MISR) depending on the quantity of input LR images. SISR is far more well-liked than MISR due to its great effectiveness. High perceptual quality HR images are employed extensively in a variety of fields, including security imaging, satellite imaging, and medical imaging because they contain more useful details. In the typical SISR framework, as depicted in Fig. **??**. Eq. **??** shows the LR image y is degraded due to blur, downsampling and noise effect.

$$y = (x \otimes k) \downarrow_s + n \tag{1.1}$$

where, x $\otimes$ k is the convolution between the blurry kernel k and the unknown HR image x, $\downarrow_s$ represents the downsampling operator with scale factor s, and n is the independent noise term. Solving equation (**??**) is an extraordinarily ill-posed problem due to the fact that a single LR input may correlate to several HR solutions. Interpolation-based approaches, reconstruction-based methods, and learning-based methods make up the majority of the prevalent SISR algorithms in recent times [**?**]. Interpolation-based SISR

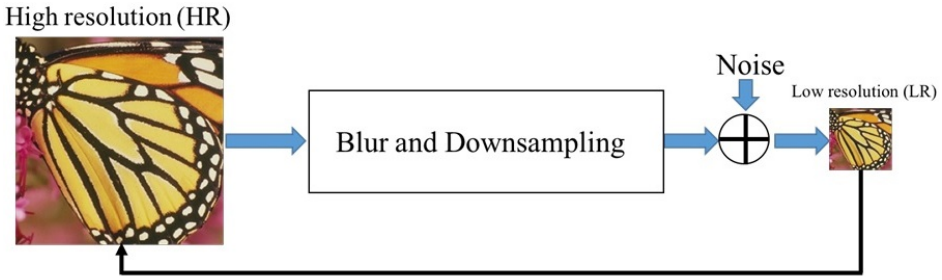


Figure 1.3: Recovering HR from its LR components. [**?**]

approaches, such as bicubic interpolation [**?**] and Lanczos resampling [**?**], are relatively quick and simple although they lack precision. Reconstruction-based SR approaches that frequently employ sophisticated previous information to restrict the available solution space, with the benefit of producing flexible and sharp details. Nevertheless, the performance of many reconstruction-based methods rapidly decreases as the scale factor grows, and these techniques are typically time-consuming.

1.3 Motivation

The visual interpretation and automated analysis of endoscopic recordings are hampered by artefacts such as motion blur, bubbles, specular reflections, floating objects, and pixel saturation. Given the increasing use of endoscopy in many clinical applications, a key medical imaging challenge is identifying such artefacts and automating the repair of damaged video frames. The doctor is likely to miss a key frame that reveals the irregularity because the video is the same length as the time it takes for the capsule to pass through the intestines. To address the prevailing procedures, Computer-Aided Detection (CAD) approaches were proposed. The use of computers to detect abnormalities in capsule endoscopy images has opened up new research opportunities in the fields of medical image processing and machine learning and deep learning. Deep learning has been proven to be one of the most efficient method to deal with images, whether it be image classification, image segmentation or object detection in images. The accuracy of these algorithms highly depend on the resolution of the images. Also, it is very easy for the doctors to detect the abnormalities with naked eye. Hence, We decided to work on this problem statement.

1.4 Objective

Wireless Capsule Endoscopy is gaining vast acceptance in society, and there is a severe need of computer aided tools to process the data obtained from the images. Our main objective is to increase the resolution of these images by a factor of 4, so that the images have better perceptual quality and the higher resolution of these images can be used to improve the accuracy of the computer aided algorithms.

1.5 Report outline

The report is organized such that Chapter 2 shows the extensive literature review which covers traditional up-sampling methods such as bi-cubic interpolation, nearest neighbour interpolation, etc. as well as deep learning based up-sampling methods, Chapter 3 describes state of the art models for image super resolution and their performance on wireless capsule dataset, while Chapter 4 demonstrates the proposed architecture and their training details. Chapter 5 shows the comparative study of results obtained during experimental work and the report ends with summary and future scope.

Chapter 2
Literature review

This chapter contains a brief overview of conventional Image upsampling methods and Image Super Resolution approaches using Deep learning. Image upsampling/Image Super Resolution is defined as increasing the spartial resolution while keeping the 2D representation of the image same. It is often used to enlarge a specific area of a picture and to remove the pixelation that results from displaying a low-resolution image over a big frame. Different Deep learning based techniques that are used for image Super Resolution are mentioned in the Fig. **??**

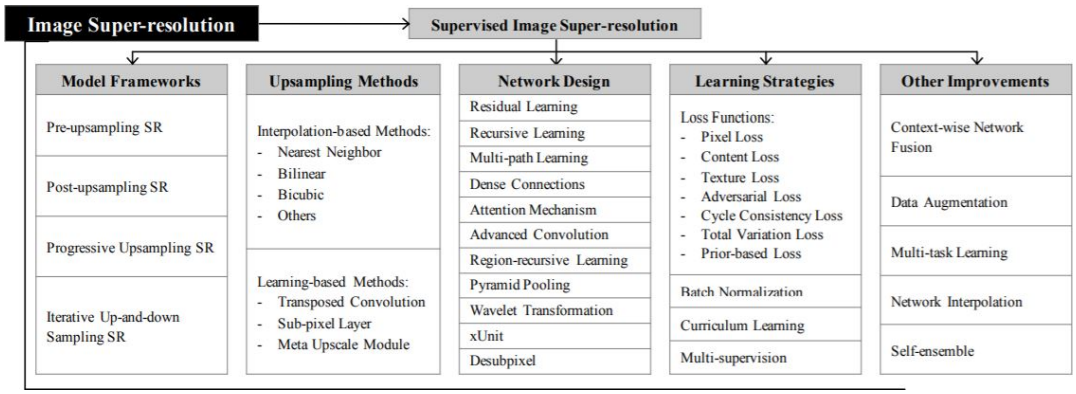


Figure 2.1: Techniques used for Supervised Image Super Resolution [**?**]

2.1 Image Upsampling

Let's first comprehend upsampling (raising the spatial resolution of images or just the number of pixel rows/columns or both in the image) and its numerous techniques in order to comprehend the rest of the theory underlying super-resolution.
Image interpolation, also known as image scaling, is the process of resizing digital images and is frequently used in applications that deal with images. The conventional techniques include linear, bilinear, bicubic, nearest-neighbor interpolation, etc.
Before understanding the rest of the theory behind the super-resolution, we need to understand upsampling (Increasing the spatial resolution of images or simply increasing the number of pixel rows/columns or both in the image) and its various methods.

Interpolation-based methods – Image interpolation (image scaling), refers to resizing digital images and is widely used by image-related applications. The traditional

methods include nearest-neighbor interpolation, linear, bilinear, bicubic interpolation.

Nearest-neighbor Interpolation – The nearest-neighbor interpolation is a simple and intuitive algorithm. It selects the value of the nearest pixel for each position to be interpolated regardless of any other pixels.

Bilinear Interpolation – The bilinear interpolation (BLI) first performs linear interpolation on one axis of the image and then performs on the other axis. Since it results in a quadratic interpolation with a receptive field-sized 2×2, it shows much better performance than nearest-neighbor interpolation while keeping a relatively fast speed.

Bicubic Interpolation – Similarly, the bicubic interpolation (BCI) performs cubic interpolation on each of the two axes Compared to BLI, the BCI takes 4×4 pixels into account, and results in smoother results with fewer artifacts but much lower speed. Refer to this for a detailed discussion.

2.1.1 Learning-based upsampling

To overcome the shortcomings of interpolation-based methods and learn upsampling in an end-to-end manner, transposed convolution layer and sub-pixel layer are introduced into the SR field.The blue boxes denote the input, and the green boxes indicate the kernel and the convolution output.

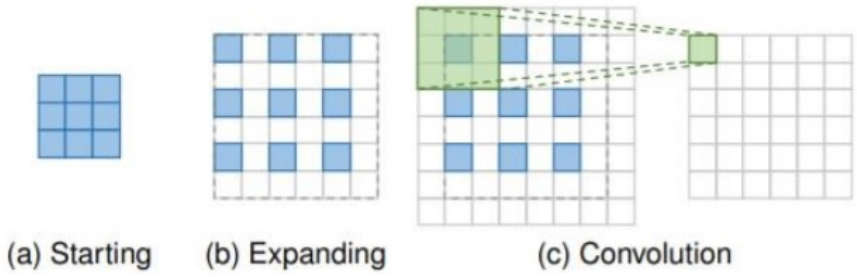


Figure 2.2: Transpose Convolution [**?**]

Transposed convolution layer, i.e. deconvolution layer, tries to perform transformation opposite a normal convolution, i.e., predicting the possible input based on feature maps sized like convolution output. Specifically, it increases the image resolution by expanding the image by inserting zeros and performing convolution. The blue boxes denote the input and the boxes with other colors indicate different convolution operations and different output feature maps.
Sub-pixel Layer: The sub-pixel layer, another end-to-end learnable upsampling layer,

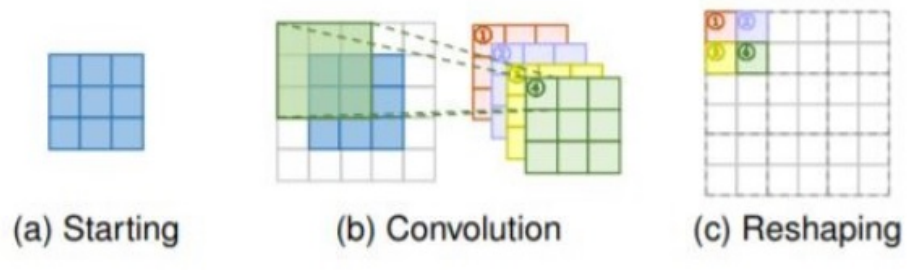


Figure 2.3: Sub-pixel Layer [**?**]

performs upsampling by generating a plurality of channels by convolution and then reshaping them shows. Within this layer, a convolution is firstly applied for producing outputs with s2 times channels, where s is the scaling factor. Assuming the input size is h × w × c, the output size will be h×w×s2c. After that, the reshaping operation is performed to produce outputs with size sh × sw × c.

2.2 Super-Resolution Frameworks

Since image super-resolution is an ill-posed problem, how to perform upsampling (i.e., generating HR output from LR input) is the key problem. There are mainly four model frameworks based on the employed upsampling operations and their locations in the model (refer to the table above).

1. Pre-upsampling Super-resolution – It does't make a direct mapping of LR images to HR images since it is considered to be a difficult task. It utilizes traditional upsampling algorithms to obtain higher resolution images and then refining them using deep neural networks is a straightforward solution. For example – LR images are upsampled to coarse HR images with the desired size using bicubic interpolation. Then deep CNNs are applied to these images for reconstructing high-quality images.

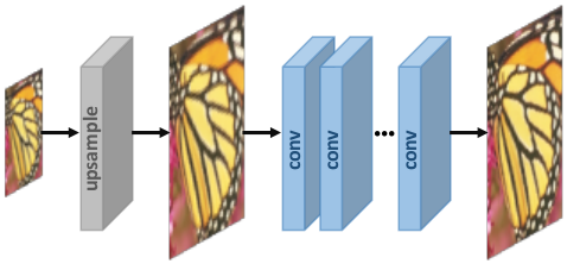


Figure 2.4: Pre-upsampling architecture [**?**]

2. Post-upsampling Super-resolution – To improve the computational efficiency and make full use of deep learning technology to increase resolution automatically, researchers propose to perform most computation in low-dimensional space by replacing the predefined upsampling with end-to-end learnable layers integrated at the end of the models. In the pioneer works of this framework, namely post-upsampling SR, the LR input images are fed into deep CNNs without increasing resolution, and end-to-end learnable upsampling layers are applied at the end of the network.

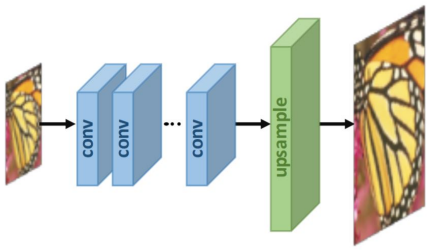


Figure 2.5: Post-upsampling architecture [**?**]

Learning Strategies In the super-resolution field, loss functions are used to measure reconstruction error and guide the model optimization. In early times, researchers usually employ the pixelwise L2 loss (mean squared error), but later discover that it cannot measure the reconstruction quality very accurately. Therefore, a variety of loss functions (e.g., content loss, adversarial loss) are adopted for better measuring the reconstruction error and producing more realistic and higher-quality results.

- Pixelwise L1 loss – Absolute difference between pixels of ground truth HR image and the generated one.
- Pixelwise L2 loss – Mean squared difference between pixels of ground truth HR image and the generated one.
- Content loss – the content loss is indicated as the Euclidean distance between high-level representations of the output image and the target image. High-level features are obtained by passing through pre-trained CNNs like VGG and ResNet.
- Adversarial loss – Based on GAN where we treat the SR model as a generator, and define an extra discriminator to judge whether the input image is generated or not.
- PSNR – Peak Signal-to-Noise Ratio (PSNR) is a commonly used objective metric to measure the reconstruction quality of a lossy transformation. PSNR is inversely proportional to the logarithm of the Mean Squared Error (MSE) between the ground truth image and the generated image.

2.2.1 Network Design

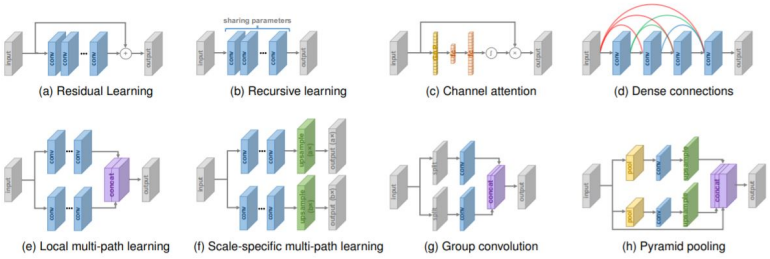


Figure 2.6: Various network designs in super-resolution architecture [**?**]

2.3 State of the Art Models

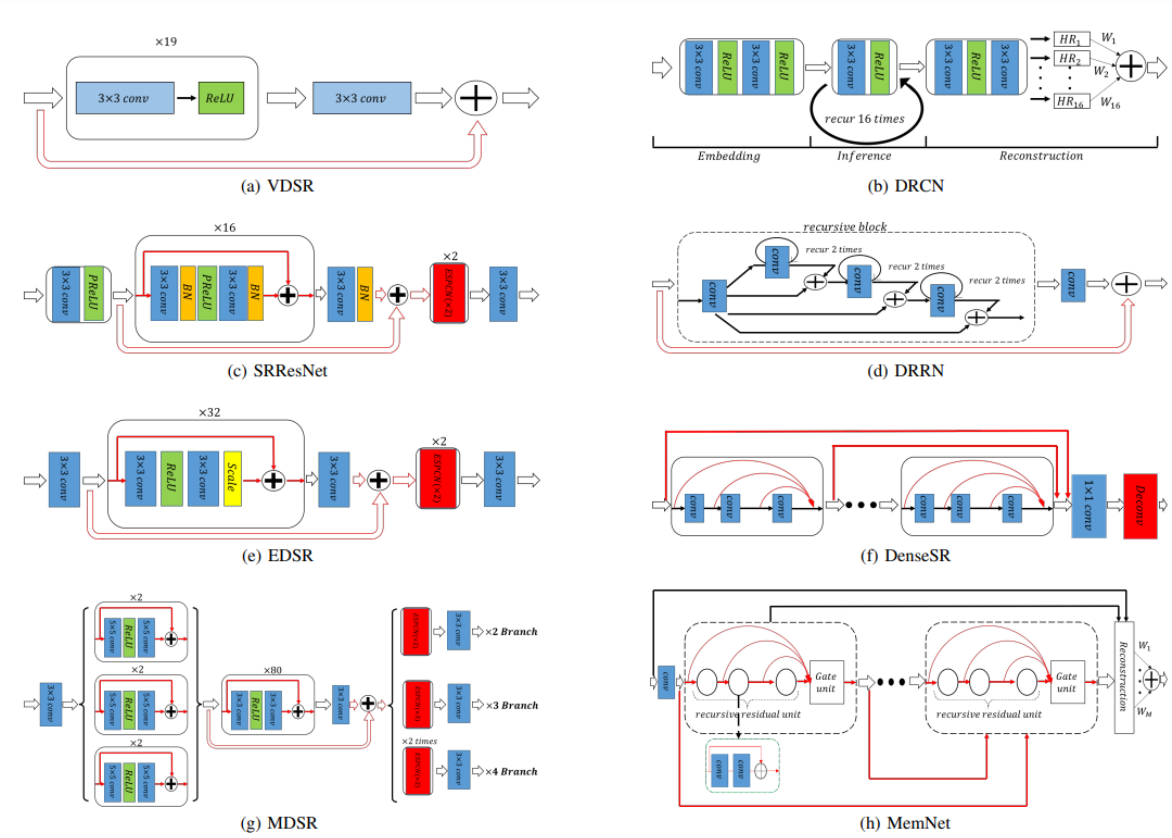


Figure 2.7: Architectures of State of the Art Models [**?**]

Chapter 3
State of The Art Models for Image Super Resolution

A series of experiment were conducted on the 10,000 images of the prepared dataset. To propose the new state of art results, various existing state of art models for Image Super Resolution were tested to get the idea of the network architecture that impeccably learns the under lying features of Capsule Endoscopy Images.
We trained on SRCNN [**?**], SRGAN [**?**], CycleGAN [**?**].

3.1 Super Resolution Generative Adversarial Network

The Generative Adversarial Network (GAN) for Super Resolution of images that is offered is called SRGAN. It is the first framework that, can infer natural images that are photorealistic when scaled up by a factor of $\times 4$. To do this, They suggest a perceptual loss function composed of an adversarial loss and a content loss. A discriminator network that is trained to distinguish between super-resolved images and original photo-realistic images pushes the solution to the natural image manifold in response to the adversarial loss. also employ content loss, which is motivated by perceptual similarity rather than pixel-space similarity.

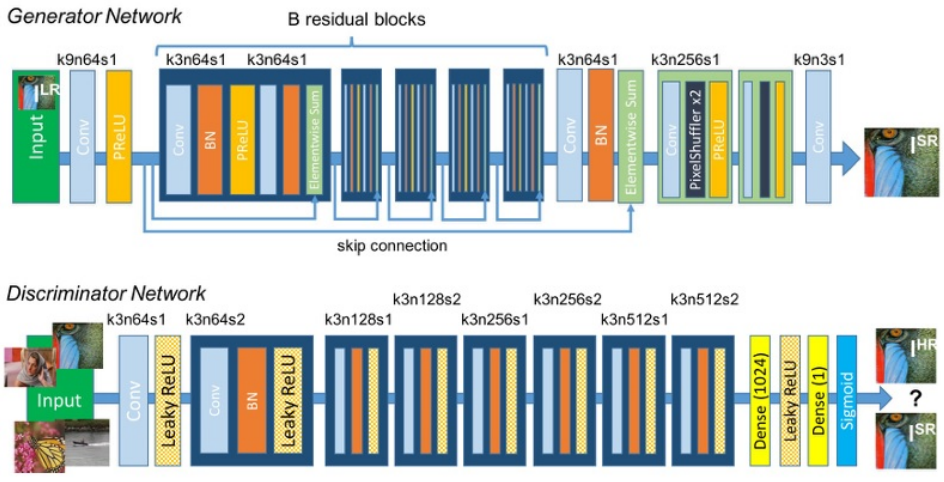


Figure 3.1: Architecture of SRGAN. [**?**]

3.2 Loss Function of SRGAN

3.2.1 Perceptual loss function

$$l^{SR} = \underbrace{\underbrace{l^{SR}_{\mathrm{X}}}_{\mathrm{X}} + \underbrace{10^{-3} l^{SR}_{Gen}}_{\text{adversarial loss}}}_{\text{perceptual loss (for VGG based content losses)}} \tag{3.1}$$

3.2.2 Content loss function

The Pixel-wise loss function is given as:

$$l^{SR}_{MSE} = \frac{1}{r^2 WH} \sum_{x=1}^{rW} \sum_{y=1}^{rH} \left(I^{HR}_{x,y} - G_{\theta_G}\left(I^{LR}\right)_{x,y} \right)^2 \tag{3.2}$$

This is the most widely used optimization target for image SR on which many state-of-the-art approaches rely on it. However, while achieving particularly high PSNR, solutions of MSE optimization problems often lack high frequency content which results in perceptually unsatisfying solutions with overly smooth textures.

3.2.3 Adversarial loss function

In addition to the content losses described so far, we also add the generative component of our GAN to the perceptual loss. This encourages our network to favor solutions that reside on the manifold of natural images, by trying to fool the discriminator network. The generative loss l^{SR}_{Gen} is defined based on the probabilities of the discriminator $D_{\theta_D}\left(G_{\theta_G}\left(I^{LR}\right)\right)$ over all training samples as:

$$l^{SR}_{Gen} = \sum_{n=1}^{N} -\log D_{\theta_D}\left(G_{\theta_G}\left(I^{LR}\right)\right) \tag{3.3}$$

3.3 CycleGAN

CycleGAN presents an approach for learning to translate an image from a source domain X to a target domain Y in the absence of paired examples. Our goal is to learn a mapping G : X $\rightarrow$ Y such that the distribution of images from G(X) is indistinguishable from the distribution Y using an adversarial loss. Because this mapping is highly under-constrained, we couple it with an inverse mapping F : Y $\rightarrow$ X and introduce a cycle consistency loss to enforce F(G(X)) $\approx$ X (and vice versa).

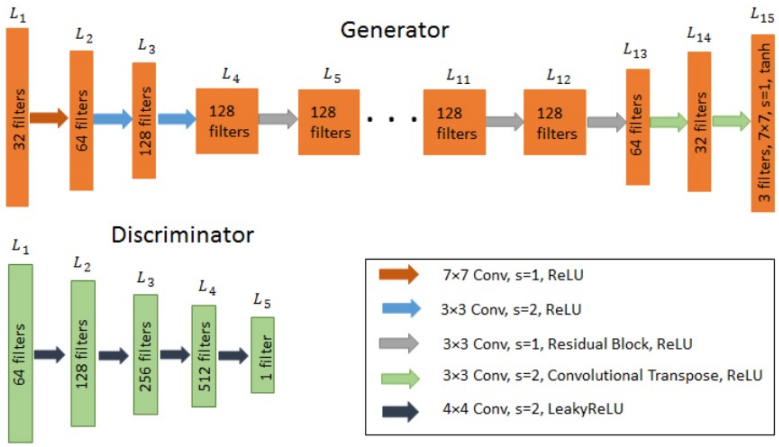


Figure 3.2: Architecture of CycleGAN. [**?**]

3.3.1 Loss in CycleGAN

Adversarial Loss

$$\begin{aligned}\mathcal{L}_{\text{GAN}}\left(G, D_Y, X, Y\right) = &\mathbb{E}_{y \sim p_{\text{data}}(y)}\left[\log D_Y(y)\right] \\ &+ \mathbb{E}_{x \sim p_{\text{data}}(x)}\left[\log\left(1 - D_Y(G(x))\right)\right]\end{aligned} \tag{3.4}$$

where G tries to generate images G(x) that look similar to images from domain Y , while D_Y aims to distinguish between translated samples G(x) and real samples y. G aims to minimize this objective against an adversary D that tries to maximize it, i.e., $min_G\ max_{DY}\ L_{GAN}$(G,D_Y,X,Y). We introduce a similar adversarial loss for the mapping function F : Y $\rightarrow$ X and its discriminator D_X as well: i.e., $min_F\ max_{DX}$ L_{GAN}(F,D_X,Y,X)

Cycle Consistency Loss

$$\begin{aligned}\mathcal{L}_{\text{cyc}}\left(G, F\right) = &\mathbb{E}_{x \sim p_{\text{data}}(x)}\left[\|F(G(x)) - x\|_1\right] \\ &+ \mathbb{E}_{y \sim p_{\text{data}}(y)}\left[\|G(F(y)) - y\|_1\right]\end{aligned} \tag{3.5}$$

Final Objective

$$\begin{aligned}\mathcal{L}\left(G, F, D_X, D_Y\right) = &\mathcal{L}_{\text{GAN}}\left(G, D_Y, X, Y\right) \\ &+ \mathcal{L}_{\text{GAN}}\left(F, D_X, Y, X\right) \\ &+ \lambda \mathcal{L}_{\text{cyc}}\left(G, F\right)\end{aligned} \tag{3.6}$$

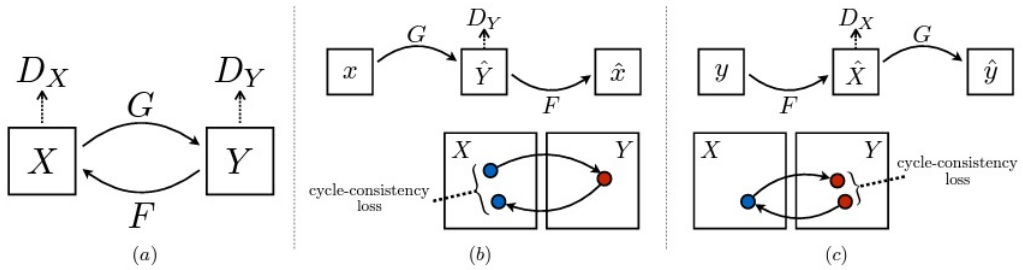


Figure 3.3: (a) CycleGAN model contains two mapping functions G : X → Y and F : Y → X, and associated adversarial discriminators D_Y and D_X. D_Y encourages G to translate X into outputs indistinguishable from domain Y , and vice versa for D_X and F. To further regularize the mappings, we introduce two cycle consistency losses that capture the intuition that translates from one domain to the other and back again we should arrive at where we started: (b) forward cycle-consistency loss: x → G(x) → F(G(x)) ≈ x,and (c) backward cycle-consistency loss: y → F(y) → G(F(y))≈ y . [**?**]

3.4 SR Dense-Net

The proposed network aims to learn an end-to-end mapping function F between the LR image and the HR image. As shown in Fig. **??**. SRDenseNet can be decomposed into several parts: the convolution layer for learning low-level feature, the blocks of DenseNet for learning highlevel features, the deconvolution layers for learning upscaling filters and the reconstruction layer for generating the HR output. Each convolution or deconvolution layer is followed by a ReLu layer for nonlinear mapping except the reconstruction layer. The ReLu activation function is applied element-wise.

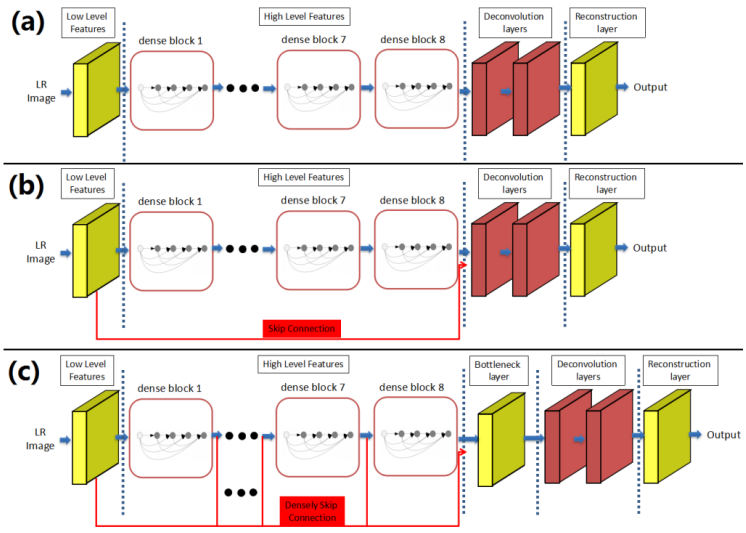


Figure 3.4: Architectures of different SR-Dense Models

Different structures of the proposed networks.
(a) SRDenseNet H: only the high-level feature maps are used as input for reconstructing the HR images.
(b) SRDenseNet HL: the low-level and the high-level features are combined as input for reconstructing the HR images.
(c) SRDenseNet All: all levels of features are combined via skip connections as input for reconstructing the HR images.
We minimize the following Mean Squared Error (MSE). Adam is used to find the optimum weights and biases in the above equation. In the following, we will describe the details of the proposed network structures.

3.4.1 SR Dense-Net blocks

After applying a convolution layer to the input LR images for learning low-level features, a set of DenseNet blocks are adopted for learning the high-level features.

In the structure of DenseNet, short paths are created between a layer and every other layer. This strengthens the flow of information through deep networks, thus alleviating the vanishing-gradient problem. In addition, DenseNet can substantially reduce the number of parameters through feature reuse, thus requiring less memory and computation to achieve high performance [7]. Here, we employ the DenseNet structure as a building block in our network. The structure of each denseNet block can be seen in Fig. **??** Specifically, there are 8 convolution layers in one DenseNet block in our work. If each convolution layer produce k feature maps as output, the total number of feature maps generated by one DenseNet block is k 8, where k is refered to as growth rate. The growth rate k regulates how much new information each layer contributes to the final reconstruction. To prevent the network from growing too wide, the growth rate k is set to 16 in this study. This results in a total number of 128 feature maps from one DenseNet block.

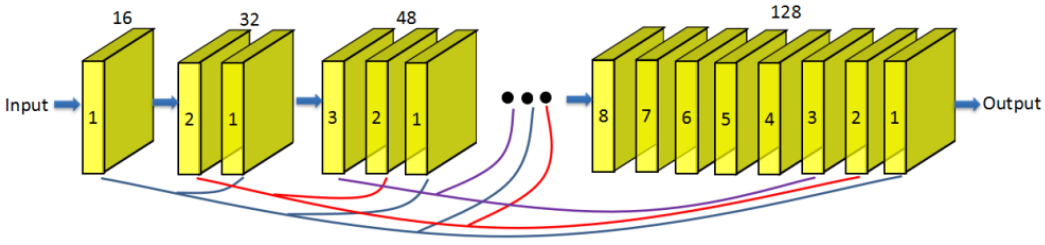


Figure 3.5: The structure of one DenseNet block.

3.5 RCAN

Most recent CNN-based methods treat channel-wise features equally, which lacks flexibility in dealing with different types of information. Image SR can be viewed as a process, where we try to recover as more high-frequency information as possible. The LR images contain most low-frequency information, which can directly forwarded to the final HR outputs. While, the leading CNN-based methods would treat each channel-wise feature equally, lacking discriminative learning ability across feature channels, and hindering the representational power of deep networks.

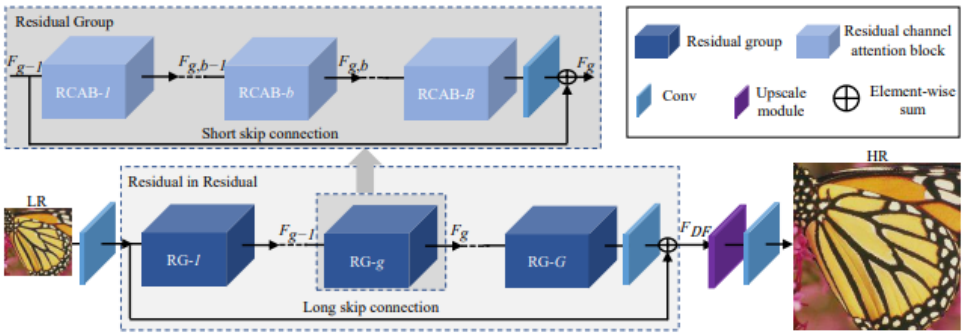


Figure 3.6: Network Architecture of RCAN

After conducting series on experiments on state of art models, we came to an conclusion that the modalities of capsule endsocopy dataset are very well learned and captured by networks with dense connections.

Chapter 4
Propsed Work

As the existing state of art model were for natural images, we trained the existing model on wireless capsule endoscopy dataset to check which kind of network is best suitable for the WCE images. after carefully studying the outputs of SOTA models, we reached a conclsion that the network having desne connections learns this images very well. Mean while we came across a reseach paper based on CycleCNN [**?**] to accomplish the task of image super resolution. we took CycleCNN as the base network and started the experimental work.

4.1 CycleCNN

Initially CycleCNN was proposed for unsupervised training method, we made relevant changes in the network to train it using supervised training method. The network architecture for the cycleCNN is shown in the Fig. **??** and Fig. **??**

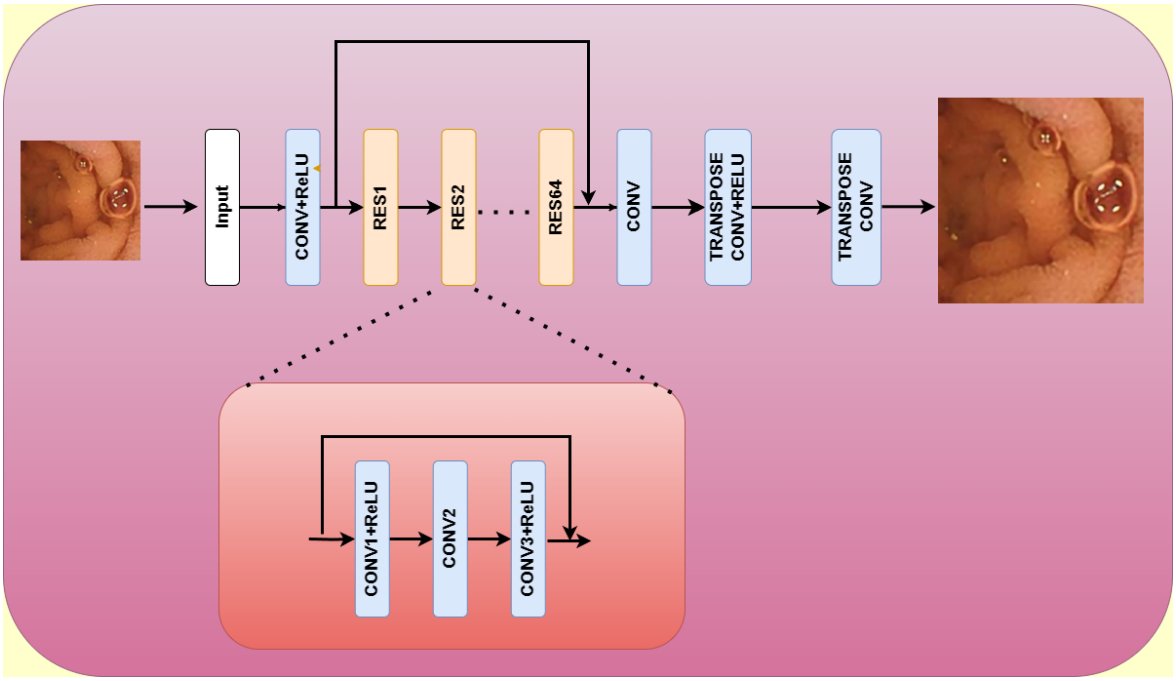


Figure 4.1: Upsampling Network of CycleCNN

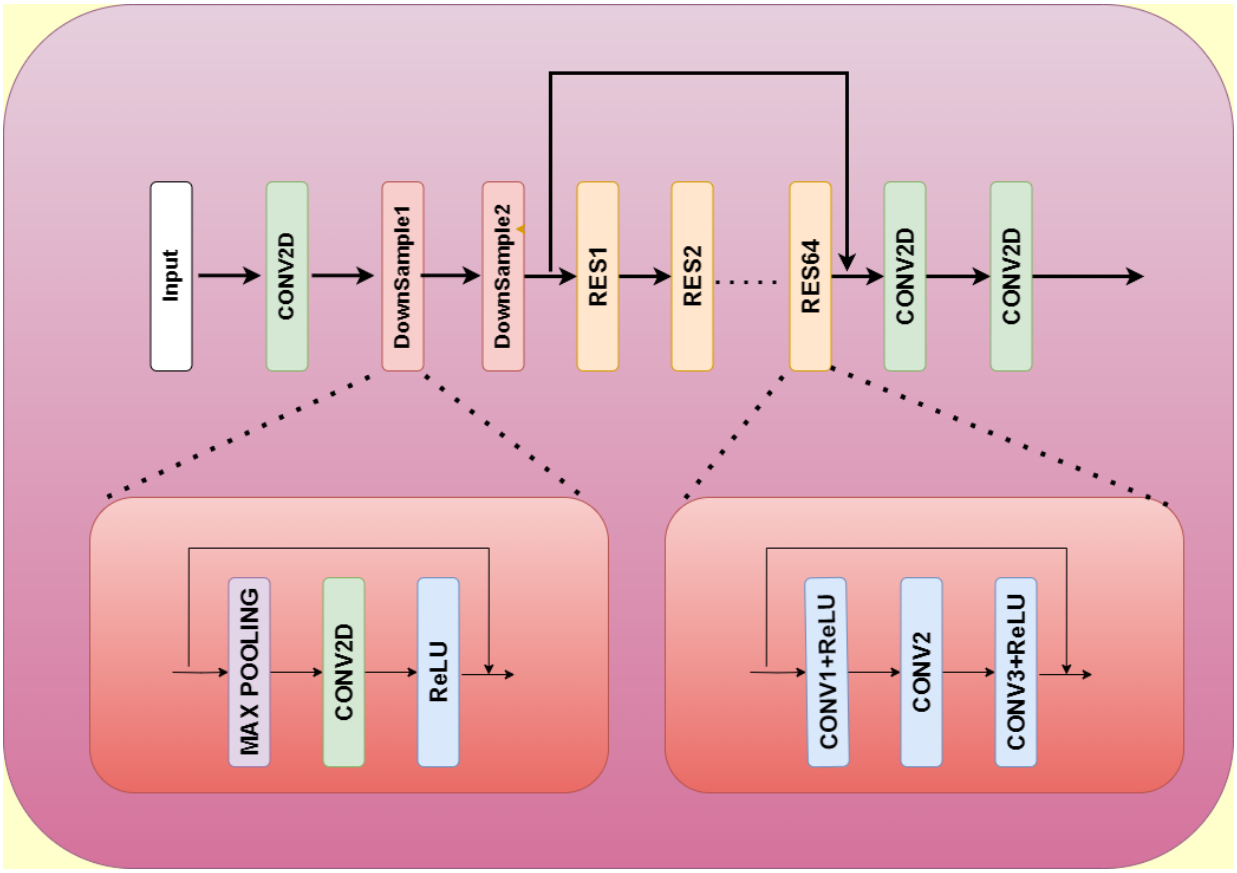


Figure 4.2: Downsampling Network of CycleCNN

Where each convolution layer has 3x3 kernel size with stride 1 and padding 1, while the transpose convolution layer has kernel size 3x3 with stride 2 and padding 1.

4.1.1 CycleCNN-MSE

The training of CycleCNN was performed on different loss functions. Firstly, it was trained on mean-squared error (MSE) loss. The same architecture was parallelly trained on cycle loss as well as MSE loss. The equations for MSE loss and Cycle loss is defined in equation **??** and **??** respectively.

$$L_{MSE} = \sum_{i=1}^{D} (x_{ti} - y_{ti})^2 \tag{5.1}$$

$$L_{cyc} = \frac{1}{N} \sum_{i=1}^{N} (\|G_2(G_1(x_{ti})) - x_{ti}\|_2 + \|G_1(G_2(y_{ti})) - y_{ti}\|_2) \tag{5.2}$$

The results for training on MSE loss as well as MSE and cycle loss are given the Fig **??**

4.2 Dense CycleCNN with GRL Block

By comparing the State of art models, it is evident that the modalities of capsule endoscopy data was very well learnt by dense connections(DenseNet). Dense connections are able to capture the high frequency details which are important for our case. To improve the results obtained from CycleCNN we introduced dense connections in the model.Global Resudial Learning(GRL) block was also introduced to resolve the issue of gradinet vanishing and to stabalize the training. The GRL block is used to propogate the input directly at the end of the model. The network architecture for this model is given in Fig. **??** and Fig. **??**

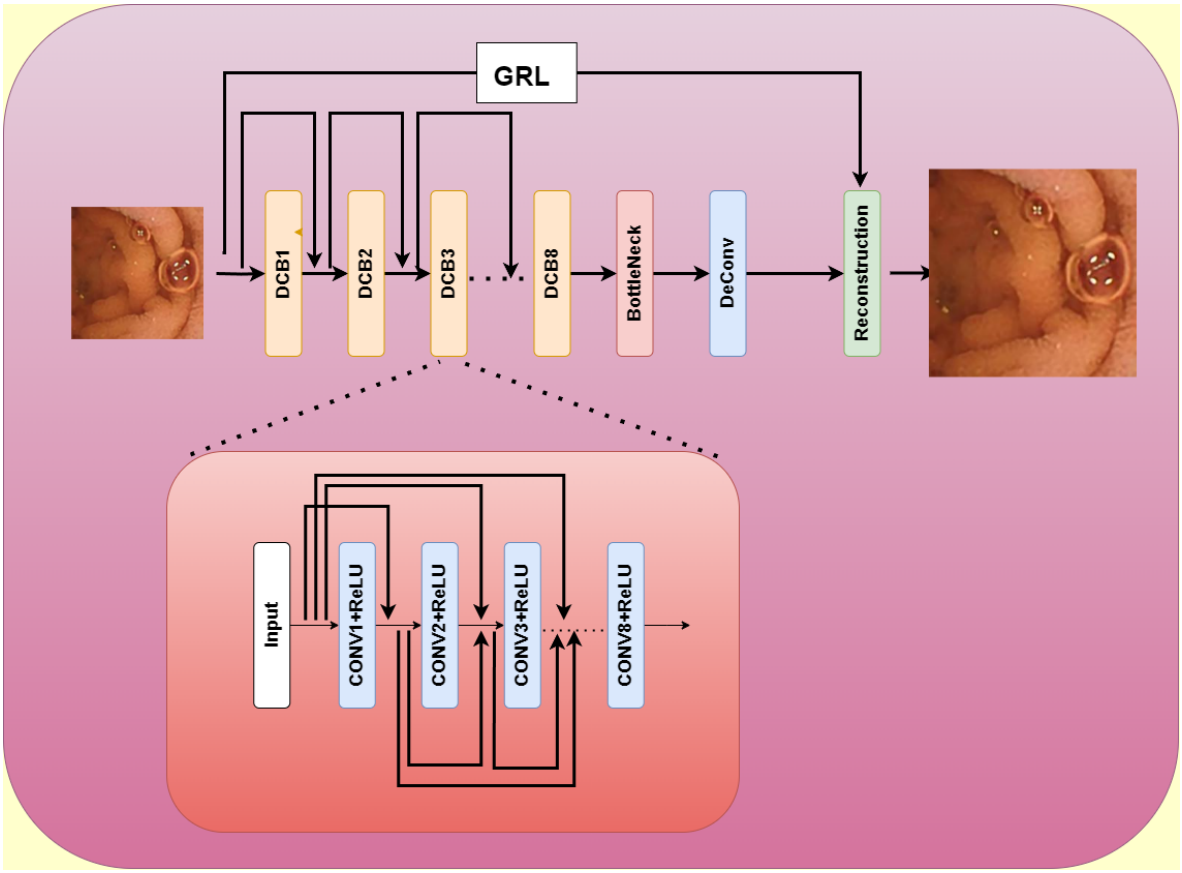


Figure 4.3: Upsampling Network for Dense CycleCNN.

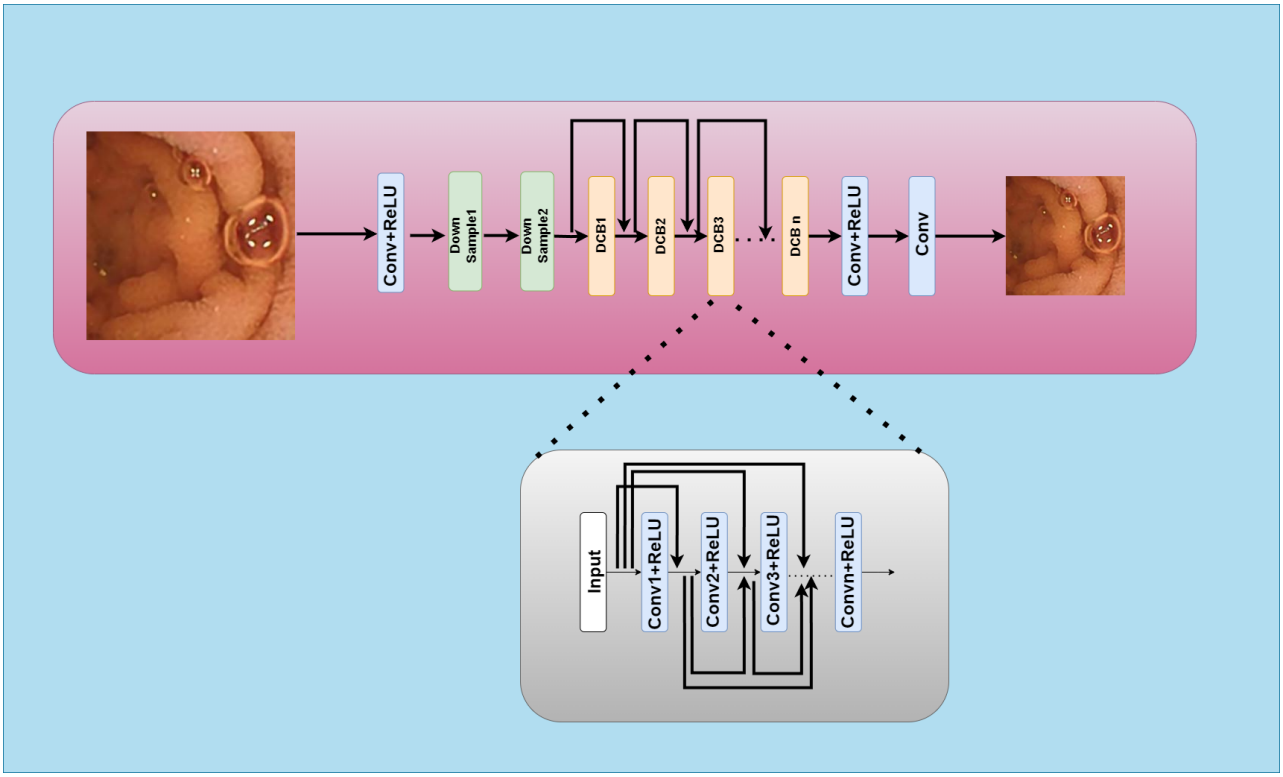
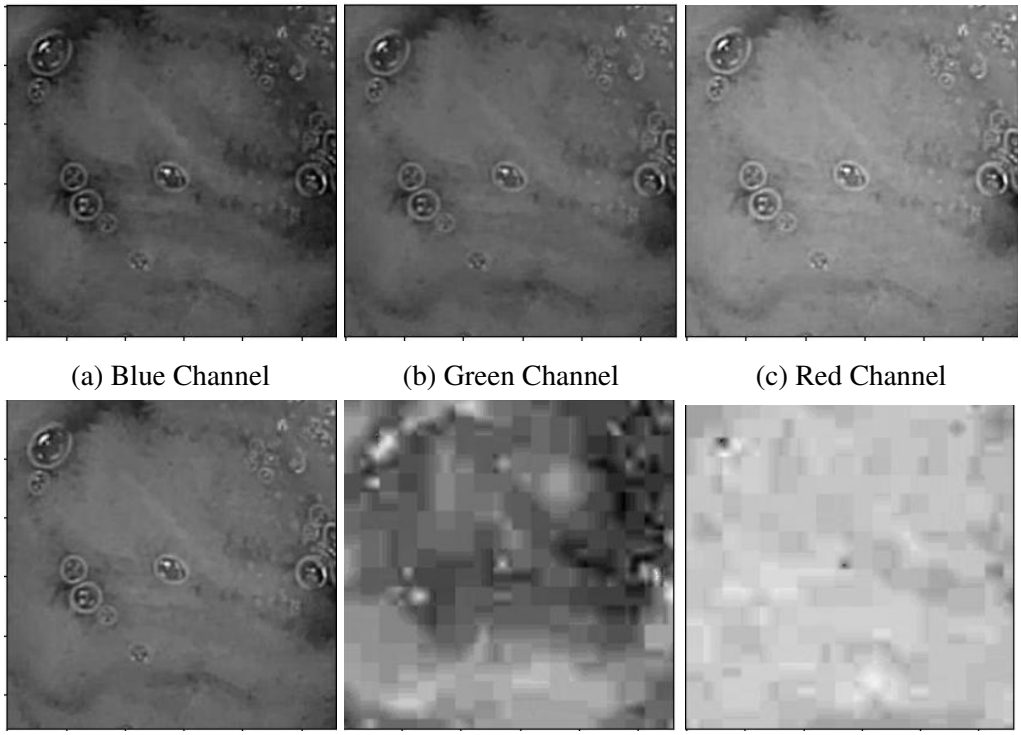


Figure 4.4: Downsampling Network for Dense CycleCNN.

4.3 Dense CycleCNN with GRL block on YCbCr Colour space

Due to increasing network complexity and training parameters, model training was getting very much slower, to speed up the model training we decided to train the network on Y-channel of YCbCr colour space. It has been shown in paper that the high frequency details in the image are easily captured in the Y-channel. Hence the image is first converted into from RGB colour space to YCbcr colour space and only training is done only on the Y-channel while Cb and Cr components are obtained by simple Bicubic interpolation.
The RGB components as well as the YCbCr components are shown in the Fig. **??**



(a) Blue Channel (b) Green Channel (c) Red Channel

(d) Y Channel (e) Cb Channel (f) Cr Channel

Figure 4.5: Image in different colour spaces.

Training on Y-channel of YCbCr colour space reduces the training parameters drastically. The results obtained after training the Dense CycleCNN model on Y channel are shown in Fig. **??**

4.3.1 Dense CycleCNN with GRL Block on Patch

Initially the training was done on whole image of 280x280. But the training was slow and it was exceeding the GPU memory. So inspired from the SRGAN [**?**], we did the training of the netwok on random patch of the image. The patch size taken is 100x100. The patch training helped in training the network really fast with same results as that of training on whole image.

4.4 DenseNET with Channel Attention Block (DCAN)

After testing various training approches and architectures mentioned above, we used the knowledge gained through these approches to design a new architecture namely DCAN-DenseNET with Channel Attention. This newtork takes into consideration the effect of Channel Attention Block(CAB) from the RCAN [**?**] and the idea of dense connections from SR-Densenet [**?**] as they proved to be effective in capturing the high frequency details from the capsule data.

4.4.1 Architecture Details

The Channel Attention Block is used to focus the training on the important features of the image. Channel attention is able to extract the important regions of the image by exploiting the inter-channel relation of the features.

For training purpose, we have taken 8 dense blocks(i.e. num_blocks=8), each consisting of 8 dense layers(num_ layers=8). The number of features in consecutive dense blocks is governed by the growth rate. We have used the growth rate of 16. The number of input features and out features for the i^{th} dense block is given by (growth_ rate * num_ layers * (i + 1), growth_ rate).
The bottle neck layer is used to compress the output features from dense connection to a lower dimension. Here, we are reducing the features to 256 features using the bottle neck layer. The deconv layer is used to upsample the images. It consist of two pixel shuffle layers each layer upsamples the image by a factor of x2. We are using pixel shuffle instead of transpose convolution because it is observed that the transpose convolution introduces the checker board effect while upsampling the image. The reconstruction layer is a convolution layer that is used to get the images number of channels that we want in our output image(i.e. num_channels=3 for training on RGB channel and num_channel=1 for training on Y channel of YCbCr colour space.) The architecture of the model used is provided in the Fig.**??**

The training is done to minimize the loss function which is the taken as the Mean Squared Loss(MSE). The training is completed for a total of 300 epochs with a batch size of 32. We did the training on patch of image, each of size 100x100, and on the Y channel of YCbCr channel. We already illustrated the benifits of training on patch of image and Y channel of YCbCr above. Adam optimizer with a learning rate of 0.0001. The quantitative analysis and qualitative analysis of the proposed model (DCAN) is shown in upcoming sections.

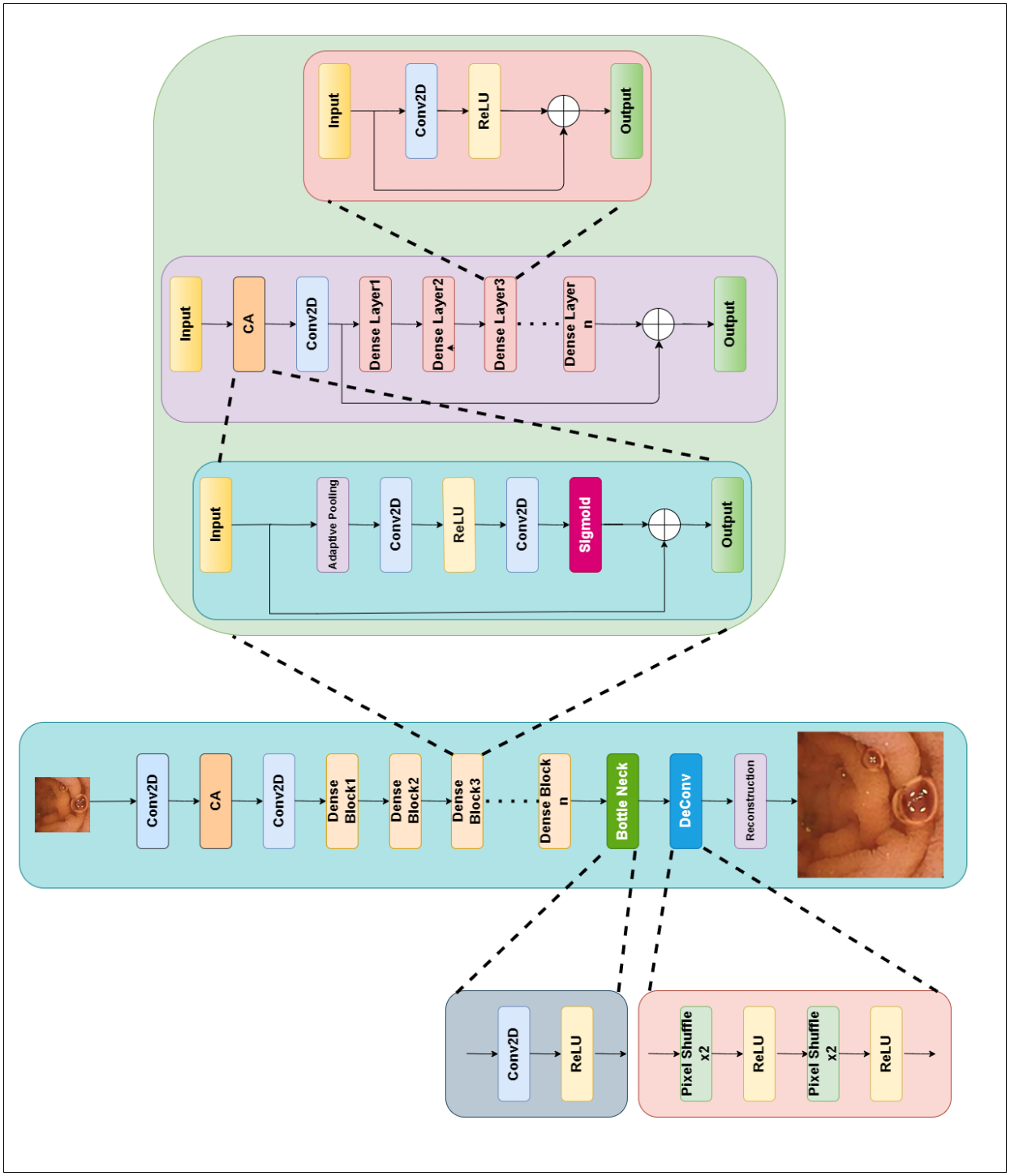


Figure 4.6: DCAN architecture

Chapter 5
Result analysis

This chapter shows the quantitative and qualitative analysis of the images generated from using different approaches.

5.1 Dataset

All the experimental work was carried out on the Kvasir Capsule Endoscopy Dataset [**?**]. A segment of Kvasir-Capsule, a video capsule endoscopy data set, is used for this project. Each image has 336x336 dimensions and is in RGB colour code. There are total of 47236 images belonging to different sub-categories of medical anomalies. The original dataset had the images as shown in Table **??**

Table 5.1: Table of original data set.

Normal clean mucosa	34338
Ileocecal valve	4189
Reduced mucosal view	2096
Pylorus	1529
Angiectasia	866
ulcer	854
Foreign body	766
Lymphangiectasia	592
Erosion	506
Blood - fresh	446
Erythema	159
Polyp	55
Blood - hematin	12
ampulla of water	10

The Experimentation was carried on multiple subsets of the original Dataset. Initially 10,000 training images were randomly sampled from the original set with 1000 testing images. Few sample images from the dataset are given in Fig.**??**

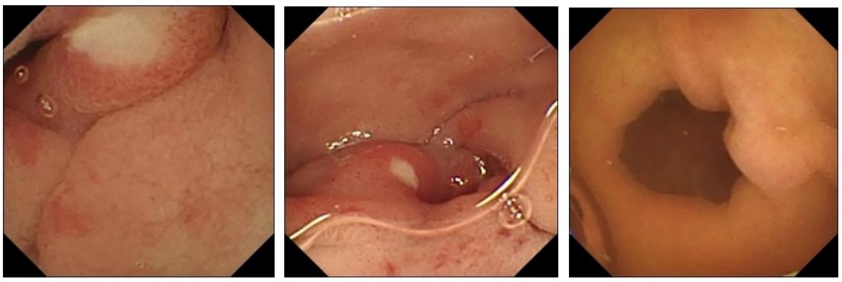


Figure 5.1: Glimpse of uncropped capsule images. [**?**]

As we can see in the sample images Fig.**??** that the corner of the images are complete black spaces, while feeding this data to the image super resolution network for learning purposes, the network was not able to learn well and an semi transparent white layer was observed on the output images. To overcome this problem, we feed the network with cropped images, that didn't have any blank spot in the corners. The glimpse of that dataset is shown in Fig.**??**

Figure 5.2: Glimpse of Cropped dataset. [**?**]

As Image super resolution is a comperatively complex task, the network was not able to perform well when the input data belonged to multiple classes, as the underlying spartial information was different for different classes. Hence a new dataset was developed which was the subset of original dataset of cropped images. The final training of all the models/networks was done on this dataset. After cropping, the resolution of the cropped images is 228×228 pixels, which was 336×336 pixels originally.

5.2 Experimental analysis on SOTA models

To set the basline for the proposed model, we trained the existing state of art models with our dataset and generated the results using those models, the results are shown in following subsections.

5.2.1 Training Details of SRGAN

On an NVIDIA Tesla GPU, we trained each network using 10,000 photos from the Kvasir Dataset [**?**]. Compared to the test images, these pictures are different. By employing a bicubic kernel and a downsampling factor of x4, we were able to derive the LR images from the HR images (BGR, C = 3). Randomly select 16 HR subimages of size 96x96 from various training images for each mini-batch were selected. It should be noted that since the generator model is completely convolutional, it may be applied to images of any size. The HR picture was scaled in range to [1,1] and the LR input image range to [0,1]. In order to determine the MSE loss, pictures with an intensity range of [1,1] were used. In order to get VGG losses on a par with MSE losses, VGG feature maps were additionally rescaled by a factor of 11.25 percentage. This is equal to adding a rescaling factor of 0.006. The Adam optimizer [**?**] was used with 1 = 0.9 for optimization. The SRResNet networks were trained using 106 update iterations and a learning rate of 0.0001. To prevent undesirable local optima, They learned MSE-based SRResNet network as initialization for the generator when training the actual GAN. A total of 105 update iterations at a learning rate of 104 and an additional 105 iterations at a slower rate of 105 were used to train all SRGAN versions. We alternate the updates to the discriminator and generator networks, which is equal to GAN's [**?**] k = 1.

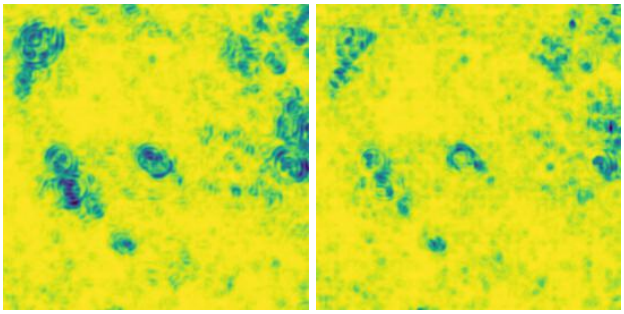
5.2.2 Result of SRGAN

The model was trained at the above mentioned parameters and the below mentioned results were obtained. The PSNR of SRGAN for cropped data was recorded as 37.217 dB which is comparatively better compared to Bicubically upsampled image having PSNR of 37.1 dB , while that for uncropped data was recorded as 36.396 dB which is comparatively better compared to Bicubically upsampled image having PSNR of 35.36 dB. SSIM score was also increased compared to the Bicubically upsampled images. The details of SSIM and PSNR for cropped as well as uncropped images is shown in table **??**. And the output results can be seen in Fig. **??**.

Table 5.2: PSNR and SSIM for SRGAN.

Method	PSNR(dB)	SSIM(dB)	Dataset
BICUBIC	35.36	0.88	uncropped
SRGAN	36.396	0.9083	uncropped
BICUBIC	37.26	0.9	cropped
SRGAN	37.06	0.9	cropped

(a) HR (b) Bicubic (c) SRGAN



(d) HR-Bicubic (e) HR-SRGAN

Figure 5.3: SRGAN Generated Image and their SSIM maps: (1) HR image (2) Bicubic Interpolated image (3) SR image (4) SSIM-map (HR-Bicubic) (5)SSIM-map (HR-SR). [**?**] [**?**]

5.2.3 Training Details of CycleGAN

Training details We apply two techniques from recent works to stabilize our model training procedure. First, for L_{GAN}, we replace the negative log likelihood objective by a least-squares loss. This loss is more stable during training and generates higher quality results. In particular, for a GAN loss L_{GAN}(G,D,X,Y), we train the G to minimize the loss and train the Discriminator. Second, to reduce model oscillation , they update the discriminators using a history of generated images rather than the ones produced by the latest generators. We keep an image buffer that stores the 50 previously created images. For all the experiments, We use the Adam solver [**?**] with a batch size of 1. All networks were trained from scratch with a learning rate of 0.0002. We keep the same learning rate for the first 100 epochs and linearly decay the rate to zero over the next 100 epochs.

5.2.4 Results for CycleGAN

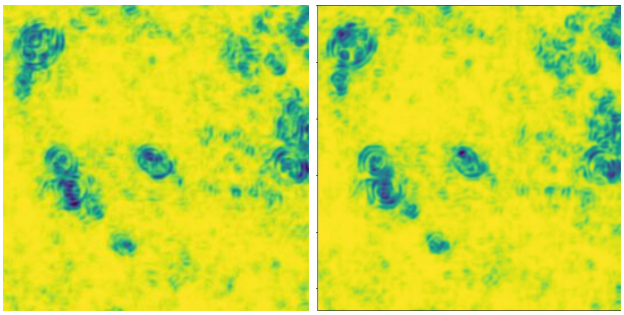
The model was trained at the above mentioned parameters and the below mentioned results were obtained. The PSNR of CycleGAN for cropped data was recorded as 36.9 dB which is comparatively better compared to Bicubically upsampled image having

PSNR of 37.1 dB , while that for uncropped data was recorded as 36.3 dB which is comparatively better compared to Bicubically upsampled image having PSNR of 37.1 dB. SSIM score was also increased compared to the Bicubically upsampled images. The details of SSIM and PSNR for cropped as well as uncropped images is shown in Table **??**

Table 5.3: PSNR and SSIM for CycleGAN.

Method	PSNR(dB)	SSIM(dB)	Dataset
BICUBIC	37.1	0.88	uncropped
CycleGAN	36.3	0.9123	uncropped
BICUBIC	36.9	0.89	cropped
CycleGAN	37.1	0.9123	cropped

(a) HR (b) Bicubic (c) CycleGAN



(d) HR-Bicubic (e) HR-CycleGAN

Figure 5.4: CycleGAN Generated Image and their SSIM maps: (1) HR image (2) Bicubic Interpolated image (3) SR image (4) SSIM-map (HR-Bicubic) (5) SSIM-map (HR-SR) [**?**] [**?**]

5.2.5 Training details of SR-Densenet

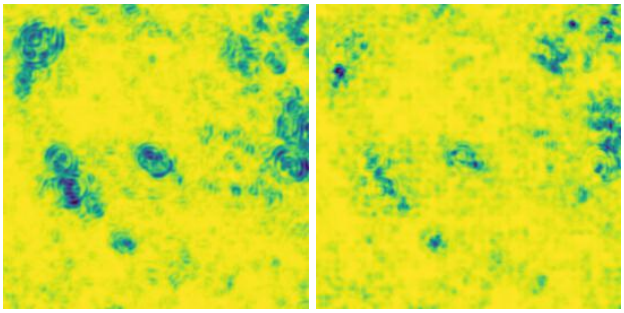
Non-overlapping sub-images with a size of 336 × 336 were cropped in the HR space. The LR images were obtained by downsampling the HR images using bicubic kernel

with a scale factor of 4×. Each image has been transformed into YCbCr space and only the Y-channel was used for training. In all networks, 8 DenseNet blocks were used, resulting in 64 convolution layers. Within each block, a growth rate of 16 was set. This generated an output of 128 feature maps from each block. The filter size was set to 3 × 3 in all weight layers. The rectified linear units (ReLu) was used as the activation function. All the networks were optimized using Adam. The learning rate was initially set to 0.0001. A mini-batch size of 32 was set during the training. The training process stopped after no improvements of the loss was observed after 200 epoches.

Table 5.4: PSNR and SSIM for SR Densenet.

Method	PSNR(dB)	SSIM(dB)	Dataset
BICUBIC	35.36	0.88	uncropped
SR Densenet	32	0.925	uncropped
BICUBIC	37.2	0.9	cropped
SR Densenet	38.86	0.92	cropped

(a) HR (b) Bicubic (c) DenseNET



(d) HR-Bicubic (e) HR-DenseNET

Figure 5.5: SR Densenet Generated Image
SR Densenet Generated Image and their SSIM maps. [**?**] [**?**]

5.2.6 Training details of RCAN

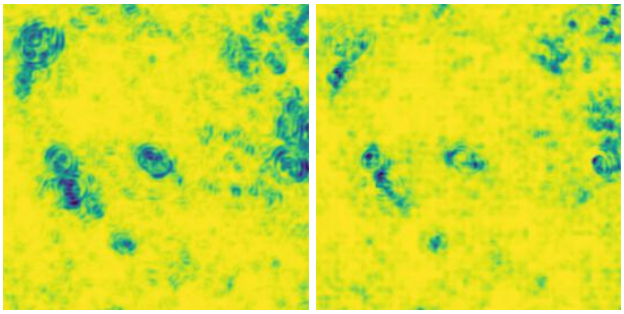
10,000 training images Kvasir Capsule Endoscopy were used as training set. Experiments with conducted using bicubic images. The SR results were evaluated with PSNR and SSIM on Y channel of transformed YCbCr space. Data augmentation is performed on the 10,000 training images, which are randomly rotated by 90 , 180 , 270 degrees and flipped horizontally. In each training batch, 16 LR color patches with the size of 48×48 are extracted as inputs. Model is trained by adam optimizer. The initial leaning rate was set to 0.0001 and then decreases to half every 2×100000 iterations of back-propagation. PyTorch was used to implement our models.

5.2.7 Results of RCAN

Table 5.5: PSNR and SSIM for RCAN.

Method	PSNR(dB)	SSIM(dB)	Dataset
BICUBIC	35.36	0.88	uncropped
RCAN	31.8	0.91	uncropped
BICUBIC	37.1	0.89	cropped
RCAN	39.47	0.92	cropped

(a) HR (b) Bicubic (c) RCAN



(d) HR-Bicubic (e) HR-RCAN

Figure 5.6: RCAN generated images
RCAN Generated Image and their SSIM maps: (1) HR image (2) Bicubic Interpolated image (3) SR image (4) SSIM-map (HR-Bicubic) (5)SSIM-map (HR-SR)

5.3 Experimental Analysis on proposed models

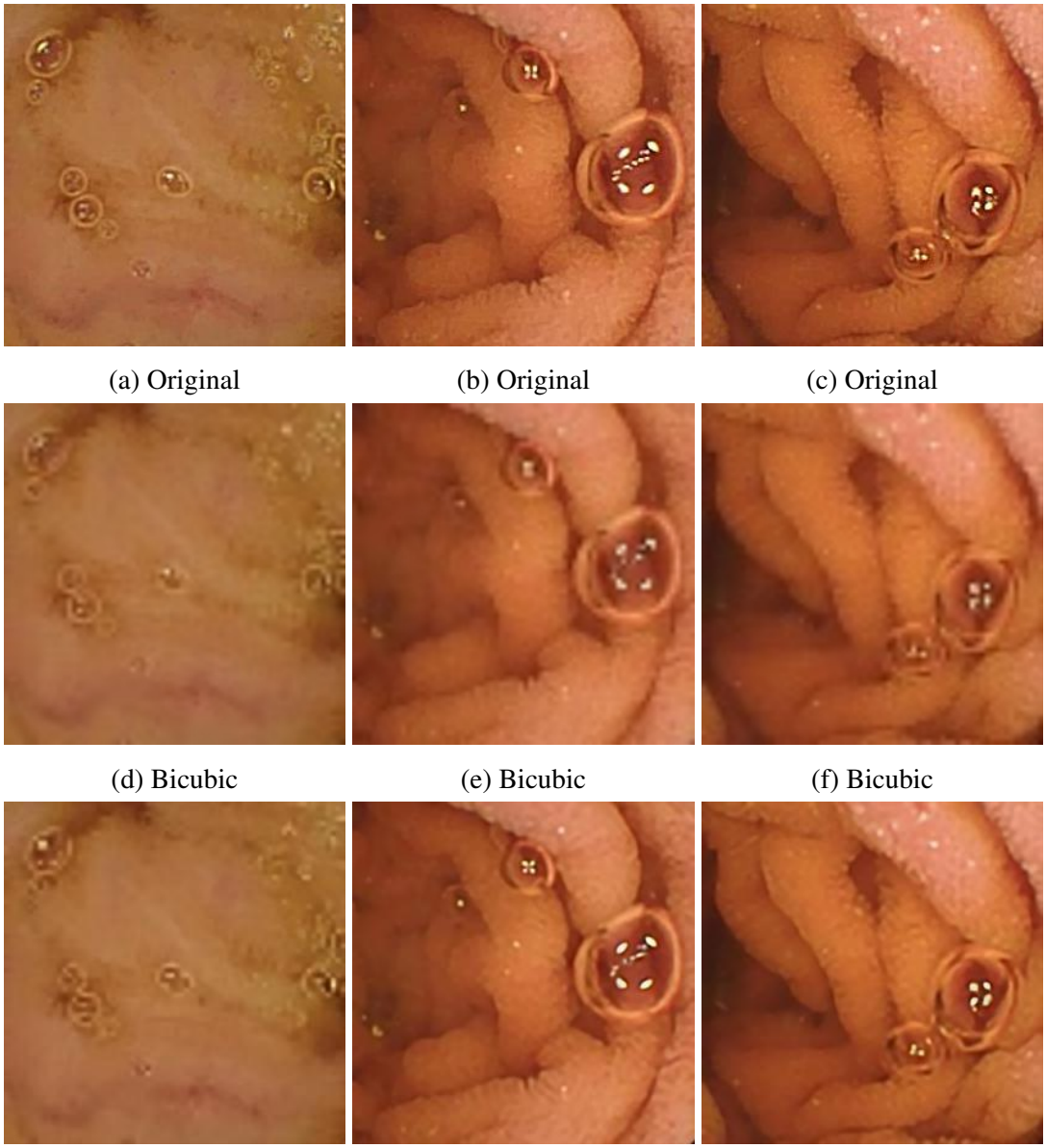
The results obtained from each model is shown subsequent figures. Fig **??** shows the images from cycleCNN on mse loss and Fig **??** shows the images from dense cycleCNN training. Fig **??** shows images from dense cycleCNN on Y channel of YCbCr scolour space and Fig **??** represents the images generated by training the dense cycleCNN network on patch of 100x100 pixels.

(a) Original (b) Original (c) Original

(d) Bicubic (e) Bicubic (f) Bicubic

(g) Cycle MSE (h) Cycle MSE (i) Cycle MSE

Figure 5.7: Images generated using MSE loss on CycleCNN Model.



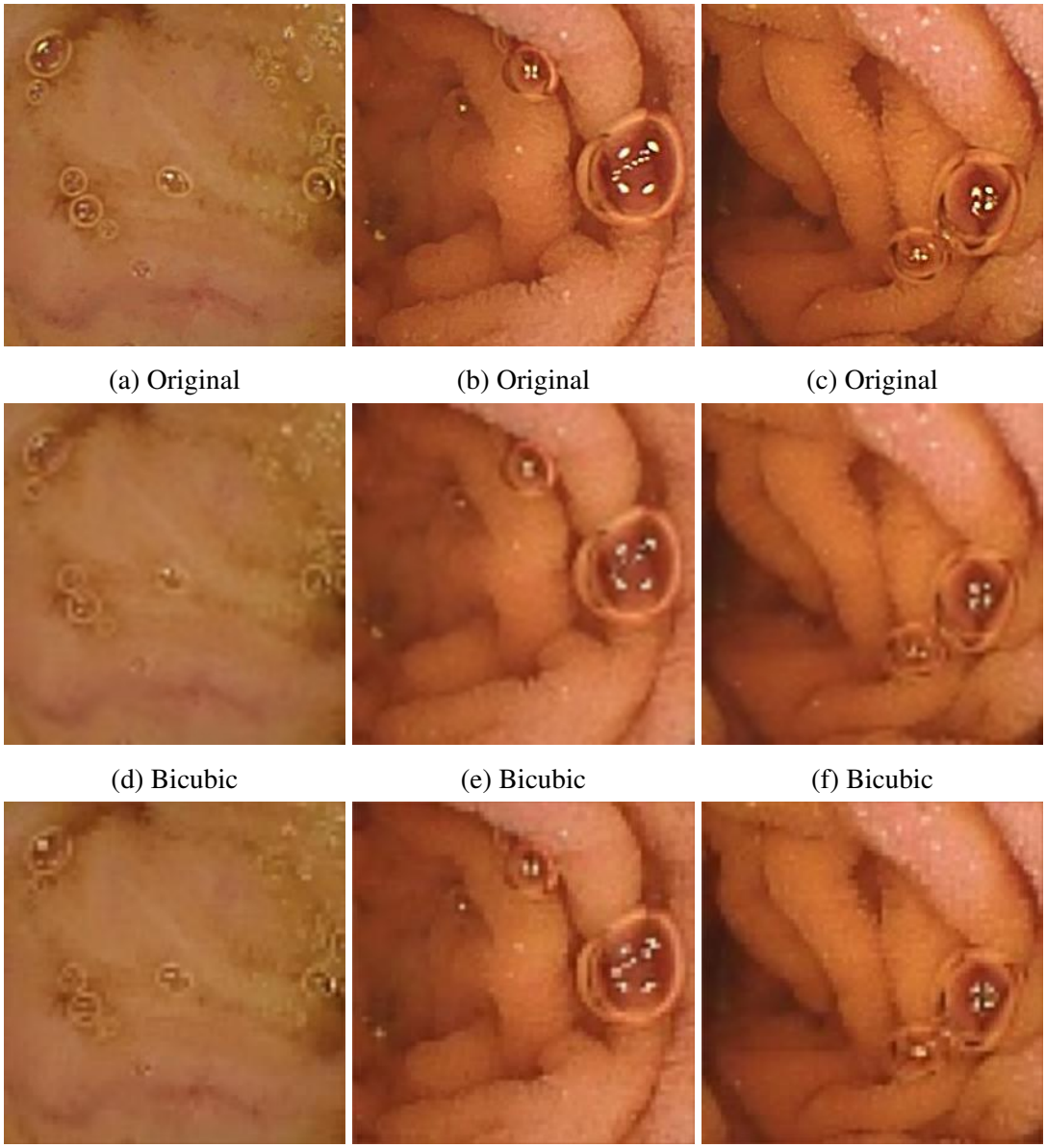
(a) Original (b) Original (c) Original

(d) Bicubic (e) Bicubic (f) Bicubic

(g) Dense GRL (h) Dense GRL (i) Dense GRL

Figure 5.8: Images generated Dense CycleCNN Model with GRL Block.

It can be observed that the images generated using dense connections and GRL block are visually more better than the traditional bicubic interpolation method. In bicubic interpolation blur effect is introduced and the high frequency details are not captured, while using this architecture we are able to capture high frequency details efficiently.



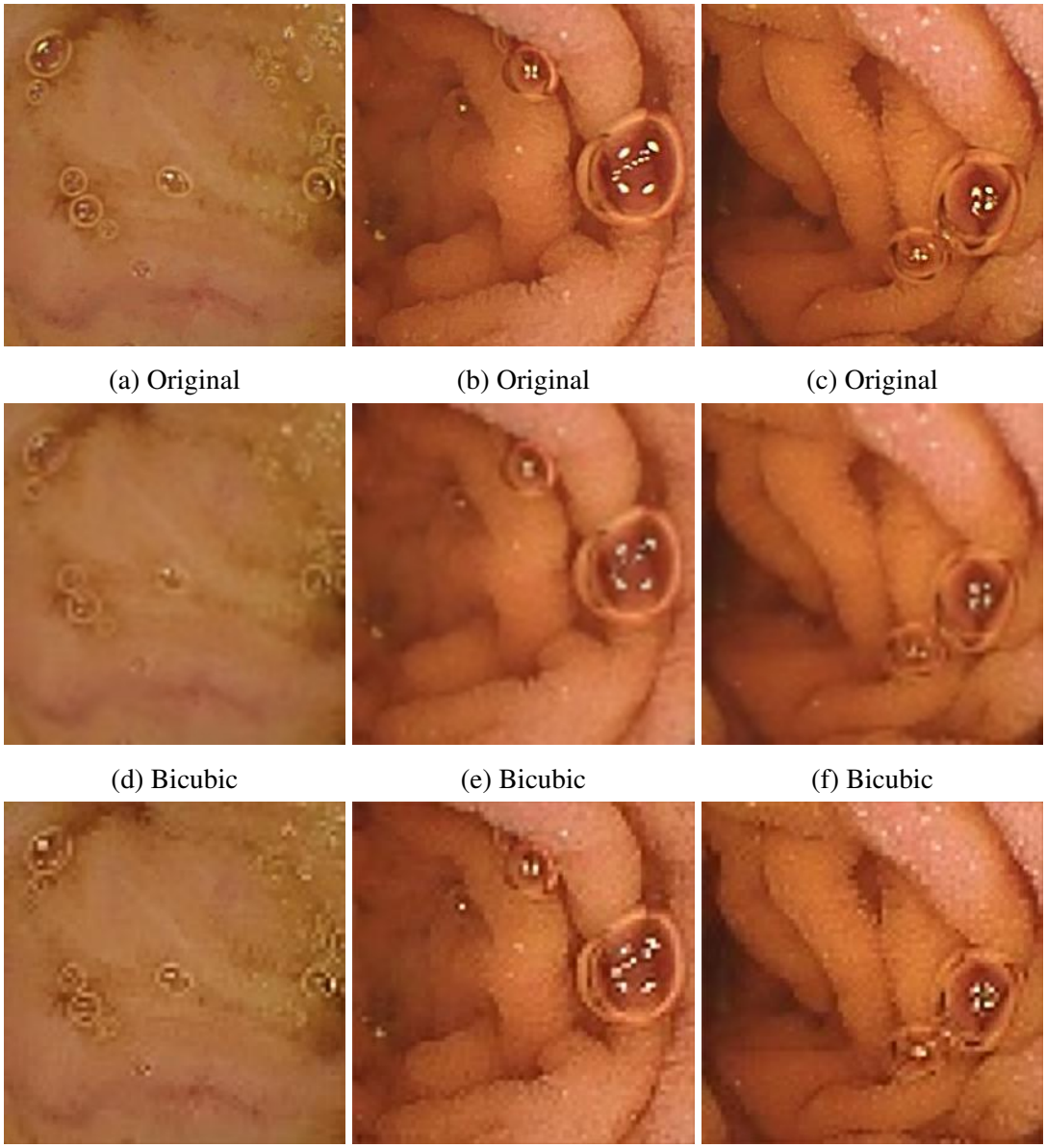
(a) Original (b) Original (c) Original

(d) Bicubic (e) Bicubic (f) Bicubic

(g) DenseGRL YCbCr (h) DenseGRL YCbCr (i) DenseGRL YCbCr

Figure 5.9: Images generated Dense CycleCNN Model with GRL Block on YCbCr.

While training on Y-channel of YCbCr colour space, the network is able to outperform the traditional bicubic approach and the results are quite similar to that of training done on RGB colour space. The advantage we gain here is the reduced training parameters and faster training of model.



(a) Original (b) Original (c) Original

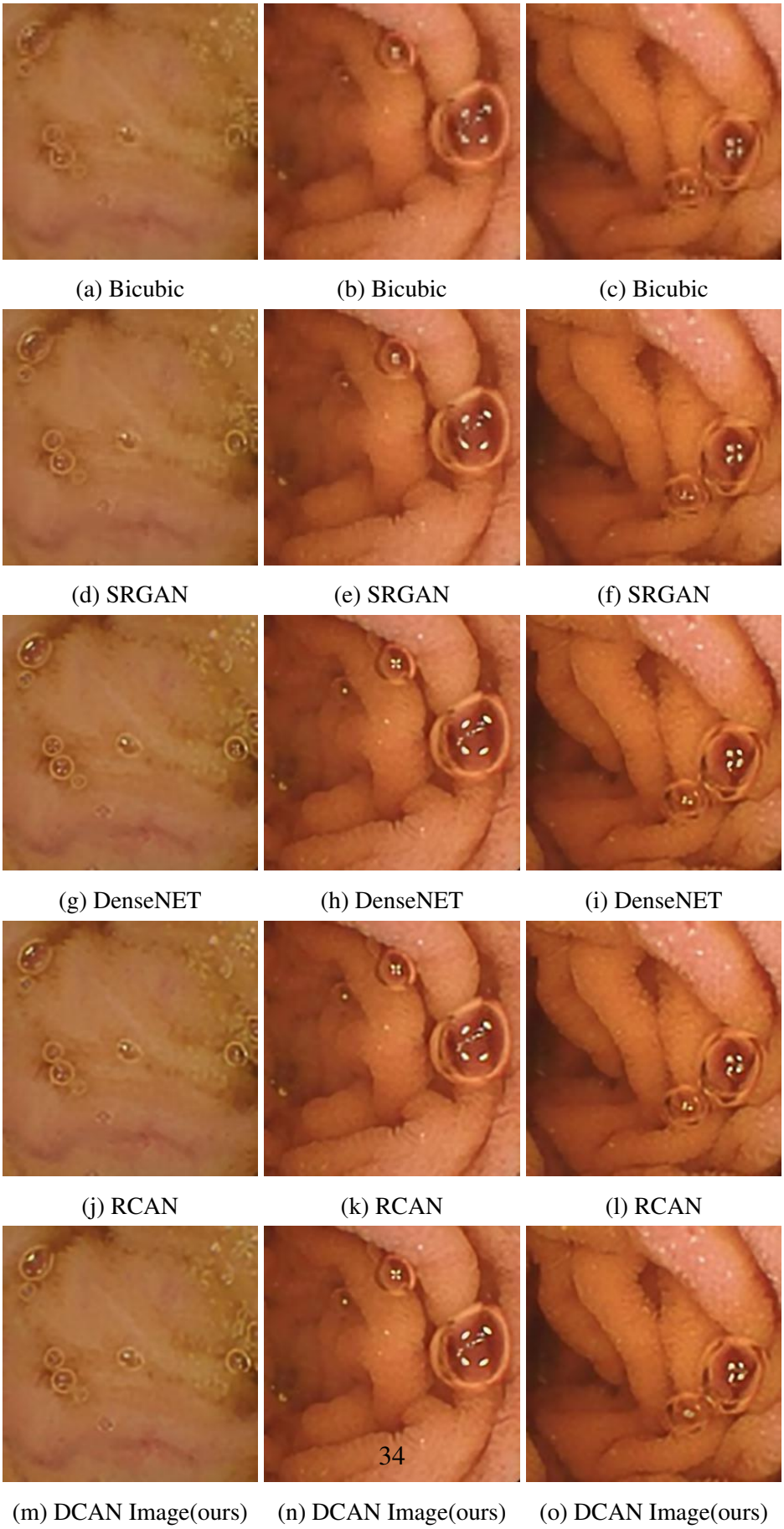
(d) Bicubic (e) Bicubic (f) Bicubic

(g) Patch Dense CycleCNN(YCbCr) (h) Patch Dense CycleCNN(YCbCr) (i) Patch Dense CycleCNN(YCbCr)

Figure 5.10: Images generated Dense CycleCNN Model with GRL Block on YCbCr with patch.

On training the network on patch size of 100x100 and Y-channel of YCbCr colour space, the network is able to beat the traditional bicubic approach and the results are similar to that of training done on RGB colour space and while image of 280x280 pixels. The advantage we gain here is faster training of model and low memory consumption.

5.3.1 DenseNET with Channel Attention Block(DCAN)



(a) Bicubic (b) Bicubic (c) Bicubic

(d) SRGAN (e) SRGAN (f) SRGAN

(g) DenseNET (h) DenseNET (i) DenseNET

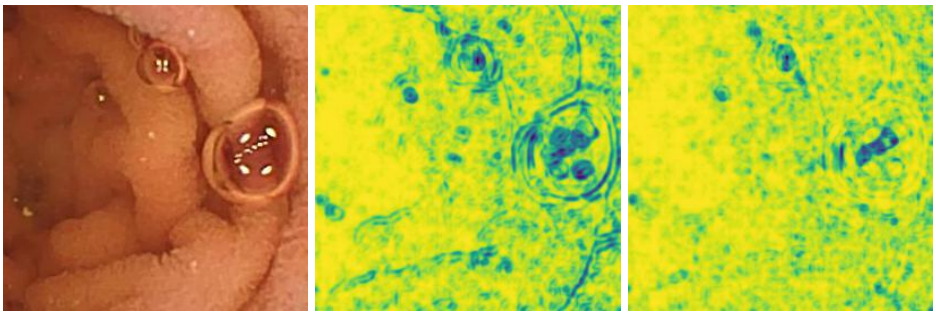
(j) RCAN (k) RCAN (l) RCAN

(m) DCAN Image(ours) (n) DCAN Image(ours) (o) DCAN Image(ours)

Figure 5.11: Comparison of images of proposed network with State of Art models

The training is done to minimize the loss function which is the taken as the Mean Squared Loss(MSE). The training is completed for a total of 300 epochs with a batch size of 32. We did the training on patch of image, each of size 100x100, and on the Y channel of YCbCr channel. We already illustrated the benifits of training on patch of image and Y channel of YCbCr above. Adam optimizer with a learning rate of 0.0001. The quantitative analysis and qualitative analysis of the proposed model (DCAN) is shown in upcoming sections.

The results obtained after training of proposed model as well as its comparison with state of art models is shown in Fig. **??** After obtaining results from different proposed models, images from each model were evaluated on different image quality matrices to carry out the comparison. The SSIM maps for each image is shown in Fig. **??**, Fig. **??**

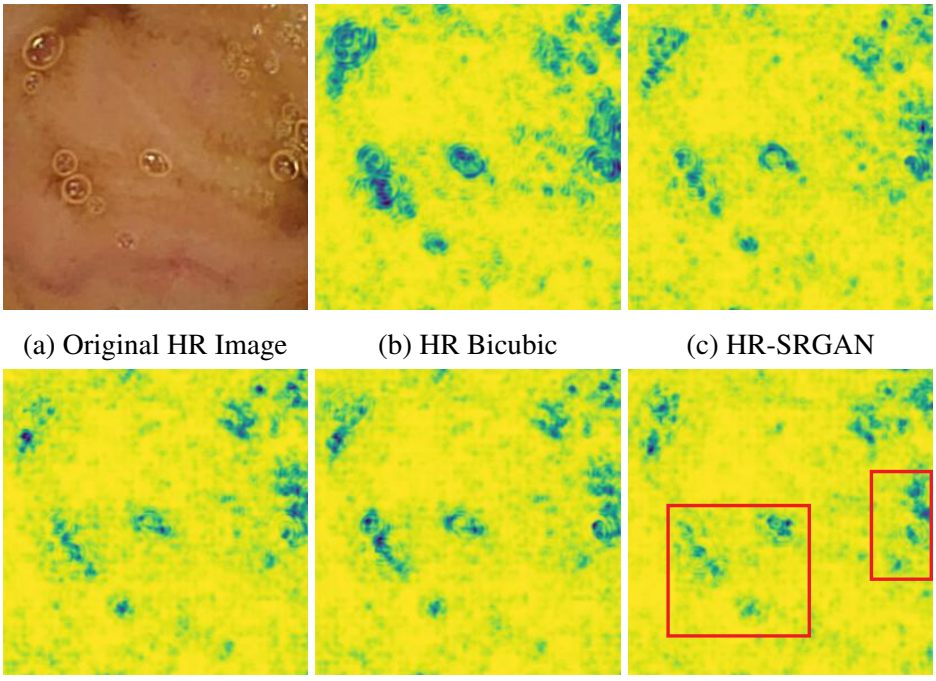


(a) Original HR Image (b) HR Bicubic (c) HR-SRGAN

(d) HR-SRDensenet (e) HR-RCAN (f) HR-Proposed

Figure 5.12: SSIM maps for Images generated from different models. Yellow region shows similarity while the blue region shows dissimilarity.

It can be observed from the SSIM maps that the our model is able to beat the state of art models like SRDensenet and RCAN. Our model exceptionally well and their outputs are very close to ground truth images.



(a) Original HR Image (b) HR Bicubic (c) HR-SRGAN

(d) HR-SRDensenet (e) HR-RCAN (f) HR-Proposed

Figure 5.13: SSIM maps for Images generated from different models: (1) Original HR image (2) HR-Bicubic (3) HR-SRGAN (4) HR-SRDensenet (5) HR-RCAN (6) HR-Propsed (Yellow region shows similarity while the blue region shows disimilarity.)

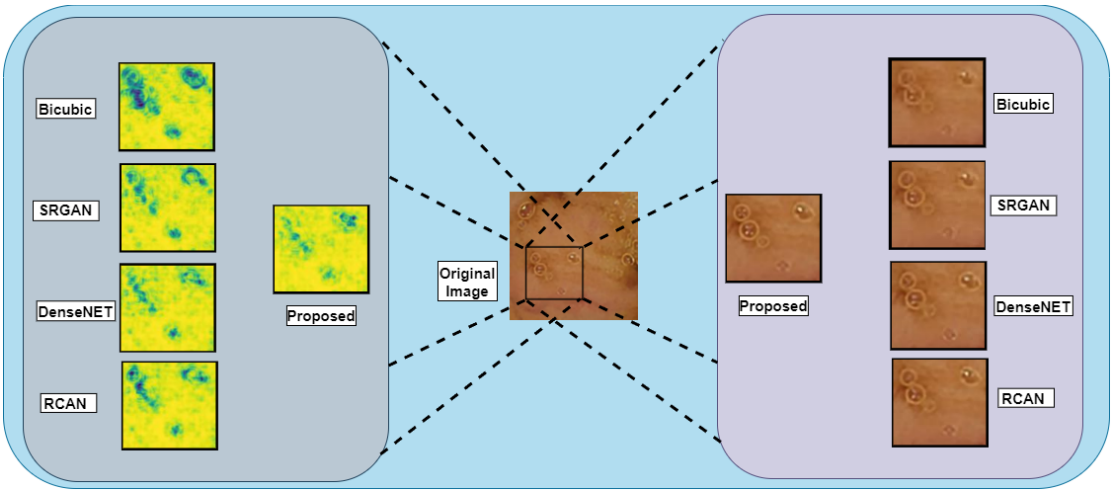


Figure 5.14: Patch Analysis

Analysis on the patch showed in Fig. **??** can be shown in Fig. **??** The right side of the image shows the patch extracted from different models such as proposed model, srgan model, rcan model, densenet model, and bicubic interpolation. The left side displays the SSIM map for the corresponding region of each image.

5.4 Quantitative Analysis

The average SSIM and PSNR values for the testing images of each model is provided in table **??**

Table 5.6: PSNR, SSIM and LPIPS of all models.

Model	PSNR ↑		SSIM ↑		LPIPS ↓
	Y-Channel	RGB	Y-Channel	RGB	RGB
Proposed	40.2261 ↑	39.5389 ↑	0.9486 ↑	0.9378 ↑	0.1346 ↓
Bicubic	38.1069	37.2111	0.9296	0.9057	0.2310
SRGAN [**?**]	38.0377	37.0021	0.9291	0.9049	0.1972
DenseNET [**?**]	39.6842	38.8596	0.9401	0.9369	0.1353
RCAN [**?**]	40.1438	39.4613	0.9427	0.9371	0.1359
CycleCNN(MSE)	38.0121	37.0143	0.9102	0.9012	0.1982
Dense CycleCNN(GRL)	38.1546	37.1453	0.9013	0.8912	0.1895
YCbCr CycleCNN	36.4098	35.5234	0.8991	0.8993	0.1783
Patch CycleCNN	36.6342	35.7452	0.9015	0.9023	0.1732

It can be observed that the PSNR and SSIM values of the proposed model are high compared to the present State of Art models, such as SR-Densenet and RCAN. The LPIPS values are also the lowest. Proposed model is providing the best values for all the image quality assessment metrices taken under consideration namely PSNR, SSIM and LPIPS.

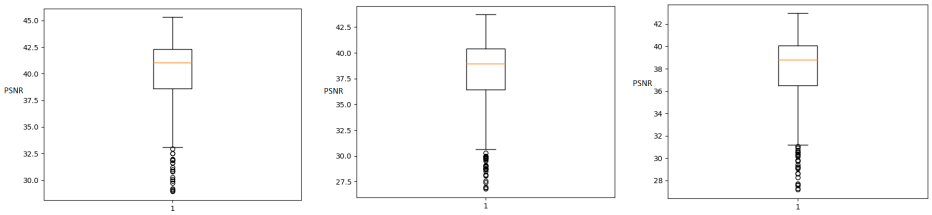
5.4.1 Statistical Analysis

Statistical analysis was also conducted on the results of the proposed model to ensure the model consistency compared to SOTA models. The values of standard deviation of each model are presented in table **??**

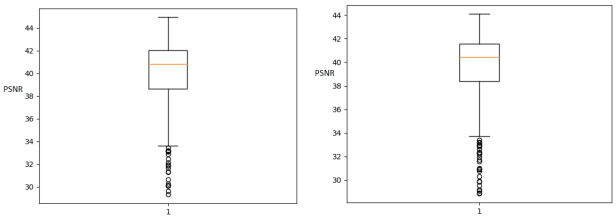
Table 5.7: Standard Deviation Values

Model	Standard deviation
Proposed	2.0186
BICUBIC	2.2102
SRGAN [**?**]	2.6924
DenseNET [**?**]	3.1996
RCAN [**?**]	2.8753

Box plots were used to observe the spread and outliers of the PSNR and SSIM values of proposed model. They are present in the Fig. **??**



(a) Proposed (b) Bicubic (c) SRGAN



(d) RCAN (e) DenseNET

Figure 5.15: Box plots for different models

We can clearly observe that the proposed model is more consistent owing to the lower standard deviation and the lower number of outlier compared to the other SOTA models.

Summary and Future scope

We analysed various state of art models for Image Super Resolution task of wireless capsule endoscopy images and started working on the proposed architecture. After conduction series of experiments with different model architectures including state of art models such as SRGAN, CycleGAN, DR-Densenet and RCAN, we proposed various models including cycleCNN, cycleCNN with GRL, dense cycleCNN with GRL on different losses including MSE loss, cycle loss, adversarial loss. But this models were not able to outperform the state of art models, After gaining knowledge from all the experiments, we were able to come up with the architecture DenseNET with Channel Attention Block(DCAN) that is able to out perform all the exiting SOTA models, not just in getting best values for all the image quality assessment metrices, but also in terms of consistency. we measured model consistency through statistical parameters such as standard deviation and box plots. The proposed model is providing better results than current state of art models DenseNet and RCAN, But it is not able to generalize over different datasets such as KID dataset and conventional endoscopy data. In future we aim to develop an deep learning based architecture that is roboust over all the datasets and we also plan to use the unsupervised approach and develop state of art models for this approach, as very limited amount of data is available for this domain.

References

[1] C. Ledig, L. Theis, F. Huszar, J. Caballero, A. Cunningham, A. Acosta, A. Aitken, A. Tejani, J. Totz, Z. Wang, and W. Shi, "Photo-realistic single image super-resolution using a generative adversarial network," 2016. [Online]. Available: https://arxiv.org/abs/1609.04802

[2] T. Tong, G. Li, X. Liu, and Q. Gao, "Image super-resolution using dense skip connections," 2017.

[3] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu, "Image super-resolution using very deep residual channel attention networks," 2018.

[4] Z. Wang, J. Chen, and S. C. H. Hoi, "Deep learning for image super-resolution: A survey," 2019.

[5] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, 2004.

[6] P. Muruganantham and S. M. Balakrishnan, "A survey on deep learning models for wireless capsule endoscopy image analysis."

[7] J. Liu, Z. Gan, and X. Zhu, "Directional bicubic interpolation," 2013.

[8] A. Kessentini, N. Bahri, N. Masmoudi, A. Samet, and M. A. Ben Ayed, "Dsp-based down-sampling process using lanczos sampling," 2014.

[9] C. Dong, C. C. Loy, K. He, and X. Tang, "Image super-resolution using deep convolutional networks," 2015. [Online]. Available: https://arxiv.org/abs/1501.00092

[10] J.-Y. Zhu, T. Park, P. Isola, and A. A. Efros, "Unpaired image-to-image translation using cycle-consistent adversarial networks."

[11] H. Zhang, P. Wang, and Z. Jiang, "Nonpairwise-trained cycle convolutional neural network for single remote sensing image super-resolution," *IEEE Transactions on Geoscience and Remote Sensing*, 2021.

[12] P. H. Smedsrud, V. Thambawita, S. A. Hicks, H. Gjestang, O. Nedrejord, E. Næss, H. Borgli, and D. Jha, "Kvasir-capsule, a video capsule endoscopy dataset," 2021.

[13] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014. [Online]. Available: https://arxiv.org/abs/1412.6980

[14] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial networks," 2014. [Online]. Available: https://arxiv.org/abs/1406.2661