UNIVERSITY OF THESSALY

SCHOOL OF ENGINEERING

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

Super-Resolution with Deep Learning Techniques for Medical Images

Diploma Thesis

Konstantinos Milas

Supervisor: George Stamoulis

September 2022

UNIVERSITY OF THESSALY

SCHOOL OF ENGINEERING

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

Super-Resolution with Deep Learning Techniques for Medical Images

Diploma Thesis

Konstantinos Milas

Supervisor: George Stamoulis

September 2022

ΠΑΝΕΠΙΣΤΗΜΙΟ ΘΕΣΣΑΛΙΑΣ

ΠΟΛΥΤΕΧΝΙΚΗ ΣΧΟΛΗ

ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

Υπερ-ανάλυση με τεχνικές βαθιάς μάθησης για την ανάλυση ιατρικών εικόνων

Διπλωματική Εργασία

Κωνσταντίνος Μήλας

Επιβλέπων/πουσα: Γεώργιος Σταμούλης

Σεπτέμβριος 2022

Approved by the Examination Committee:

Supervisor **George Stamoulis**
Professor, Department of Electrical and Computer Engineering, University of Thessaly

Member **Vassilis Plagianakos**
Professor, Department of Computer Science and Biomedical Informatics, University of Thessaly

Member **Sotirios Tasoulis**
Assistant Professor, Department of Computer Science and Biomedical Informatics, University of Thessaly

Acknowledgements

I want to thank my family and friends for their support all these years.

DISCLAIMER ON ACADEMIC ETHICS AND INTELLECTUAL PROPERTY RIGHTS

«Being fully aware of the implications of copyright laws, I expressly state that this diploma thesis, as well as the electronic files and source codes developed or modified in the course of this thesis, are solely the product of my personal work and do not infringe any rights of intellectual property, personality and personal data of third parties, do not contain work / contributions of third parties for which the permission of the authors / beneficiaries is required and are not a product of partial or complete plagiarism, while the sources used are limited to the bibliographic references only and meet the rules of scientific citing. The points where I have used ideas, text, files and / or sources of other authors are clearly mentioned in the text with the appropriate citation and the relevant complete reference is included in the bibliographic references section. I also declare that the results of the work have not been used to obtain another degree. I fully, individually and personally undertake all legal and administrative consequences that may arise in the event that it is proven, in the course of time, that this thesis or part of it does not belong to me because it is a product of plagiarism».

The declarant

Konstantinos Milas

Diploma Thesis

Super-Resolution with Deep Learning Techniques for Medical Images

Konstantinos Milas

Abstract

Single Image Super-Resolution is an important low-level computer vision process to enhance and denoise images and videos.This thesis is about how Deep Learning and more significantly Convolutional Neural Networks and Generative Adversarial Networks dominated the field of SISR.Firstly,the low resolution images are produced from bicubic interpolation of high resolution images so only bicubic degradation is learned by the SISR model.Secondly, state of the art techniques implemented like Enhanced Deep Residual Networks for Single Image Super-Resolution (EDSR) Residual Channel Attention Networks (RCAN) and Enhanced Super-Resolution Generative Adversarial Networks (ESRGAN).Lastly, experiments are conducted in famous Image Super-Resolution and Medical image datasets showing that EDSR and RCAN can have better PSNR and SSIM scores but ESRGAN outperfomrs all in perceptual quality.

Keywords:

Single Image Super-Resolution

Convolutional Neural Networks

Generative Adversarial Networks

Medical Image Analysis

Διπλωματική Εργασία

Υπερ-ανάλυση με τεχνικές βαθιάς μάθησης για την ανάλυση ιατρικών εικόνων

Κωνσταντίνος Μήλας

Περίληψη

Η Υπερανάλυση εικόνας είναι μια σημαντική διαδικασία υπολογιστικής όρασης χαμηλού επιπέδου για τη βελτίωση των εικόνων και των βίντεο. Αυτή η διατριβή έχει να κάνει με το πώς οι μέθοδοι Βαθιάς Μάθησης και πιο σημαντικά τα Συνελικτικά Νευρωνικά Δίκτυα και τα Παραγωγικά αντιπαραθετικά Δίκτυα κυριάρχησαν στο πεδίο της υπερανάλυσης. Πρώτον, οι εικόνες χαμηλής ανάλυσης είναι που παράγονται από δικυβική παρεμβολή εικόνων υψηλής ανάλυσης, έτσι μόνο η δικυβική υποβάθμιση μαθαίνεται από το μοντέλο SISR. Δεύτερον, οι τεχνικές αιχμής που εφαρμόζονται όπως τα Enhanced Deep Residual Networks for Single Image Super-Resolution (EDSR), Residual Channel Attention Networks (RCAN) και το Enhanced Super-Resolution Generative Adversarial Networks (ESRGAN). Τέλος, διεξάγονται πειράματα σε διάσημα σύνολα δεδομένων και ιατρικές εικόνες που δείχνουν ότι τα EDSR και RCAN μπορούν να έχουν καλύτερες βαθμολογίες PSNR και SSIM, αλλά το ESRGAN υπερτερεί όλων σε αντιληπτική ποιότητα.

Λέξεις-κλειδιά:

Υπερ-ανάλυση εικόνας

Συνελικτικά Νευρωνικά Δίκτυα

Παραγωγικά αντιπαραθετικά Δίκτυα

Επεξεργασία Ιατρικών εικόνων

Table of contents

List of figures

List of tables

Abbreviations

LR	Low resolution image
SR	Super-Resolution
GT	Ground Truth
ISR	Image Super-Resolution
SISR	Single Image Super-Resolution
MFSR	Multi-Frame Super-Resolution
CNNs	Convolutional Neural Networks
GANs	Generative Adversarial Networks
G	Generator
D	Discriminator
ESPCN	Efficient Sub-pixel Convolutional Neural Network
SRGAN	Super-Resolution Generative Adversarial Network
SRResNet	Super-Resolution Residual Network
EDSR	Enhanced Deep Super-Resolution Network
RCAN	Residual Channel Attention Networks
ESRGAN	Enhanced Super-Resolution Generative Adversarial Network
PSNR	Peak Signal-to-Noise ratio
SSIM	Structural Similarity index
MSE	Mean squared error
MRI	Magnetic resonance imaging
Swin Transformer	Shifted window vision transformer
SwinIR	Shifted window vision transformer for Image Restoration

Chapter 1

Εισαγωγή

1.1 Introduction

Image Super-Resolution is very helpful technique for modern digital images.Nowadays, many applications need high quality images from a civilian with a smartphone camera to a medical doctor that need to take a careful look to an Magnetic resonance imaging (MRI). Spatial Resolution can enhance image quality and is crucial for applications that detail matters like satelite imagery and medical imaging. Image Super-Resolution tries to solve the problem of upscaling spatial resolution of an image and create more pixels and sufficient detail to an image. This problem can have multiple images as input or only one low resolution image, for most image enhancement tasks is difficult to have multiple image of the same scene and time is crucial so Single Image Super-Resolution is a more interesting problem. A low resolution image can have a infinite number of high resolution image pairs so the Image Super-Resolution model tries to find the most probable HR image.This probability can be learned using Convolutional Neural Networks (CNNs) and Generative Adversarial Networks (GANs) to reconstruct more perceptually pleasing details.

1.1.1 Main contributions

Our main contributions are:

- CNN-based architectures are implemented for SISR.
- Trained with different loss fuctions.
- GANs models are used to enhance perceptual quality.

- Test and evaluate our models in different ISR benchmarks and also in medical images.

1.2 Thesis Structure

In chapter 2 introduces the Super-Resolution problem and explains different interpolation,upsampling techniques.Also, presents three different metrics to evaluate SR images with GT images.
In chapter 3 CNNs and GANs are explained different training procedures and loss functions are suggested.
In chapter 4 different state of the art techniques are proposed as a solution to the problem.
In chapter 5 results and evaluation metrics are showed for different digital images and in different scales.

Chapter 2

Super Resolution and Resampling

2.1 Super Resolution

Super-Resolution is the operation of recostructing a high resolution image from either a sequence of low (noisy) resolution images or only from one image alone.There are two most commonly used techniques Multi-Frame Super-Resolution (MFSR) and Single Image Super-Resolution(SISR).

- Multi-Frame Super-Resolution : This method uses multiple images of the same object or texture that have a slightly different position, different time taken or different image degradation to generate the higher resolution image pair.The advantages of MFSR are that there is a lot of information to process and from every image can generate different high frequencies.The main disadvantage is the computational complexity and that is not applicable for real-time tasks.

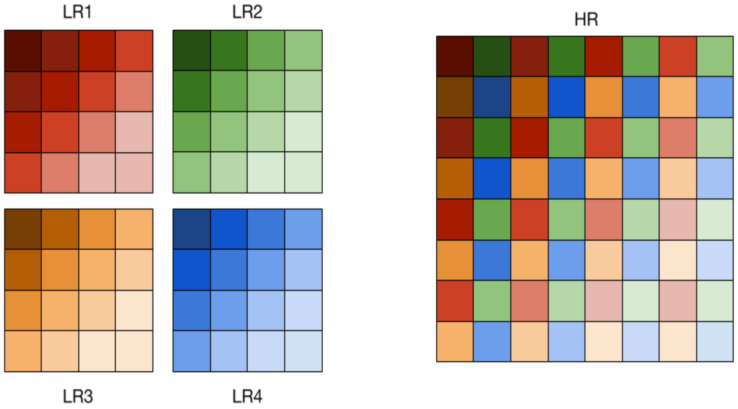

Figure 2.1: MFSR model

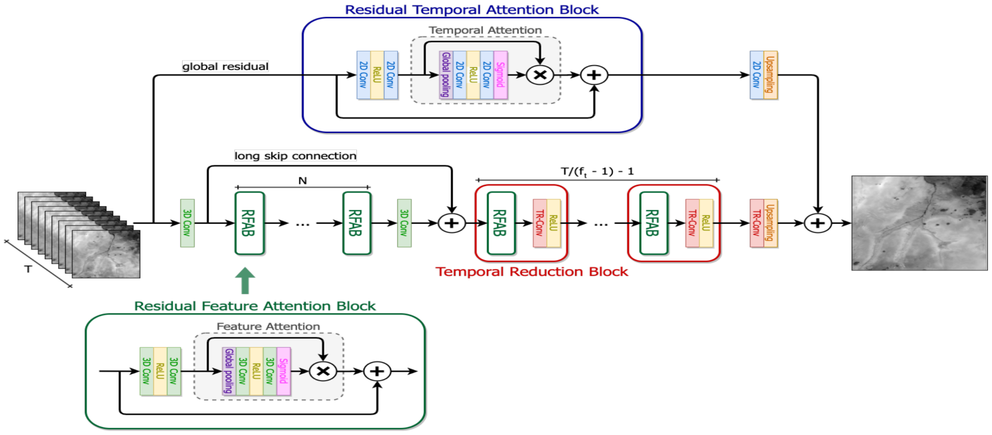

Figure 2.2: RAMS model

- Single Image Super-Resolution : This method uses only one low resolution image to generate the high resolution pair.The pipeline goes with downsampling the HR image with a interpolation technique like bicubic and then try to approximate the HR from the LR image using a model like CNN.The advantages of SISR is that is computational efficiently and can generate great results.The disadvantages is that it is difficult to generate high resolution images from different degradations in low resolution.

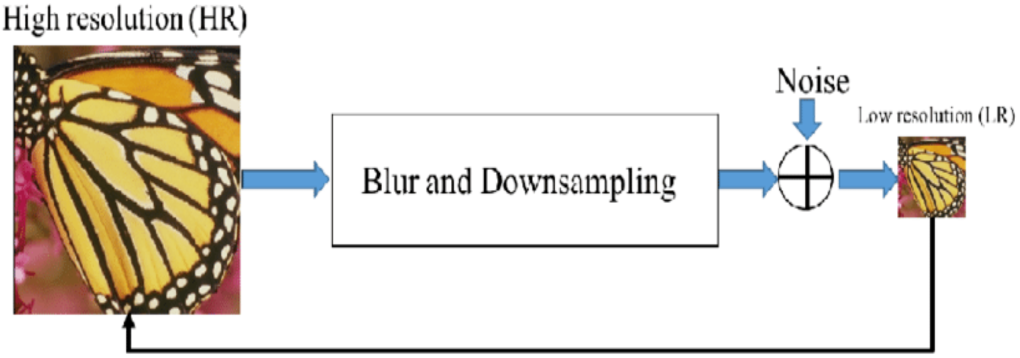

Figure 2.3: SISR pipeline

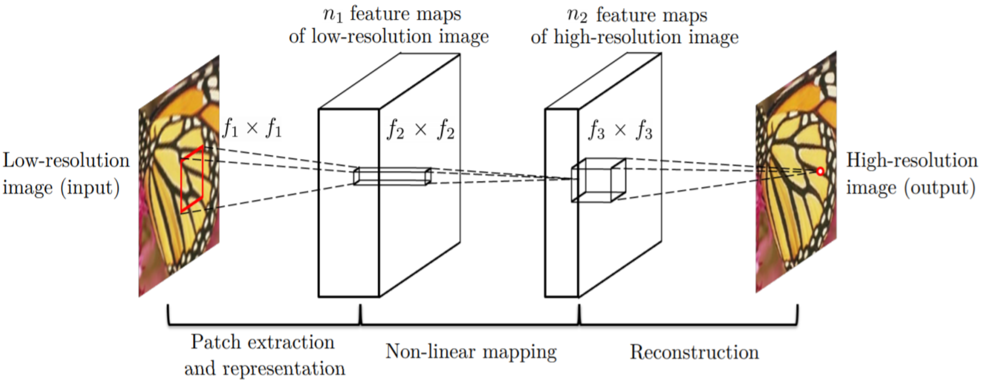
n_1 feature maps of low-resolution image
n_2 feature maps of high-resolution image
$f_1 \times f_1$
$f_2 \times f_2$
$f_3 \times f_3$
Low-resolution image (input)
High-resolution image (output)
Patch extraction and representation
Non-linear mapping
Reconstruction

Figure 2.4: SRCNN model

2.2 Interpolation Methods

Interpolation is the method to estimate new data values of known data values that are based to a different discrete set.These methods are applicable either for upsampling or downsampling of an image.The problem with those methods are that although they are extremely fast they don't pay attention to generate high frequencies and to eliminate noisy low frequencies, so in downsampling a deblurring method is used like a Gaussian smoothing.The big advantage of these methods that make them computationally efficient is that all can be calculated as a matrix multiplication or as a convolutional operation.

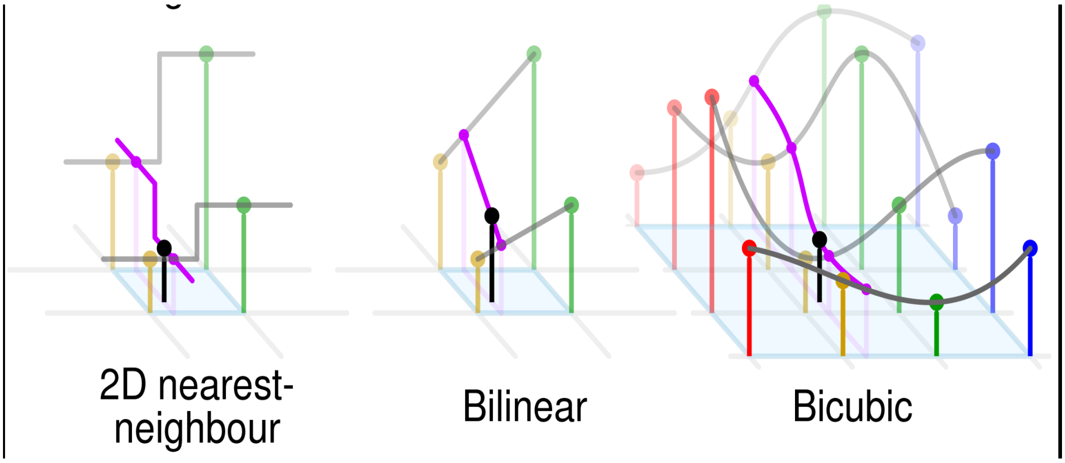

Figure 2.5: Interpolation Methods

2.2.1 Nearest-Neighbour

Nearest neighbour interpolation is the simplest approach to interpolation. Rather than calculate an average value by some weighting criteria or generate an intermediate value based on complicated rules, this method simply determines the "nearest" neighbouring pixel, and assumes the intensity value of it.The major drawback of this algorithm is that generate a poor quality and the image content isn't smooth.A formula for this algorithm is shown below:

$$A = Distance[(x,y)(i,j)]$$
$$B = Distance[(x,y)(i+1,j)]$$
$$C = Distance[(x,y)(i,j+1)]$$
$$D = Distance[(x,y)(i+1,j+1)]$$

$$nearest\_neighbour\_pixel_{D_{min}} = min\{A_{pixel}, B_{pixel}, C_{pixel}, D_{pixel}\}$$

So the point with the minimum value will be selected as the pixel value to the interpolated image.

2.2.2 Bilinear

In mathematics, bilinear interpolation is a method for interpolating functions of two variables (e.g., x and y) using repeated linear interpolation. It is usually applied to functions sampled on a 2D rectilinear grid.

Bilinear interpolation is performed using linear interpolation first in one direction, and then again in the other direction. Although each step is linear in the sampled values and in the position, the interpolation as a whole is not linear but rather quadratic in the sample location. Bilinear interpolation is one of the basic resampling techniques in computer vision and image processing. The algorithm is working by taking a weighted average of the nearest 4 pixels p1,p2,p3,p4.The weights are determined from the distances of the four points in the x and y direction.

$$f(x, y_2) = \frac{x_2 - x}{x_2 - x_1} \cdot p1 + \frac{x - x_1}{x_2 - x_1} \cdot p2$$

$$f(x, y_1) = \frac{x_2 - x}{x_2 - x_1} \cdot p3 + \frac{x - x_1}{x_2 - x_1} \cdot p4$$

$$f(x, y) = \tfrac{y - y_1}{y_2 - y_1} \cdot f(x, y_2) + \tfrac{y_2 - y}{y_2 - y_1} \cdot f(x, y_1)$$

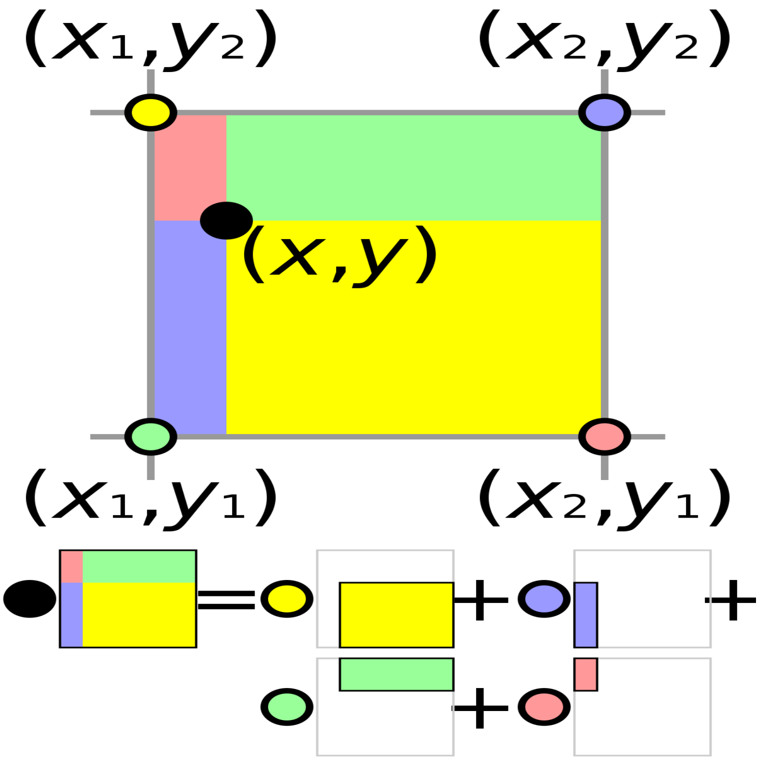

Figure 2.6: 2D Bilinear Interpolation

2.2.3 Bicubic

In mathematics, bicubic interpolation is an extension of cubic interpolation (not to be confused with cubic spline interpolation, a method of applying cubic interpolation to a data set) for interpolating data points on a two-dimensional regular grid. The interpolated surface is smoother than corresponding surfaces obtained by bilinear interpolation or nearest-neighbor interpolation. Bicubic interpolation can be accomplished using either Lagrange polynomials, cubic splines, or cubic convolution algorithm.

In image processing, bicubic interpolation is often chosen over bilinear or nearest-neighbor interpolation in image resampling, when speed is not an issue. In contrast to bilinear interpolation, which only takes 4 pixels (2×2) into account, bicubic interpolation considers 16 pixels (4×4). Images resampled with bicubic interpolation are smoother and have fewer interpolation artifacts.Also,16 coefficient it is needed to calculate in order to fit the new points in the interpolated space and takes into account except from distance and orientation of every pixel so thats why 16 pixels are needed to compute the gradients.

$$f(x,y) = \sum_{i=1}^{3}\sum_{i=1}^{3} a_{ij}x^i y^j$$

The a_{ij} coefficients can be calculated by the 16 equations where p_x is the derivative in the x direction p_y is the derivative in the y direction and p_{xy} are the partial derivatives.

$$f(0,0) = a_{00}$$

$$f(0,1) = a_{00} + a_{01} + a_{02} + a_{03}$$

$$f(1,0) = a_{00} + a_{10} + a_{20} + a_{30}$$

$$f(1,1) = \sum_{i=1}^{3}\sum_{i=1}^{3} a_{ij}$$

$$f_x(0,0) = a_{10}$$

$$f_x(0,1) = a_{10} + a_{11} + a_{12} + a_{13}$$

$$f_x(1,0) = a_{00} + a_{10} + 2a_{20} + 3a_{30}$$

$$f_x(1,1) = \sum_{i=1}^{3}\sum_{i=1}^{3} a_{ij} i$$

$$f_y(0,0) = a_{01}$$

$$f_y(0,1) = a_{01} + 2a_{02} + 3a_{03}$$

$$f_y(1,0) = a_{01} + a_{11} + 2a_{21} + 3a_{31}$$

$$f_y(1,1) = \sum_{i=1}^{3}\sum_{i=1}^{3} a_{ij} j$$

$$f_{xy}(0,0) = a_{01}$$

$$f_{xy}(0,1) = a_{11} + 2a_{12} + 3a_{13}$$

$$f_{xy}(1,0) = a_{11} + 2a_{21} + 3a_{31}$$

$$f_{xy}(1,1) = \sum_{i=1}^{3}\sum_{i=1}^{3} a_{ij} ij$$

Those equation can be transformed in a linear equation system $Aa = x$ and calculate the inverse matrix $A^{-1}x = a$

2.2.4 Interpolation Methods Comparison

The difference between the three methods is obvious nearest neighbour creates the most noisy example , bilinear and bicubic create smoother but also noisy data points. However, bicubic has the most texture between the three.

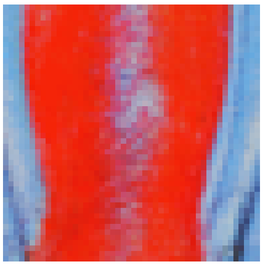

Figure 2.7: Nearest Neighbour

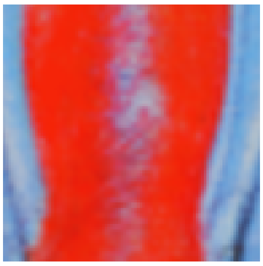

Figure 2.8: Bilinear

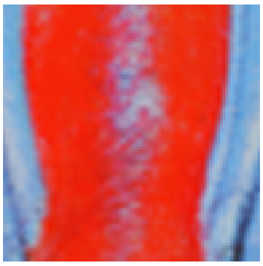

Figure 2.9: Bicubic

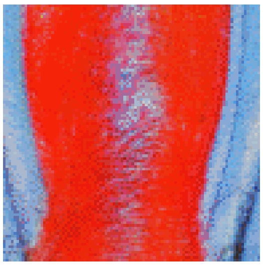

Figure 2.10: Ground Truth

2.3 Learnable Upsample Modules

To learn how to upsample an image without using heuristic methods like interpolation it is common to use the convolutional layer and with different tricks to generate feature maps of scale S in this chapter will represent two popular methods in SR upsampling :

- Transposed or Backwards Convolutions
- Efficient Sub-pixel Convolution

2.3.1 Transposed Convolution-Backward Convolution

Transposed and backward convolution[1] are another type of convolutional layers for upsampling and can be implemented as one module.As an operation are a convolution but we need to insert zeros between values to upscale the input in order to simulate fractional-strided convolution and to define the padding and the stride as in regular convolution.A picture is presented for a better understanding.

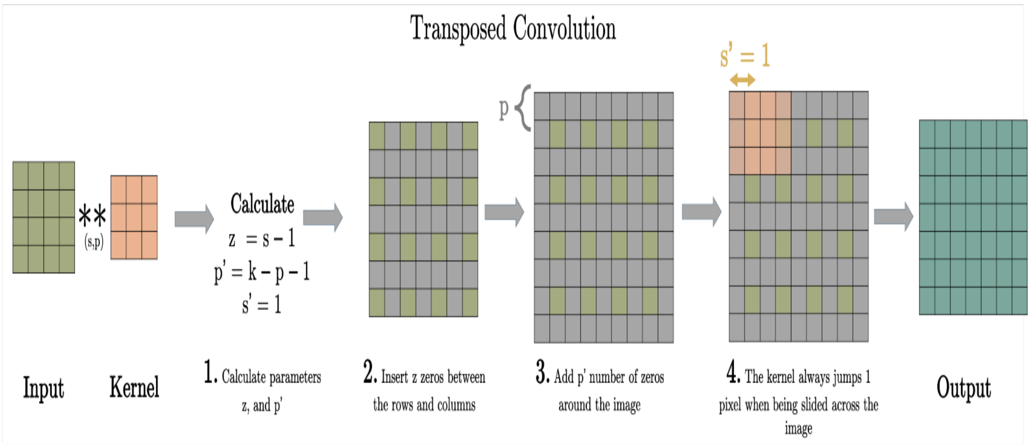
Transposed Convolution
s' = 1
p
**
(s,p)
Calculate
z = s - 1
p' = k - p - 1
s' = 1
Input
Kernel
1. Calculate parameters z, and p'
2. Insert z zeros between the rows and columns
3. Add p' number of zeros around the image
4. The kernel always jumps 1 pixel when being slided across the image
Output

Figure 2.11: Transposed Convolution

2.3.2 Efficient Sub-Pixel Convolution

Efficient sub-pixel convolution [2] is a more computational efficient upsample module than backward convolution because those operations do convolution in a high resolution image and zeros are added, so unnecessary operations are done to pixels that will not be activated.Efficient sub-pixel convolution also uses a convolutional layers to create feature maps of R^2 depth and then rearrange those feature maps to one feature map to a high resolution space where R is the upsample factor.A visualization is presented for better understanding.

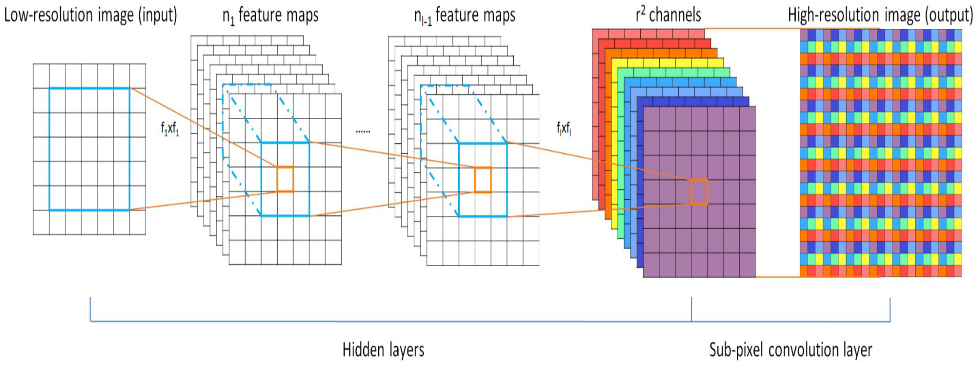
Low-resolution image (input)
n1 feature maps
nl-1 feature maps
r2 channels
High-resolution image (output)
f1xf1
......
flxfl
Hidden layers
Sub-pixel convolution layer

Figure 2.12: Efficient sub-pixel Convolution

2.4 Image Quality Assessment (IQA)

Image Quality Assessment are different methods that quantify image quality using human experience and how perceive images and colors.

2.4.1 Peak Signal-to-Noise Ratio (PSNR)

PSNR is a simple method to evaluate the quality of a noisy image with the ground truth image.This method uses mean squared error to be computed where I is the original image and I’ is the noisy image.

$$MSE = \frac{1}{m \cdot n} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} [I(i,j) - I'(i,j)]^2$$

PSNR three differnet formulas :

$$PSNR = 10 \cdot \log_{10} \frac{MAX_I^2}{MSE}$$

$$PSNR = 20 \cdot \log_{10} \frac{MAX_I}{\sqrt{MSE}}$$

$$PSNR = 20 \cdot \log_{10} MAX_I - 10 \cdot \log_{10} MSE$$

PSNR as a metric suits very well optimization techniques that are used in CNNs because it is proven to reduce MSE value.
If I and I' are the same images with same pixel and shape that means MSE=0 and PSNR will be infinite.

2.4.2 Structural Similarity index (SSIM)

SSIM[3] is a perception-based model that considers image degradation as perceived change in structural information, while also incorporating important perceptual phenomena, including both luminance masking and contrast masking terms. The difference with other techniques such as MSE or PSNR is that these approaches estimate absolute errors. Structural information is the idea that the pixels have strong inter-dependencies especially when they are spatially close. These dependencies carry important information about the structure of the objects in the visual scene. Luminance masking is a phenomenon whereby image distortions (in this context) tend to be less visible in bright regions, while contrast masking is a phenomenon whereby distortions become less visible where there is significant activity or ”texture” in the image.

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}$$

- Where μ_x is the average of x and μ_y is the average of y.
- Where σ_x is the variance of x , σ_y is the variance of y.
- Where σ_{xy} is the covariance of x and y.
- Where $c_1 = (k_1L)^2$, $c_2 = (k_2L)^2$ two variables to stabilize the division with weak denominator.
- Where $k_1 = 0.01$ and $k_2 = 0.03$ by default.

The SSIM formula is based on three comparison measurements between the samples of x and y : luminance l, contrast c and structure s. The individual comparison functions are:

- $$l(x,y) = \frac{(2\mu_x\mu_y + c_1)}{(\mu_x^2 + \mu_y^2 + c_1)}$$
- $$c(x,y) = \frac{(2\sigma_x\sigma_y + c_2)}{(\sigma_x^2 + \sigma_y^2 + c_2)}$$

- $$s(x,y) = \frac{(\sigma_{xy} + c_3)}{(\sigma_x \sigma_y + c_3)}$$

- $$c_3 = \frac{c_2}{2}$$

- $$SSIM(x,y) = l(x,y)^{\alpha} \cdot c(x,y)^{\beta} \cdot s(x,y)^{\gamma}$$

Chapter 3

Super Resolution and Deep Learning

3.1 Deep Learning

The deep learning era started by Alexnet [4] winning the ImageNet LSVRC-2012 [5] classification contest by a large margin. Alexnet was the first CNN model that won Imagenet which is a very large dataset and have different visual tasks like classification and localization. Some components that helped this CNN to succeed in the early 2010s was the use of GPU device for faster training ,the use of ReLU as activation function for better convergence ability of gradient descent based optimization algorithms and different techniques to reduce overfitting like Dropout and image augmentation.

3.1.1 CNN

CNNs are a special class of neural network developed for image application.Yann LeCun and his collaborators was the first ones to develop a dataset (MNIST[6]) and a system that can be used in production for handwritten digit recognition that in its core was a CNN named LeNet-5 [7]. LeNet architecture introduced some building blocks of modern CNN architectures like Convolution Layers,Pooling Layers and Fully connected layers.Convolution is very important in visual tasks that can extract features in an image by incorporating this ability with different convolutional filters at every layer of a neural network a feature detector is created with great potential to learn recognize a lot of complex features that are present in natural images.

In a high-level vision task like classification and object detection the first layers of the CNN learn abstract features like edges and corners in the middle layers learn more meaningful features like colors and textures and at the end learn to recognize and seperate different objects.Pooling layer or convolution with bigger stride that downsample the features space help the network when it goes deeper to have a global view of the image space and that helps the network to find complex features.In ISR this operation is not useful because spatial information is very important for those tasks in order to keep the same structure and try to reconstruct a HR space.So most of the times the architectures first upsample the images to HR space or upsample in the last layers of the network.

The first architectures like SRCNN [8] at first upsample the image with bicubic interpolation and then forward pass to generate an SR near the HR space, but this solution is computational inefficient and the network does not learn to upsample. So faster methods like FSRCNN [9] use learnable upsample methods like transposed convolution.ESPCN [2] introduced a more efficient upsample module with learnable parameters with the use of sub-pixel convolution.From there most deep learning methods for SR use this type of upsampling because of the speed and the training time and not necessarily for being a better reconstruction operation. Also VDSR introduced a very deep architecture with 20 layers and residual learning.Lastly, in the paper [10] where SRResNet and SRGAN were introduced two important methods were used residual blocks [11] and GAN framework where we will explain in depth in the next section.

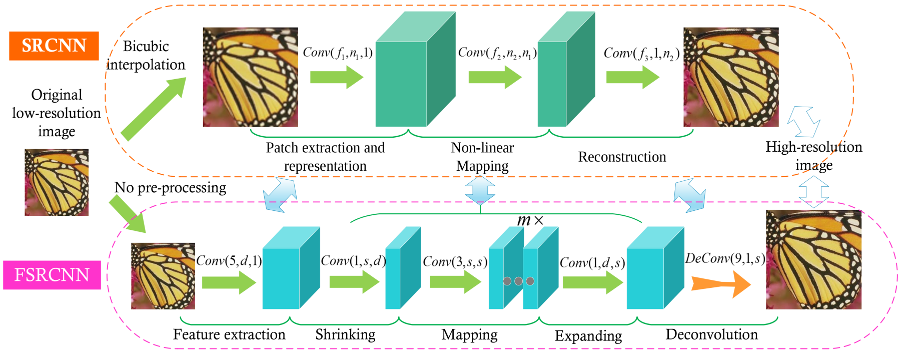

Figure 3.1: SRCNN vs FSRCNN

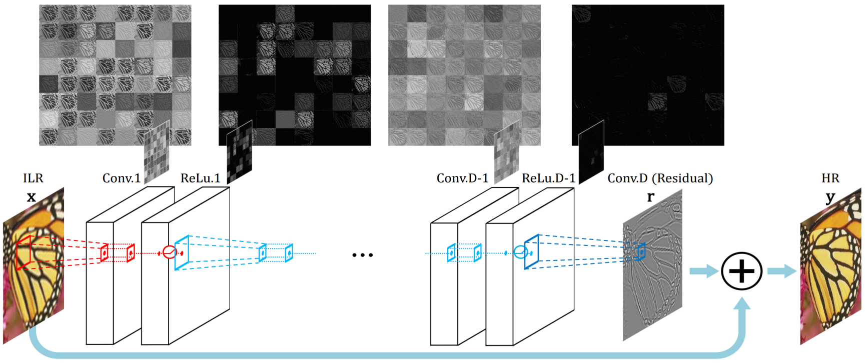

Figure 3.2: VDSR

3.1.2 GANs

Generative adversarial network [12] is a framework that makes two networks antagonise each other in a min-max game.The one network is called Generator G and tries to create data that is near the data distribution but different and trick the second network called Discriminator D. Discriminator tries to predict if the Generator gives real or fake examples and in that way pushes the Generator to make better examples.

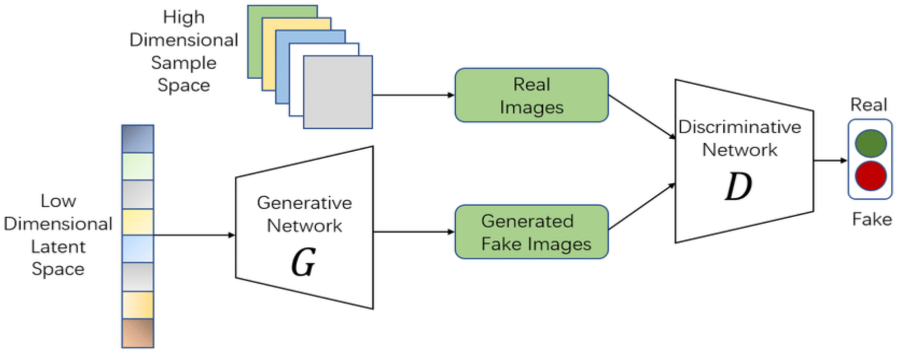

Figure 3.3: GAN framework

The framework tries to optimize the adversarial loss function :

$$\min_G \max_D V(D,G) = E_{x \sim p_{data}(x)}[\log(D(x))] + E_{z \sim p_z(z)}[\log(1 - D(G(z)))]$$

Where x is a sample from p_{data} distribution and z is random noise.

The first GAN used multi layer perceptron for D and G ,but DCGAN showed that convolution and transposed convolution can be a very important building block for this framework. DCGAN was able to have an exceptional performance in image synthesis and that shows that GANs are able to generate images with details and with meaning and transform noise to images.SRGAN took advantage of this framework by firstly using VGG[13] perceptual loss.VGG loss is the distance between the feature space of VGG reconstructed from the G and the GT from the data distribution.

$$Loss_{VGG} = MSE(I_{HR}, G(I_{LR}))$$

This is the adversarial loss is used and not the classic one , because gradients behave better.

$$l_{gen}^{SR} = \sum_{n=1}^{N} -\log D_{\theta_D}(G_{\theta_G}(I_{LR}))$$

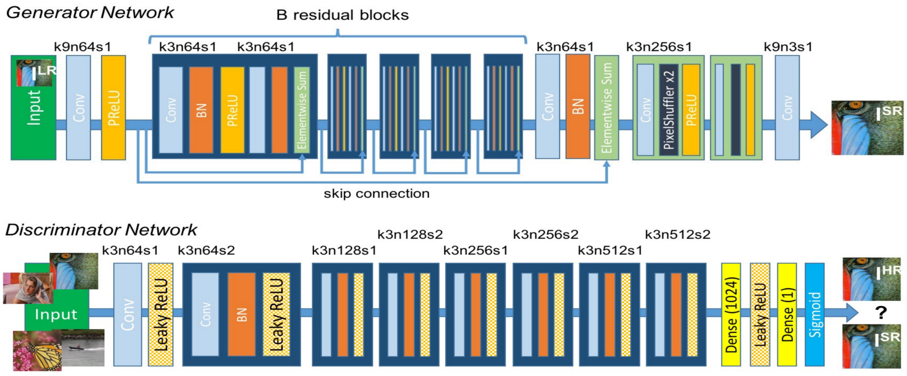

Figure 3.4: SRGAN architecture

Chapter 4

Proposed Methods

4.1 EDSR

EDSR[14] is an an enhanced deep super-resolution network which won the NTIRE2017 Super-Resolution Challenge[15]. EDSR proposed an enhanced residual block which enables a bigger and deeper architecture. Deeper architecture improve metrics performance (PSNR/SSIM) and is also possible with tricks that stabilize training procedure.

4.1.1 Residual Block

The model's residual block is based on SRResNet which use a original ResNet block with minor differences.EDSR try to optimize this block by removing unnecessary modules and by testing they came into conclusion that the Batchnormalization layer is not necessary for low-level vision tasks like SR.The model by removing batch normalization has a better memory because it does not need extra computation for this layer and is possible to train a bigger model. Also,by normalizing the features they lack of range flexibility and batch normalization can create unpleasant artifacts and reduces generalization ability. It is shown that by removing batch normalization EDSR can save 40% of memory usage during training.

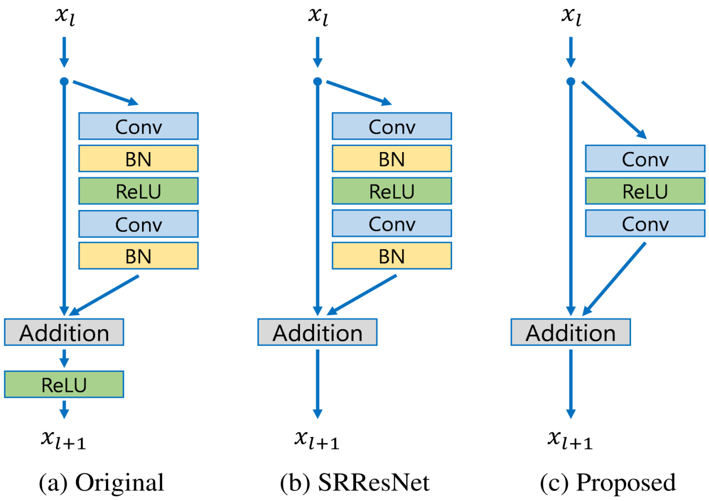

Figure 4.1: Comparison of residual blocks

4.1.2 Training

This model introduces some techniques that stabilize training procedure and is also used by the other two models that will be presented.Data augmentation is used like random horizontal flip and 90 rotation.They train with Adam optimizer with default settings, a learning rate of $2 \cdot 10^{-4}$ that is divided by 2 every $2 \cdot 10^{5}$ mini-batches.Lastly, L1 norm is used as a loss function instead of MSE.Although MSE is popular because it can optimized PSNR , it is shown by experiments that L1 has a faster convergence.

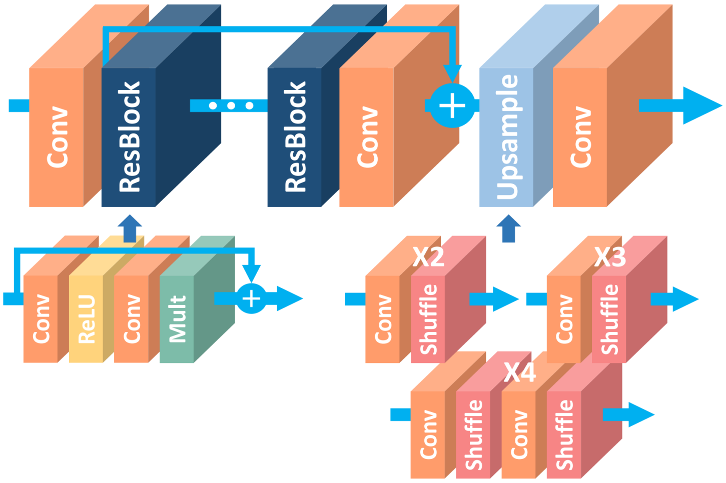

Figure 4.2: EDSR Single scale architecture

4.2 RCAN

RCAN[16] accomplished to train a very deep CNN to do that used short and long skip connections.Another important method of this architecture was channel attention that helps the model to pay attention on more useful channels.

4.2.1 Channel Attention

LR images have a lot of low-frequency information and a little but valuable high frequency information.To bring out this information a special mechanism is needed like CA which uses global average pooling and two Convolutional layer are used to down-scale the channels ,then up-scale them and at the end a sigmoid function is used as gating mechanism.The output of the CA layer are used to rescale the input and that is done by multiplying input with output.

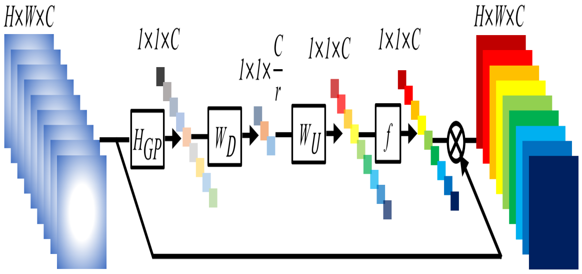

Figure 4.3: Channel attention layer. $\otimes$ is the multiplication symbol.

Where H_{GP} is the global average pooling layer,W_D the weights of a convolutional layer that down-scales features, W_U the weights of a convolutional layer that upscales features and f the gating function that is sigmoid.

4.2.2 Residual Channel Attention Block (RCAB)

The RCAB is the basic module of the architectures is more like a general case of a residual block proposed from EDSR because it uses CA to enhance information of every channel and residual connections to help construct a deeper network.

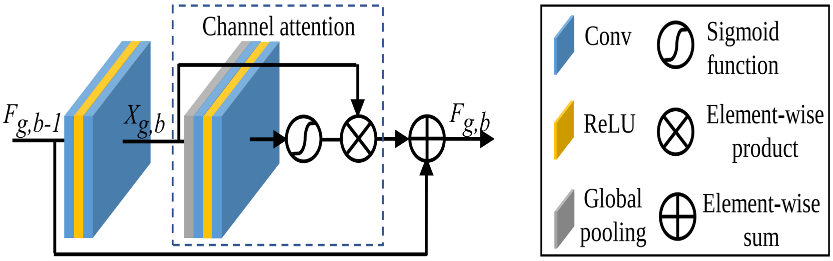

Figure 4.4: RCAB

4.2.3 Residual in Residual (RIR)

RIR is the basic block of the architectures.RIR consists of stacked RCAB modules and a short skip connection.Also the general architecture are RIR structures stacked the one after the other , a long skip connection after the stacked RIR and the upsample module which is like EDSR.All three techniques (CA,long skip connection,short skip connection) proposed by the authors give a better performance to the network.

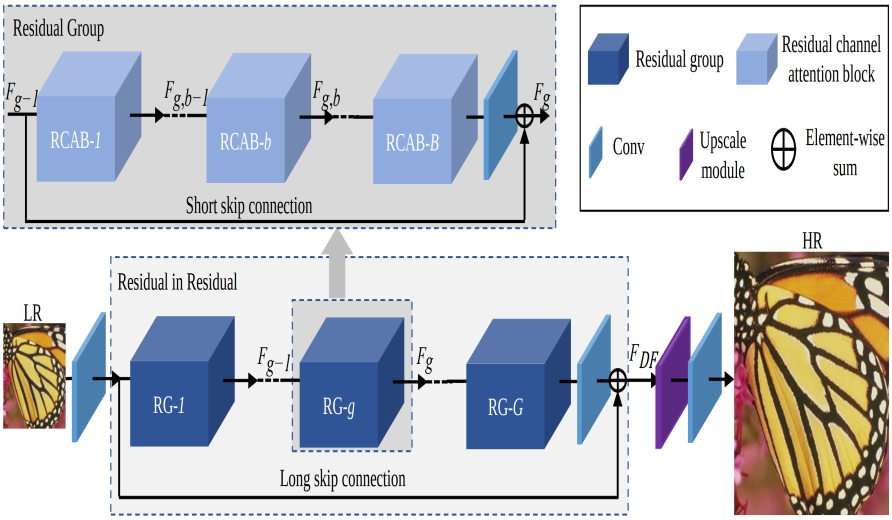

Figure 4.5: RCAN architecture

4.3 ESRGAN

ESRGAN tries to optimize the perceptual quality of an SR image and not the PSNR score.The model tries to enhance the performance of SRGAN[2] by optimizing the architecture of Generator and Discriminator and change the loss function of GAN.

4.3.1 Residual in Residual Dense Block (RRDB)

The building block of the model is RRDB which make use of residual connections,dense connections as was proposed by DenseNet and residual scaling using a β parameter of 0.2.Also the residual block of RRDB does not batch normalization as for the same reasons as EDSR and RCAN and because is difficul for GAN framework to create high frequencies without artifacts.There is a small network with 16 RRDB and a bigger one with 23 RRDB that has a great performance when is optimized with L1 loss for PSNR and SSIM metrics.

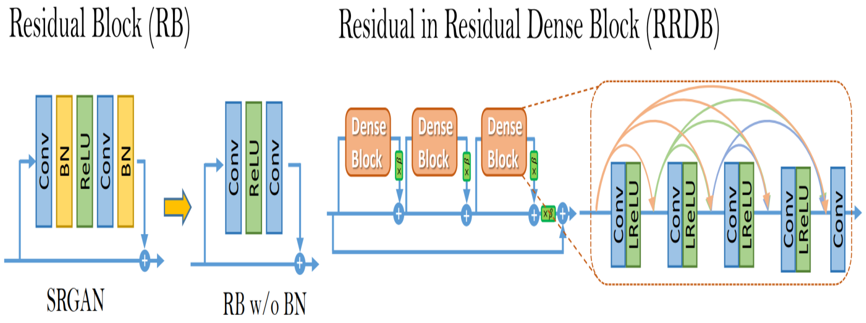

Figure 4.6: Residual block and RRDB

4.3.2 Perceptual Loss

Perceptual loss to optimize the feature space of a pretrained deep network like VGG.The SRGAN proposed a perceptual loss after activation layer.ESRGAN showed that using perceptual loss of VGG before activation layer that boosts performance because after activation more features become inactive and information is removed.

4.3.3 Relativistic GAN

SRGAN used a standard discriminator that tries to predict if the input image is real or fake.ESRGAN changed this framework by predicting that a real image is more realistic than a fake one.

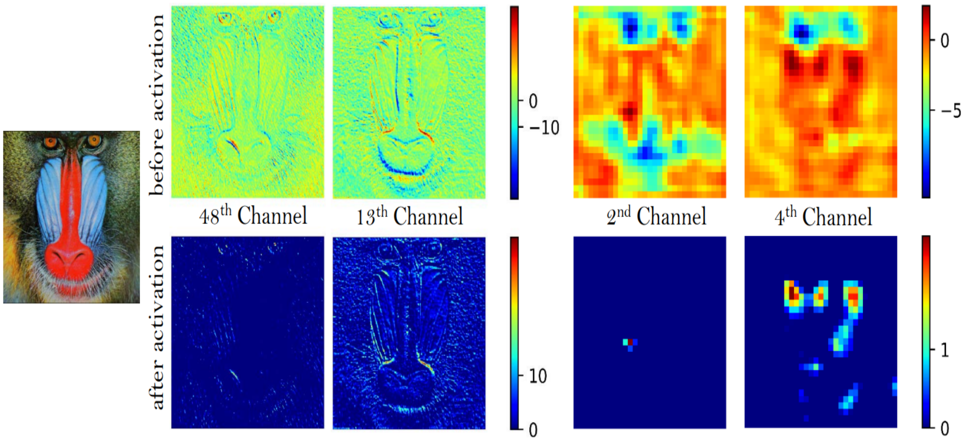

Figure 4.7: Comparison of features before and after activation

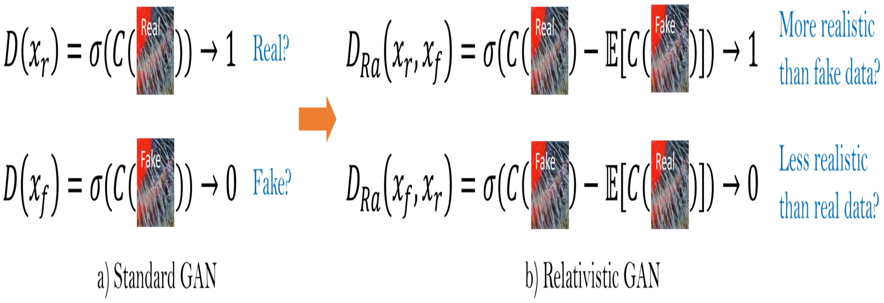
$D(x_r) = \sigma(C(\text{Real})) \rightarrow 1$
Real?
$D(x_f) = \sigma(C(\text{Fake})) \rightarrow 0$
Fake?
$D_{Ra}(x_r, x_f) = \sigma(C(\text{Real}) - \mathbb{E}[C(\text{Fake})]) \rightarrow 1$
More realistic than fake data?
$D_{Ra}(x_f, x_r) = \sigma(C(\text{Fake}) - \mathbb{E}[C(\text{Real})]) \rightarrow 0$
Less realistic than real data?
a) Standard GAN
b) Relativistic GAN

Figure 4.8: Standard vs Relativistic GAN

Chapter 5

Training Process and Results

5.1 Training

5.1.1 Dataset

The dataset that are used for training are DIV2K[17] and Flickr2k (also known as DF2K).The LR images were obtained by using bicubic intertpolation from MATLAB.Also for the retinal images the MESSIDOR-2 dataset [18] is used with 1000 images for training and 100 images for testing.

5.1.2 Augmentations

During training patches of the image are used and not the whole image because it is not feasible from the computational complexity.Big patches are important the give larger receptive field and boost the visual performance of the networks ,so 192x192 HR patches are used for the PSNR-oriented models and 128x128 are used for the GAN-based models.All the RGB channels of the images are used and augmented by randomly flipped vertical and horizontal.

5.1.3 Settings

The ADAM optimizer are used with the default values $\beta_1 = 0.9 \beta_2 = 0.999 \epsilon = 10^{-8}$ and the mini-batch size is set to 16.The learning rate is initialized with $2 \cdot 10^{-4}$ and is halved every 200k iterations.RCAN and EDSR are trained for 300k iterations ,RRDBNet for 500k and ESRGAN for 400k iterations.Lastly,the models are implemented and trained using Pytorch[19],Mixed Precision and an NVIDIA RTX 3070.

5.2 Results

5.2.1 PSNR and SSIM based Performance

The PSNR and SSIM are calculated by converting the RGB images to YCbCr and using only the Y channel for evaluation.RCAN outperforms in most test sets except from the scale 4 where RRDBNet is a clear winner.All the scores in the tests are very close with the original implementations.For the MESSIDOR-2 results first the models are trained in DF2K and then are finetuned in the MESSIDOR2 train images for less than half the iterations used for training.

Method	Scale	Set5		Set14		BSDS100		BSDS200		Manga109		Urban100	
		PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM	PSNR	SSIM
EDSR	x2	37.95	0.9609	33.53	0.9173	32.16	0.9003	33.04	0.9149	38.44	0.9771	31.86	0.9262
RCAN	x2	38.17	0.9618	33.86	0.9199	32.31	0.9022	33.24	0.9170	39.27	0.9789	32.65	0.9334
RRDBNet	x2	37.89	0.9605	33.54	0.9168	32.17	0.8999	33.05	0.9145	38.57	0.9766	32.02	0.9278
EDSR	x3	34.33	0.9270	30.26	0.8415	29.09	0.8065	29.72	0.8301	33.44	0.9438	28.01	0.8504
RCAN	x3	34.66	0.9296	30.53	0.8470	29.25	0.8106	29.94	0.8350	34.22	0.9485	28.73	0.8652
EDSR	x4	32.14	0.8949	28.54	0.7813	27.55	0.7372	28.04	0.7643	30.34	0.9065	25.97	0.7828
RCAN	x4	32.55	0.9001	28.82	0.7878	27.76	0.7437	28.31	0.7725	31.09	0.9162	26.70	0.8050
RRDBNet	x4	32.59	0.9001	28.87	0.7888	27.78	0.7444	28.32	0.7731	31.26	0.9173	26.78	0.8073

Method	Scale	MESSIDOR2	
		PSNR	SSIM
EDSR	x4	45.76	0.9723
RCAN	x4	45.58	0.9725
RDDBNet	x4	45.17	0.9712

5.2.2 Perceptual Performance x4 scale

As an example the x4 scale is used where the differences are more obvious.ESRGAN is able to reconstruct complex information in a perceptual pleasing way and the problem of all the other networks is that over-smooth those details and stop to look realistic.An example is the fur in the baboon photo.But not all the photos are easy to reconstruct small details like the photo with the two men where critical facial information is missed.In the MESSIDOR2 tests most of the critical information is reconstructed with not serious error.One error that we can catch is with the ESRGAN in the second photo where one region is more yellow than it should.

Figure 5.1: GT Manga109

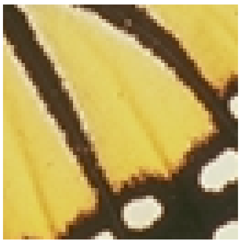

Figure 5.2: ESRGAN Set5

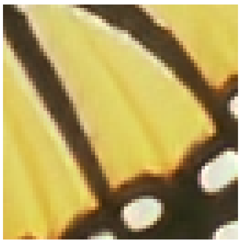

Figure 5.3: RRDBNet Set5

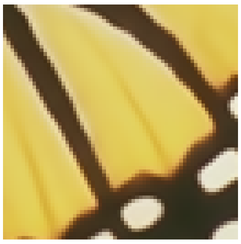

Figure 5.4: EDSR Set5

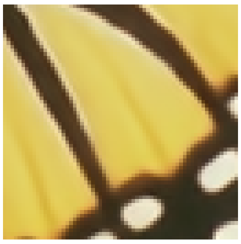

Figure 5.5: RCAN Set5

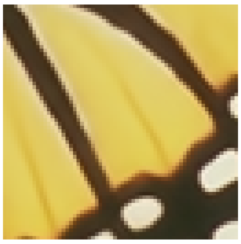

Figure 5.6: GT Set14

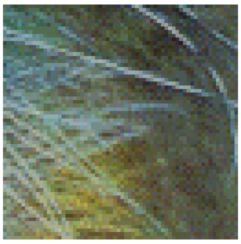

Figure 5.7: ESRGAN Set14

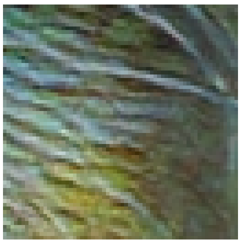

Figure 5.8: RRDBNet Set14

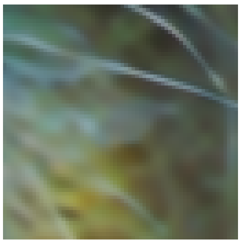

Figure 5.9: EDSR Set14

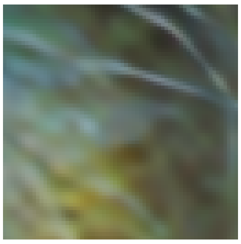

Figure 5.10: RCAN Set14

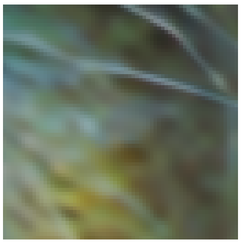

Figure 5.11: GT BSDS100

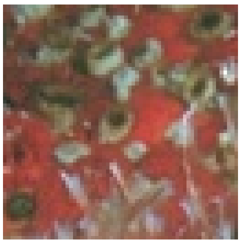

Figure 5.12: ESRGAN BSDS100

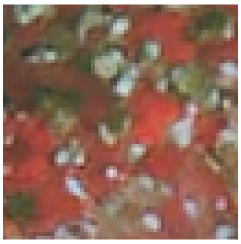

Figure 5.13: RRDBNet BSDS100

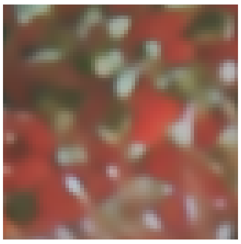

Figure 5.14: EDSR BSDS100

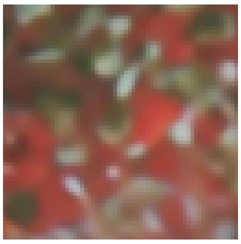

Figure 5.15: RCAN BSDS100

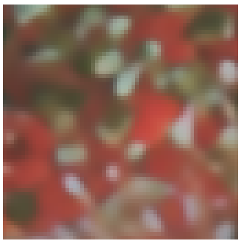

Figure 5.16: GT BSDS200

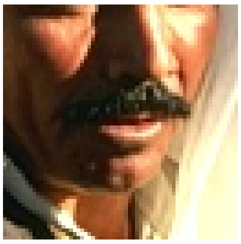

Figure 5.17: ESRGAN BSDS200

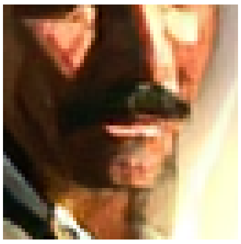

Figure 5.18: RRDBNet BSDS200

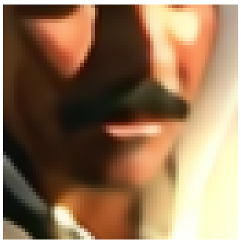

Figure 5.19: EDSR BSDS200

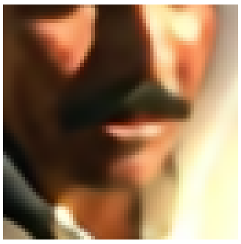

Figure 5.20: RCAN BSDS200

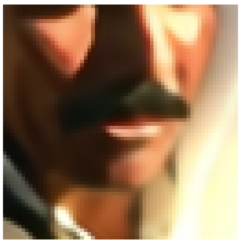

Figure 5.21: GT Urban100

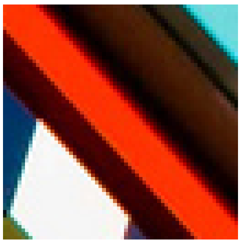

Figure 5.22: ESRGAN Urban100

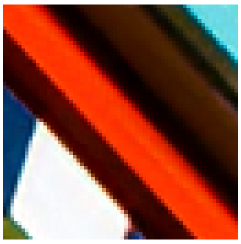

Figure 5.23: RRDBNet Urban100

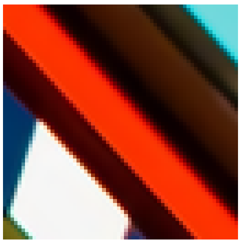

Figure 5.24: EDSR Urban100

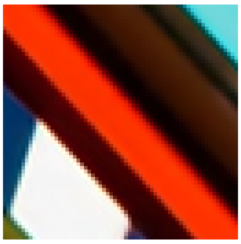

Figure 5.25: RCAN Urban100

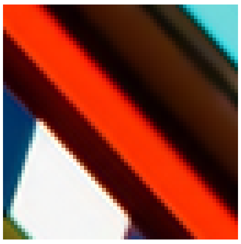

Figure 5.26: GT Manga109

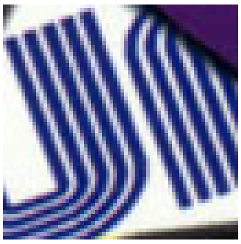

Figure 5.27: ESRGAN Manga109

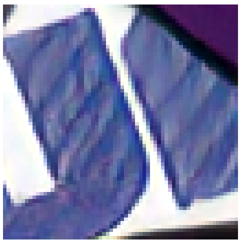

Figure 5.28: RRDBNet Manga109

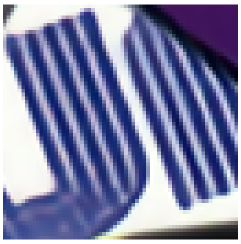

Figure 5.29: EDSR Manga109

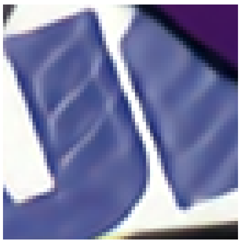

Figure 5.30: RCAN Manga109

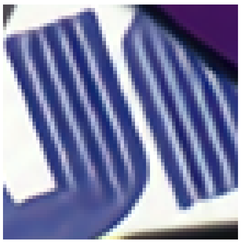

Figure 5.31: GT MESSIDOR2_1

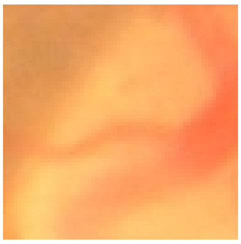

Figure 5.32: ESRGAN MESSIDOR2_1

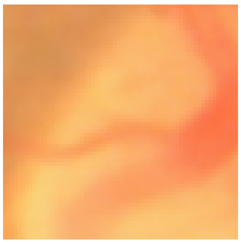

Figure 5.33: RRDBNet MESSIDOR2_1

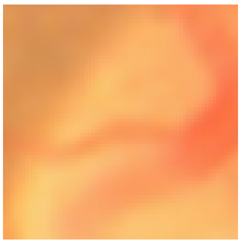

Figure 5.34: EDSR MESSIDOR2_1

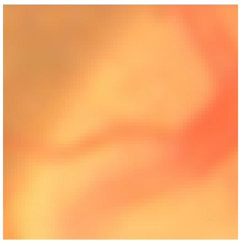

Figure 5.35: RCAN MESSIDOR2_1

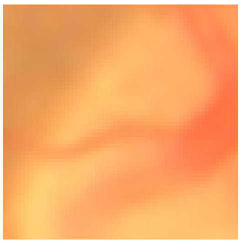

Figure 5.36: GT MESSIDOR2_2

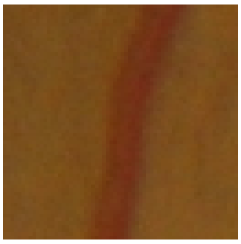

Figure 5.37: ESRGAN MESSIDOR2_2

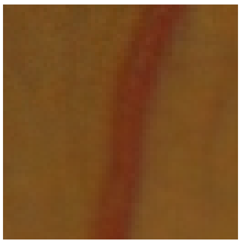

Figure 5.38: RRDBNet MESSIDOR2_2

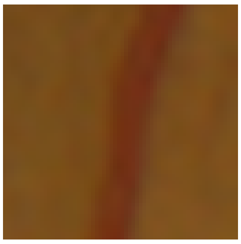

Figure 5.39: EDSR MESSIDOR2_2

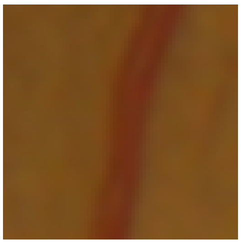

Figure 5.40: RCAN MESSIDOR2_2

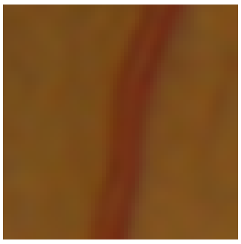

Figure 5.41: GT MESSIDOR2_3

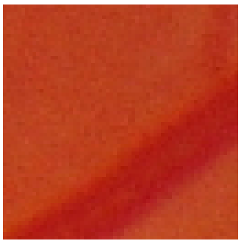

Figure 5.42: ESRGAN MESSIDOR2_3

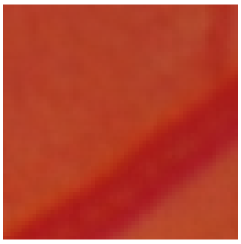

Figure 5.43: RRDBNet MESSIDOR2_3

Figure 5.44: EDSR MESSIDOR2_3

Figure 5.45: RCAN MESSIDOR2_3

Chapter 6

Conclusion

We have three different deep learning SR method, the two[16][14] being PSNR oriented and the other one [20] focuses on perceptual quality. We tested the ability to generate detailed SR images in popular test datasets like Set5,Set14,Urban100,others and in medical images in three different scales (x2,x3,x4).

6.1 Future work

Some future work could be using an unknown degradation model as most natural images have and SwinIR[21] model showed that transformers like Swin[22] can have excellent result for Image Restoration and Super Resolution models.

Bibliography

[1] Jonathan Long, Evan Shelhamer, and Trevor Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3431–3440, 2015.

[2] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1874–1883, 2016.

[3] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

[4] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, 25, 2012.

[5] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015.

[6] Yann LeCun and Corinna Cortes. MNIST handwritten digit database. 2010.

[7] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. In *Proceedings of the IEEE*, volume 86, pages 2278–2324, 1998.

[8] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Learning a deep convolutional network for image super-resolution. In *European conference on computer vision*, pages 184–199. Springer, 2014.

[9] Chao Dong, Chen Change Loy, and Xiaoou Tang. Accelerating the super-resolution convolutional neural network. In *European conference on computer vision*, pages 391–407. Springer, 2016.

[10] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4681–4690, 2017.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[12] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Advances in neural information processing systems*, 27, 2014.

[13] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[14] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, July 2017.

[15] Radu Timofte, Eirikur Agustsson, Luc Van Gool, Ming-Hsuan Yang, and Lei Zhang. Ntire 2017 challenge on single image super-resolution: Methods and results. In *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pages 114–125, 2017.

[16] Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun Fu. Image super-resolution using very deep residual channel attention networks. In *ECCV*, 2018.

[17] Eirikur Agustsson and Radu Timofte. Ntire 2017 challenge on single image super-resolution: Dataset and study. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, July 2017.

[18] Messidor-2 dataset. `https://www.adcis.net/en/third-party/messidor2/`.

[19] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.

[20] Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Yu Qiao, and Chen Change Loy. Esrgan: Enhanced super-resolution generative adversarial networks. In *The European Conference on Computer Vision Workshops (ECCVW)*, September 2018.

[21] Jingyun Liang, Jiezhang Cao, Guolei Sun, Kai Zhang, Luc Van Gool, and Radu Timofte. Swinir: Image restoration using swin transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1833–1844, 2021.

[22] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10012–10022, 2021.

Chapter A

Mathematics

A.1 Convolution

Although CNN the say that use convolution in reality they use cross corellation formula $\star$.

$$I \star k = \sum_{i=-N}^{N} \sum_{i=-N}^{N} I(x+i, y+j)k(i,j)$$

Where I is the image and k is convolution kernel.

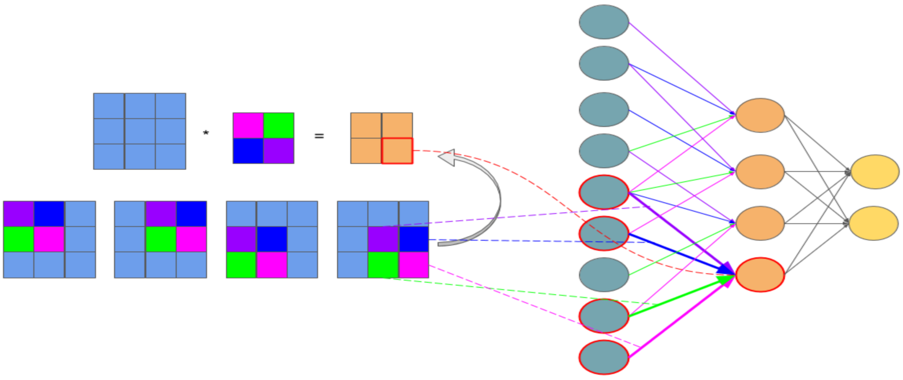

Figure A.1: Visualize convolution operation

A.2 Activation functions

Sigmoid function :

$$S(x) = \frac{1}{1 + e^{-x}}$$

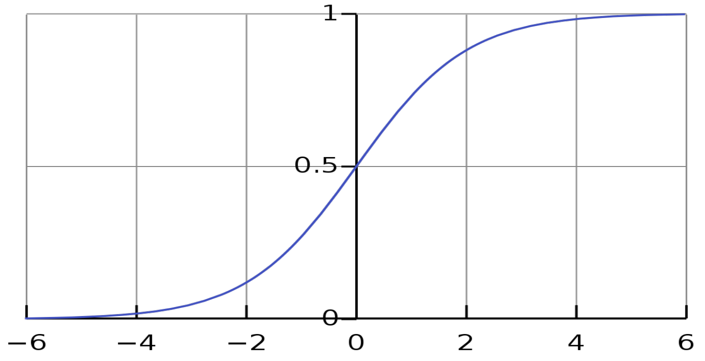

Figure A.2: Sigmoid function

ReLU and PReLU function :

$$ReLU(x) = \begin{cases} 0 & if\ x < 0 \\ x & if\ x \geq 0. \end{cases}$$

$$PReLU(x) = \begin{cases} a \cdot x & if\ x < 0 \\ x & if\ x \geq 0. \end{cases}$$

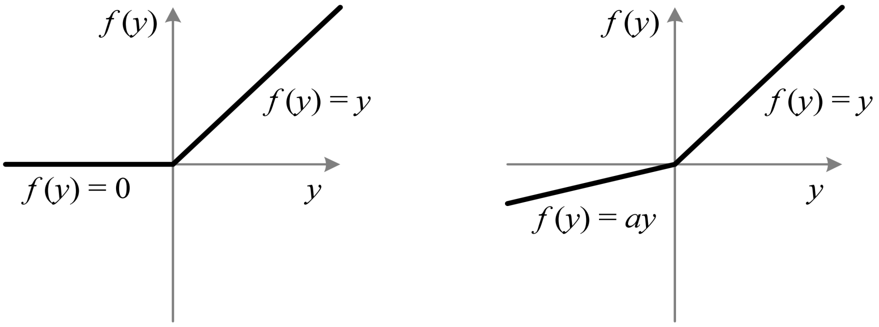

Figure A.3: ReLU and PReLU