A

Major

Project On

A GENERATIVE ADVERSARIAL NETWORK BASED DEEP LEARNING METHOD FOR LOW QUALITY DEFECT IMAGE RECONSTRUCTION AND RECOGNITION

Submitted to

Jawaharlal Nehru Technological University, Hyderabad

In Partial fulfillment of the requirements for the award of Degree

of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE & ENGINEERING

by

S.Ruchitha (187R1A05N7)

T.Akhila (187R1A05P0)

T.Sony (187R1A05N9)

Under the Guidance of

D.SANDHYA RANI

(Assistant Professor)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CMR TECHNICAL CAMPUS

UGC AUTONOMOUS

(Accredited by NAAC, NBA, Permanently Affiliated to JNTUH, Approved by AICTE, New Delhi) Recognized Under Section 2(f) & 12(B) of the UGCAct.1956,

Kandlakoya (V), Medchal Road, Hyderabad-501401.

2018-22

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

CERTIFICATE

This is to certify that the project entitled "**A GENERATIVE ADVERSARIAL NETWORK BASED DEEP LEARNING METHOD FOR LOW QUALITY DEFECT IMAGE RECONSTRUCTION AND RECOGNITION**" being submitted by **S.Ruchitha (187R1A05N7), T.Akhila (187R1A05P0) & T.Sony (187R1A05N9)** in partial fulfillment of the requirements for the award of the degree of B.Tech in Computer Science and Engineering to the Jawaharlal Nehru Technological University Hyderabad, is a record of bonafide work carried out by them under our guidance and supervision during the year 2021-22.

The results embodied in this thesis have not been submitted to any other University or Institute for the award of any degree or diploma.

D.Sandhya Rani
Assistant Professor
INTERNAL GUIDE

Dr. A. Raji Reddy
DIRECTOR

Dr. K. Srujan Raju
HoD

EXTERNAL EXAMINER

Submitted for viva voice Examination held on ________________

ACKNOWLEDGEMENT

Apart from the efforts of us, the success of any project depends largely on the encouragement and guidelines of many others. We take this opportunity to express our gratitude to the people who have been instrumental in the successful completion of this project.

We take this opportunity to express my profound gratitude and deep regard to my guide **D.Sandhya Rani,** Assistant Professor for her exemplary guidance, monitoring and constant encouragement throughout the project work.The blessing, help and guidance given by him shall carry us a long way in the journey of life on which we are about to embark.

We also take this opportunity to express a deep sense of gratitude to the Project Review Committee (PRC) **Mr. A. Uday Kiran, Mr. J. Narasimha Rao, Dr. T. S. Mastan Rao, Mrs. G. Latha ,Mr.A.Kiran Kumar** for their cordial support, valuable information and guidance, which helped us in completing this task through various stages.

We are also thankful to **Dr. K. Srujan Raju,** Head, Department of Computer Science and Engineering for providing encouragement and support for completing this project successfully.

We are obliged to **Dr. A. Raji Reddy,** Director for being cooperative throughout the course of this project. We also express our sincere gratitude to Sri. **Ch. Gopal Reddy,** Chairman for providing excellent infrastructure and a nice atmosphere throughout the course of this project.

The guidance and support received from all the members of **CMR Technical Campus** who contributed to the completion of the project. We are grateful for their constant support and help.

Finally, we would like to take this opportunity to thank our family for their constant encouragement, without which this assignment would not be completed. We sincerely acknowledge and thank all those who gave support directly and indirectly in the completion of this project.

S.RUCHITHA (187R1A05N7)

T.AKHILA (187R1A05P0)

T.SONY (187R1A05N9)

ABSTRACT

Machine vision significantly improves the efficiency, quality, and reliability of defect detection of image.In the further development of the field of visual inspection, the application of deep learning will play an increasingly important role. In vision-based defect recognition, deep learning (DL) is a research hotspot. However, DL is sensitive to image quality, and it is hard to collect enough high-quality defect images. The low-quality images usually lose some useful information and may mislead the DL methods into poor results. To overcome this problem, this project proposes a generative adversarial network (GAN)-based DL method for low-quality defect image recognition.Generative Adversarial Networks (GANs) are a thriving unsupervised machine learning technique that has led to significant advances in various fields such as computer vision, natural language processing, among others methods. A GAN is used to reconstruct the low-quality defect images, and a VGG16 network is built to recognize the reconstructed images. The experimental results under low-quality defect images show that the proposed method achieves very good performances, which has accuracies of 95.53-99.62% with different masks and noises, and they are improved greatly compared with the other methods. Furthermore, the results on PSNR, SSIM, cosine, and mutual information indicate that the quality of the reconstructed image is improved greatly, which is very helpful for defect analysis.

LIST OF FIGURES/TABLES

LIST OF SCREENSHOTS

TABLE OF CONTENTS

1.INTRODUCTION

1.INTRODUCTION

1.1 PROJECT SCOPE

This System is based on Generative Adversarial Network. GAN architectures have been devised, with their own unique features for solving specific image processing problems.The aim of our Project is to convert low resolution images in to high quality images so there is no data loss which consist in that image and we can reduce the noise in the images by using this system. Our System has the capability to reconstruct the low quality image and we can also able to recognize whether the reconstructed images of high quality or low quality and we can also able to identify it is real or fake .

1.2 PROJECT PURPOSE

The Purpose of this system is to detect the low quality image and also reconstruct the low quality image to high resolution using a deep learning method GAN. The low-quality images usually lose some useful information and may mislead the DL methods into poor results. To overcome this problem, this project proposes a generative adversarial network (GAN)-based DL method for low-quality defect image recognition.Generative Adversarial Networks (GANs) are a thriving unsupervised machine learning technique that has led to significant advances in various fields such as computer vision, natural language processing, among others.

1.3 PROJECT FEATURES

To Overcome the problem of the data loss this system is used .The main reason for introducing this system is to get high quality image so that it can used in many crime scenes.If we found a image which is of low quality in any crime scene it is hard to collect information from that.By using this system we can get high quality image it is easy to identify the information which we want

2.SYSTEM ANALYSIS

2.SYSTEM ANALYSIS

2.SYSTEM ANALYSIS

System Analysis is the important phase in the system development process.The System is studied to the minute details and analyzed.The system analyst plays an important role of an interrogator and dwells deep into the working of the present system. In analysis, a detailed study of these operations performed by the system and their relationships within and outside the system is done.A key question considered here is,"what must be done to solve the problem?"The system is viewed as a whole and the inputs to the system are identified. Once analysis is completed the analyst has a firm understanding of what is to be done.

2.1 PROBLEM DEFINITION

Recovering a high-resolution image from a single low-resolution image, is a classical problem in computer vision. This problem is inherently ill-posed since a multiplicity of solutions exist for any given low-resolution pixel. In other words, it is an underdetermined inverse problem, of which solution is not unique. Such a problem is typically mitigated by constraining the solution space by strong prior information. To learn the prior, recent state-of-the-art methods mostly adopt the example-based strategy. These methods either exploit internal similarities of the same image, or learn mapping functions from external low- and high-resolution exemplar pair. The external example-based methods can be formulated for generic image super-resolution, or can be designed to suit domain specific tasks, i.e., face hallucination, according to the training samples provided.

2.2 EXISTING SYSTEM

Image Super resolution has been implemented in several different ways using Technologies like Matlab. The system shows acceptable accuracy for image super resolution by implementing Neural Networks in the system.The low-quality images usually lose some useful information and may mislead the DL methods into poor results.And it does not provide good performance.So when there is a low quality image there is loss of the data.

2.2.1 LIMITATIONS OF EXISTING SYSTEM

- The existing systems are implemented on MATLAB and hence are not opensource.
- It takes up more resources and overall gives less accuracy.
- The accuracy of these conventional SR system is not very high and also they work using simple mathematics.
- Significant amount of data is lost while using these algorithms.

2.3 PROPOSED SYSTEM

The proposed system uses a generative adversarial network (GAN)-based DL method for low-quality defect image recognition.The results under low-quality defect images show that the proposed method achieves very good performances, which has accuracies of 95.53-99.62%With different masks and noises, and they are improved greatly compared with the other methods.Furthermore, the results on PSNR, SSIM indicate that the quality of the reconstructed image is improved greatly, which is very helpful for defect analysis.

2.3.1 ADVANTAGES OF THE PROPOSED SYSTEM

- Implementation of Machine Learning and Neural Network increases the efficiency and Accuracy of the system.
- High accuracy,Good performance.
- High flexible compared to other methods.

2.4 FEASIBILITY STUDY

The feasibility of the project is analyzed in this phase and the business proposal is put forth with a very general plan for the project and some cost estimates.During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the user.

Three key considerations involved in the feasibility analysis are

- Economic Feasibility
- Technical Feasibility
- Social Feasibility

2.4.1 ECONOMIC FEASIBILITY

The developing system must be justified by cost and benefit. Criteria to ensure that effort is concentrated on a project, which will give best, return at the earliest. One of the factors, which affect the development of a new system, is the cost it would require.

The following are some of the important financial questions asked during preliminary investigation:

- The costs conduct a full system investigation.
- The cost of the hardware and software.
- The benefits in the form of reduced costs or fewer costly errors.

Since the system is developed as part of project work, there is no manual cost to spend for the proposed system. Also all the resources are already available, it gives an indication that the system is economically possible for development.

2.4.2 TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources.The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

2.4.3 BEHAVIORAL FEASIBILITY

This includes the following questions:

- Is there sufficient support for the users?
- Will the proposed system cause harm

The project would be beneficial because it satisfies the objectives when developed and installed.All behavioral aspects are considered carefully and conclude that the project is behaviorally feasible.

2.5 HARDWARE & SOFTWARE REQUIREMENTS

2.5.1 HARDWARE REQUIREMENTS:

Hardware interfaces specify the logical characteristics of each interface between the software product and the hardware components of the system.The following are some hardware requirements.

- Processor :intel i3/i5
- Harddisk : 128GB
- RAM : 4GB and Above.

2.5.2 SOFTWARE REQUIREMENTS:

Software Requirements specifies the logical characteristics of each interface and software components of the system.The following are some software requirements.

- Operating System : Windows 7+
- Server Side Script : Python 6.6+
- Libraries Used : Pandas,Numpy
- IDE : PyCharm

3.ARCHITECTURE

3.ARCHITECTURE

3.1 PROJECT ARCHITECTURE

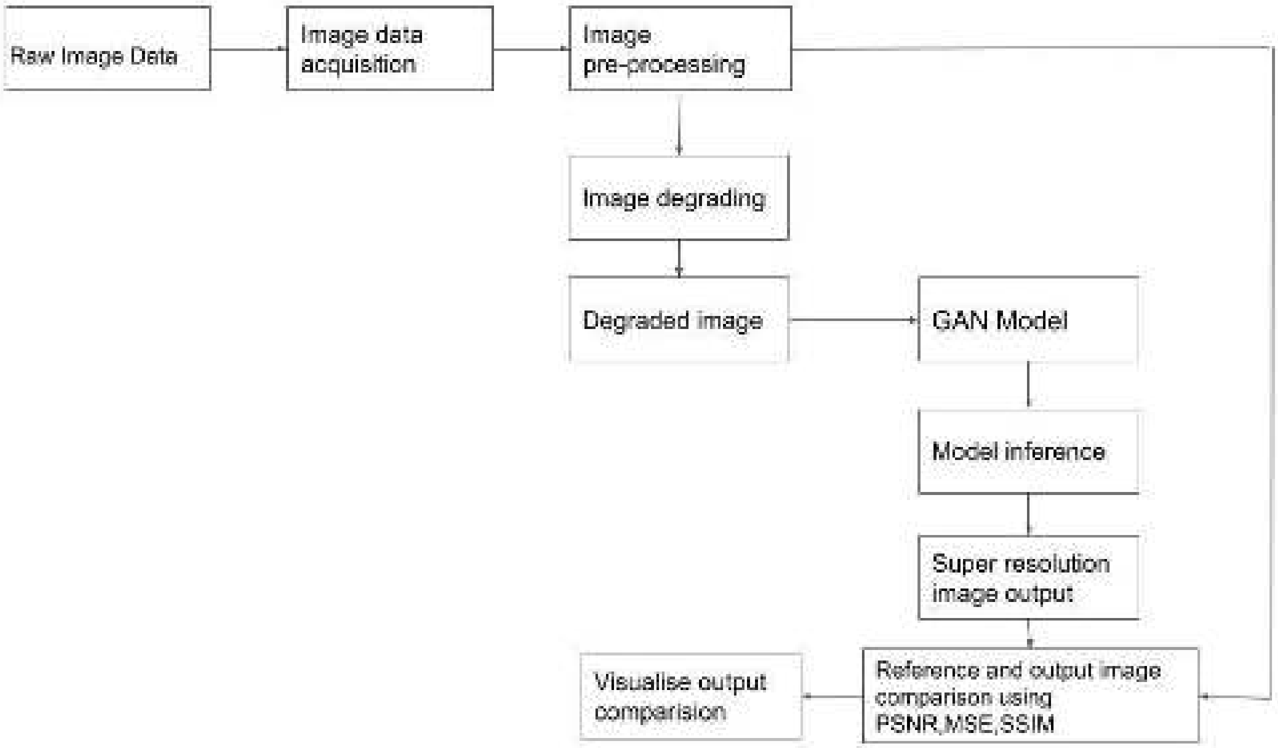


Fig 3.1 Project Architecture

3.2 DESCRIPTION

Raw Image Data: Input data is generally in image format.

Image Data Acquisition : In image processing, image acquisition is an action of retrieving an image from an external source for further processing. It's always the foundation step in the workflow since no process is available before obtaining an image.

Image Preprocessing: Image preprocessing are the steps taken to format images before they are used by model training and inference. This includes, but is not limited to, resizing, orienting, and color corrections.

Image Degrading : Image degradation is the act of loss of quality of an image due to different noises,motion blur etc.

GAN Model: Generative Adversarial Networks (GANs) are a thriving unsupervised machine learning technique.It is basically a system where two competing neural network compete with each other to create or generate variation in data.It consists of two networks:**Generator Network,Discriminator Network.**

Model Inference: Model Inference allows user to interact with the model by changing model inputs,running the simulations and viewing the output.

PSNR:The PSNR block computes the peak signal-to-noise ratio, in decibels, between two images. This ratio is used as a quality measurement between the original and a compressed image. The higher the PSNR, the better the quality of the compressed, or reconstructed image.

SSIM: The structural similarity index measure (SSIM) is a method for predicting the perceived quality of digital television and cinematic pictures, as well as other kinds of digital images and videos. SSIM is used for measuring the similarity between two images.

MSE: The Mean Square Error(MSE) represents the cumulative squared error between the compressed and the original image.

ALGORITHM:

SRGAN (SUPER RESOLUTION GAN)

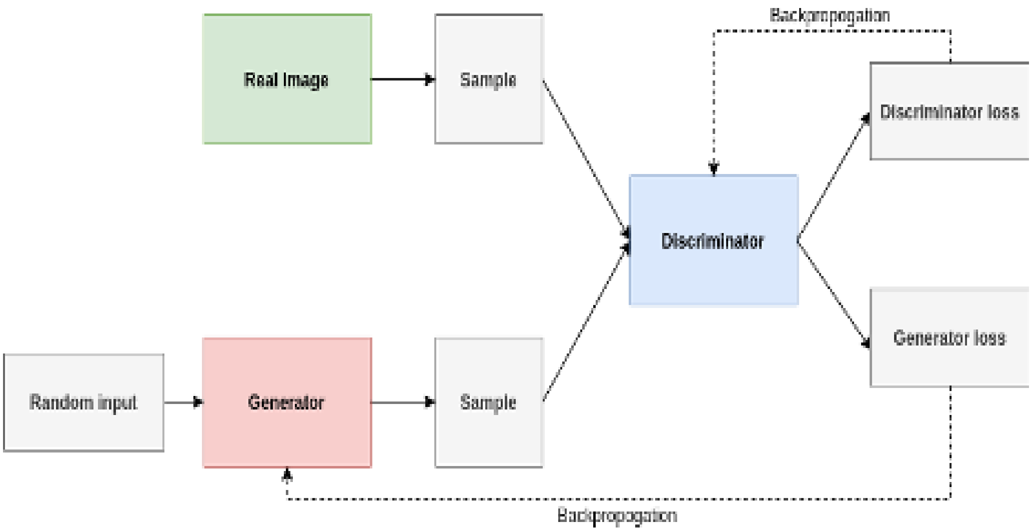


Figure 3.2: SRGAN Network

SRGAN was proposed by researchers at Twitter. The motive of this architecture is to recover finer textures from the image when we upscale it so that it's quality cannot be compromised. There are other methods such as Bilinear Interpolation that can be used to perform this task but they suffer from image information loss and smoothing. In this paper, the authors proposed two architectures the one without GAN (SRResNet) and one with GAN (SRGAN). It is concluded that SRGAN has better accuracy and generate image more pleasing to eyes as compared to SRGAN.

Architecture: Similar to GAN architectures, the Super Resolution GAN also contains two parts Generator and Discriminator where generator produces some data based on the probability distribution and discriminator tries to guess weather data coming from input dataset or generator.Generator than tries to optimize the generated data so that it can fool the discriminator. Below are the generator and discriminator architectural details:

Generator Architecture:

The generator architecture contains residual network instead of deep convolution networks because residual networks are easy to train and allows them to be substantially deeper in order to generate better results. This is because the residual network used a type of connections called skip connections.

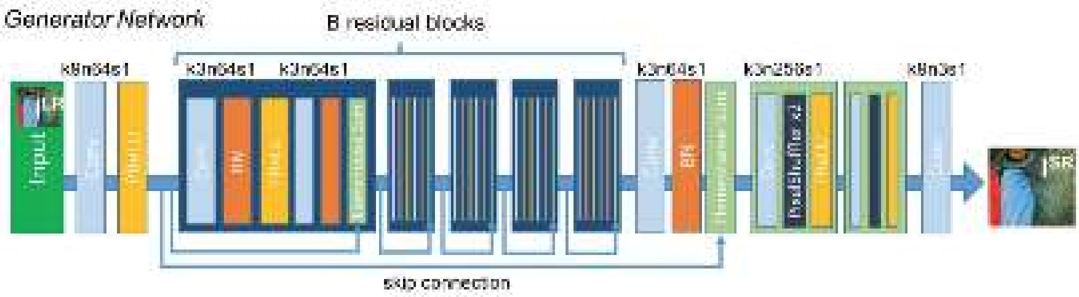


Figure 3.2.1:Generator Network

There are B residual blocks (16), originated by ResNet. Within the residual block, two convolutional layers are used, with small 3×3 kernels and 64 feature maps followed by batch-normalization layers and ParametricReLU as the activation function.

The resolution of the input image is increased with two trained sub-pixel convolution layers.

This generator architecture also uses parametric ReLU as an activation function which instead of using a fixed value for a parameter of the rectifier (alpha) like LeakyReLU. It adaptively learns the parameters of rectifier and improves the accuracy at negligible extra computational cost

During the training, A high-resolution image (HR) is downsampled to a low-resolution image (LR). The generator architecture than tries to upsample the image from low resolution to super-resolution. After then the image is passed into the discriminator, the discriminator and tries to distinguish between a super-resolution and High-Resolution image and generate the adversarial loss which then backpropagated into the generator architecture.

Discriminator Architecture:

The task of the discriminator is to discriminate between real HR images and generated SR images. The discriminator architecture used in this paper is similar to DC- GAN architecture with LeakyReLU as activation. The network contains eight convolutional layers with of 3×3 filter kernels, increasing by a factor of 2 from 64 to 512 kernels. Strided convolutions are used to reduce the image resolution each time the number of features is doubled. The resulting 512 feature maps are followed by two dense layers and a leakyReLU applied between and a final sigmoid activation function to obtain a probability for sample classification.

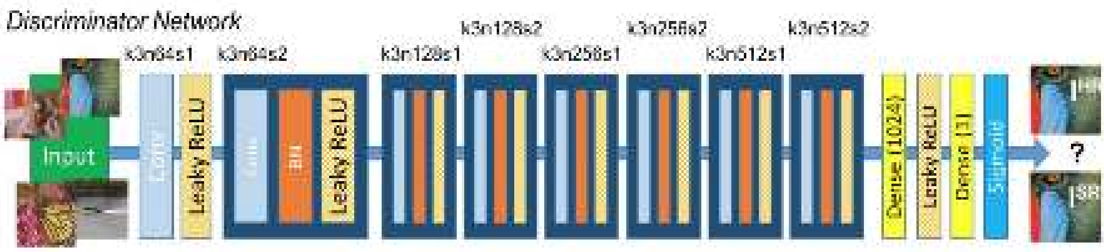


Figure 3.2.2:Discriminator Network

Loss Function:

The SRGAN uses perpectual loss function (LSR) which is the weighted sum of two loss components : content loss and adversarial loss. This loss is very important for the performance of the generator architecture:

Content Loss: We use two types of content loss in this paper : pixelwise MSE loss for the SRResnet architecture, which is most common MSE loss for image Super Resolution. However MSE loss does not able to deal with high frequency content in the image that resulted in producing overly smooth images. Therefore the authors of the paper decided to use loss of different VGG layers. This VGG loss is based on the ReLU activation layers of the pre-trained 19 layer VGG network. This loss is defined as follows:

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

$$l_{VGG/i.j}^{SR} = \frac{1}{W_{i,j}H_{i,j}} \sum_{x=1}^{W_{i,j}} \sum_{y=1}^{H_{i,j}} (\phi_{i,j}(I^{HR})_{x,y} - \phi_{i,j}(G_{\theta_G}(I^{LR}))_{x,y})^2$$

Adversarial Loss: The Adversarial loss is the loss function that forces the generator to image more similar to high resolution image by using a discriminator that is trained to differentiate between high resolution and super resolution images.

3.3 USE CASE DIAGRAM

In use case diagram we have basically one actor who is user.The user have access to the system through which picture is uploaded and result is displayed on the screen.

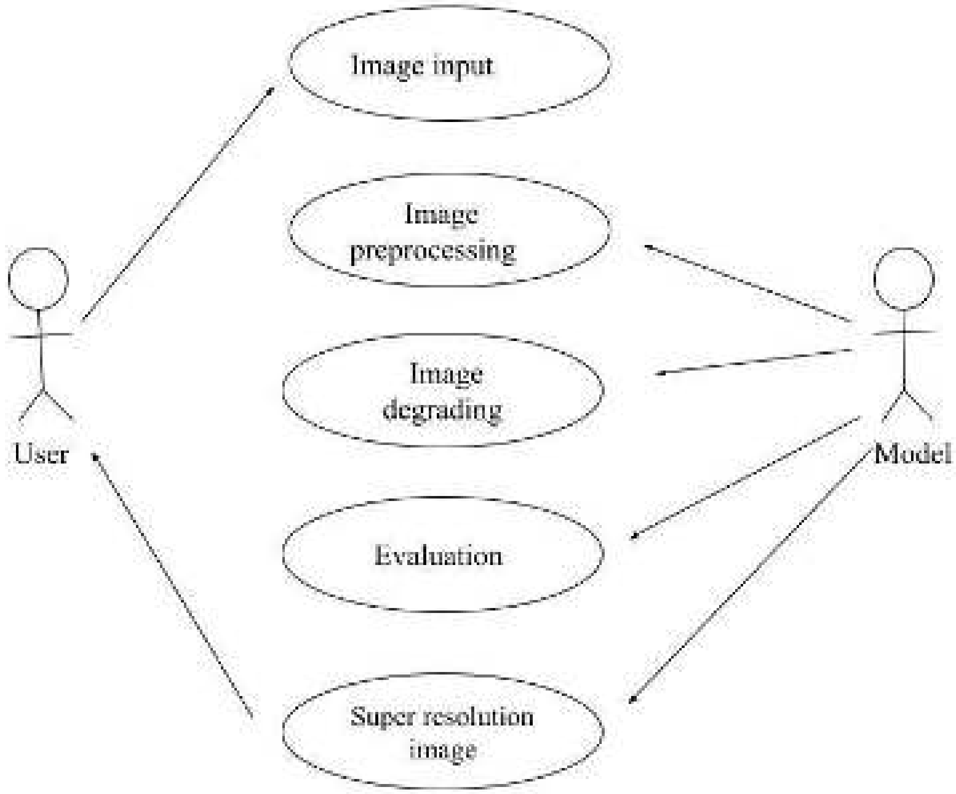


Figure 3.3: Use Case Diagram

3.4 CLASS DIAGRAM

Class Diagram is a collection of classes and objects.

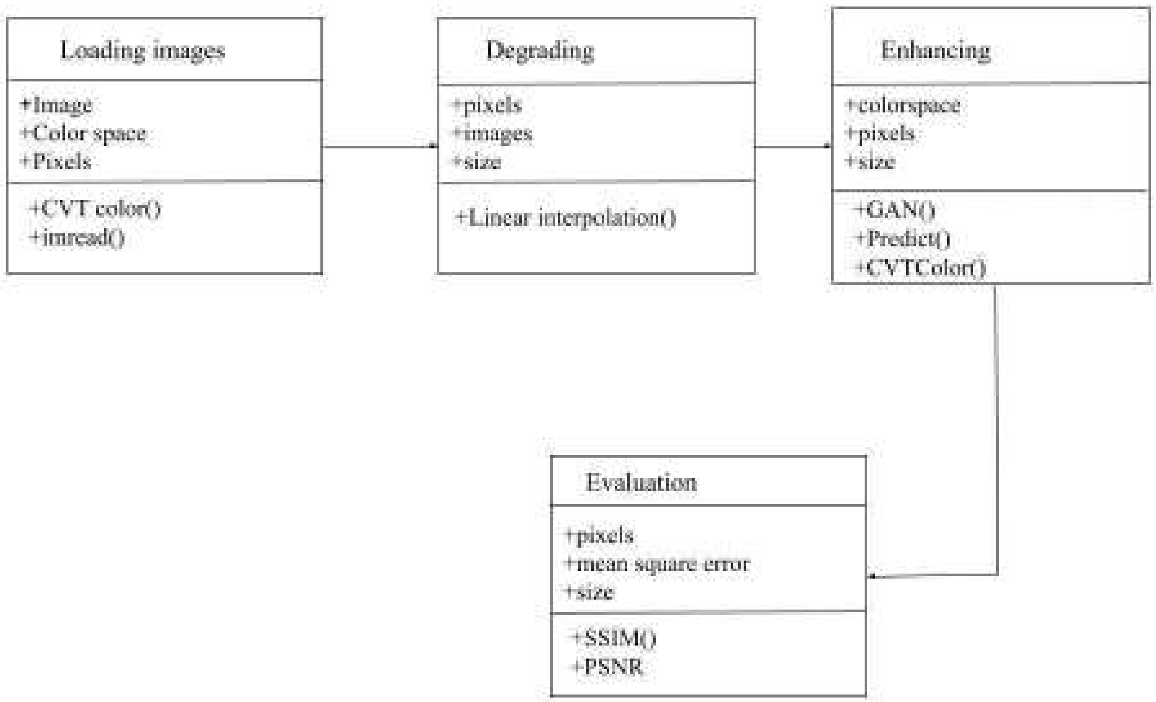


Figure 3.4: Class Diagram

3.5 SEQUENCE DIAGRAM

The sequence diagram shows the sequence in which different tasks are being carried out by the actors.

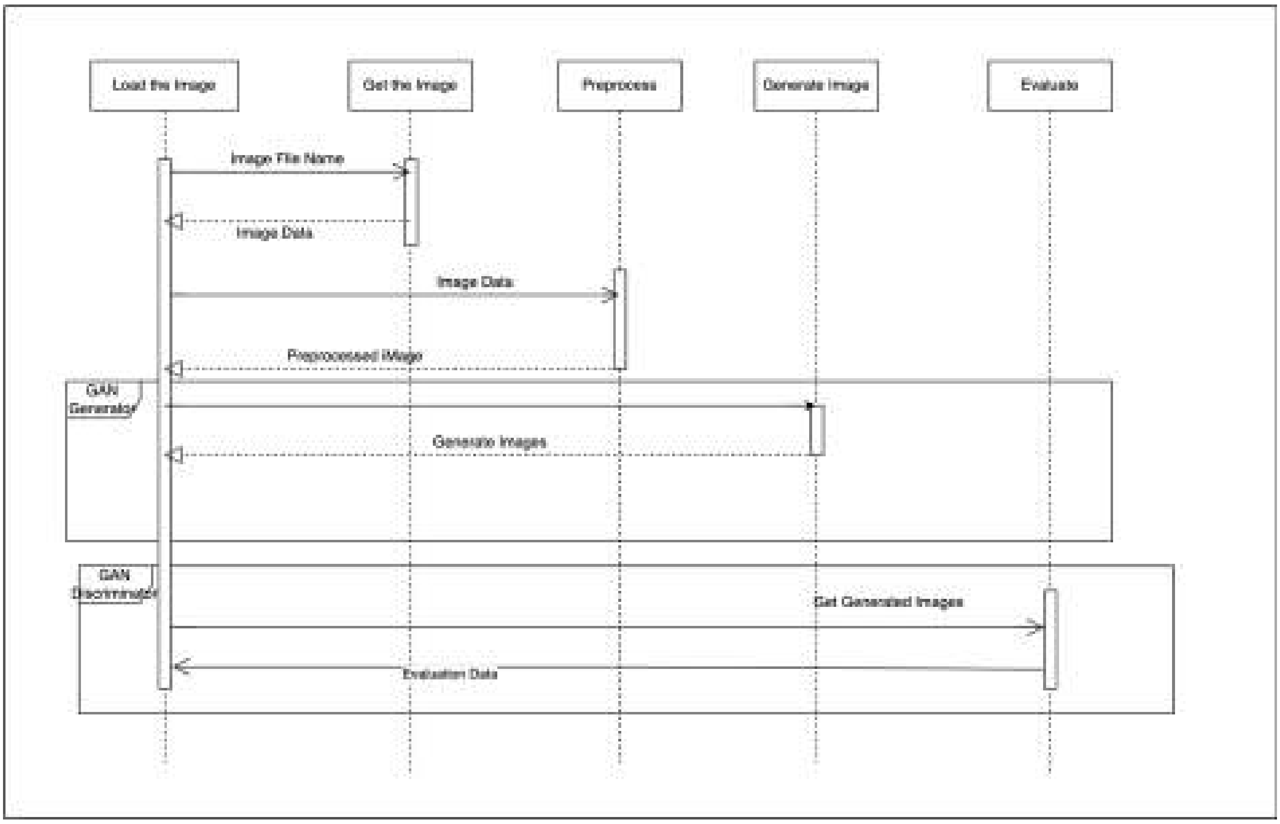


Figure 3.5: Sequence Diagram

3.6 ACTIVITY DIAGRAM

It describes the flow of activity states.

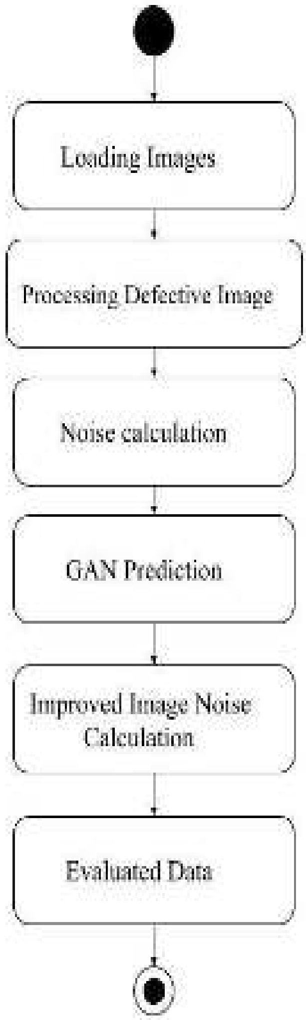


Figure 3.6: Activity Diagram

3.7 DATA FLOW DIAGRAM

Data flow diagram is a way of representing a flow of data through the process or a system

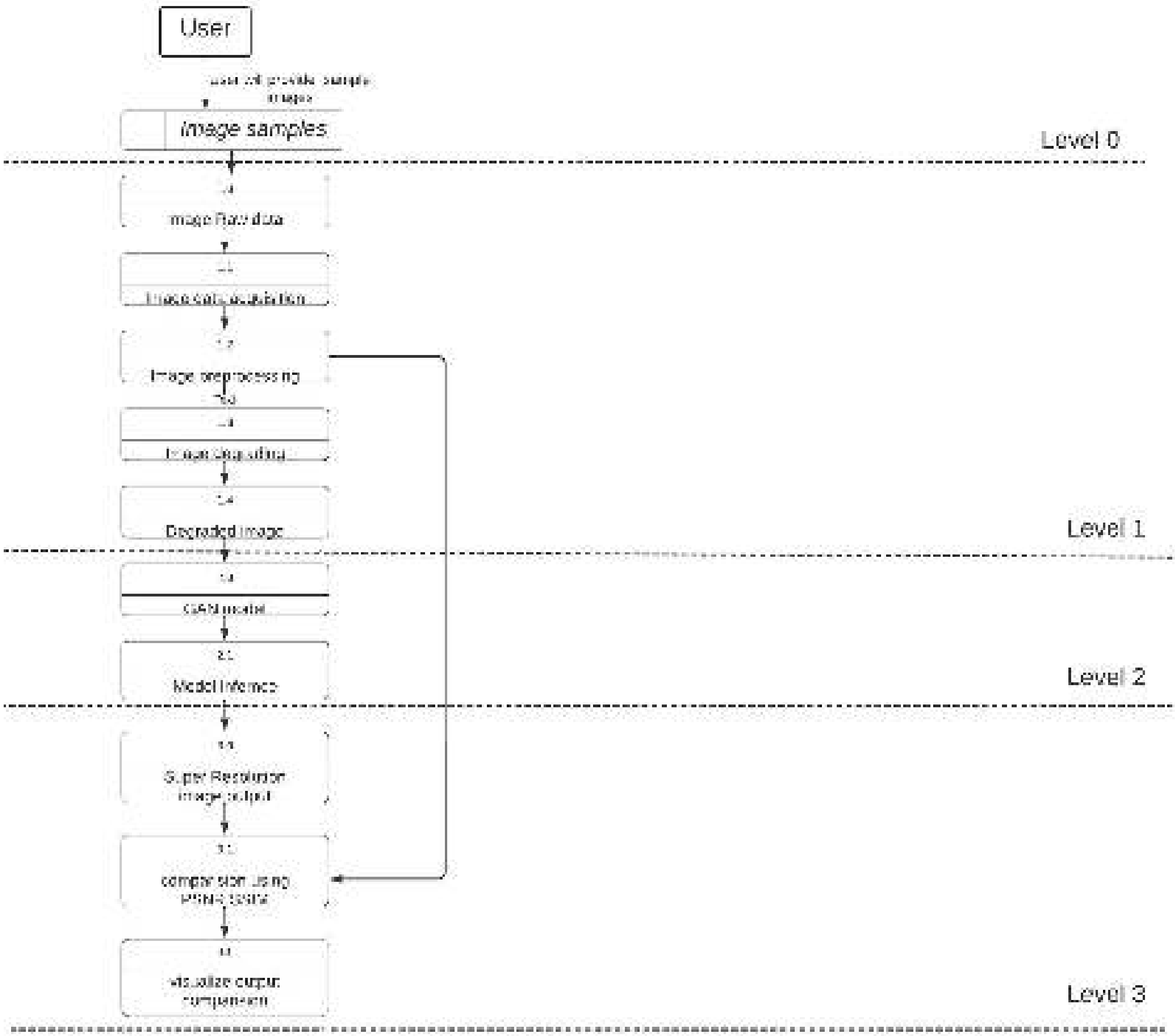


Figure 3.7: Data Flow Diagram

4.IMPLEMENTATION

4. IMPLEMENTATION

4.1 SAMPLE CODE

```
from flask import
Flask,render_template,request,json,jsonify,session,redirect,send_file,url_for,flash

import os

from werkzeug.utils import secure_filename

import SRGAN_module as srgan

import systemcheck


app=Flask(__name__)

app.secret_key="secure"

app.config['UPLOAD_FOLDER'] = str(os.getcwd())+'/static/uploads'


ALLOWED_EXTENSIONS = set(['png', 'jpg', 'jpeg', 'gif','bmp'])


def allowed_file(filename):

    return '.' in filename and filename.rsplit('.', 1)[1].lower() in ALLOWED_EXTENSIONS


@app.route('/',methods=["post","get"])

def first_page():

    if request.method=="POST":

        global image_name,image_data


        file = request.files['file']

        is_degrade = request.form.get("degrade")

        if file.filename == '':

            flash('No image selected for uploading')
```

```
  return redirect(request.url)
        if file and allowed_file(file.filename) :

        filename = secure_filename(file.filename)

        file.save(os.path.join(app.config['UPLOAD_FOLDER'], filename))

        if is_degrade=="on":
            in_file_add = os.path.join(app.config['UPLOAD_FOLDER'], filename)
            res_file_add = os.path.join(app.config['UPLOAD_FOLDER'],
"srgan_degraded_image.jpg")
            srgan.predict_degrade(in_file_add,res_file_add)
            result = "srgan_degraded_image.jpg"
            return render_template("data_page.html", result = result)
        elif is_degrade == None:
            in_file_add = os.path.join(app.config['UPLOAD_FOLDER'], filename)
            res_file_add = os.path.join(app.config['UPLOAD_FOLDER'],
"srgan_normal_image.jpg")
            srgan.predict_normal(in_file_add,res_file_add)
            result = "srgan_normal_image.jpg"
            return render_template("data_page.html", result = result)
        else:
            return redirect(request.url)
    else:
        flash('Allowed image types are -> png, jpg, jpeg, gif')
        return redirect(request.url)
    else:
    return render_template("form_page.html")
```

```
@app.route('/display/<filename>')
def display_image(filename):
    #print('display_image filename: ' + filename)
    return redirect(url_for('static', filename='uploads/' + filename))
app.run(debug=True)
```

5.RESULTS

5.1.1 HOMEPAGE SCREEN

This is the website screen for our model which allows user to upload image by clicking "input image" option

Screenshot 5.1.1: Home Page screen

5.1.2 INPUT IMAGE RESULT

This is the result of the user giving input

Screenshot 5.1.2: Input Image Result

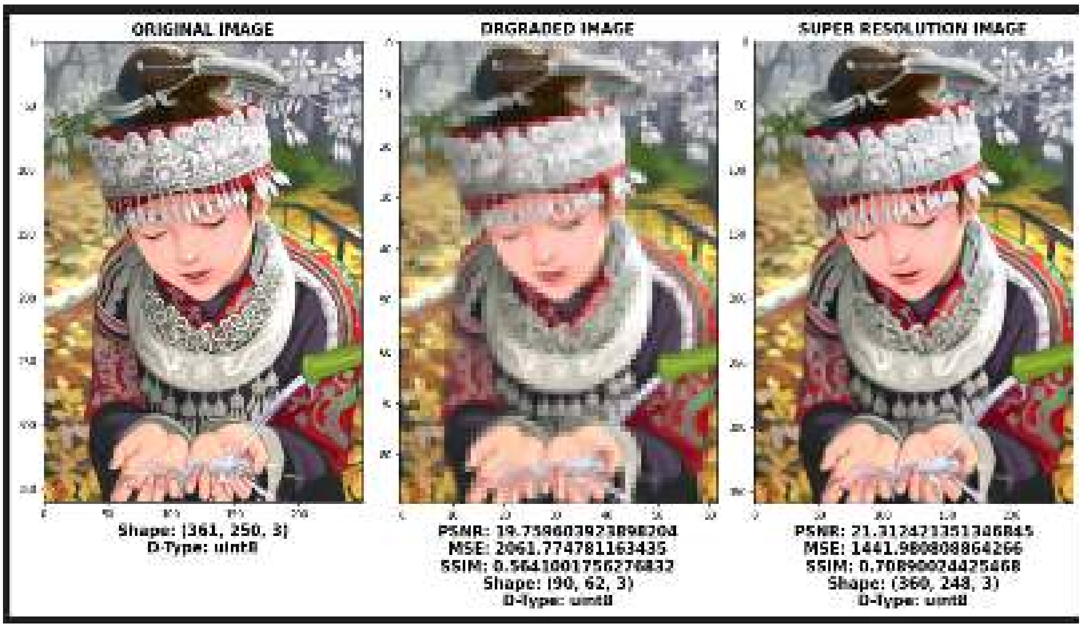
5.1.3 DEGRADE OPTION RESULT

This is the option in which user want to press degrade option

Screenshot 5.1.3: Degrade Option Result

5.1.4 OUTPUT WITH DEGRADING

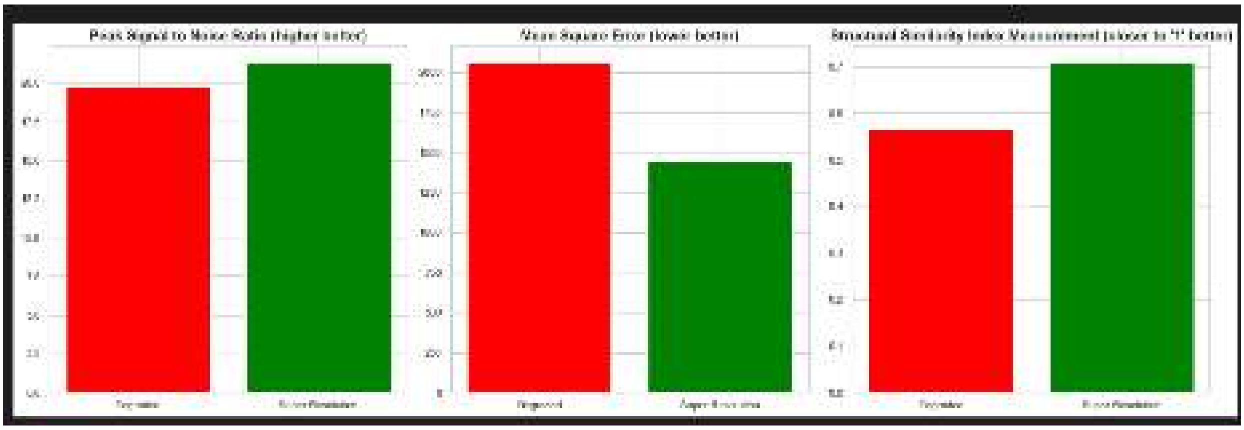
This is where the user gives input image ,he will get the result of degraded image and super resolution image.



Screenshot 5.1.4:Output With Degrading

5.2.1 TEST GRAPH:

This graph depicts the difference between the MSE,PSNR,SSIM values of degraded image and super resolution image.



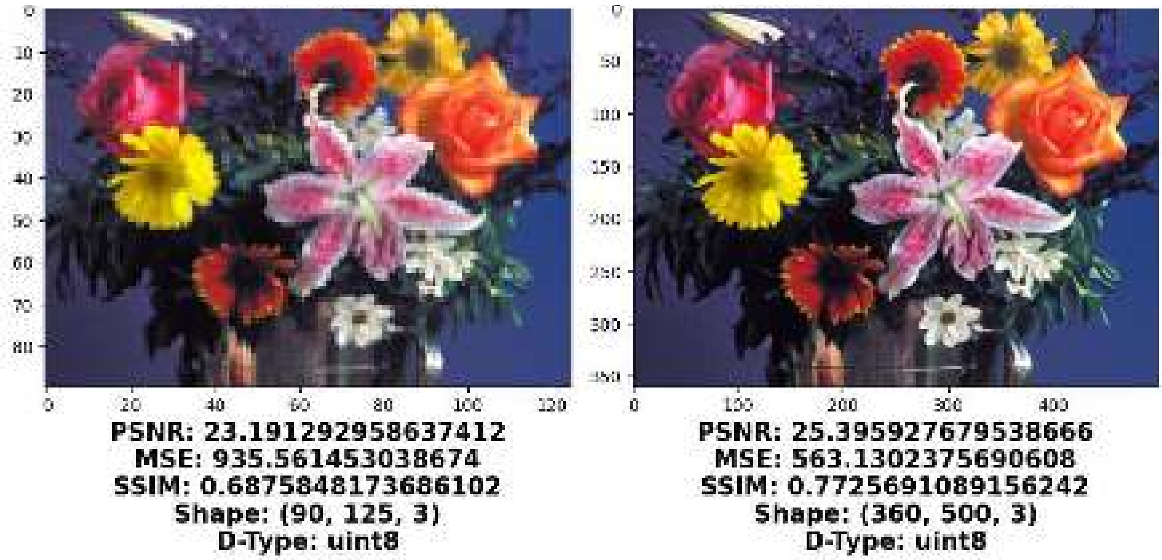
Screenshot 5.2.1: Test Graph

5.1.5 INPUT IMAGE 2 RESULT

The user give another input image without degrade option.

Screenshot 5.1.5:Input Image2 Result

5.1.6 OUPUT WITHOUT DEGRADING



Screenshot 5.1.6:output Without Degrading

6.TESTING

6. TESTING

6.1 INTRODUCTION TO TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product.It provides a way to check the functionality of components, subassemblies, assemblies and/or a finished product.It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner.There are various types of test.Each test type addresses a specific testing requirement.

6.2 TYPES OF TESTING

6.2.1 UNIT TESTING

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs.All decision branches and internal code flow should be validated.It is the testing of individual software units of the application.It is done after the completion of an individual unit before integration.This is a structural testing,that relies on knowledge of its construction and is invasive.Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration.Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

6.2.2 INTEGRATION TESTING

Integration tests are designed to test integrated software components to determine if they actually run as one program.Testing is event driven and is more concerned with the basic outcome of screens or fields.Integration tests demonstrate that although the components were individually satisfaction,as shown by successfully unit testing,the combination of components is correct and consistent.Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

6.2.3 FUNCTIONAL TESTING

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.

Invalid Input : identified classes of invalid input must be rejected.

Functions : identified functions must be exercised.

Output : identified classes of application outputs must be exercised.

Systems/Procedures: interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases.In addition, systematic coverage pertaining to identifying Business process flows; data fields, predefined processes.

6.3 TESTING STRATEGIES

Structural similarity (SSIM) index for measuring image quality (SSIM):

The difference with respect to other techniques mentioned previously such as MSE or PSNR is that these approaches estimate *absolute errors*; on the other hand, SSIM 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. 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.

Peak signal-to-noise ratio (PSNR):

The term peak signal-to-noise ratio (PSNR) is an expression for the ratio between the maximum possible value (power) of a signal and the power of distorting noise that affects the quality of its representation. Because many signals have a very wide dynamic range, (ratio between the largest and smallest possible values of a changeable quantity) the PSNR is usually expressed in terms of the logarithmic decibel scale.

Image enhancement or improving the visual quality of a digital image can be subjective. Saying that one method provides a better-quality image could vary from person to person. For this reason, it is necessary to establish quantitative/empirical measures to compare the effects of image enhancement algorithms on image quality.

Using the same set of tests images, different image enhancement algorithms can be compared systematically to identify whether a particular algorithm produces better results. The metric under investigation is the peak-signal-to-noise ratio. If we can show that an algorithm or set of algorithms can enhance a degraded known image to more closely resemble the original, then we can more accurately conclude that it is a better algorithm.

7.CONCLUSION

7.CONCLUSION & FUTURE SCOPE

7.1 PROJECT CONCLUSION

In vision-based defect recognition, deep learning (DL) is a research hotspot. However, DL is sensitive to image quality, and it is hard to collect enough high-quality defect imagesAlthough state-of-the-art non-machine learning algorithms for image denoising exist, we are constantly wondering that we can achieve better performance with the assistance of deep learning. This system proposes a generative adversarial network (GAN)-based DL method for low-quality defect image recognition.Generative Adversarial Networks, or GAN's for short, are an approach to generative modeling using deep learning methods, such as convolutional neural networks. Furthermore, the results on PSNR, SSIM, cosine, and mutual information indicate that the quality of the reconstructed image has improved greatly, which is very helpful for defect analysis.

7.2 FUTURE SCOPE

We have presented a novel deep learning approach for single image super-resolution (SR). We show that conventional sparse-coding-based SR methods can be reformulated into a deep convolutional neural network. The proposed approach, SRCNN, learns an end-to-end mapping between low- and high-resolution images, with little extra pre/post-processing beyond the optimization. With a lightweight structure, the SRCNN has achieved superior performance than the state-of-the-art methods. Weconjecturethatadditionalperformancecanbefurther gained by exploring more filters and different training strategies. Besides, the proposed structure, with its advantages of simplicity and robustness, could be applied to other low-level vision problems, such as image deblurring or simultaneous SR+denoising. One could also investigate a network to cope with different upscaling factors.

8.BIBLIOGRAPHY

8.BIBLIOGRAPHY

8.1 REFERENCES

1. Aharon, M., Elad, M., Bruckstein, A.: K-SVD: An algorithm for designing overcomplete dictionaries for sparse representation. TSP 54(11), 4311–4322 (2006)
2. Bevilacqua, M., Roumy, A., Guillemot, C., Morel, M.L.A.: Low-complexity singleimage super-resolution based on nonnegative neighbor embedding. In: BMVC (2012)
3. Burger, H.C., Schuler, C.J., Harmeling, S.: Image denoising: Can plain neural networks compete with BM3D? In: CVPR. pp. 2392–2399 (2012)
4. Chang, H., Yeung, D.Y., Xiong, Y.: Super-resolution through neighbor embedding. In: CVPR (2004)
5. Deng, J., Dong, W., Socher, R., Li, L.J., Li, K., Fei-Fei, L.: ImageNet: A large-scale hierarchical image database. In: CVPR. pp. 248–255 (2009)
6. Eigen, D., Krishnan, D., Fergus, R.: Restoring an image taken through a window covered with dirt or rain. In: ICCV. pp. 633–640 (2013)
7. Freedman, G., Fattal, R.: Image and video upscaling from local self-examples. TOG 30(2), 12 (2011)
8. Freeman, W.T., Jones, T.R., Pasztor, E.C.: Example-based super-resolution. Computer Graphics and Applications 22(2), 56–65 (2002)
9. Freeman, W.T., Pasztor, E.C., Carmichael, O.T.: Learning low-level vision. IJCV 40(1), 25–47 (2000)
10. Glasner, D., Bagon, S., Irani, M.: Super-resolution from a single image. In: ICCV. pp. 349–356 (2009)
11. Irani, M., Peleg, S.: Improving resolution by image registration. CVGIP 53(3), 231–239 (1991)
12. Jain, V., Seung, S.: Natural image denoising with convolutional networks. In: NIPS. pp. 769–776 (2008)
13. Jia, K., Wang, X., Tang, X.: Image transformation based on learning dictionaries across image spaces. TPAMI 35(2), 367–380 (2013)
14. Krizhevsky, A., Sutskever, I., Hinton, G.: ImageNet classification with deep convolutional neural networks. In: NIPS. pp. 1097–1105 (2012)
15. LeCun, Y., Boser, B., Denker, J.S., Henderson, D., Howard, R.E., Hubbard, W., Jackel, L.D.: Backpropagation applied to handwritten zip code recognition. Neural computation pp. 541–551 (1989)
16. LeCun, Y., Bottou, L., Bengio, Y., Haffner, P.: Gradient-based learning applied to document recognition. Proceedings of the IEEE 86(11), 2278–2324 (1998)
17. Lee, H., Battle, A., Raina, R., Ng, A.Y.: Efficient sparse coding algorithms. In: NIPS. pp. 801–808 (2006)
18. Nair, V., Hinton, G.E.: Rectified linear units improve restricted Boltzmann machines. In: ICML. pp. 807–814 (2010)
19. Schuler, C.J., Burger, H.C., Harmeling, S., Scholkopf, B.: A machine learning approach for non-blind image deconvolution. In: CVPR. pp. 1067–1074 (2013)

20. Timofte, R., De Smet, V., Van Gool, L.: Anchored neighborhood regression for fast example-based super-resolution. In: ICCV. pp. 1920–1927 (2013)
21. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. TIP 13(4), 600–612 (2004)
22. Yang, C.Y., Yang, M.H.: Fast direct super-resolution by simple functions. In: ICCV. pp. 561–568 (2013)
23. Yang, J., Lin, Z., Cohen, S.: Fast image super-resolution based on in-place example regression. In: CVPR. pp. 1059–1066 (2013)

8.2 WEBSITES

www.google.com
www.geeksforgeeks.org/generative-adversarial-network-gan
https://code.visualstudio.com/

8.3 DRIVE LINK

https://drive.google.com/drive/folders/19F0_lwpea_NtjSLVa6SpFClpFvtVccgG?usp=sharing