PRELIMENERY REPORT ON

"GRAY SCALE IMAGE COLORIZATION USING DEEP LEARNING"

SUBMITTED TO THE SAVITRIBAI PHULE UNIVERSITY, PUNE
IN THE PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE AWARD OF THE DEGREE

OF

BACHELOR OF ENGINEERING (COMPUTER ENGINEERING)

SUBMITTED BY

Miss Divya Dinesh Pathak	Exam No. 64
Miss Vaidehi Sudhir Patil	Exam No. 35
Miss Gauri Pravin Sangale	Exam No. 17
Miss Vranda Vikash Gupta	Exam No. 29

AISSMS
INSTITUTE OF INFORMATION TECHNOLOGY
ADDING VALUE TO ENGINEERING
Approved by AICTE New Delhi, Recognized by the Government of Maharashtra
and Affiliated to Savitribai Phule Pune University.
Accredited by NAAC with A grade

DEPARTMENT OF COMPUTER ENGINEERING

ALL INDIA SHRI SHIVAJI MEOMRIAL SOCIETY'S

INSTITUTE OF INFORMATION TECHNOLOGY

KENNEDY ROAD, NEAR R.T.O. PUNE-411001

SAVITRIBAI PULE PUNE UNIVERSITY

2020-2021

AISSMS
INSTITUTE OF INFORMATION TECHNOLOGY
ADDING VALUE TO ENGINEERING
Approved by AICTE New Delhi, Recognized by the Government of Maharashtra
and Affiliated to Savitribai Phule Pune University.
Accredited by NAAC with A grade

CERTIFICATE

This is to certify that the project report entitles

"GRAY SCALE IMAGE COLORIZATION USING DEEP LEARNING"

Submitted by

Miss Divya Dinesh Pathak	Exam No. 64
Miss Vaidehi Sudhir Patil	Exam No. 35
Miss Gauri Pravin Sangale	Exam No. 17
Miss Vranda Vikash Gupta	Exam No. 29

is a bonafide student of this institute and the work has been carried out by him/her under the supervision of **Dr. S.N. Zaware** and it is approved for the partial fulfillment of the requirement of Savitribai Phule Pune University, for the award of the degrees of **Bachelor of Engineering** (Computer Engineering).

(Dr. S.N. Zaware)	**(Dr. S.N. Zaware)**
Guide	Head
Department of Computer Engineering	Department of Computer Engineering

(Dr. P.B. Mane)
Principal
AISSMS Institute of Information Technology, Pune-411001

Place: PUNE
Date:

ABSTRACT

Former approaches to the gray scale image colorization problem rely on manual methods with human intervention that produced some de-saturated results that are not likely to be true colorizations. Some of these methods were example based colorization, scribble based colorization. To fully automate the process of colorization, enormous amount of computation power and datasets are required which is daunting. Many approaches used Deep Learning methods such as CNN to automate the process and could successfully train on colossal datasets, but due to inaccessibility of the above mentioned requirements and studying various approaches we have defined two approaches to solve the problem. The first approach: Transfer learning using Encoder-Decoder, uses a pre-trained network called “VGG-16” as an encoder.We explored the fact that a colorization problem is basically how meritoriously we could classify an image and then colorize it accurately. Using transfer learning, decoder is able to predict the color channels of the images based on features extracted by VGG-16 encoder. The second approach: Generative Adversarial Network, uses two networks: a generator and a discriminator. The generator network will generate fake colorized images and will try to fool the discriminator network. The discriminator network compares the fake colorized image produced by the generator network and the real ground truth image. If it is able to distinguish between the above mentioned two images, it sends a feedback to the Generator Network that a “fake image is produced”, along with the losses.

The outcome of our project is the colorized image. Many approaches are proposed that uses CNN to colorize images. Very less study has been done on using GAN in image colorization problem. We use two approaches i.e, the first approach help solve the problem of computation and dataset requirement, the second helps solve the problem of false colorization as the accuracy of GAN is very high. Out Scope of the project is to combine both the approaches to increase the accuracy in terms of

loss functions and number of epochs.

0.1 Technical Keywords

Technical Key words related to our Proposed System are:

- Vgg-16
- GAN
- Transfer Learning
- Encoder
- Decoder
- Generator
- Discriminator
- Generator Loss
- Discriminator Loss
- Google Colab
- Image Colorization
- Autoencoders

ACKNOWLEDGMENT

It gives us immense pleasure in presenting the preliminary project report on **"GRAY SCALE IMAGE COLORIZATION USING DEEP LEARNING"** The success and the outcome of this report required a lot of guidance. We are very grateful to our guide **Dr. S.N. Zaware** who has provided expertise and encouragement. We thank madam who provided vision and knowledge that was very helpful throughout the research. All that we have done is only due to the great guidance.

We would also like to extend our sincere thanks to Principal Dr.P.B. Mane, for his dynamic and valuable guidance throughout the project and providing the necessary facilities that helped us to complete our dissertation work.

Miss Divya Dinesh Pathak
Miss Vaidehi Sudhir Patil
Miss Gauri Pravin Sangale
Miss Vranda Vikash Gupta

INDEX

List of Figures

Chapter 1

INTRODUCTION

Image colorization is an interesting topic in image to - image translation. Nowadays, many cameras are still capturing greyscale images, like surveillance cameras and satellite cameras. In this project, we implemented image colorization framework based on deep learning methods. The first approach: Transfer learning using Encoder-Decoder, uses a pre-trained network called "VGG-16" as an encoder. VGG-16 is a CNN which is developed by the Oxford University and has won an ImageNet competition in the year 2014 which is able to classify images based on 1000 classes. The second approach: Generative Adversarial Network, uses two networks: a generator and a discriminator. The generator network will generate fake colorized images and will try to fool the discriminator network. The discriminator network compares the fake colorized image produced by the generator network and the real ground truth image. If it is able to distinguish between the above mentioned two images, it sends a feedback to the Generator Network that a "fake image is produced", along with the losses. Very less study has been done on using GAN in image colorization problem. We use two approaches as mention above. We verify their colorization performance based on different number of epochs used, and we find Generative Adversarial Network is a powerful model to produce plausible colored images.

Gray Scale image colorization is a very old research topic which mainly had focused on to reinstate the ancient images and documents. But the scope of our project does not limit itself to just restore old images and documents, there are many cameras that do not have color sensors and fails to capture color images, to name a few: Surveillance cameras,MRI imaging,etc. We need to colorize images produced by these cameras with plausible coloring.

1.1 MOTIVATION

The motivation behind choosing the domain is that gray Scale image colorization is a very old research topic which mainly had focused on to reinstate the ancient images and documents. But the scope of our project does not limit itself to just restore old images and documents, there are many cameras that do not have color sensors and fails to capture color images, to name a few: Surveillance cameras,MRI imaging,etc. We need to colorize images produced by these cameras with plausible coloring. Also, ancient black and white flims and photographs colorization are our domain of interest.

1.2 PROBLEM DEFINITION

1. To automatically colorize gray scale images using Vgg-16 and GAN without human intervention.
2. To perform transfer learning.
3. To effectively choose and apply loss functions so as to decrease the level of desaturation of the output colorized image.
4. To understand the accuracy of different classes based upon number of datasets used for training, epochs and batch size.

1.3 Input to the Project

1.3.1 Grayscale image

- Each pixel is represented using 8 bits which denotes the intensity of that pixel
- The higher the value, the greater the intensity. Current displays support 256 distinct shades of gray. Each one just a little bit lighter than the previous one
- In the memory, a gray scale image is represented by a two dimensional array of bytes. Technically, this array is a ”channel”. So, a gray scale image has only one channel. And this channel represents the intensity of whites

1.3.2 Datasets

The following are some sources for input images for training and testing purposes:

- Imagenet
- cfair-100
- Kaggle

Figure 1.1: Input Gray Scale Image

Chapter 2

LITERATURE SURVEY

1. **Colourful image colorization**
   The report explains the importance and accuracy of pre trained CNN on Image Net to solve this problem. ImageNet is a database consisiting of large number of datasets for common classes like dogs, cats, humans, etc.

2. **Image colorization with deep convolutional neural networks.**
   The report uses the VGG-16 model to colorize images. Vgg-16 is a pre trained network which has been trained on 1000 classes from ImageNet. It has won an ImageNet competition for successfully classifying images based on 1000 classes with an accuracy of approx. 93 percent.

3. **End-to-End Conditional GAN-based Architectures for Image Colourisation**
   The paper proves to be an evidence that GAN is the best model to use for colorization problem. GAN is tough to train as it requires lot of data and computation power. Hence, training GAN on GoogleColab proves to be best.

4. **Implementation of image colorization with convolutional neural network.**
   The paper uses VGG-19 and GAN to colorize images. This method proves to give good accuracy for the images. Here, it is understood that both the models are good for some classes. VGG shows good results for dog, cat, classes.

Chapter 3

SOFTWARE REQUIREMENTS SPECIFICATION

3.1 Project Scope

1. To automatically colorize gray scale images using Vgg-16 and GAN without human intervention.
2. To perform transfer learning.
3. To effectively choose and apply loss functions so as to decrease the level of desaturation of the output colorized image.

3.2 System Requirements

3.2.1 Software Requirements

1. **Jupyter-Notebook/Spyder** Its an IDE used for the computing various aspects in different programming languages. Data cleaning, data transformation, numerical simulation, modeling, visualization, machine learning.
2. **Keras** Keras is an open-source software library that provides a Python interface for artificial neural networks.
3. **NumPy** NumPy is a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays.
4. **Scikit-image** scikit-image is an open-source image processing library for the Python programming language. It includes algorithms for segmentation, geometric transformations, color space manipulation, analysis, filtering, morphology, feature detection, and more

5. **Tensorflow** TensorFlow is a free and open-source software library for machine learning. It can be used across a range of tasks but has a particular focus on training and inference of deep neural networks.

6. **Google Colab** Colab allows anybody to write and execute arbitrary python code through the browser, and is especially well suited to machine learning, data analysis and education.

3.2.2 Hardware Requirements

1. **Disk Space:** Minimum disk space of 500 GB is expected for computations and storage means.

2. **Processor** i5 CPU @1.60 GHz 1.80 GHz, 32-bit x32 OR 64-bit x64 processor is preferable.

3. **Memory:** 4 GB RAM and above .

4. **GPU:** 2GB GPU and above.

3.3 System Implementation Plan

In the system plan implementation the input is the GrayScale image. Using the two approaches - Approach 1: Transfer Learning using Encoder Decoder and Approach 2; Generative Adversarial Network , ab color channel is predicted. The expected output is the colorized image.

Figure 3.1: System Plan

Chapter 4

SYSTEM DESIGN

4.1 System Architecture

4.1.1 Approach 1: Transfer Learning using Encoder Decoder

1. Input Gray scale image
2. Encoder network
3. Decoder network
4. Output is the colorized image

4.1.2 Approach 2: Generative Adversarial Network

1. Input Gray scale image
2. Generator network
3. Discriminator network
4. Output is the colorized image

4.2 UML Diagrams

4.2.1 Activity Diagram

- Activity Diagram is a behavioral diagram presenting the actors their functions performed.

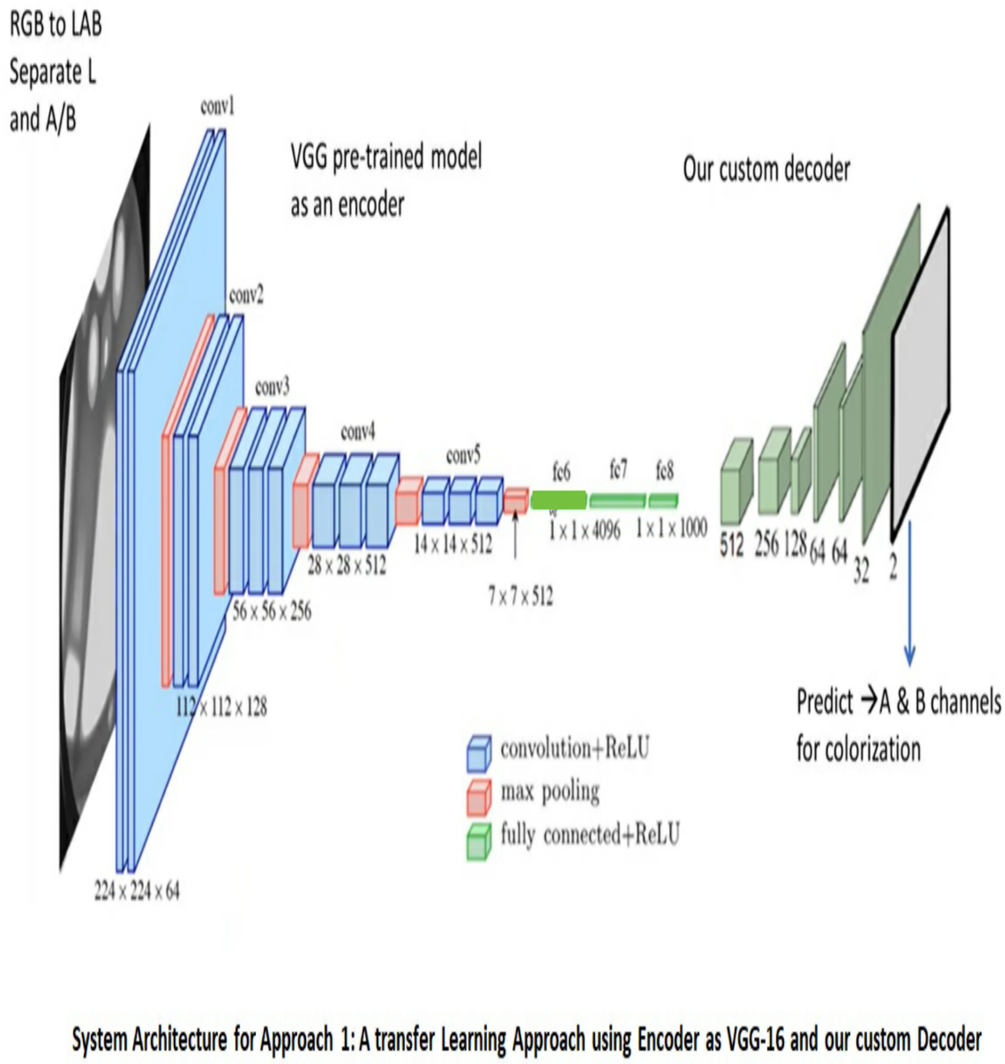


Figure 4.1: System Architecture for Approach 1

- They also include the swim lanes and the forks and joins.
- The represent the individual lane as their entire activities and the functionality carried out by that particular actor in the respective lanes.
- Input is being provided by the user and the output is being given back to the user.

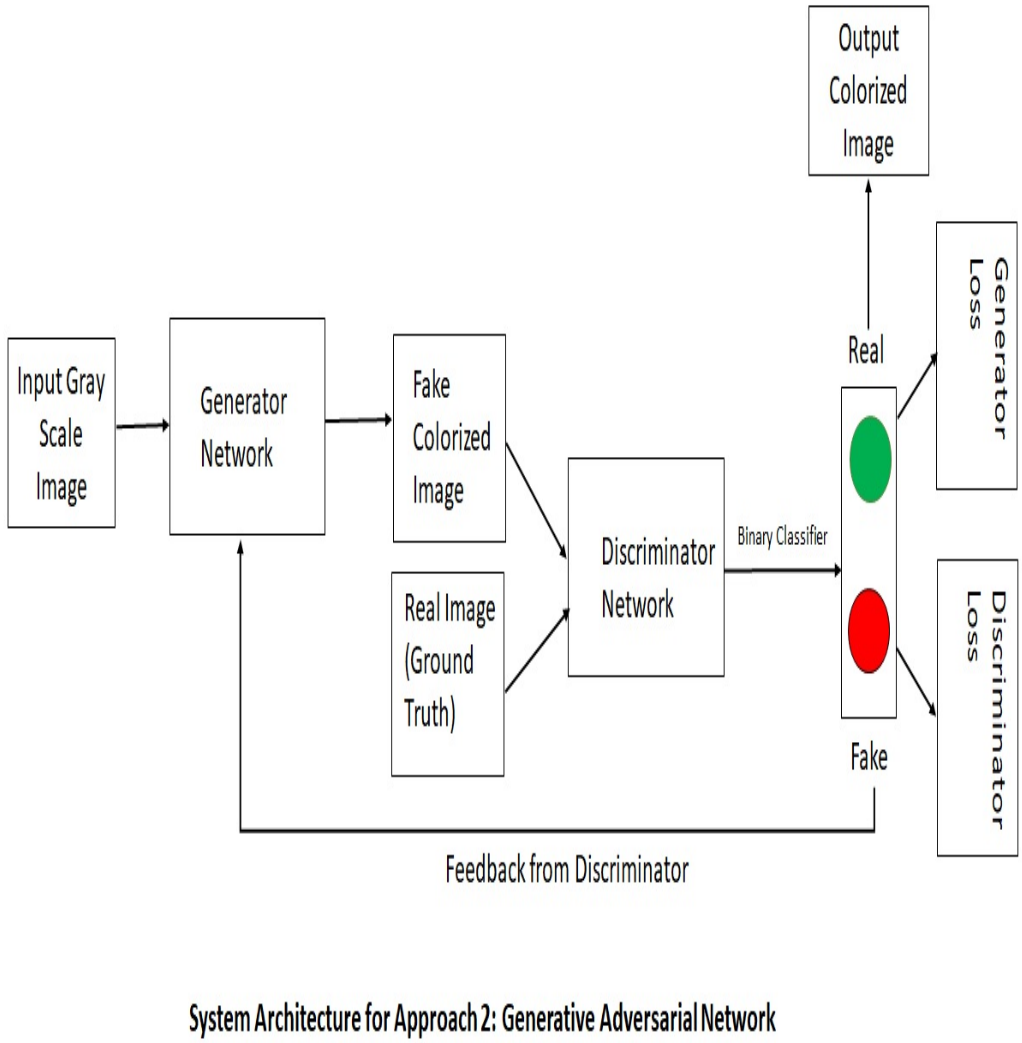


Figure 4.2: System Architecture for Approach 2

4.2.2 Use case Diagram

Use Case diagram is used for representing the problem statement that is the actors in it, their functionality in an behavioral manner. They are useful when the system is to be in the programmatic execution.

- Use Case diagram has the actors and all the functional blocks.

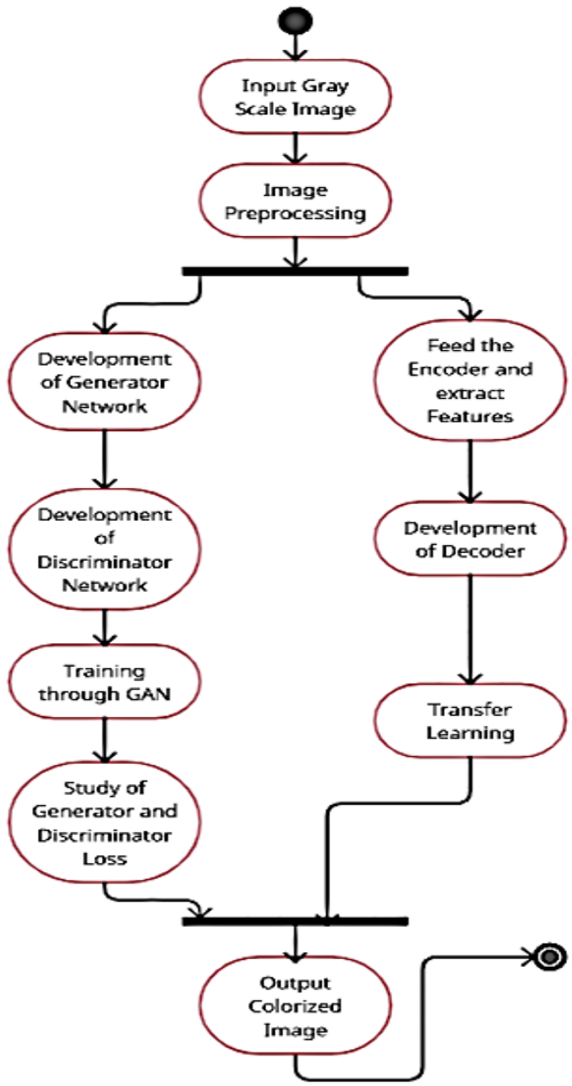


Figure 4.3: Activity Diagram

- **Actors**
  1. User
  2. Colorization System
- Functional Blocks include the stepwise execution of the entire system that is the form of different procedures (functionality).

- Input to the every function in this diagram is the output from the previous state.

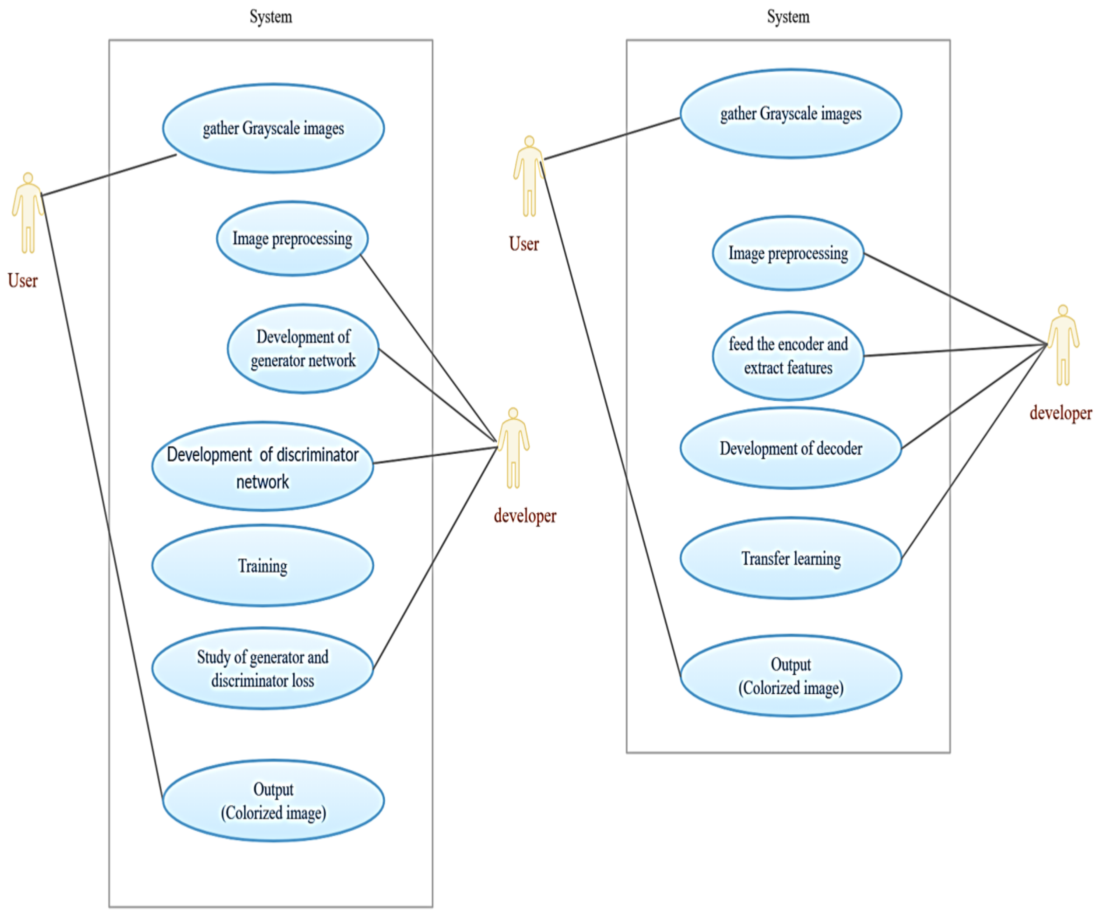


Figure 4.4: Use Case Diagram

4.3 Data Flow Diagrams

4.3.1 DFD Level 0

In the data dependency diagram at level zero the input is the gray scale dataset and the output is the colorized image.

4.3.2 DFD Level 1

In the data dependency diagram at level one the input is the gray scale dataset and the output is the colorized image.The input image is fed in 2 modules-VGG or GAN.

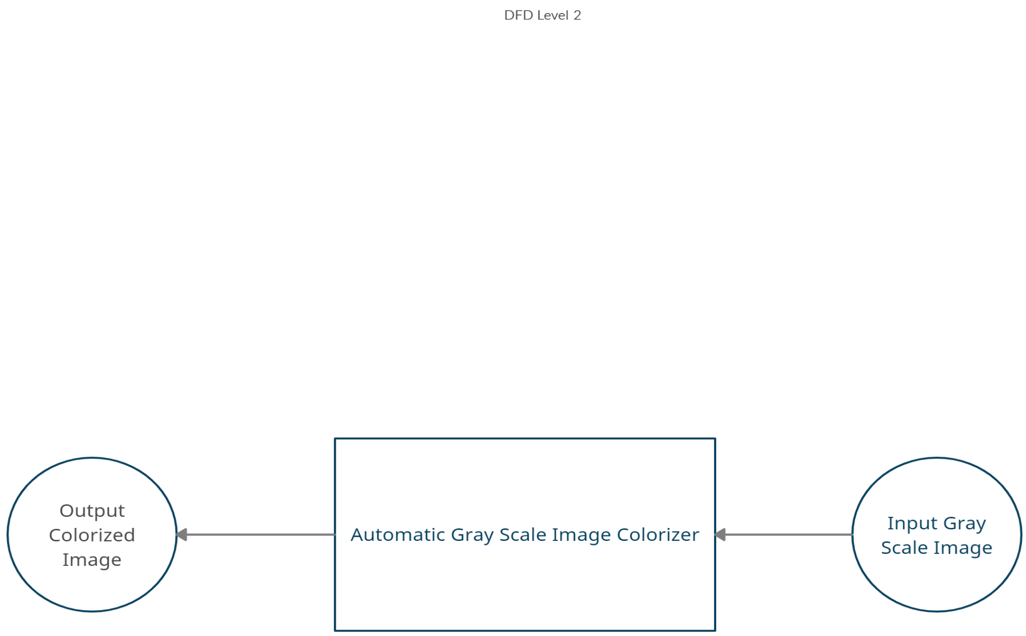


Figure 4.5: DFD Level 0

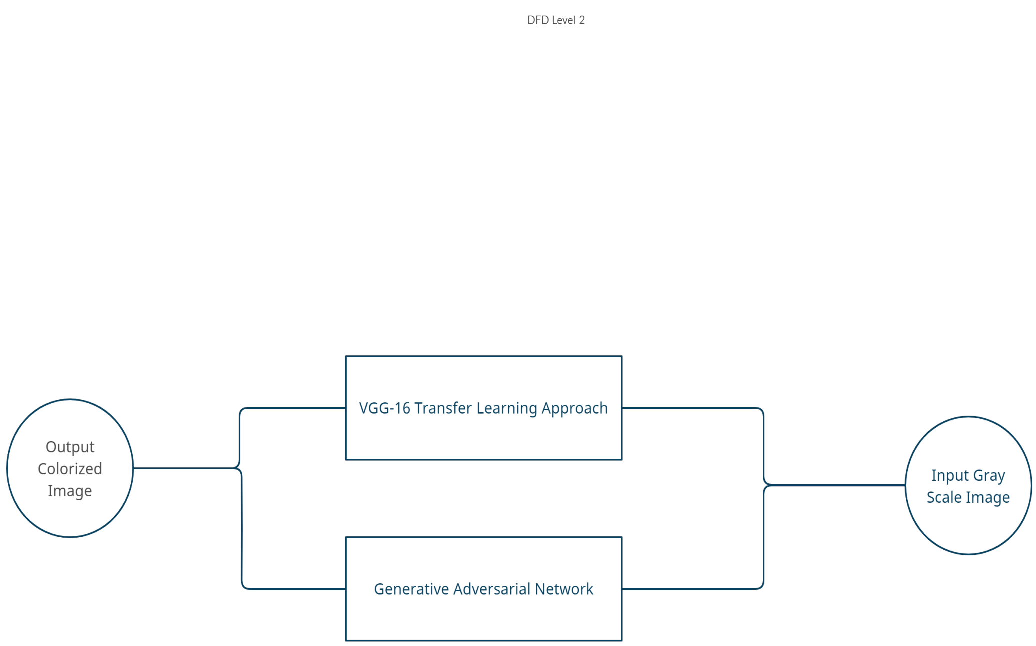


Figure 4.6: DFD Level 1

Chapter 5

LIST OF MODULES AND FUNCTIONALITY

5.1 For Approach 1: Transfer Learning using Encoder-Decoder

- **Input Gray scale image pre-processing:**

  1. Perform dimension scaling of the input image based on the Dataset used for training.
  2. Convert the input Gray Scale image (1 channel) into RGB channel by replicating input channel 3 times. Then, convert RGB image into CIE Lab image. Now, the input Gray scale is the L (Lightness channel).

- **Feeding the Encoder and extract features :** Feed the L channel Gray scale image into VGG-16 and extract features of the image.

- **Development of Decoder :** A decoder network consists of convolutional layers with up sampling layers to restore the dimensions of the original input image. A decoder network is developed using Python.

- **Perform Transfer learning from encoder to decoder and predict ab channels :**

  1. The extracted features from VGG-16 encoder is fed into the decoder network.
  2. The decoder network will then predict the ab channels using the extracted features.

- Combine input channel L and predicted channels ab from the decoder to form CIE Lab output image.

- Conversion of CIE Lab color space output image to RGB color space image.

5.2 For Approach 2: Generative Adversarial Network

- **Input Gray scale image pre-processing:** Perform dimension scaling of the input image based on the Dataset used for training.

- **Development of the Generator network:** A generator network is developed using Python. The generator network will generate fake colorized images and will try to fool the discriminator network. It generates fake colorized images until the discriminator network cannot distinguish between the fake colorized and the real ground truth image and thinks as if both are same.

- **Development of the Discriminator network:** A discriminator network is developed using Python. It compares the fake colorized image produced by the generator network and the real ground truth image. If it is able to distinguish between the above mentioned two images, it sends a feedback to the Generator Network that a "fake image is produced", along with the losses.

- **Train the images through GAN:** The process of Generator trying to fool Discriminator is the training of GAN. Every time the generator gets a feedback from the Discriminator, it learns more and generates fake colorized images until the discriminator network cannot distinguish between the fake colorized and the real ground truth image and thinks as if both are same. Once, the discriminator cannot distinguish between real and fake images, training step is successfully accomplished.

- **Comparative Study of Generator and Discriminator Loss on different number of epochs:** As the no of epochs increases, the losses decreases. So, a study is performed based on no of epochs and losses in generator and discriminator network.

- **Extract the Output Image after successful training:** After the GAN is trained successfully, we get the output colorized image for the input gray scale images.

Chapter 6

ALGORITHMS

- **Convolutional Neural Network -** Visual Graphics Group(VGG)-16 VGG16 is a convolutional neural network model. The input to cov1 layer is of fixed size 224 x 224 RGB image. The image is passed through a stack of convolutional (conv.) layers, where the filters were used with a very small receptive field: 3×3. Spatial pooling is carried out by five max-pooling layers, which follow some of the conv. Layers. Three Fully-Connected (FC) layers follow a stack of convolutional layers. The final layer is the soft-max layer. We are not going to use the last three predicting layers of VGG-16

- **GAN -** Generative modelling is an unsupervised learning task in machine learning that trains a generative model by framing the problem as a supervised learning problem with two sub-models: the generator model that we train to generate new examples, and the discriminator model that tries to classify examples as either real (from the domain) or fake (generated). The two models are trained together in a zero-sum game, adversarial, until the discriminator model is fooled about half the time, meaning the generator model is generating plausible examples.

Chapter 7

RELEVANT MATHEMATICS ASSOCIATED

SYSTEM DESCRIPTION:

- **Input :** Gray Scale Image
- **Output :** Colorized image
- **Functions :** We are using two functions in our project i.e -
  1. **Loss Functions:** The goal is to minimize the loss over the training set. In the colorization problem, the training data consists of thousands of color images and their grayscale versions. Examples are L1, Cross entropy.
  2. **Color rebalancing:** It is used to rebalance the loss based on the rarity of the color class. This contributes towards getting more vibrant and saturated colors in the output. It makes many images very lively and vibrant. Most of them are plausible colors.

Chapter 8

OTHER SPECIFICATION

8.1 Advantages

- Colorization results are very accurate for the classes which are already been trained on VGG-16 like dog, macaw, rabbit, etc.

8.2 Limitations

- The system requires a very large dataset for training and hence, lot of computation power and time.

8.3 Applications

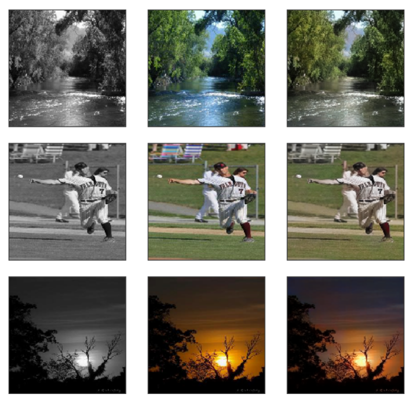
1. Can be used for colorization of Black and White old flims
2. Can be used to reinstate old documents
3. Can be used for colorization of survellience images, MRI images and satellite images.

Chapter 9

CONCLUSIONS AND FUTURE WORK

9.1 Conclusion

Grayscale Images are successfully colorized by the two approaches and accuracy comparisons per class is evaulated in accordance with the number of datasets used for training, epochs and batch size.



9.2 Future work

1. Use different color spaces.

2. Train the model for images of variable size/type.

3. Reduce the size of quantized bins.

ANNEXURE A

1.

ANNEXURE B

1.

ANNEXURE C: PLAGIARISM

Bibliography

[1] An, Jiancheng, Koffi Gagnon Kpeyiton, and Qingnan Shi."Grayscale images colorization with convolutional neural networks." Soft Computing 24.7 (2020): 4751-4758.

[2] Kiani, Leila, Masoud Saeed, and Hossein Nezamabadi-pour. "Image Colorization Using Generative Adversarial Networks and Transfer Learning." 2020 International Conference on Machine Vision and Image Processing (MVIP). IEEE, 2020.

[3] Sindhuja Kotala; Srividya Tirumalasetti; Vudaru Nemitha; Swapna Munigala."Automatic Colorization of Black and White Images using Deep Learning"

[4] Górriz, Marc, et al. "End-to-End Conditional GAN-based Architectures for Image Colourisation." arXiv preprint arXiv:1908.09873 (2019).

[5] Dabas, Chetna, et al. "Implementation of image colorization with convolutional neural network." International Journal of System Assurance Engineering and Management (2020): 1-10.

[6] Zhang, Richard, Phillip Isola, and Alexei A. Efros. "Colorful image colorization." European conference on computer vision. Springer, Cham, 2016.

[7] Hwang, Jeff, and You Zhou. "Image colorization with deep convolutional neural networks." Stanford University, Tech. Rep.. 2016.

[8] Fu, Qiwen, Wei-Ting Hsu, and Mu-Heng Yang. "Colorization Using ConvNet and GAN." Stanford University. 2017. 1-8.