SUPERVISED COLORIZATION OF GRAYSCALE IMAGES USING DEEP NEURAL NETWORKS

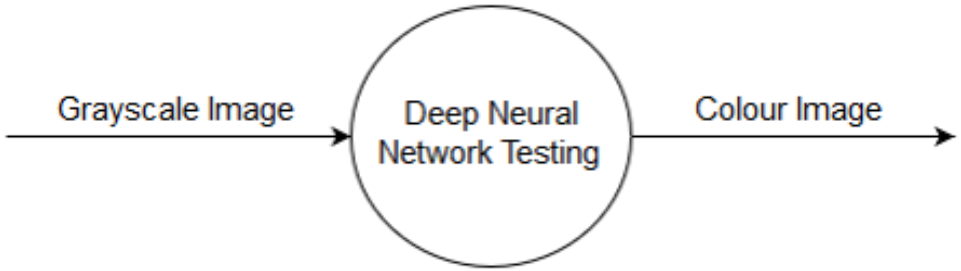
TEAM – 2
GUIDED BY
JUMANA NAHAS

INTRODUCTION

The aim of the project is the supervised colorization of grayscale images in RGB colorspace using deep convolutional neural networks. The input to the project is a grayscale image in RGB colorspace and the output is the colorized version of the same image in the RGB colorspace

- **Input** : Grayscale image in RGB colour space.

- **Output** : Colorized image in RGB colour space



PROBLEM STATEMENT

The proposed system converts the input which is given as grayscale image to a more realistic colourful image without user intervention. The existing system need constant user guidance to identify the true colour of objects in image, which is time consuming and staggered. The technology has advanced significantly, but the existing system fails to implement it. Thus it leads to the wastage of resources.

PROPOSED SOLUTION

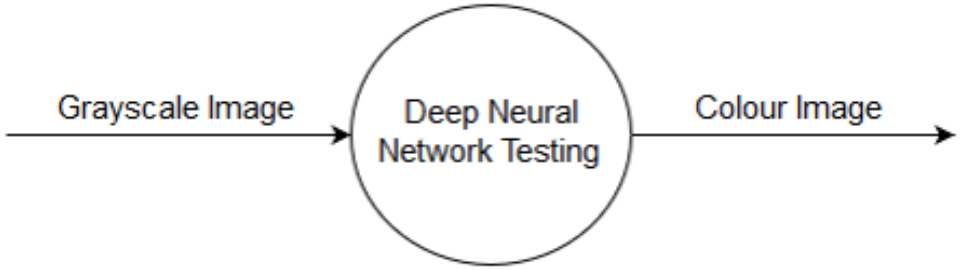
A deep convolutional neural network is used to colorize the grayscale image by employing an extremely large reference database. A chrominance refinement step is used for increasing the quality of the predicted image. The model is trained using the machine learning library Tensorflow.

COLORIZATION

Luminance (Grayscale image) + **Chrominance (Predicted Colour Values)** = **Colorized Image**

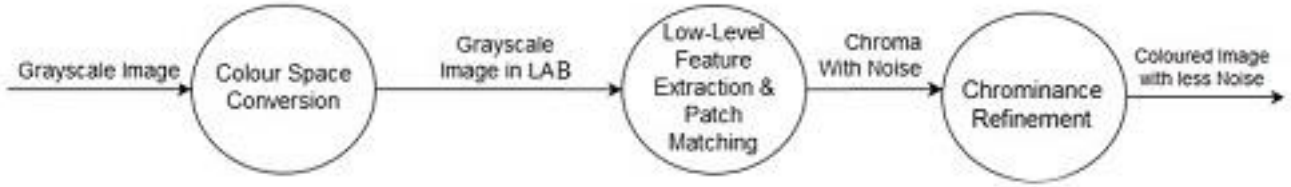
DFD

- DFD 0

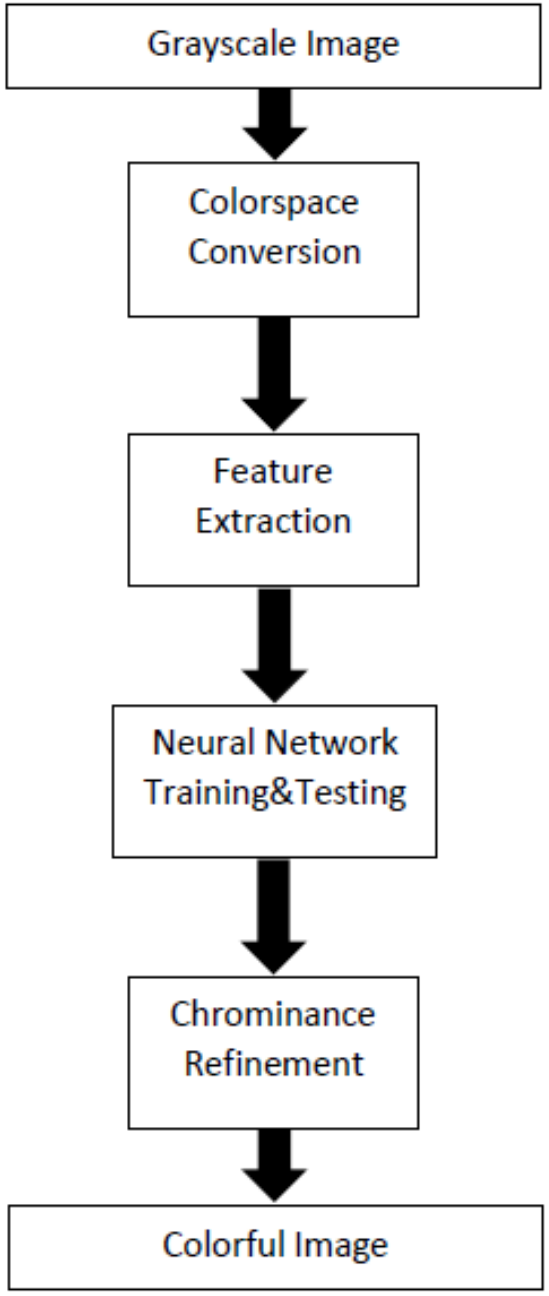


- DFD 1

- DFD 2



SYSTEM ARCHITECTURE



MODULES

The project is divided into 4 modules:

1. Conversion of grayscale image from RGB colorspace to LAB colorspace
2. Feature Extraction
3. Neural network model training & testing
4. Chrominance Refinement

Module 1 :

Conversion of Grayscale image from RGB colorspace to LAB colorspace

- **Input** : Grayscale image in RGB colorspace
- **Output** : Grayscale image in LAB colorspace

Module 2 :

Feature Extraction

- **Input** : Grayscale image in LAB colorspace
- **Output** : SIFT features extracted from the grayscale image

Module 3 :

Neural Network Model Training & Testing

- **Input :** Extracted SIFT features
- **Output :** Colorized image in RGB colorspace

Module 4 :
Chrominance Refinement

- **Input :** Colorized image in RGB colorspace
- **Output :** Colorized image with refined chrominance in RGB colorspace

Module 1

RGB to LAB Conversion Algorithm

1. Convert RGB to XYZ colorspace.
2. Set the white point, that is a set of tristimulus values or chromaticity coordinates.
3. With the value of white point, convert XYZ to LAB colorspace.

SIFT Algorithm

1. **Constructing a scale space** This is the initial preparation. Internal representations of the original image is created to ensure scale invariance. This is done by generating a "scale space".

2. **LoG Approximation** The Laplacian of Gaussian is great for finding interesting points (or key points) in an image. But it's computationally expensive. So we cheat and approximate it using the representation created earlier.

3. **Finding key points** With the super fast approximation, we now try to find key points. These are maxima and minima in the Difference of Gaussian image we calculate in step 2.

4. **Get rid of bad key points** Edges and low contrast regions are bad key points. Eliminating these makes the algorithm efficient and robust.

5. **Assigning an orientation to the key points** An orientation is calculated for each key point. Any further calculations are done relative to this orientation. This effectively cancels out the effect of orientation, making it rotation invariant.

6. **Generate SIFT features** Finally, with scale and rotation invariance in place, one more representation is generated. This helps uniquely identify features.

Module 3

Pickling

It is the process by which python objects are saved to disk from memory. And unpickling is the process of loading python objects to memory from disk.

For a given image in training data, the following properties are pickled :

- SIFT feature
- Luminance
- A channel
- B channel

Module 3

Training Algorithm

- Using pickled data, two separate models are trained.
- One model to predict the Channel A luminance, and the other to predict the Channel B luminance.
- The trained models are written to disk separately.
- Training step is done.

Testing Algorithm

- A grayscale image in RGB colorspace is given as test input.

- For predicting the channel A:
  - Extract and load the SIFT features and luminance to memory
  - Load the neural network model for channel A
  - Use this model to predict the channel A of the given input.
  - Pickle the predicted output channel A

- For predicting the channel B:
  - Extract and load the SIFT features and luminance to memory
  - Load the neural network model for channel B
  - Use this model to predict the channel B of the given input.
  - Pickle the predicted output channel B

Contd.

Testing Algorithm (Contd.)

- Extract the luminance of the given image.
  - Load the predicted channel A, and predicted channel B of the given image from disk.
  - Fuse the luminance, channel A, and channel B, to obtain the colorized image.
  - Convert the image from LAB colorspace to RGB colorspace.
  - Write the image in RGB colorspace to disk.

Neural Network Model

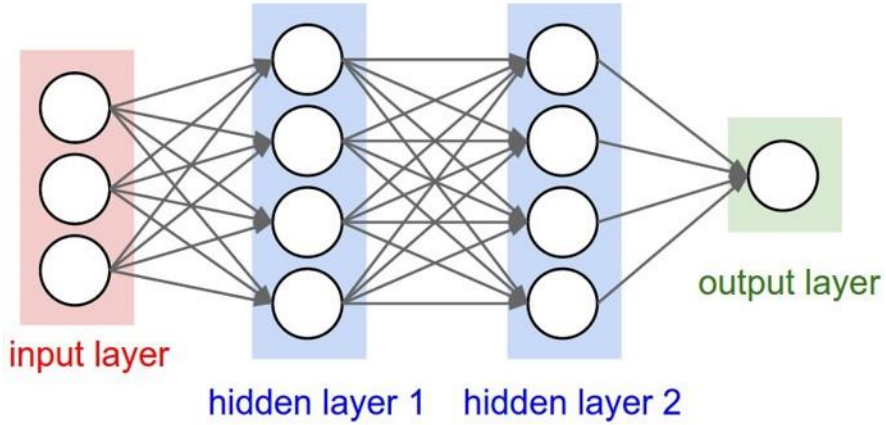
Model

- The model consists of 10 layers.
- Each layer 128 neurons

- All the layers are fully connected

  In a fully connected layer each neuron is connected to every neuron in the previous layer, and each connection has it's own weight.

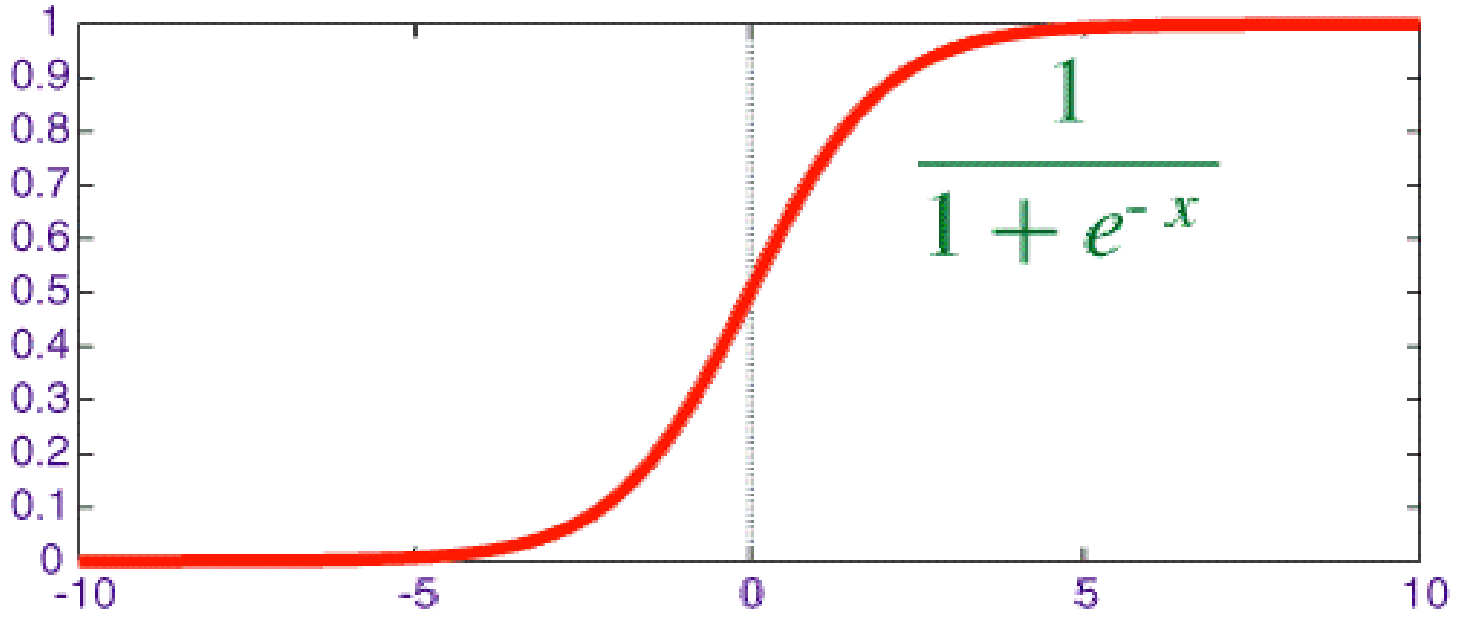
- The activation function used is sigmoid



Sigmoid function :

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

It's output boundaries are (0, 1)



Chrominance Refinement

Algorithm

- Luminance value L=l

- Predicted A channel value = x
- Predicted B channel Value = y

- A channel value after refinement = X
- B channel value after refinement = Y

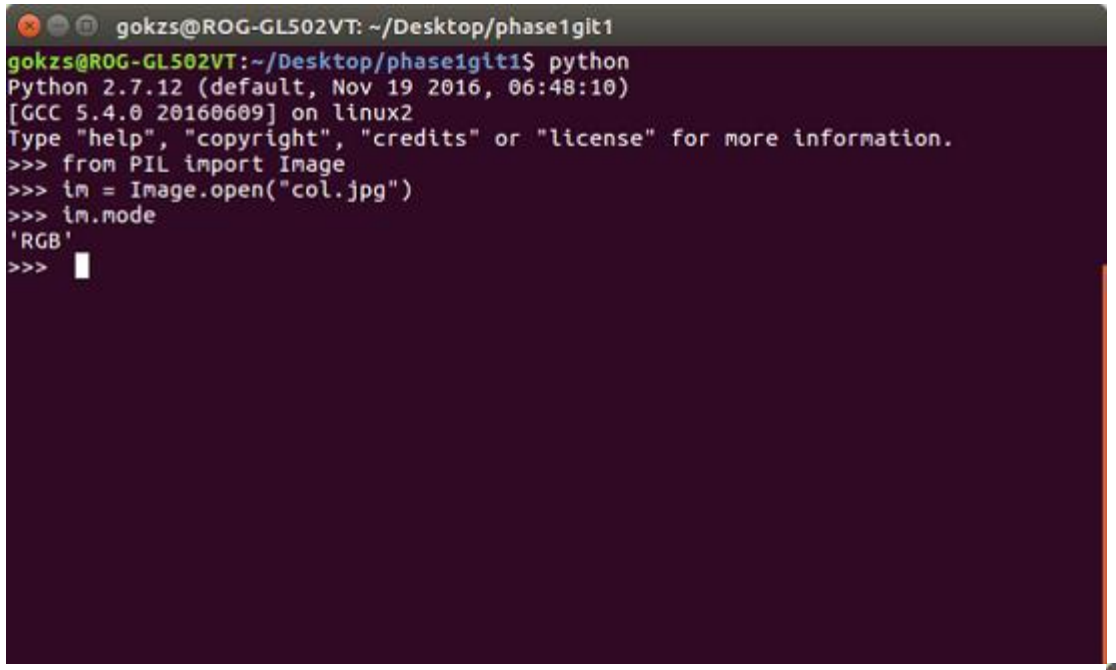
- Ratio A to B Must remain same
  x/y=X/Y

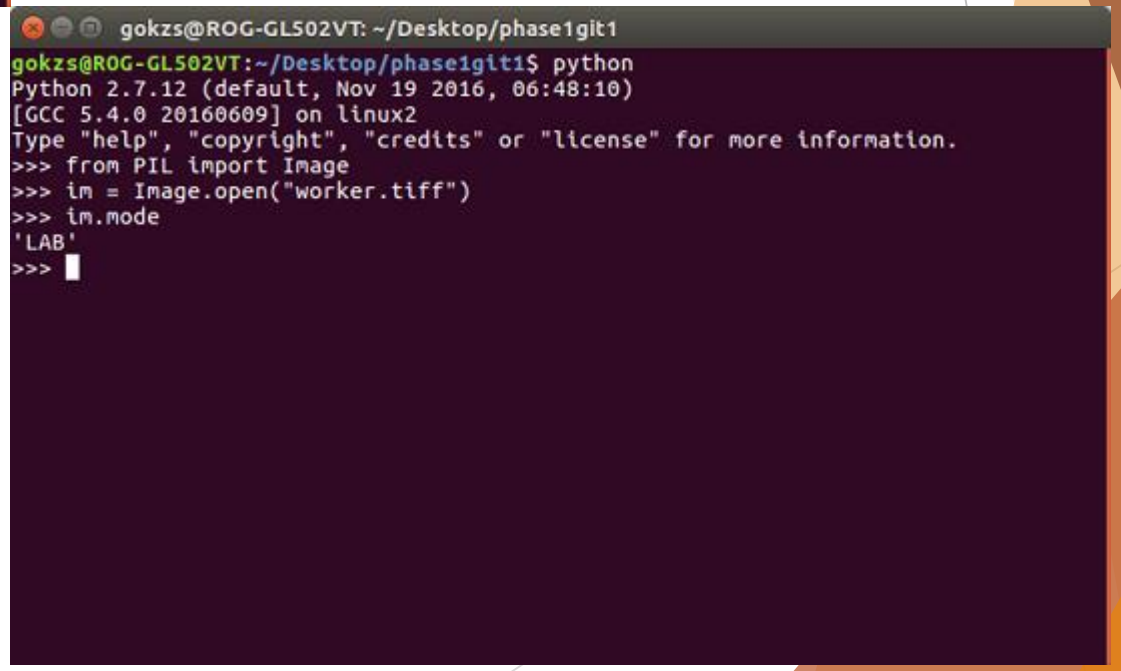
- Intensity of each pixel in grayscale image = Intensity of corresponding pixel in color image

$$X = \frac{2L}{1+\left(\frac{y}{x}\right)} \qquad Y = \frac{2L}{1-\left(\frac{x}{y}\right)}$$

OUTPUTS

Module 1


gokzs@ROG-GL502VT: ~/Desktop/phase1git1
gokzs@ROG-GL502VT:~/Desktop/phase1git1$ python
Python 2.7.12 (default, Nov 19 2016, 06:48:10)
[GCC 5.4.0 20160609] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> from PIL import Image
>>> im = Image.open("col.jpg")
>>> im.mode
'RGB'
>>>


gokzs@ROG-GL502VT: ~/Desktop/phase1git1
gokzs@ROG-GL502VT:~/Desktop/phase1git1$ python
Python 2.7.12 (default, Nov 19 2016, 06:48:10)
[GCC 5.4.0 20160609] on linux2
Type "help", "copyright", "credits" or "license" for more information.
>>> from PIL import Image
>>> im = Image.open("worker.tiff")
>>> im.mode
'LAB'
>>>

Output Image in LAB Colour space

Module 2

Output Image with SIFT Features

Overall Output

- Black & White Image

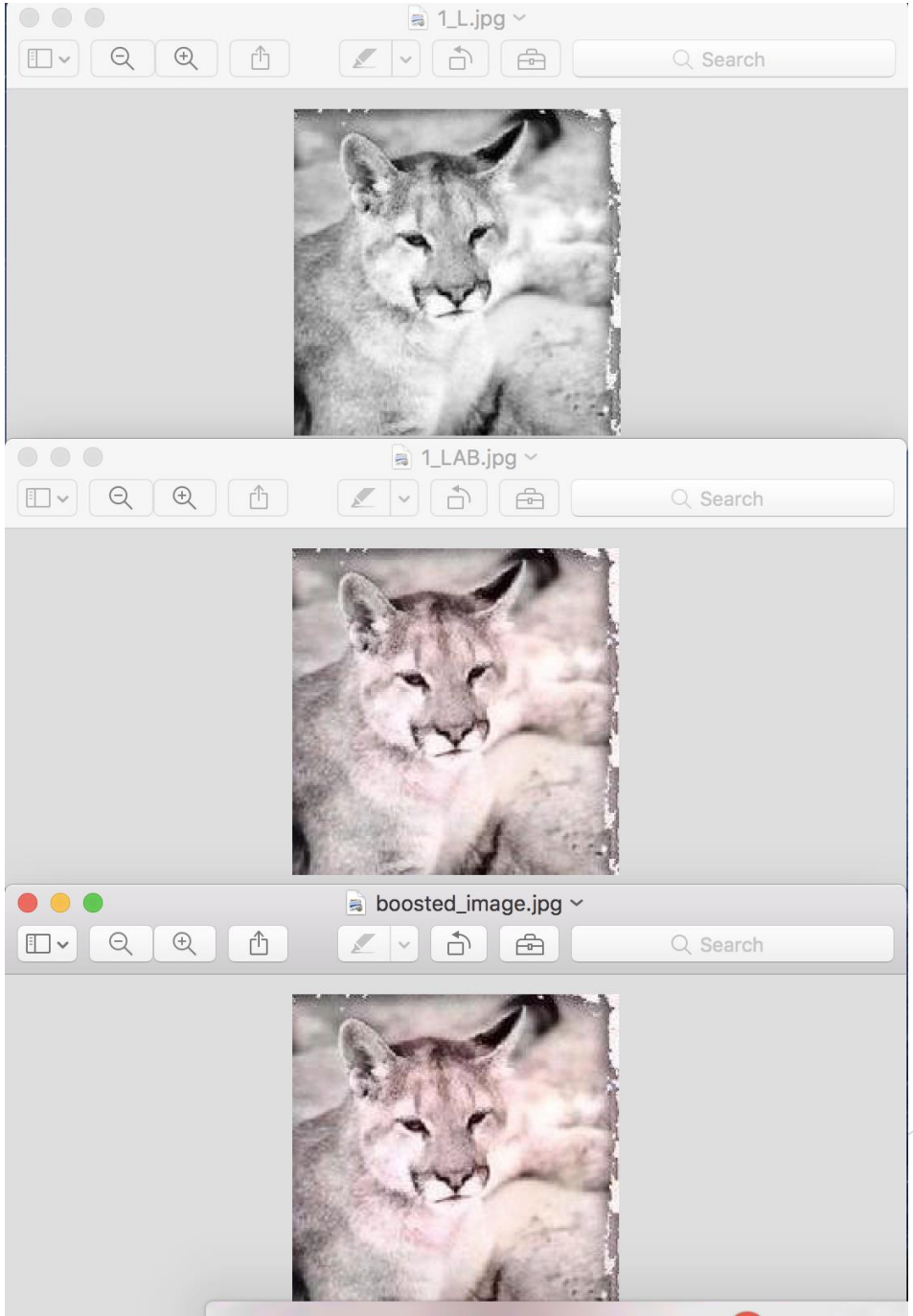
- Ground Truth Image



- Predicted colour image

Image after chrominance refinement

- Grayscale image
- Colorized image
- Colorized image after chrominance refinement



Thank you