Performance Analysis of Deep Learning Models for Image Colorization

Group 12

Team Members

Amrita Varshini E R
AM.EN.U4AIE19010

Ann Maria John
AM.EN.U4AIE19013

Arunima Divya
AM.EN.U4AIE19016

Devi Parvathy Nair
AM.EN.U4AIE19026

Namitha S
AM.EN.U4AIE19042

Table of Contents

01 Introduction

Abstract

Use various deep learning models to compare and analyse the colorization excellence

Motivation

Can derive more information from color images than grayscale images for application on historical images, b&w movies, etc.

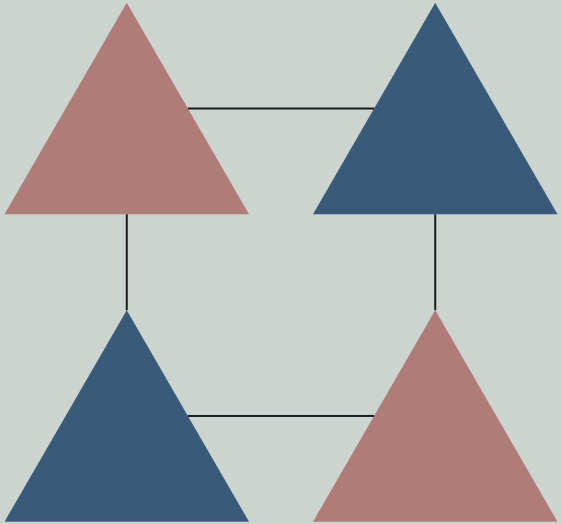
Problem Statement

Predict the color components of an image given the luminance component and combine both to get the color scale image

Why Image Colorization?

Improve visual appeal of images

Add relevant information to images



Color accuracy, finer details

Make images more understandable

02 Literature Review

Literature Review

01 | **Deep Koalarization: Image Colorization using CNNs and Inception-Resnet-v2**
Amrita Varshini E R

02 | **Grayscale Image Colorization using a Convolutional Neural Network**
Ann Maria John

03 | **Image Colorization with Deep Convolutional Neural Networks**
Arunima Divya

04 | **Image Colorization with Convolutional Neural Networks | IEEE Conference Publication**
Devi Parvathy Nair

05 | **Colorful Image Colorization**
Namitha S

Literature Review

Deep Koalarization: Image Colorization using CNNs and Inception-Resnet-v2	Grayscale Image Colorization using a Convolutional Neural Network
• Architecture combines features of pretrained Inception model and their Deep CNN model	• Backbone of the architecture: Modified FusionNet model
• Identify a mapping that connects the luminance version of the image	• 3 modules: Feature Extraction, Bridge, Reconstruction
• Inception model performs high level feature extraction	• Feature Extraction: 4 encoding blocks; each block with four convolution layer + residual block
• 4 modules: Encoder, Feature extractor, Fusion and Decoder	• Reconstruction: 4 decoding blocks with similar structure of encoding block

Literature Review

Image Colorization with Deep Convolutional Neural Network	Colorful Image Colorization
• Architecture uses a regression based baseline model combining features of pretrained VGG16 model.	• Suggest a completely automated method for creating vivid and realistic colorizations
• 2 modules: Summarizing Encoding Part and Creating Decoding Part	• Employ class-rebalancing during training boosting the diversity of colors in the output
• Uses ReLU activation function Batch Normalization for faster convergence	• Create objective function taking into account the multimodal uncertainty of the colorization problem capturing a wide range of hues

Literature Review

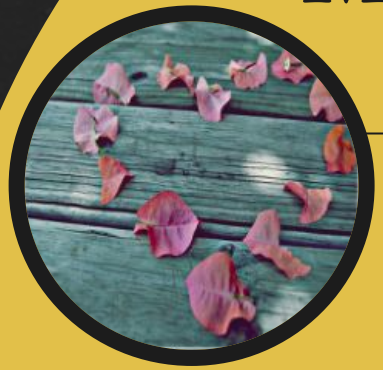
Image Colorization with Convolutional Neural Networks

- Approach based on deep neural networks to color the image in grayscale
- Several combinations of neural networks and loss functions compared
- VGG-16 CNN model based with cross entropy loss compared with learning-based methods

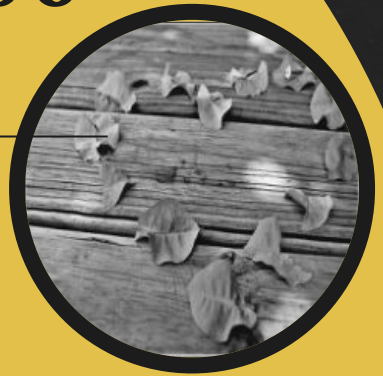
03 Dataset

DATASET

MIRFLICKR-25000



25K Images Dataset - .npy files



CIELAB Colorspace

Green-red (*a)

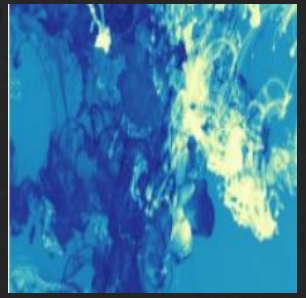
Values ranging from -128 (green) to 127 (red)

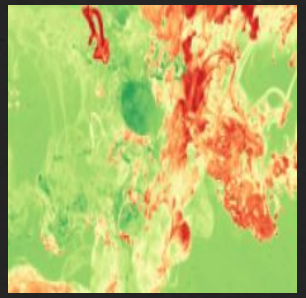
Lightness (L*)

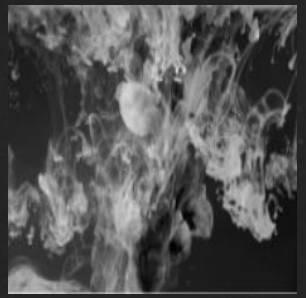
Lightness on a scale from 0 (black) to 100 (white)

Blue-yellow

Values ranging from -128 (blue) to 127 (yellow)

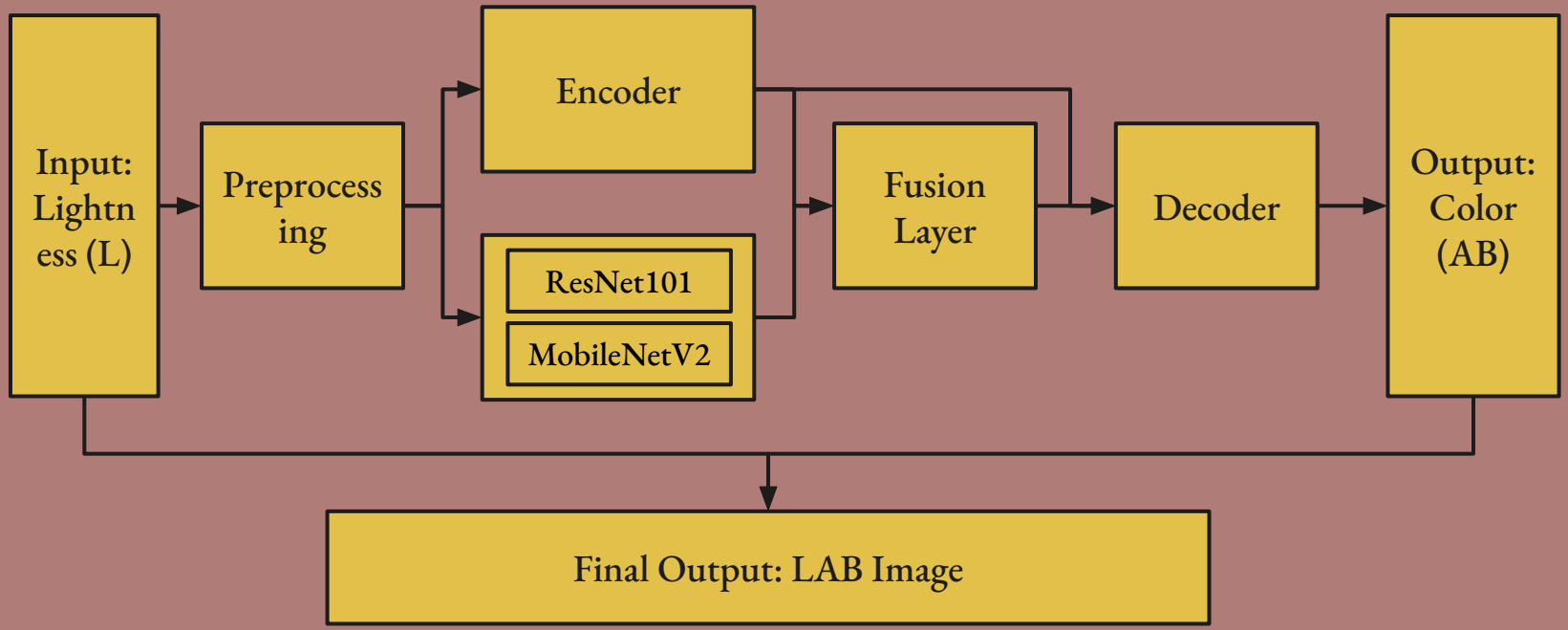






04 Methodology

Basic Architecture



Deep Learning Models

MobileNetV2 with Custom CNN

- A combination of MobileNetV2 with a Convolutional encoder and decoder
- MSE with a tanh activation

Resnet with Custom CNN

- A combination of ResNet with a Convolutional encoder and decoder
- MSE with a tanh activation

Pre-trained models (Baseline)

ResNet 101

- Pre-trained on ImageNet dataset
- 152 layers, but lesser complexity
- Learning residual functions with reference to layer inputs

MobileNet-V2

- Pre-trained on ImageNet dataset
- Key features: inverted residual blocks with bottleneck features
- 53 layers deep

Modules

01

Preprocessing

- Converting input L images to 3D vectors to match input size of models
- Normalisation of L and *a*b images

02

Encoder

- 5 Conv2D layers
- Initially LeakyReLU and later ReLU activations
- Input: 224x224
- Output: 7x7

03

Fusion

- Connects ResNet101/MobileNetV2 with encoder
- 1 Conv2D layer with ReLU activation
- A skip connection from encoder

04

Decoder

- Uses skip connections from encoder with added dropouts
- Conv2DTranspose layers
- ReLU and a final tanH activation

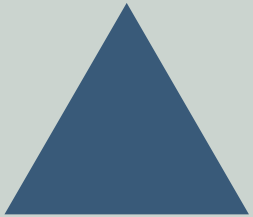
05 Results

Results

CNN Models	Custom CNN with MobileNetV2	Custom CNN with ResNet-101
Training Accuracy	0.7278	0.7843
Validation Loss	0.0168	0.0163
Validation Accuracy	0.6080	0.6103

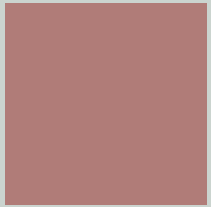
Research Challenges/Open Issues

Images often have brownish shade



Models lack historical knowledge when selecting colors and applying color theory

Inaccurate colorization for rare images/objects



Uses too much RAM during training and evaluation

Thank you!