ASU
ARIZONA STATE UNIVERSITY

Automatic Image Colorization

Ahmed Nasrallah, Touseef Ali, Hitesh Kumar

Team #8

EEE 511 Final Project

Outline

- Introduction
  - Motivation
  - Problem Definition
- Background
  - Color spaces
    - From RGB to Lab
  - Methods
    - Parametric and Non-Parametric
- Model Architecture
- Evaluation
  - Results, Performance, and Discussion
- Limitations and Challenges
- Conclusions and Recommendations

Introduction

Challenge

Specific objects can take on many different valid and “good” colors

Luminance, L

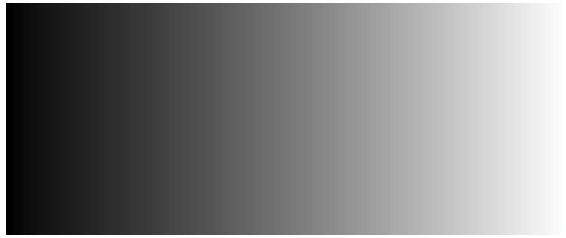
?

+

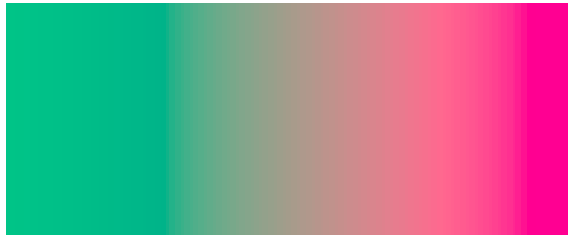
GB)

“L-ab” Color space

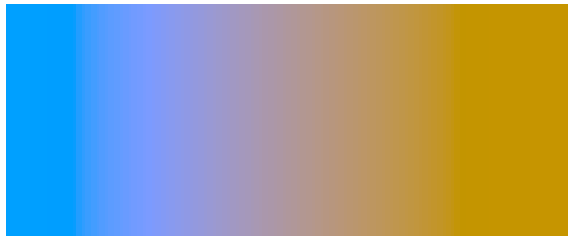
L channel

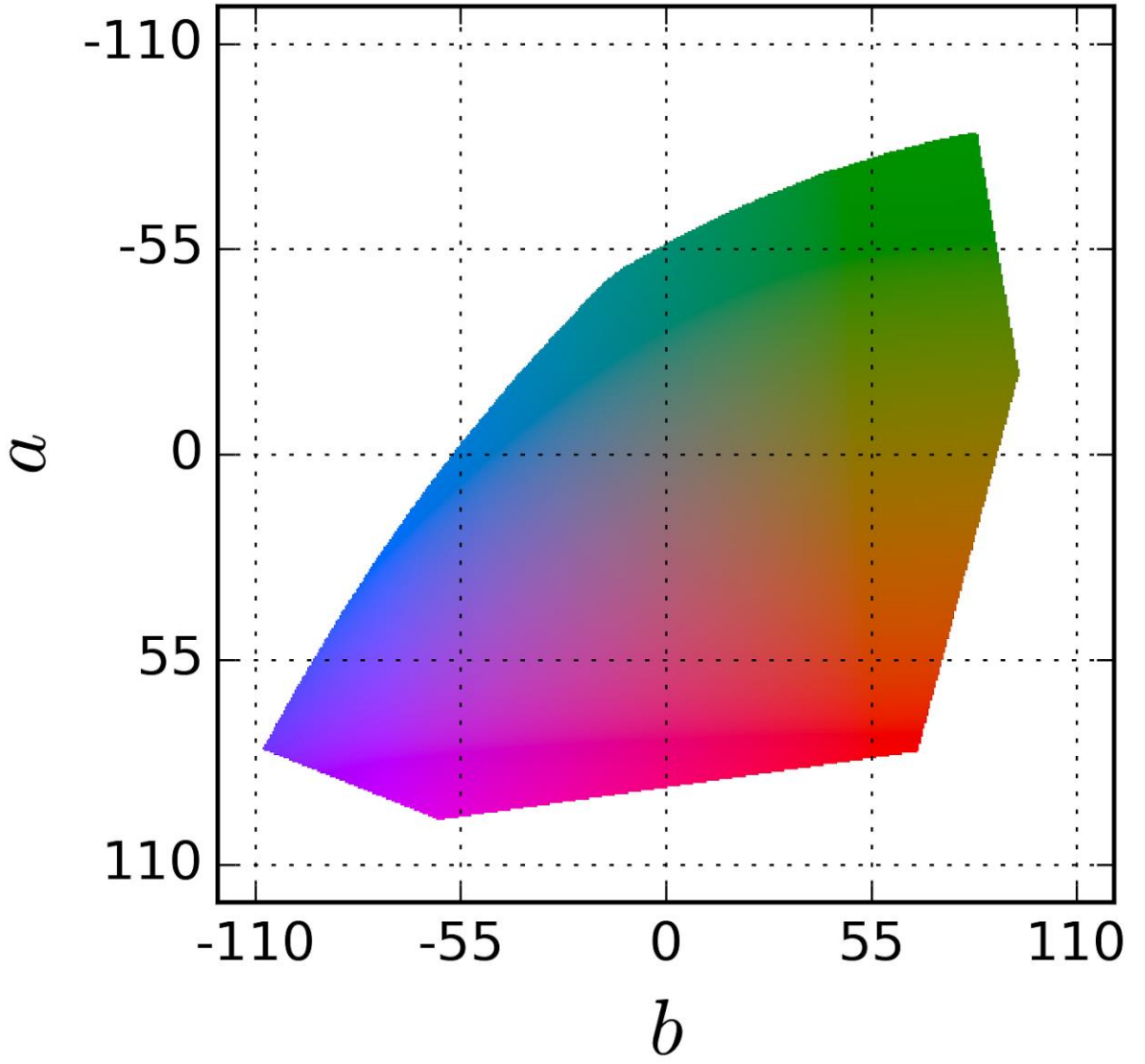


a channel

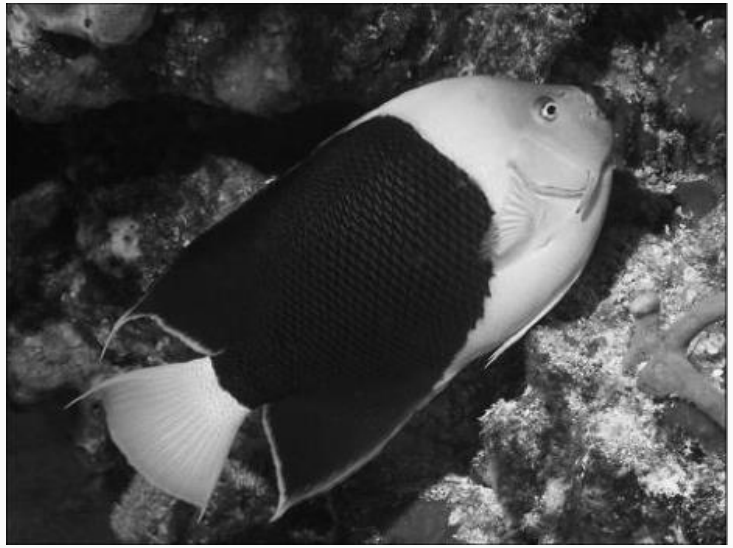


b channel





Problem Description



$\mathcal{F}$

Grayscale image: L channel

$$\mathbf{X} \in \mathbb{R}^{H \times W \times 1}$$

Color information: ab channels

$$\hat{\mathbf{Y}} \in \mathbb{R}^{H \times W \times 2}$$

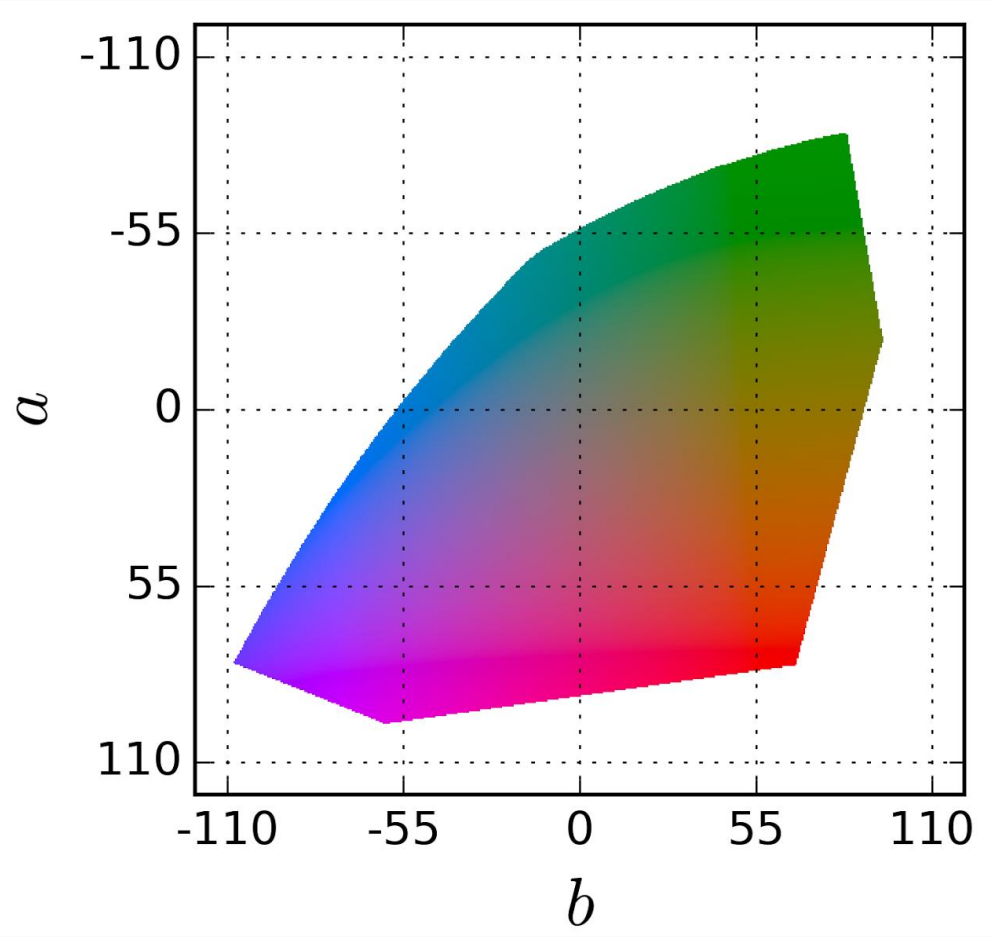
Loss Function

- Learning a mapping function $\mathcal{F}$ which predicts $\hat{\mathbf{Y}}$

- Loss Function:

$$\mathrm{L}_2(\hat{\mathbf{Y}}, \mathbf{Y}) = \frac{1}{2} \sum_{h,w} \|\mathbf{Y}_{h,w} - \hat{\mathbf{Y}}_{h,w}\|_2^2$$

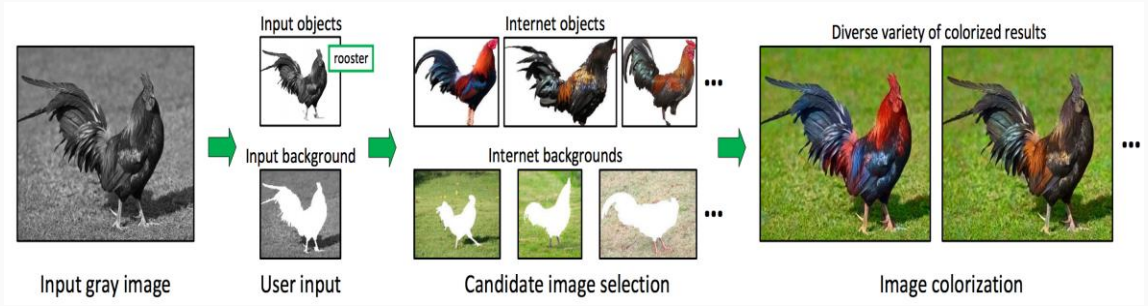
$$\hat{\mathbf{Y}} \in \mathbb{R}^{H \times W \times 2}$$



Methods

Non-parametric

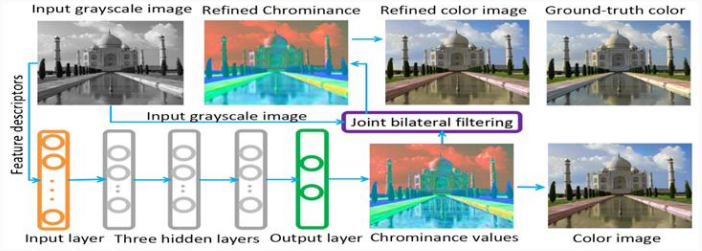
Hertzmann et al. In SIGGRAPH, 2001.
Welsh et al. In TOG, 2002.
Irony et al. In Eurographics, 2005.
Liu et al. In TOG, 2008.
Chia et al. In ACM 2011.
Gupta et al. In ACM, 2012.



Parametric

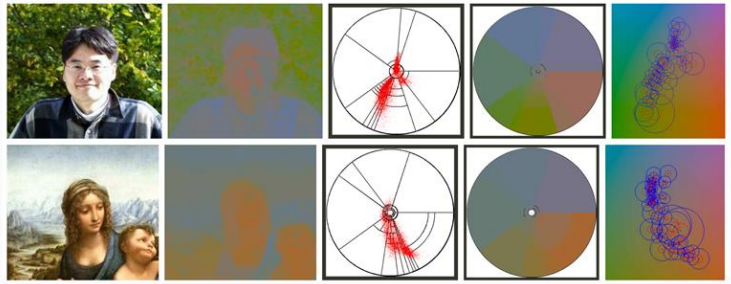
Hand-engineered Features

L2 Regression



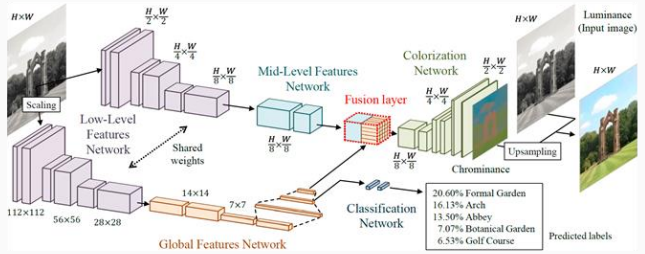
Deshpande et al. Cheng et al. In ICCV 2015.

Classification

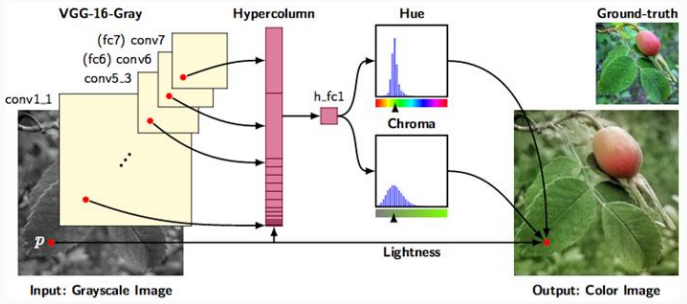


Charpiat et al. In ECCV 2008.

Deep Networks



Dahl. Jan 2016. Iizuka et al. In SIGGRAPH, 2016.



Larsson et al. In ECCV 2016.

Framework/Model Summary

Data Preprocessing
lab = rgb2lab(image)
L = lab[:,:,0]

→

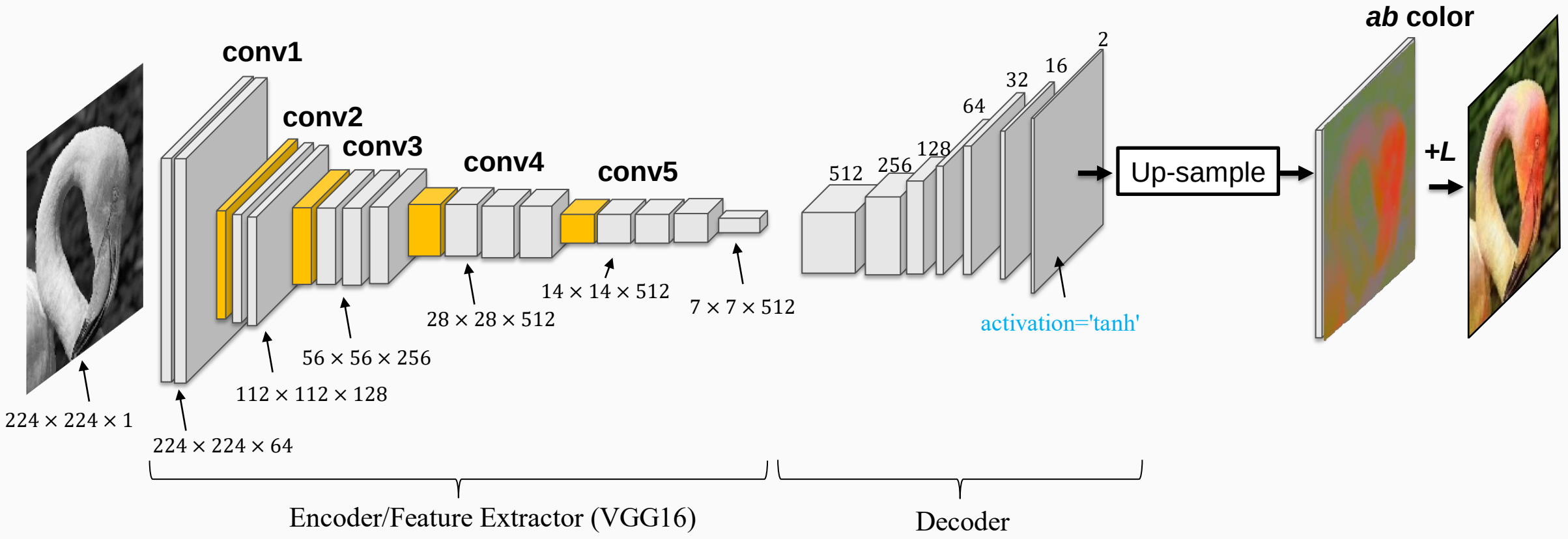
Feature Extraction
keras.applications.vgg16.VGG16()
vggpred = newmodel.predict(L)

→

Selecting *ab* based on Feature learned
tf.keras.layers.Conv2D
ab = model.predict(vggpred)

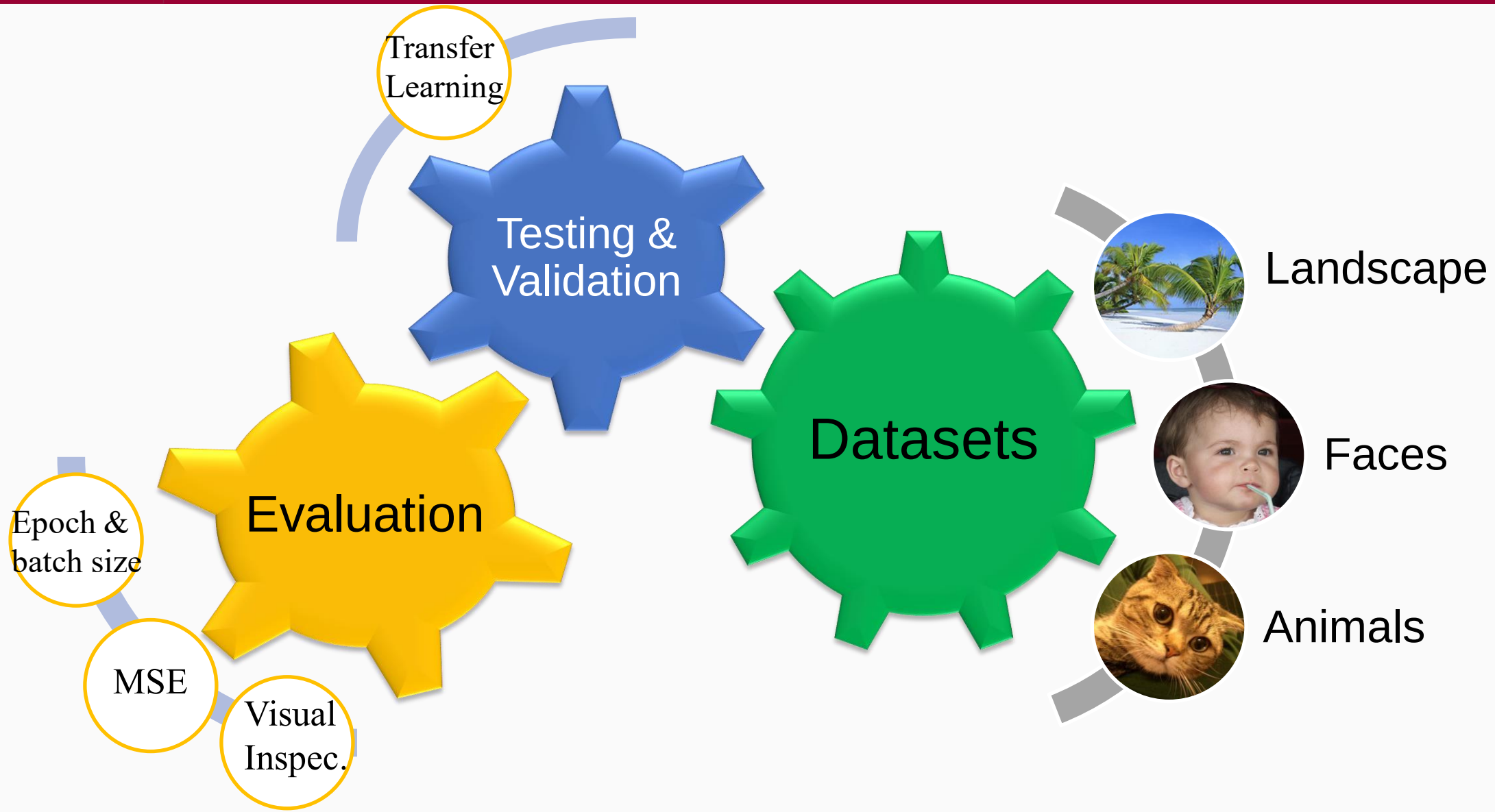
→

Up-scaling and combining all channels
temp = np.zeros((224, 224, 3))
temp[:,:,0] = L
temp[:,:,1:] = ab
out= lab2rgb(temp)



https://blog.floydhub.com/colorizing-b-w-photos-with-neural-networks/
https://medium.com/@mahmoudeljiddawi/auto-colorization-of-black-and-white-images-using-machine-learning-auto-encoders-technique-a213b47f7339

Evaluation


Transfer Learning
Testing & Validation
Datasets
Landscape
Faces
Animals
Evaluation
Epoch & batch size
MSE
Visual Inspec.

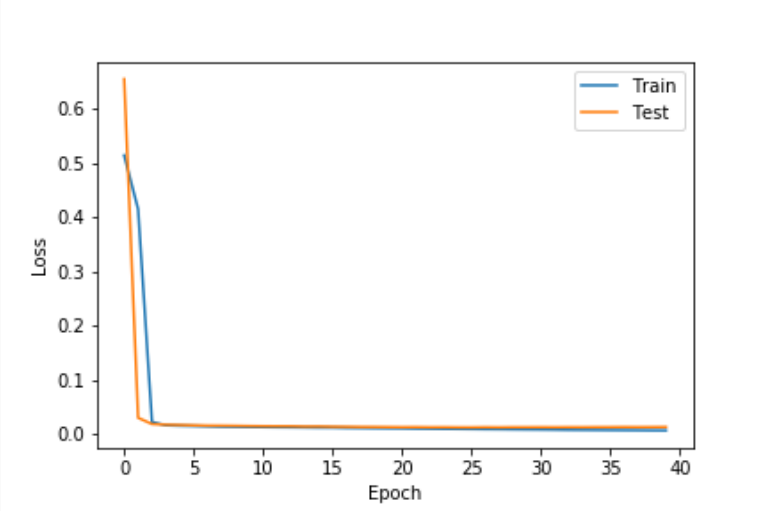
Results and Discussion






Grayscale | Ground Truth | Output Image

- Image Quality reduction
  - Down sampling
  - Dimensionality reduction
- Training set of **Landscape** images (924 images)

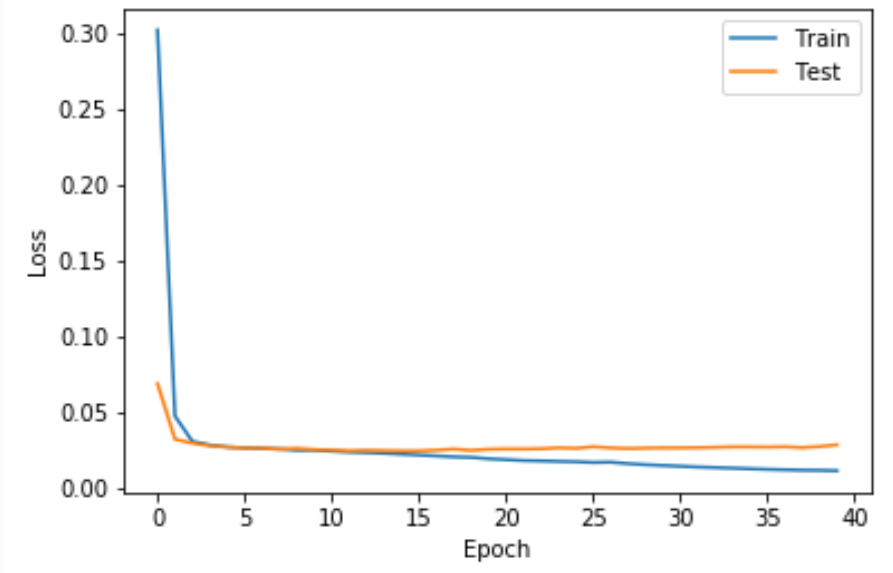


MSE loss during each epoch with 10% validation test set

Results and Discussion Cont.

Input Ground Truth Output Image

- Image Quality reduction
  - Down sampling
  - Dimensionality reduction
- Training set of **Animal** images (539 images)

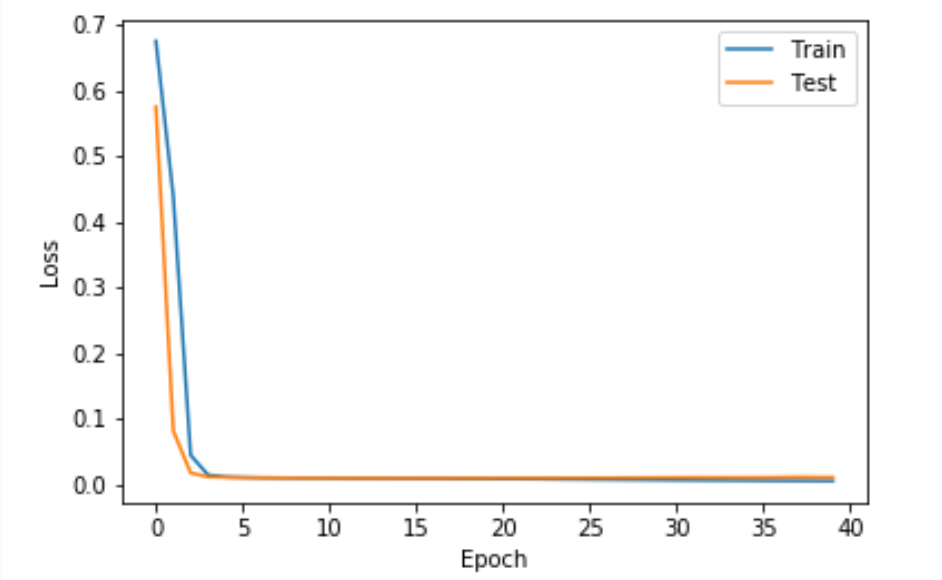


MSE loss during each epoch with 10% validation test set

Results and Discussion Cont.

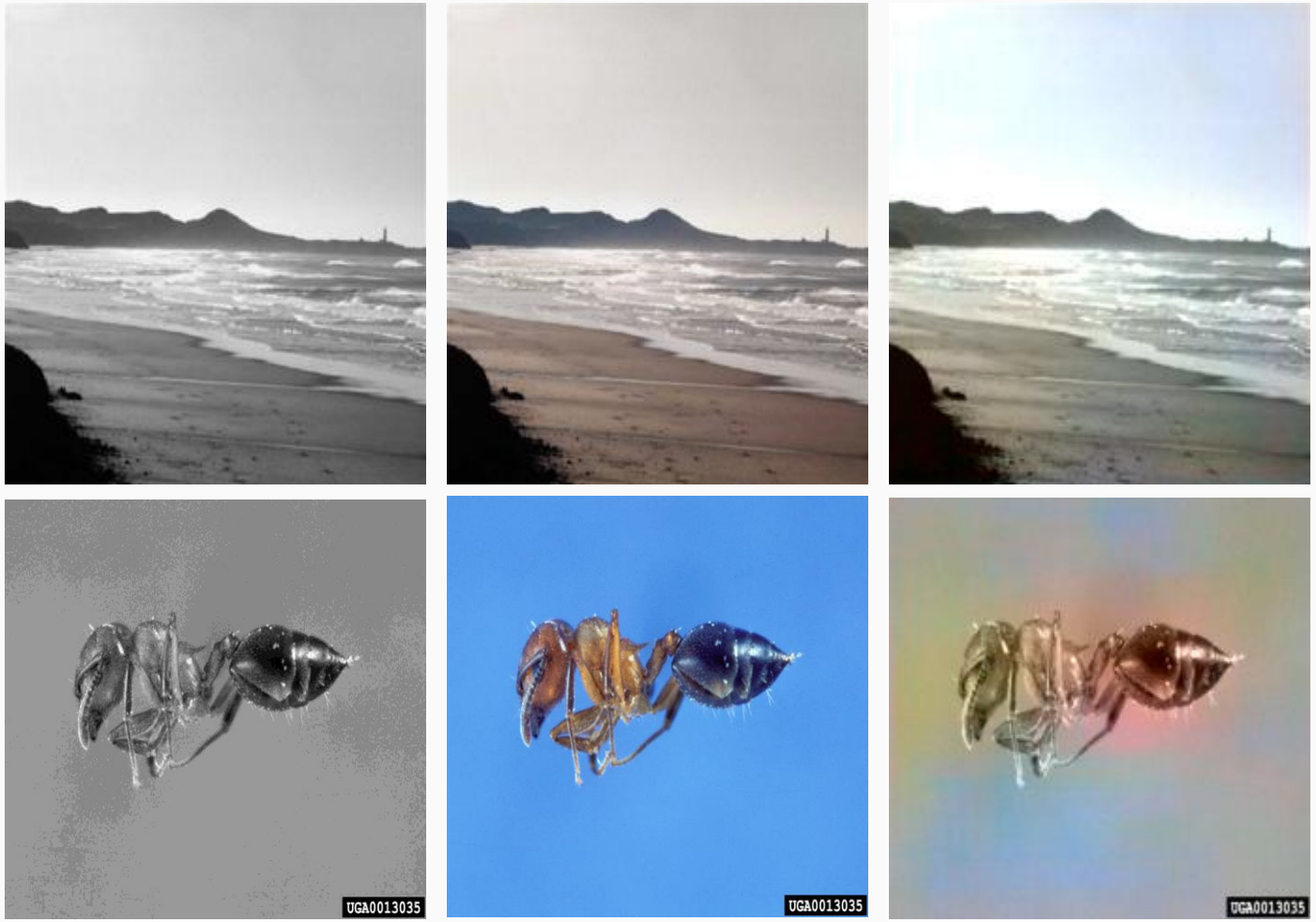
Input | Ground Truth | Output Image

- Image Quality reduction
  - Down sampling
  - Dimensionality reduction
- Training set of **Faces** images (996 images)



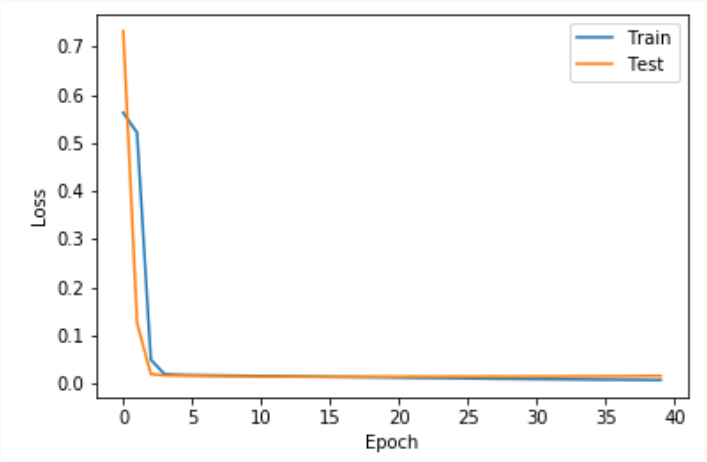
MSE loss during each epoch with 10% validation test set

Results and Discussion Cont.

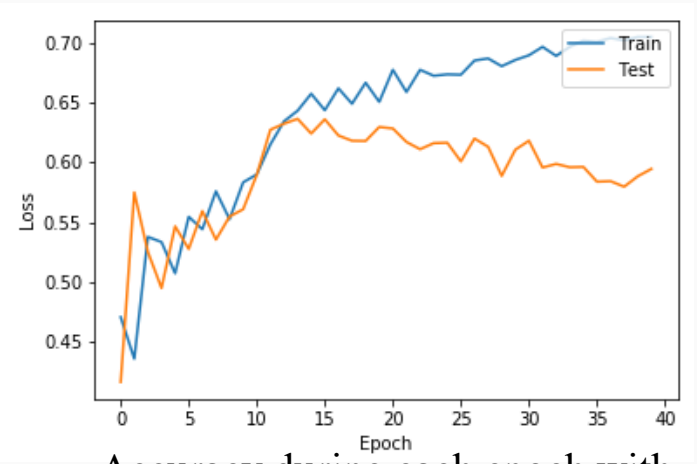


Input Ground Truth Output Image

- Training set of **All** images (2500 images)



MSE loss during each epoch with 10% validation test set



Accuracy during each epoch with 10% validation test set

Limitations and Challenges

- The main limitation is the test set given
  - Larger and more diverse set → better feature extractor →better results in determining *ab* spaces
  - Model can learn “too” well (overfit) data set
    - Skin tone foreground color on background objects (e.g., sky, blue background)
- VGG16 pretrained on RGB images
- Two color channel (Lab) estimation lowers performance overhead but reduces vibrancy of output
- Longer training time does not necessary corresponds to better results
  - Accuracy drops under validation set
- Quality and performance
  - Inverse relationship trade-off

Conclusions and Recommendations

- Image colorization using CNN with auto encoder
- Trained using Keras with TensorFlow backend
- Different test cases and conditions are evaluated
  - Animal
  - Faces
  - Landscapes
- Evaluation based on subjective user inspection and MSE and accuracy quantification
- Only maximum of 2500 images were used
  - Typically, need around 50K images for reasonable colorization
  - Model underperformed in some images given as input
  - More testing and training
  - Using custom encoder with training instead of VGG16 feature extractor

Thank you

Questions and Suggestions!