
WATCH
YOUR STEP


WATCH
YOUR STEP








Image Colorization

Presented by

BA865 Group 6

Crystal Leatvanich, Jeonghee (Christina) Son,
Kuang-Ching (Amanda) Ting, Tharfeed Ahmed Unus

Project Overview

Background

- Black and white images are still widely used in various areas.
- Color provides richer visual context, enhancing understanding and engagement.

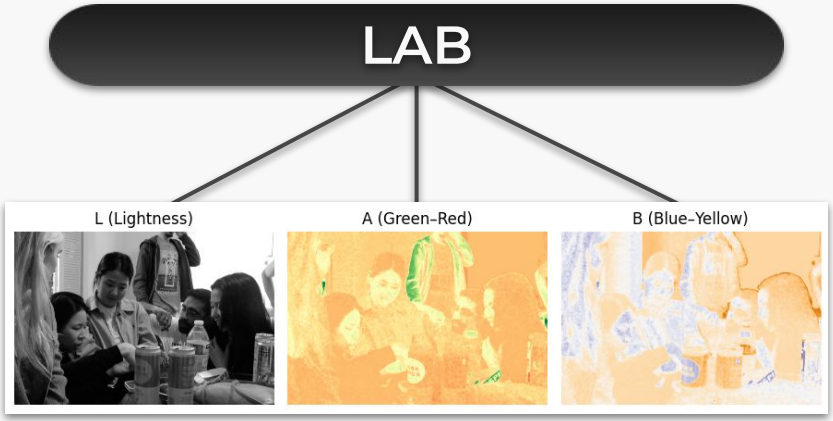
Motivation

Museums and
Historical Archives

Media and
Film Industry

Education
and Medical Field

Color Spaces



Initial Dataset

Full-Scene Portraits

265 Files, 505 after augmentation

Data Preprocessing

Neural Networks

- Model 1: Sequential RGB
- Model 2: Simple CNN
- Model 3: Improved CNN
- Model 4: U-Net Implementation
- Model 5: Sequential LAB
- Model 6: Transformer with VGG

Model Evaluation

Loss Functions: MAE/MSE

Metrics: MSE/MAE

Final Evaluation: Manual Verification

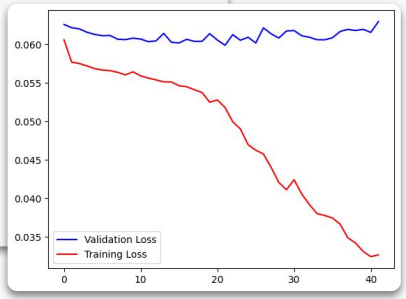
U-Net Implementation

```python
def build_unet_model(input_shape=(120, 176, 1)):
    inputs = layers.Input(shape=input_shape)

    # Encoder
    conv1 = layers.Conv2D(64, (3, 3), padding='same')(inputs)
    conv1 = layers.BatchNormalization()(conv1)
    conv1 = layers.ReLU()(conv1)
    conv1 = layers.Conv2D(64, (3, 3), padding='same')(conv1)
    conv1 = layers.BatchNormalization()(conv1)
    conv1 = layers.ReLU()(conv1)
    pool1 = layers.MaxPooling2D((2, 2))(conv1)

    conv2 = layers.Conv2D(128, (3, 3), padding='same')(pool1)
    conv2 = layers.BatchNormalization()(conv2)
    conv2 = layers.ReLU()(conv2)
    conv2 = layers.Conv2D(128, (3, 3), padding='same')(conv2)
    conv2 = layers.BatchNormalization()(conv2)
    conv2 = layers.ReLU()(conv2)
    pool2 = layers.MaxPooling2D((2, 2))(conv2)

    # Bottleneck
    bottleneck = layers.Conv2D(256, (3, 3), padding='same')(pool2)
    bottleneck = layers.BatchNormalization()(bottleneck)
    bottleneck = layers.ReLU()(bottleneck)

    # Decoder
    up2 = layers.Conv2DTranspose(128, (2, 2), strides=2, padding='same')(bottleneck)
    up2 = layers.Concatenate()([up2, conv2])
    up2 = layers.Conv2D(128, (3, 3), padding='same')(up2)
    up2 = layers.BatchNormalization()(up2)
    up2 = layers.ReLU()(up2)

    up1 = layers.Conv2DTranspose(64, (2, 2), strides=2, padding='same')(up2)
    up1 = layers.Concatenate()([up1, conv1])
    up1 = layers.Conv2D(64, (3, 3), padding='same')(up1)
    up1 = layers.BatchNormalization()(up1)
    up1 = layers.ReLU()(up1)

    outputs = layers.Conv2D(3, (3, 3), activation='sigmoid', padding='same')(up1)

    model = models.Model(inputs=inputs, outputs=outputs)
    model.compile(optimizer='adam', loss='mse', metrics=['mae'])
    return model
```

The model failed to capture true color details - output images appear as grayscale inputs with a subtle color overlay rather than accurate color restorations.

Sequential LAB

```python
#Encoder
seq_lab_model = Sequential()
seq_lab_model.add(Conv2D(64, (3, 3), activation='relu', padding='same', strides=2, input_shape=(120, 176, 1)))
seq_lab_model.add(Conv2D(128, (3, 3), activation='relu', padding='same'))
seq_lab_model.add(Conv2D(128, (3,3), activation='relu', padding='same', strides=2))
seq_lab_model.add(Conv2D(256, (3,3), activation='relu', padding='same'))
seq_lab_model.add(Conv2D(512, (3,3), activation='relu', padding='same'))
seq_lab_model.add(Conv2D(512, (3,3), activation='relu', padding='same'))
seq_lab_model.add(Conv2D(256, (3,3), activation='relu', padding='same', strides=2))
seq_lab_model.add(Conv2D(512, (3,3), activation='relu', padding='same'))
seq_lab_model.add(Conv2D(512, (3,3), activation='relu', padding='same'))
seq_lab_model.add(Conv2D(256, (3,3), activation='relu', padding='same'))
#Decoder
seq_lab_model.add(Conv2D(128, (3,3), activation='relu', padding='same'))
seq_lab_model.add(UpSampling2D((2, 2)))
seq_lab_model.add(Conv2D(64, (3,3), activation='relu', padding='same'))
seq_lab_model.add(UpSampling2D((2, 2)))
seq_lab_model.add(Conv2D(32, (3,3), activation='relu', padding='same'))
seq_lab_model.add(Conv2D(16, (3,3), activation='relu', padding='same'))
seq_lab_model.add(Conv2D(2, (3, 3), activation='tanh', padding='same'))
seq_lab_model.add(UpSampling2D((2, 2)))
seq_lab_model.compile(optimizer=Adam(learning_rate=1e-4), loss='mae', metrics=['mse'])
```



This was the first model to show promising signs of learning on the training data, but it exhibited clear signs of overfitting and failed to generalize effectively to new images.

Transformer with VGG

```python
# Input grayscale image
input_l = Input(shape=(120, 176, 1), name='grayscale_input')
x_rgb = Lambda(lambda x: tf.image.grayscale_to_rgb(x))(input_l)

# VGG encoder
vgg_base = VGG16(include_top=False, weights='imagenet', input_tensor=x_rgb)
for layer in vgg_base.layers:
    layer.trainable = False

vgg_features = vgg_base.get_layer('block3_conv3').output  # (15, 22, 256)

# Transformer block
x = transformer_encoder(vgg_features, num_heads=4, ff_dim=512)

# Decoder block
x = build_decoder(x)

# Final resizing to match input
output_rgb = Lambda(lambda x: tf.image.resize_with_crop_or_pad(x, 120, 176))(x)

# Final model
trans_vgg_model = Model(inputs=input_l, outputs=output_rgb)

# Feature extractor for perceptual loss
vgg_feat_model = VGG16(include_top=False, weights='imagenet', input_shape=(120, 176, 3))
vgg_feat_model.trainable = False
feature_extractor = Model(inputs=vgg_feat_model.input, outputs=vgg_feat_model.get_layer('block3_conv3').output)
```

This model appeared to associate objects with general color patterns, but the outputs lacked clarity and structure, making the original scene nearly impossible to interpret from the prediction alone.

Homogeneous Image Set

Full Scene Portraits

MSBA Headshots

U-Net Implementation

Grayscale Input

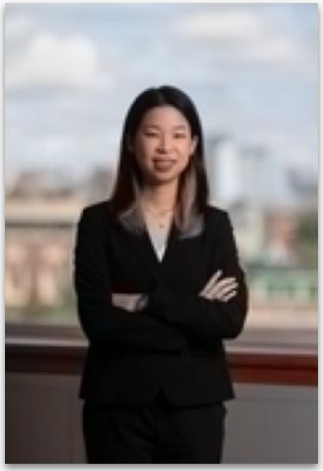
Predicted Image

Ground Truth

This model performs significantly better than earlier, aided by the simpler dataset.
However, some areas of the output (on the subject and background) still remained desaturated.

Sequential LAB Implementation

Grayscale Input

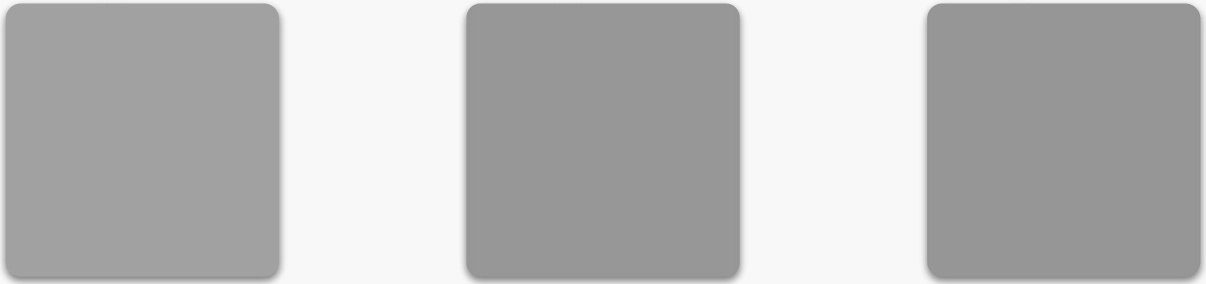


Predicted Image

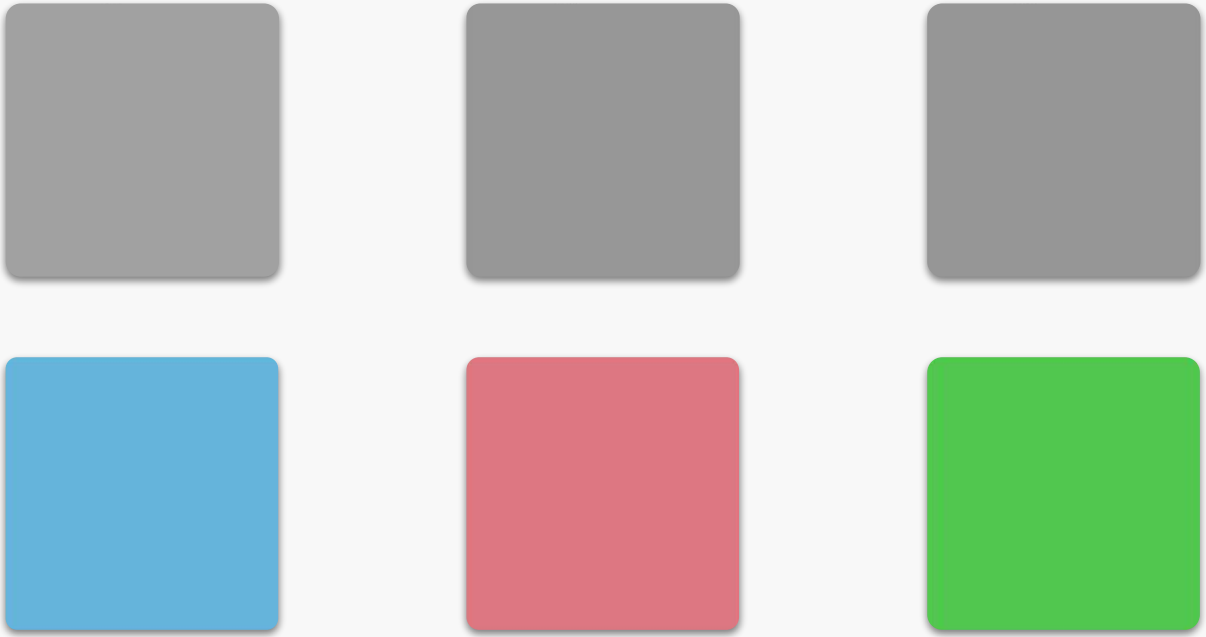
Ground Truth

The model performs well, successfully colorizing both the subject and the background with natural-looking results.

Why is this problem so difficult to solve?



They Look the Same
But Are They?



Future Steps

Create a Generalizable Colorization Network

Expand the Dataset

Incorporate more diverse images

Improve the model's ability to handle unfamiliar inputs

Increase Model Complexity

Build deeper networks

Requires more computational resources

Add Textual Context

Pair images with descriptive text labels

Teach the model associations between objects and typical colors

a group of five women posing on golden sand with the ocean and a white suspension bridge in the background, under a clear blue sky.

Questions?

Appendix

References & Additional Links

References:
1. Zhang, Z., Li, Y., & Shin, B. S. (2022). Robust Medical Image Colorization with Spatial Mask-Guided Generative Adversarial Network. Bioengineering (Basel, Switzerland), 9(12), 721. https://doi.org/10.3390/bioengineering9120721

2. Pai, Nick. (2024). Understanding RGB, YCbCr and Lab Color Spaces. Medium. https://medium.com/@weichenpai/understanding-rgb-ycbcr-and-lab-color-spaces-f9c4a5fe485a

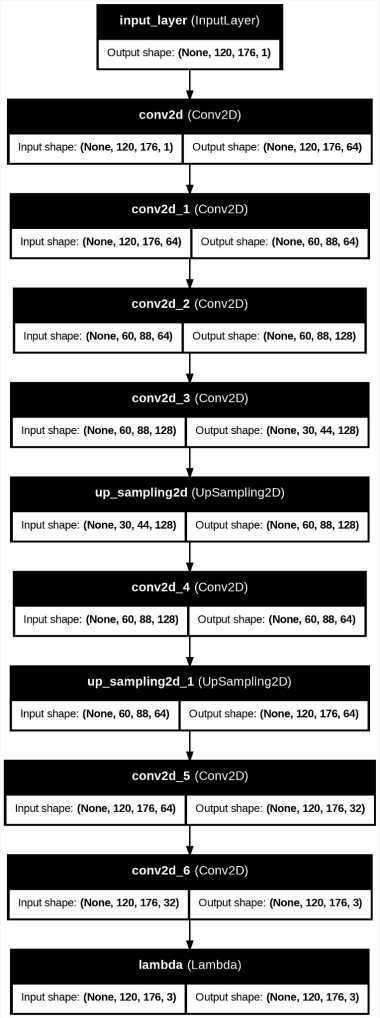
Additional Links:
Link to our Dataset: BA865 - Group 6: Image Colorization Dataset
Link to our Notebook: BA865 - Group 6: Image Colorization Notebook
Link to our Medium Article: Image Colorization Using Neural Networks

Sequential RGB

Model: "functional_1"

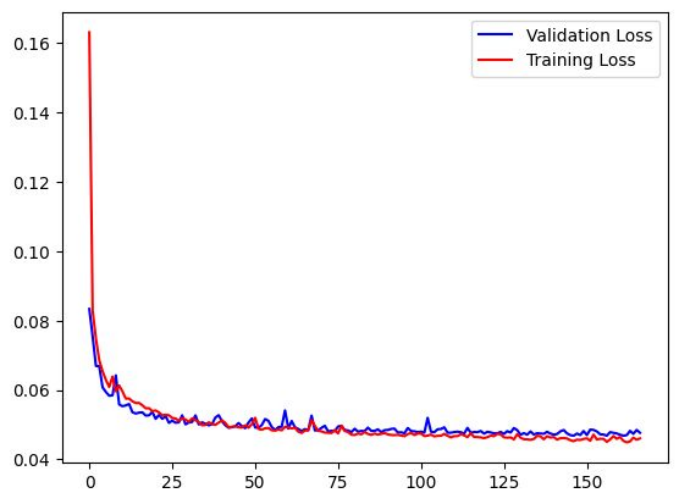
Layer (type)	Output Shape	Param #
input_layer_1 (InputLayer)	(None, 120, 176, 1)	0
conv2d_7 (Conv2D)	(None, 120, 176, 64)	640
conv2d_8 (Conv2D)	(None, 60, 88, 64)	36,928
conv2d_9 (Conv2D)	(None, 60, 88, 128)	73,856
conv2d_10 (Conv2D)	(None, 30, 44, 128)	147,584
up_sampling2d_2 (UpSampling2D)	(None, 60, 88, 128)	0
conv2d_11 (Conv2D)	(None, 60, 88, 64)	73,792
up_sampling2d_3 (UpSampling2D)	(None, 120, 176, 64)	0
conv2d_12 (Conv2D)	(None, 120, 176, 32)	18,464
conv2d_13 (Conv2D)	(None, 120, 176, 3)	867
lambda_1 (Lambda)	(None, 120, 176, 3)	0

Total params: 352,131 (1.34 MB)
Trainable params: 352,131 (1.34 MB)
Non-trainable params: 0 (0.00 B)

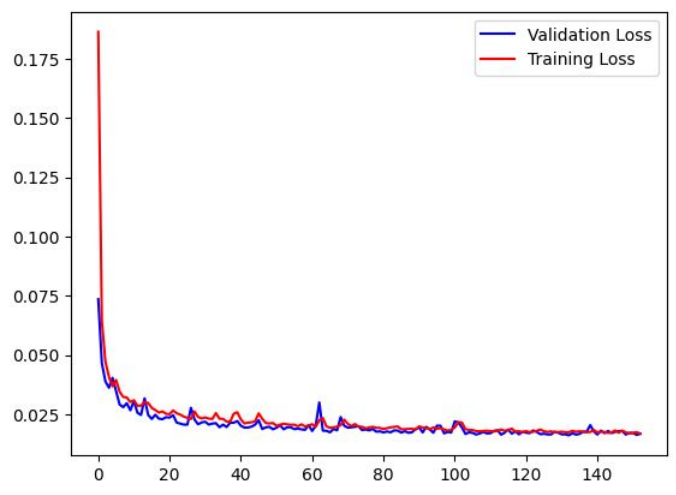


Sequential RGB

Full Scene Portraits



MSBA Headshots



Loss over Epochs

Sequential RGB

Full Scene Portraits

Training Set

Test Set

MSBA Headshots

Training Set

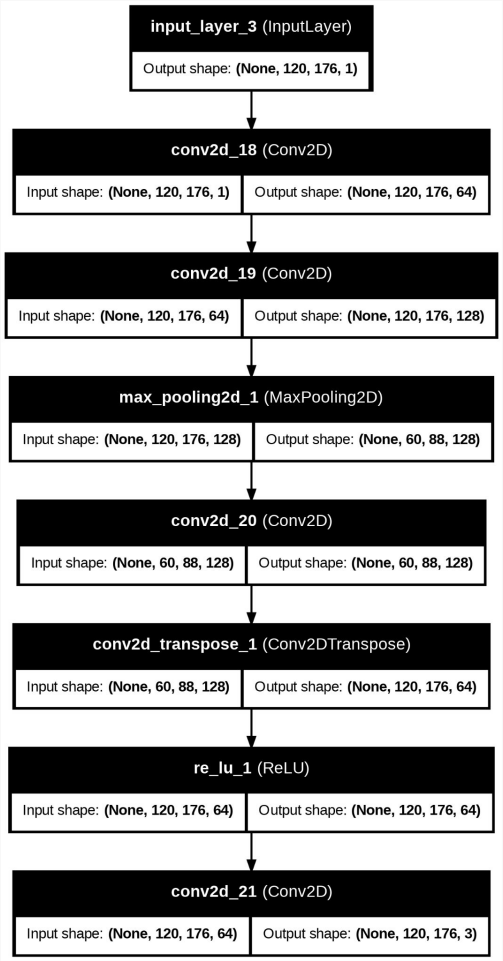
Test Set

Simple CNN

Model: "functional_2"

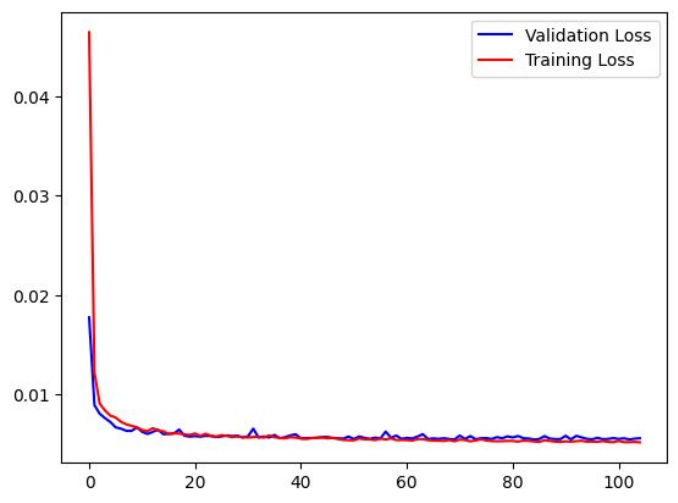
Layer (type)	Output Shape	Param #
input_layer_2 (InputLayer)	(None, 120, 176, 1)	0
conv2d_14 (Conv2D)	(None, 120, 176, 64)	640
conv2d_15 (Conv2D)	(None, 120, 176, 128)	73,856
max_pooling2d (MaxPooling2D)	(None, 60, 88, 128)	0
conv2d_16 (Conv2D)	(None, 60, 88, 128)	147,584
conv2d_transpose (Conv2DTranspose)	(None, 120, 176, 64)	32,832
re_lu (ReLU)	(None, 120, 176, 64)	0
conv2d_17 (Conv2D)	(None, 120, 176, 3)	1,731

Total params: 256,643 (1002.51 KB)
Trainable params: 256,643 (1002.51 KB)
Non-trainable params: 0 (0.00 B)

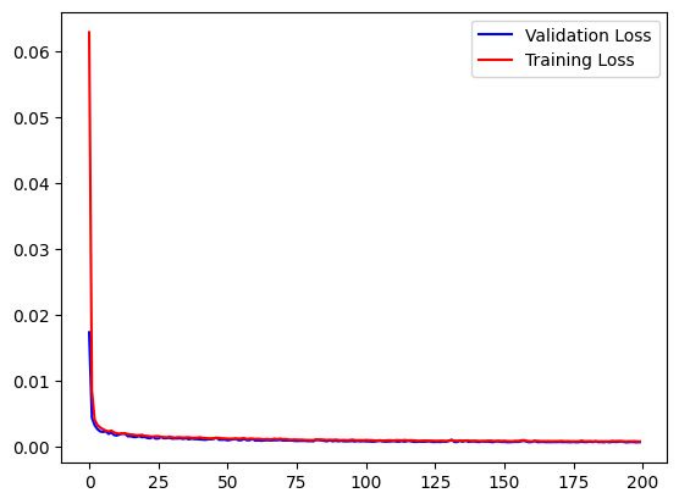


Simple CNN

Full Scene Portraits



MSBA Headshots



Loss over Epochs

Simple CNN

Full Scene Portraits

MSBA Headshots

Training Set

Test Set

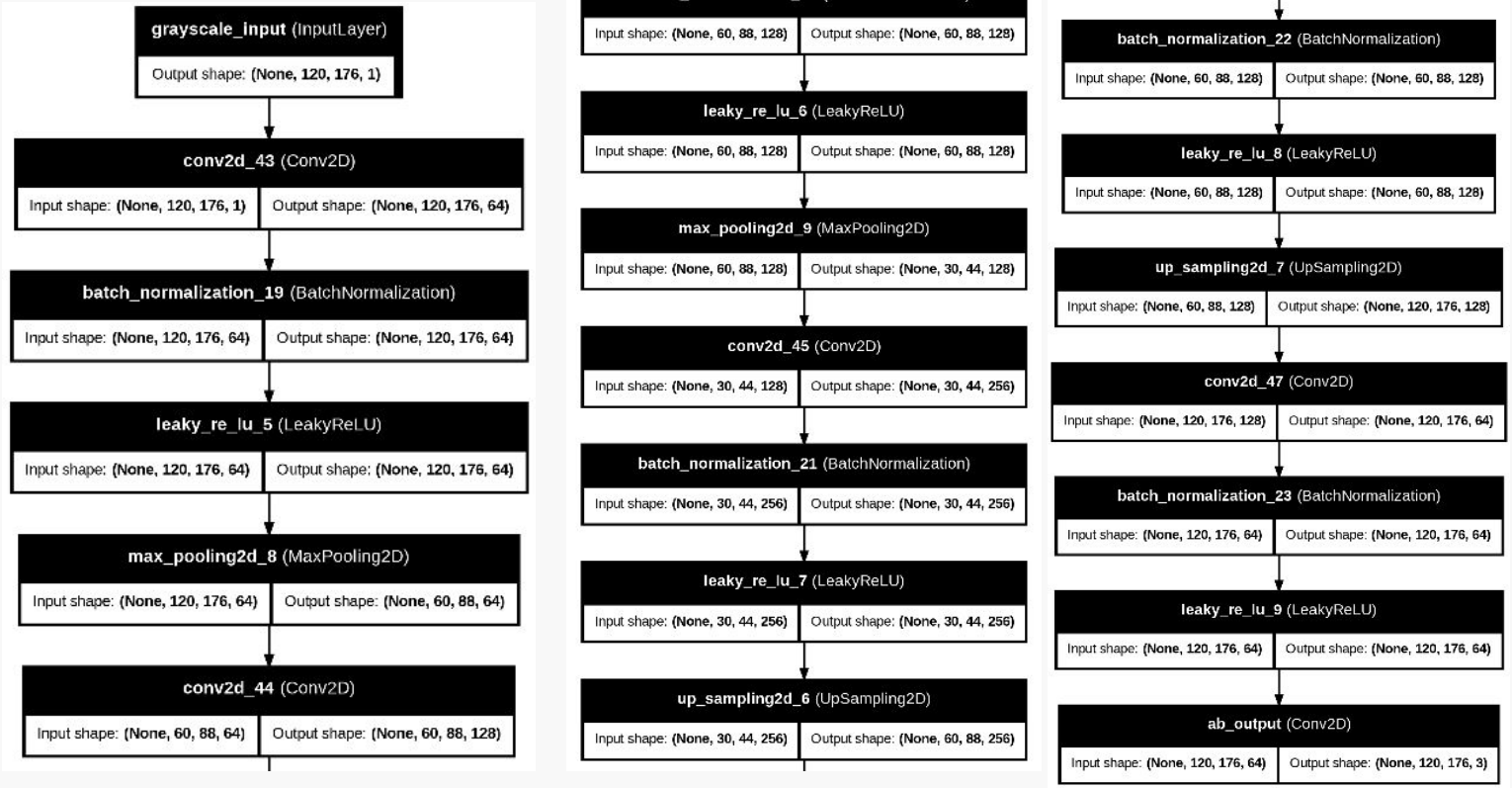
Improved CNN

Model: "ColorizationModel"

Layer (type)	Output Shape	Param #
grayscale_input (InputLayer)	(None, 120, 176, 1)	0
conv2d_38 (Conv2D)	(None, 120, 176, 64)	640
batch_normalization_14 (BatchNormalization)	(None, 120, 176, 64)	256
leaky_re_lu (LeakyReLU)	(None, 120, 176, 64)	0
max_pooling2d_6 (MaxPooling2D)	(None, 60, 88, 64)	0
conv2d_39 (Conv2D)	(None, 60, 88, 128)	73,856
batch_normalization_15 (BatchNormalization)	(None, 60, 88, 128)	512
leaky_re_lu_1 (LeakyReLU)	(None, 60, 88, 128)	0
max_pooling2d_7 (MaxPooling2D)	(None, 30, 44, 128)	0
conv2d_40 (Conv2D)	(None, 30, 44, 256)	295,168
batch_normalization_16 (BatchNormalization)	(None, 30, 44, 256)	1,024
leaky_re_lu_2 (LeakyReLU)	(None, 30, 44, 256)	0
up_sampling2d_4 (UpSampling2D)	(None, 60, 88, 256)	0
conv2d_41 (Conv2D)	(None, 60, 88, 128)	295,040
batch_normalization_17 (BatchNormalization)	(None, 60, 88, 128)	512
leaky_re_lu_3 (LeakyReLU)	(None, 60, 88, 128)	0
up_sampling2d_5 (UpSampling2D)	(None, 120, 176, 128)	0
conv2d_42 (Conv2D)	(None, 120, 176, 64)	73,792
batch_normalization_18 (BatchNormalization)	(None, 120, 176, 64)	256
leaky_re_lu_4 (LeakyReLU)	(None, 120, 176, 64)	0
ab_output (Conv2D)	(None, 120, 176, 3)	1,731

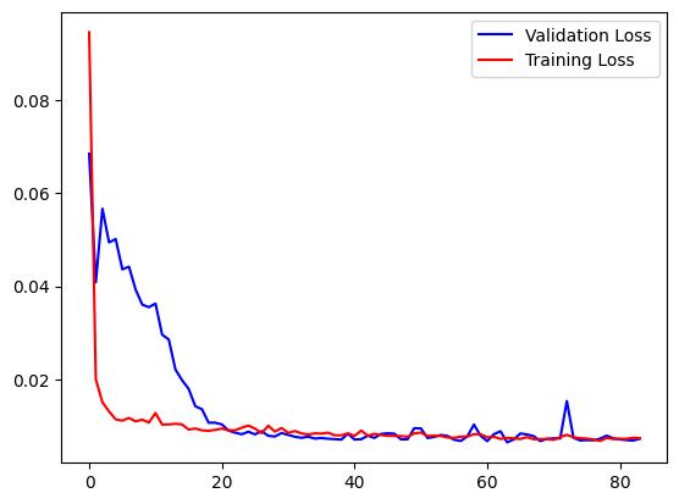
Total params: 742,787 (2.83 MB)
Trainable params: 741,507 (2.83 MB)
Non-trainable params: 1,280 (5.00 KB)

Improved CNN

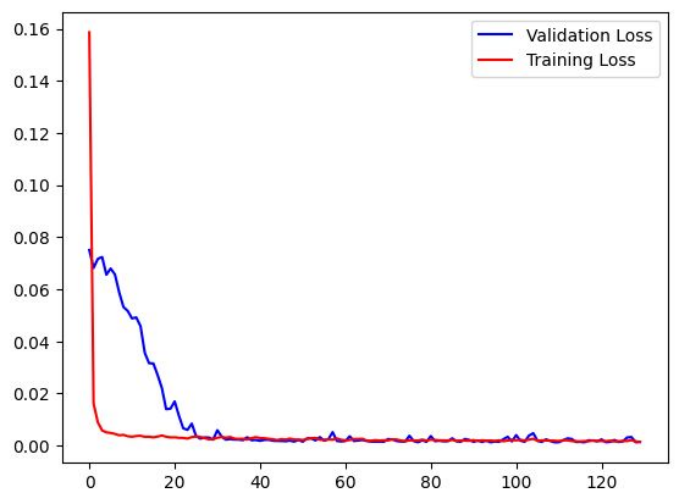


Improved CNN

Full Scene Portraits



MSBA Headshots



Loss over Epochs

Improved CNN

Full Scene Portraits

MSBA Headshots

Training Set

Test Set

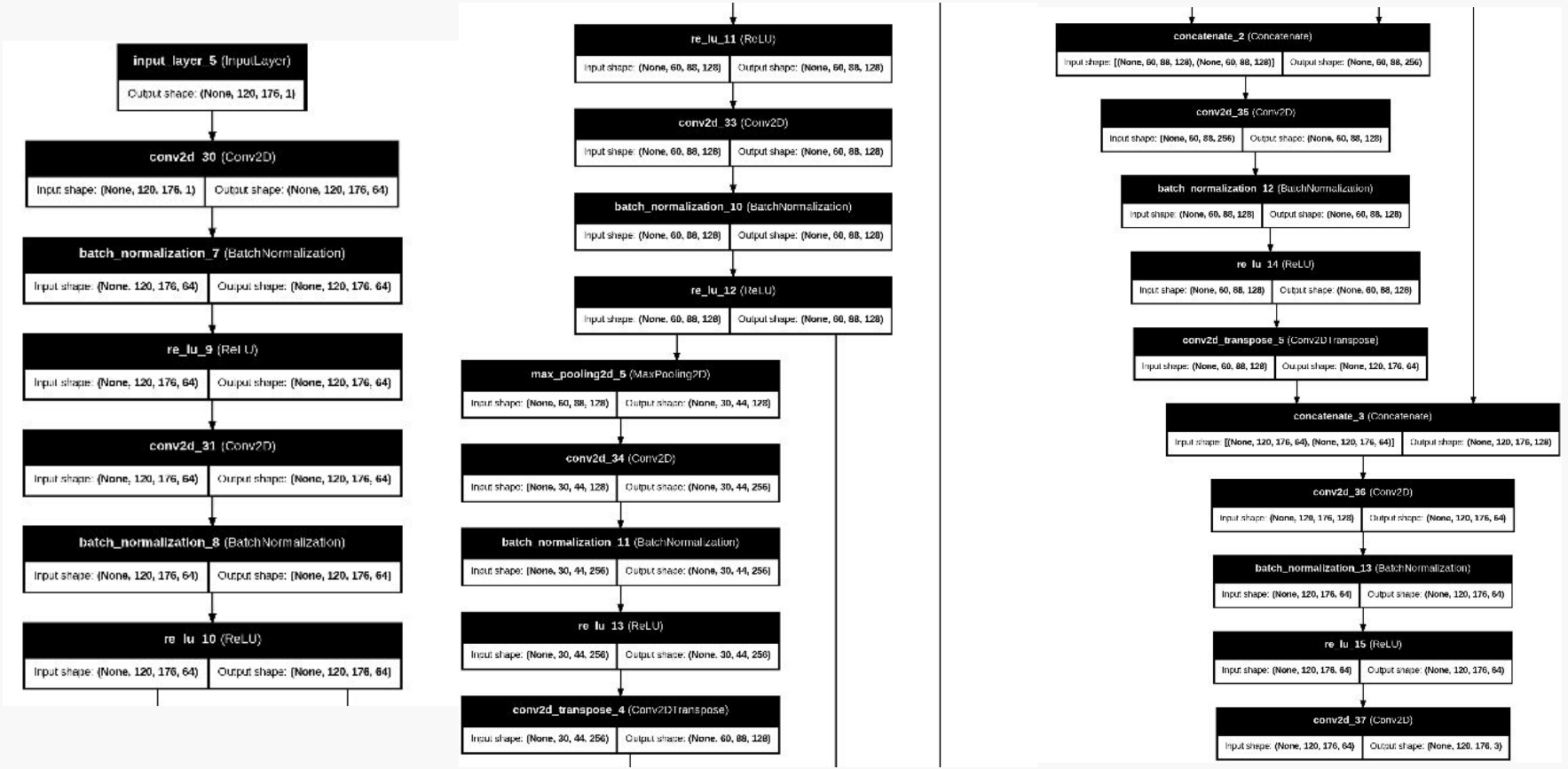
U-Net Implementation

Model: "functional_4"

Layer (type)	Output Shape	Param #	Connected to
input_layer_4 (InputLayer)	(None, 120, 176, 1)	0	–
conv2d_22 (Conv2D)	(None, 120, 176, 64)	640	input_layer_4[0]…
batch_normalization (BatchNormalizatio…	(None, 120, 176, 64)	256	conv2d_22[0][0]
re_lu_2 (ReLU)	(None, 120, 176, 64)	0	batch_normalizat…
conv2d_23 (Conv2D)	(None, 120, 176, 64)	36,928	re_lu_2[0][0]
batch_normalizatio… (BatchNormalizatio…	(None, 120, 176, 64)	256	conv2d_23[0][0]
re_lu_3 (ReLU)	(None, 120, 176, 64)	0	batch_normalizat…
max_pooling2d_2 (MaxPooling2D)	(None, 60, 88, 64)	0	re_lu_3[0][0]
conv2d_24 (Conv2D)	(None, 60, 88, 128)	73,856	max_pooling2d_2[…
batch_normalizatio… (BatchNormalizatio…	(None, 60, 88, 128)	512	conv2d_24[0][0]
re_lu_4 (ReLU)	(None, 60, 88, 128)	0	batch_normalizat…
conv2d_25 (Conv2D)	(None, 60, 88, 128)	147,584	re_lu_4[0][0]
batch_normalizatio… (BatchNormalizatio…	(None, 60, 88, 128)	512	conv2d_25[0][0]
re_lu_5 (ReLU)	(None, 60, 88, 128)	0	batch_normalizat…
max_pooling2d_3 (MaxPooling2D)	(None, 30, 44, 128)	0	re_lu_5[0][0]
conv2d_26 (Conv2D)	(None, 30, 44, 256)	295,168	max_pooling2d_3[…
batch_normalizatio… (BatchNormalizatio…	(None, 30, 44, 256)	1,024	conv2d_26[0][0]
re_lu_6 (ReLU)	(None, 30, 44, 256)	0	batch_normalizat…
conv2d_transpose_2 (Conv2DTranspose)	(None, 60, 88, 128)	131,200	re_lu_6[0][0]
concatenate (Concatenate)	(None, 60, 88, 256)	0	conv2d_transpose… re_lu_5[0][0]
conv2d_27 (Conv2D)	(None, 60, 88, 128)	295,040	concatenate[0][0]
batch_normalizatio… (BatchNormalizatio…	(None, 60, 88, 128)	512	conv2d_27[0][0]
re_lu_7 (ReLU)	(None, 60, 88, 128)	0	batch_normalizat…
conv2d_transpose_3 (Conv2DTranspose)	(None, 120, 176, 64)	32,832	re_lu_7[0][0]
concatenate_1 (Concatenate)	(None, 120, 176, 128)	0	conv2d_transpose… re_lu_3[0][0]
conv2d_28 (Conv2D)	(None, 120, 176, 64)	73,792	concatenate_1[0]…
batch_normalizatio… (BatchNormalizatio…	(None, 120, 176, 64)	256	conv2d_28[0][0]
re_lu_8 (ReLU)	(None, 120, 176, 64)	0	batch_normalizat…
conv2d_29 (Conv2D)	(None, 120, 176, 3)	1,731	re_lu_8[0][0]

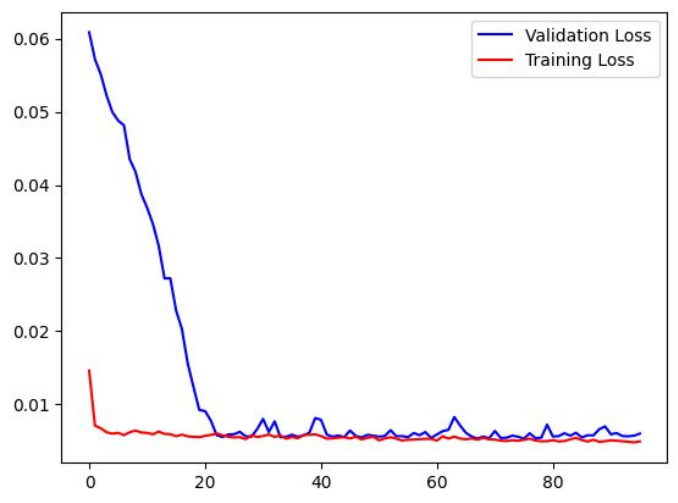
Total params: 1,092,099 (4.17 MB)
Trainable params: 1,090,435 (4.16 MB)
Non-trainable params: 1,664 (6.50 KB)

U-Net Implementation

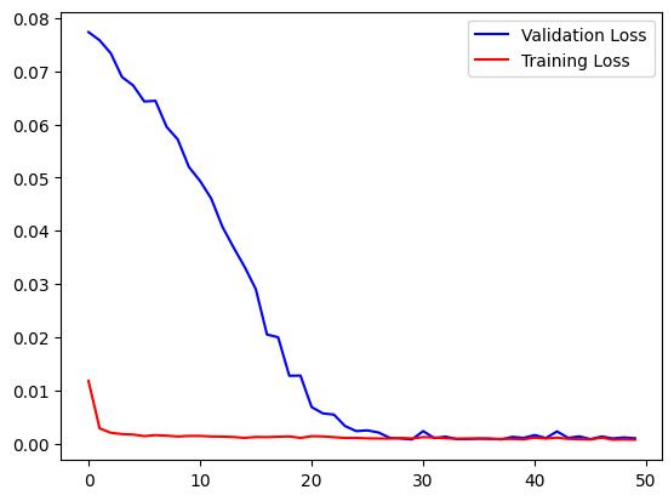


U-Net Implementation

Full Scene Portraits



MSBA Headshots



Loss over Epochs

U-Net Implementation

Full Scene Portraits

MSBA Headshots

Training Set

Test Set

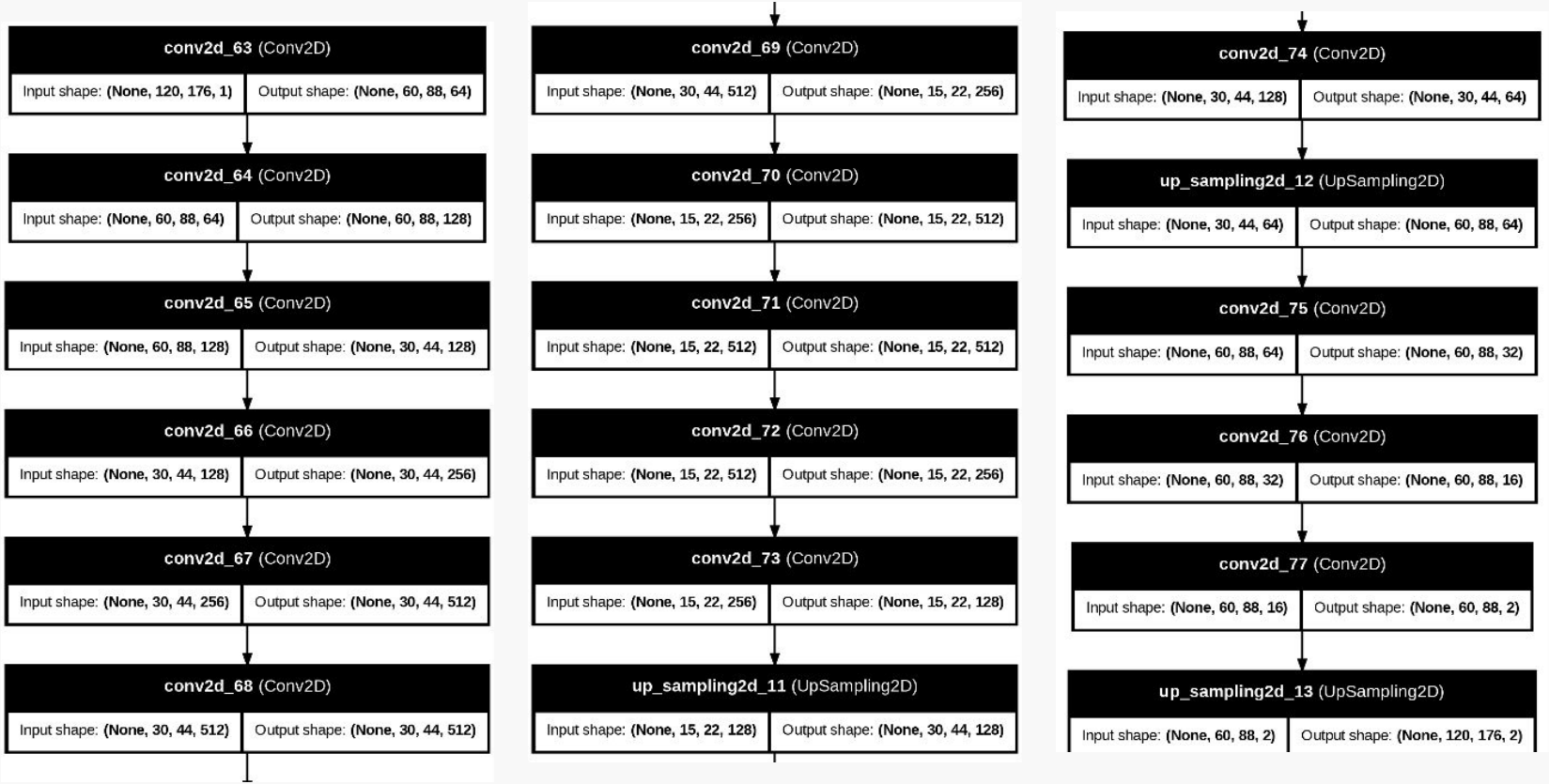
Sequential LAB

Model: "sequential"

Layer (type)	Output Shape	Param #
conv2d_48 (Conv2D)	(None, 60, 88, 64)	640
conv2d_49 (Conv2D)	(None, 60, 88, 128)	73,856
conv2d_50 (Conv2D)	(None, 30, 44, 128)	147,584
conv2d_51 (Conv2D)	(None, 30, 44, 256)	295,168
conv2d_52 (Conv2D)	(None, 30, 44, 512)	1,180,160
conv2d_53 (Conv2D)	(None, 30, 44, 512)	2,359,808
conv2d_54 (Conv2D)	(None, 15, 22, 256)	1,179,904
conv2d_55 (Conv2D)	(None, 15, 22, 512)	1,180,160
conv2d_56 (Conv2D)	(None, 15, 22, 512)	2,359,808
conv2d_57 (Conv2D)	(None, 15, 22, 256)	1,179,904
conv2d_58 (Conv2D)	(None, 15, 22, 128)	295,040
up_sampling2d_8 (UpSampling2D)	(None, 30, 44, 128)	0
conv2d_59 (Conv2D)	(None, 30, 44, 64)	73,792
up_sampling2d_9 (UpSampling2D)	(None, 60, 88, 64)	0
conv2d_60 (Conv2D)	(None, 60, 88, 32)	18,464
conv2d_61 (Conv2D)	(None, 60, 88, 16)	4,624
conv2d_62 (Conv2D)	(None, 60, 88, 2)	290
up_sampling2d_10 (UpSampling2D)	(None, 120, 176, 2)	0

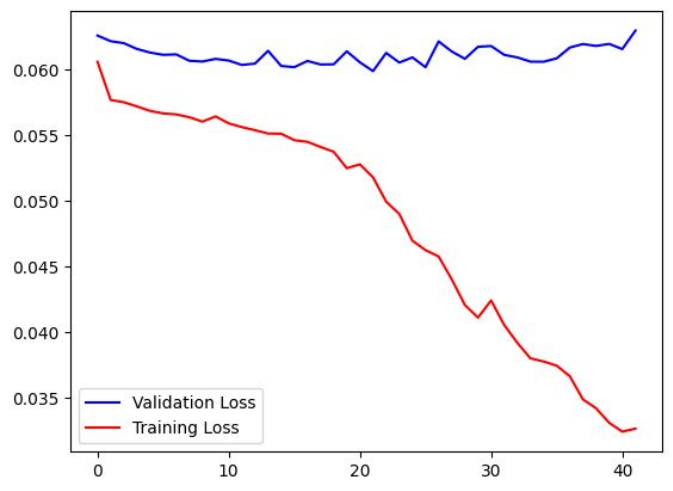
Total params: 10,349,202 (39.48 MB)
Trainable params: 10,349,202 (39.48 MB)
Non-trainable params: 0 (0.00 B)

Sequential LAB

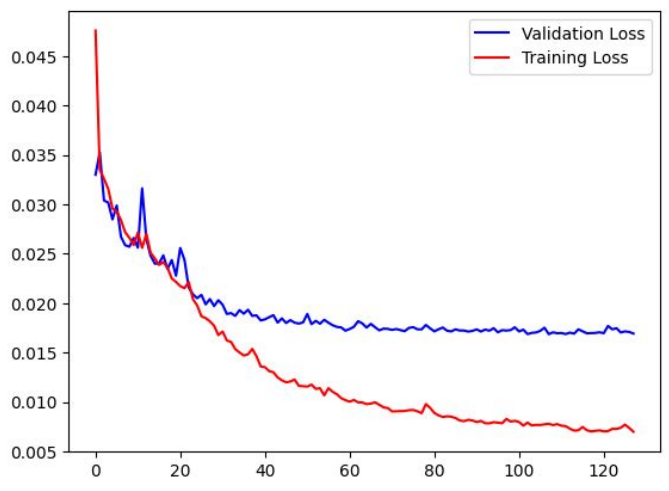


Sequential LAB

Full Scene Portraits



MSBA Headshots



Loss over Epochs

Sequential LAB

Full Scene Portraits

MSBA Headshots

Training Set

Test Set

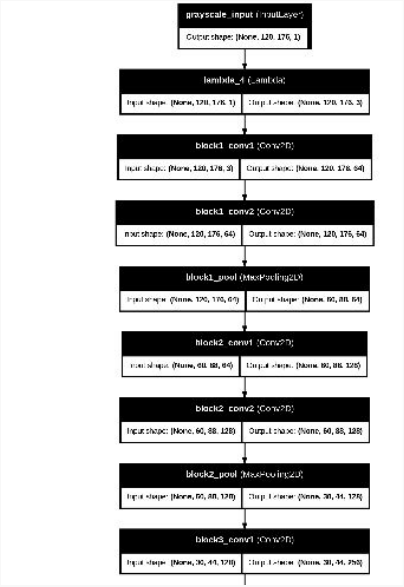
Transformer with VGG

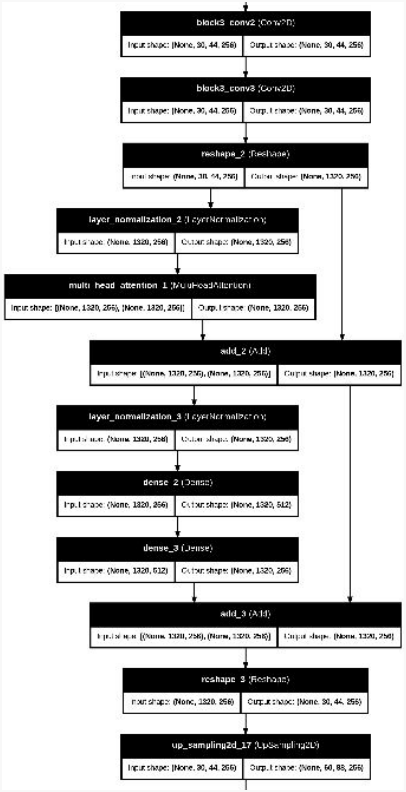
Model: "functional_42"

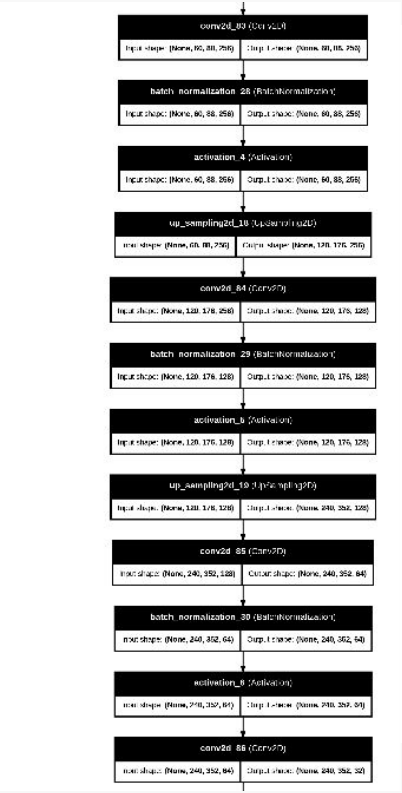
Layer (type)	Output Shape	Param #	Connected to
grayscale_input (InputLayer)	(None, 120, 176, 1)	0	–
lambda_2 (Lambda)	(None, 120, 176, 3)	0	grayscale_input[…
block1_conv1 (Conv2D)	(None, 120, 176, 64)	1,792	lambda_2[0][0]
block1_conv2 (Conv2D)	(None, 120, 176, 64)	36,928	block1_conv1[0][…
block1_pool (MaxPooling2D)	(None, 60, 88, 64)	0	block1_conv2[0][…
block2_conv1 (Conv2D)	(None, 60, 88, 128)	73,856	block1_pool[0][0]
block2_conv2 (Conv2D)	(None, 60, 88, 128)	147,584	block2_conv1[0][…
block2_pool (MaxPooling2D)	(None, 30, 44, 128)	0	block2_conv2[0][…
block3_conv1 (Conv2D)	(None, 30, 44, 256)	295,168	block2_pool[0][0]
block3_conv2 (Conv2D)	(None, 30, 44, 256)	590,080	block3_conv1[0][…
block3_conv3 (Conv2D)	(None, 30, 44, 256)	590,080	block3_conv2[0][…
reshape (Reshape)	(None, 1320, 256)	0	block3_conv3[0][…
layer_normalization (LayerNormalizatio…	(None, 1320, 256)	512	reshape[0][0]
multi_head_attenti… (MultiHeadAttentio…	(None, 1320, 256)	1,051,904	layer_normalizat… layer_normalizat…
add (Add)	(None, 1320, 256)	0	reshape[0][0], multi_head_atten…
layer_normalizatio… (LayerNormalizatio…	(None, 1320, 256)	512	add[0][0]
dense (Dense)	(None, 1320, 512)	131,584	layer_normalizat…
dense_1 (Dense)	(None, 1320, 256)	131,328	dense[0][0]
add_1 (Add)	(None, 1320, 256)	0	add[0][0], dense_1[0][0]
reshape_1 (Reshape)	(None, 30, 44, 256)	0	add_1[0][0]
up_sampling2d_14 (UpSampling2D)	(None, 60, 88, 256)	0	reshape_1[0][0]
conv2d_78 (Conv2D)	(None, 60, 88, 256)	590,080	up_sampling2d_14…
batch_normalizatio… (BatchNormalizatio…	(None, 60, 88, 256)	1,024	conv2d_78[0][0]
activation (Activation)	(None, 60, 88, 256)	0	batch_normalizat…
up_sampling2d_15 (UpSampling2D)	(None, 120, 176, 256)	0	activation[0][0]
conv2d_79 (Conv2D)	(None, 120, 176, 128)	295,040	up_sampling2d_15…
batch_normalizatio… (BatchNormalizatio…	(None, 120, 176, 128)	512	conv2d_79[0][0]
activation_1 (Activation)	(None, 120, 176, 128)	0	batch_normalizat…
up_sampling2d_16 (UpSampling2D)	(None, 240, 352, 128)	0	activation_1[0][…
conv2d_80 (Conv2D)	(None, 240, 352, 64)	73,792	up_sampling2d_16…
batch_normalizatio… (BatchNormalizatio…	(None, 240, 352, 64)	256	conv2d_80[0][0]
activation_2 (Activation)	(None, 240, 352, 64)	0	batch_normalizat…
conv2d_81 (Conv2D)	(None, 240, 352, 32)	18,464	activation_2[0][…
batch_normalizatio… (BatchNormalizatio…	(None, 240, 352, 32)	128	conv2d_81[0][0]
activation_3 (Activation)	(None, 240, 352, 32)	0	batch_normalizat…
conv2d_82 (Conv2D)	(None, 240, 352, 3)	867	activation_3[0][…
lambda_3 (Lambda)	(None, 120, 176, 3)	0	conv2d_82[0][0]

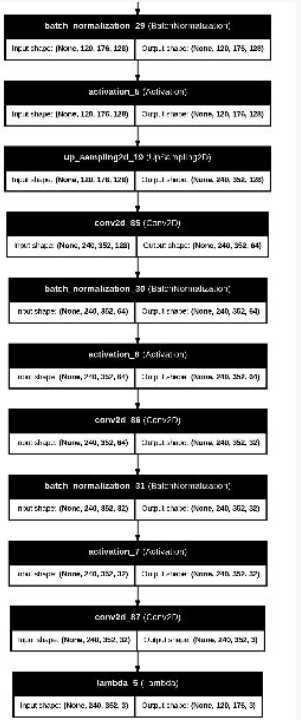
Total params: 4,031,491 (15.38 MB)
Trainable params: 2,295,043 (8.75 MB)
Non-trainable params: 1,736,448 (6.62 MB)

Transformer with VGG



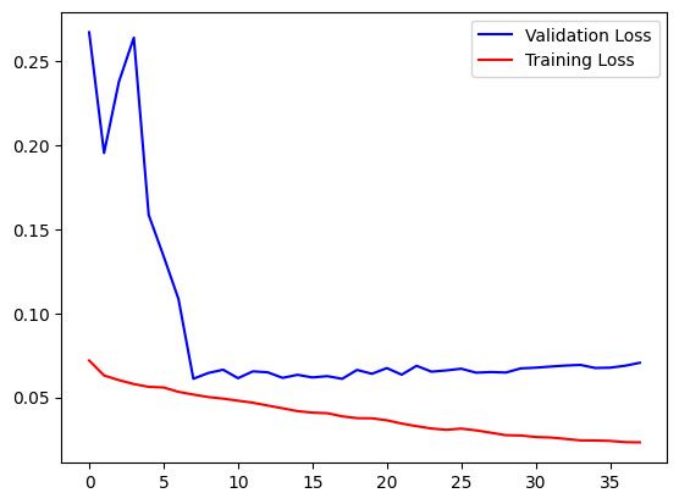




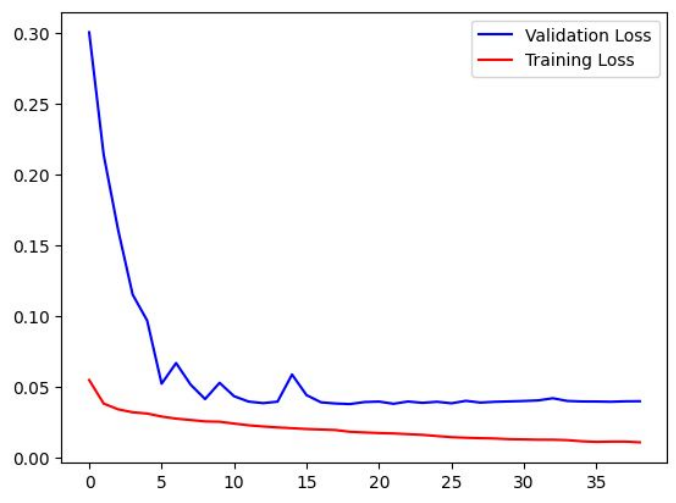


Transformer with VGG

Full Scene Portraits



MSBA Headshots



Loss over Epochs

Transformer with VGG

Full Scene Portraits

MSBA Headshots

Training Set

Test Set