Predictor struct:

predictor.add(Input(trainX.shape[1:]))

predictor.add(Conv2D(32, (9, 9), (2, 2), padding = 'same'))

predictor.add(BatchNormalization())

predictor.add(PReLU(shared\_axes = [1, 2]))

predictor.add(Conv2D(32, (5, 5), (2, 2), padding = 'same'))

predictor.add(BatchNormalization())

predictor.add(PReLU(shared\_axes = [1, 2]))

predictor.add(Conv2D(16, (5, 5), (1, 1), padding = 'same'))

predictor.add(BatchNormalization())

predictor.add(PReLU(shared\_axes = [1, 2]))

predictor.add(Conv2D(16, (5, 5), (1, 1), padding = 'same'))

predictor.add(BatchNormalization())

predictor.add(PReLU(shared\_axes = [1, 2]))

predictor.add(Conv2D(8 , (5, 5), (1, 1), padding = 'same'))

predictor.add(BatchNormalization())

predictor.add(PReLU(shared\_axes = [1, 2]))

predictor.add(Conv2D(8 , (5, 5), (1, 1), padding = 'same'))

predictor.add(PReLU(shared\_axes = [1, 2]))

predictor.add(Flatten())

predictor.add(Dense(256, activation = 'relu'))

predictor.add(Dense(64, activation = 'relu'))

predictor.add(Dense(stages\_count))

Trying to learn the behavior of JSCC\_1

Trained on 1000 images of Imagenet\_a

Stages\_count = 5

Patch\_size = 32

Train SNR = 13

Inputs pixels range: {0, 1}

Output: PSNR