

1 Training the best model

For training the best model over all possible model (or a subsample of them) we have to apply the model selection principles.

1.1 Model Selection

Model selection involves training different models using the same training dataset. This could include using the same model architecture with different parameters or employing entirely different architectures. After training, we utilize a validation set that does not contain any samples from the training set to calculate the **loss** and **metrics** for each model.

Once we have selected the **best model** based on these metrics, we continue training only that model using the combined training and validation set, as the validation set is no longer needed. To clarify, the final performance metrics of the model (e.g., **loss**, **PSNR**) have not yet been evaluated at this stage. The proportion of the splitting is reported in Table 1

Table 1: Data Splitting proportion

Dataset Split	Value
Train	0.50
Validation	0.30
Test	0.20

For simplicity (and lack of computational resources), we perform model selection on the two possible parameter of the Network, i.g. the number of channels (feature channels), and the number of residual block (that regulates the depth of the network). The combinations used are in Table 2.

Table 2: Validation Parameters

Number of Channels	Number of Residual Blocks
16	4
16	8
16	16
32	4
32	8
32	16
64	4
64	8
64	16

1.2 Training for Validation

The training needs to have a training set, a Loss and an Optimizer. We choose L1 Loss and ADAM optimizer with hyperparameters reported in Table 3. We

Table 3: Adam Hyperparameters

Hyperparameter	Value
Learning Rate	1×10^{-4}
Betas	(0.9, 0.999)
Epsilon	1×10^{-8}

trained each model for 50 epochs and choose the best model based on the result found in validation. The state of the model is saved for future use and demonstration. The best model is described in Table 4, and we can see that the model is smaller than the one in the paper that used 16 Residual Blocks. Indeed we are working with much smaller images.

To recap, what has been used during all the trainings is reported in Table 5. A visual comparison of the result of the best large model in validation can be observed in Figure 1 and in Figure 2 for the smaller model. Even if the difference in db is not marked, this model, compared to algorithmic techniques like Biliner , gives a less blurry image.

Table 4: Validation Results

Metric	Value
L1 Loss	0.017452
PSNR	30.2676 dB
Number of Channels	64
Number of Residual Blocks	8

Table 5: Training Recap

Parameter	Value
Loss Function	L1
Optimizer	Adam
Epochs	50
DataLoader	training set

Figure 1: Validation output large model

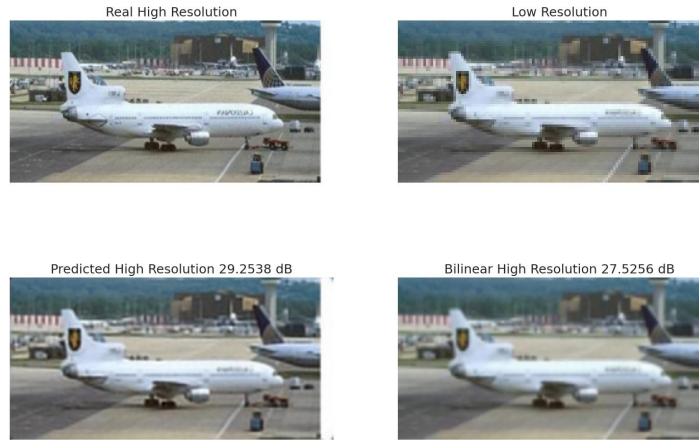
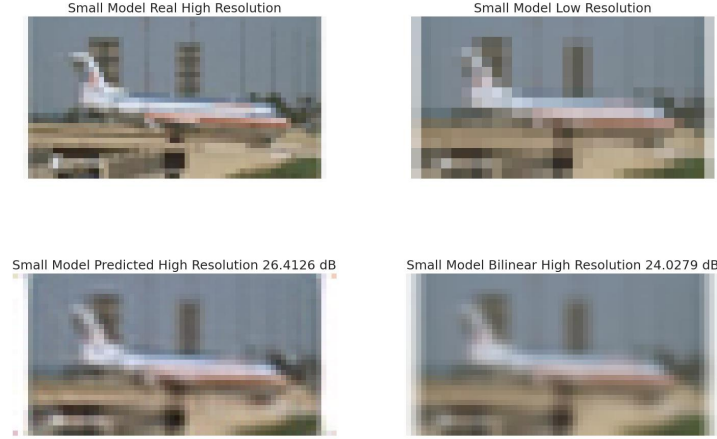


Figure 2: Validation output small model



Continuing to train the best model for an additional 150 epochs using the combined training and validation set leads to the results shown in terms of loss and PSNR, as seen in Figure 3 and Figure 4. The classification of image quality levels as low, medium, and high based on PSNR values is a matter of convention and does not relate to resolution; it only indicates how much the generated image differs from the original.

Figure 3: L1 Loss

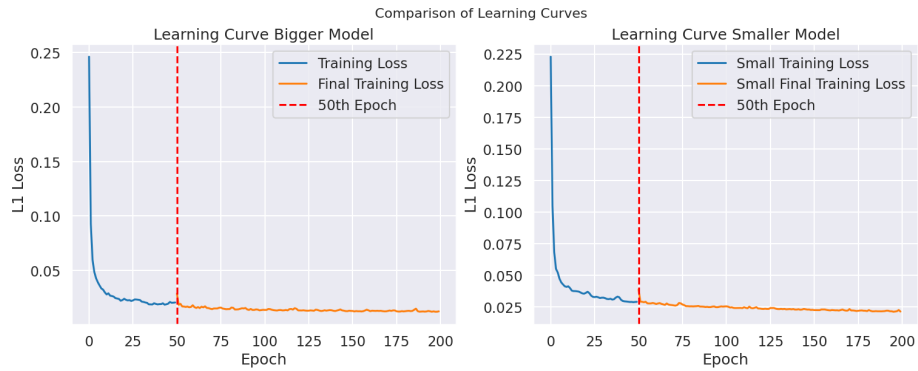


Figure 4: PSNR

