ETH Zürich
Mathematical Foundations of
Computer Graphics and Vision
Spring 2019

M. Arda Düzceker
18-950-485
ardad@student.ethz.ch

EXERCISE 6

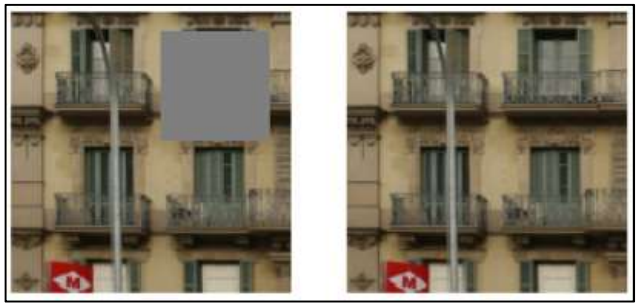
NEURAL NETWORKS AND MACHINE LEARNING TECHNIQUES

Task 1.1 and Task 1.2: Fixed Location Crop and Random Location Crop:

The authors of [1] suggest, "cropped" area of the image should be filled with zero when the input image is zero centered. In our case, input image is scaled such that each pixel value is between 0.0 and 1.0, therefore I opted for filling the cropped areas with 0.5 which corresponds to the center value. Such strategy has another advantage. While generating desired output of this assignment (image triples – input, groundtruth, prediction), one has to find the cropped area in order to replace it predictions. In the case of fixed center crop, this is not an issue. However, when random crops are utilized, 0.5 pixel value is helpful to locate the cropped area since such value in the input image is unique to the cropped area (127/255 = 0.498 and 128/255 = 0.502).

Fixed Center Crop Examples:

Random Crop Examples:



Also, I changed the input pipeline such that both types of crops are performed on the air (both are like random crops). The motivation behind is to be able to apply random augmentation to the dataset while feeding it. These augmentations are:

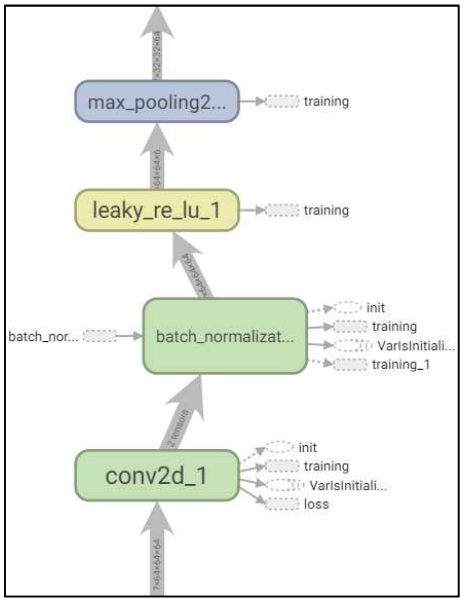
1. 50% chance of left-right flip
2. 25% chance of additive Gaussian noise
3. 50% chance of slight brightness, contrast and hue alteration

Principal reason for pursuing online augmentation is to increase the robustness of the model and prevent overfitting on such a small dataset.

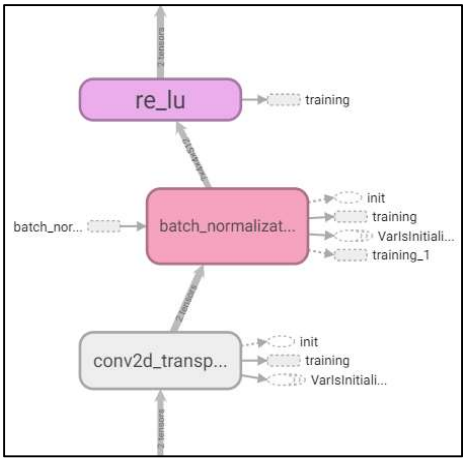
M. Arda Düzceker
18-950-485
ardad@student.ethz.ch

Task 2.1: Create the encoder decoder graph

Encoder Block:

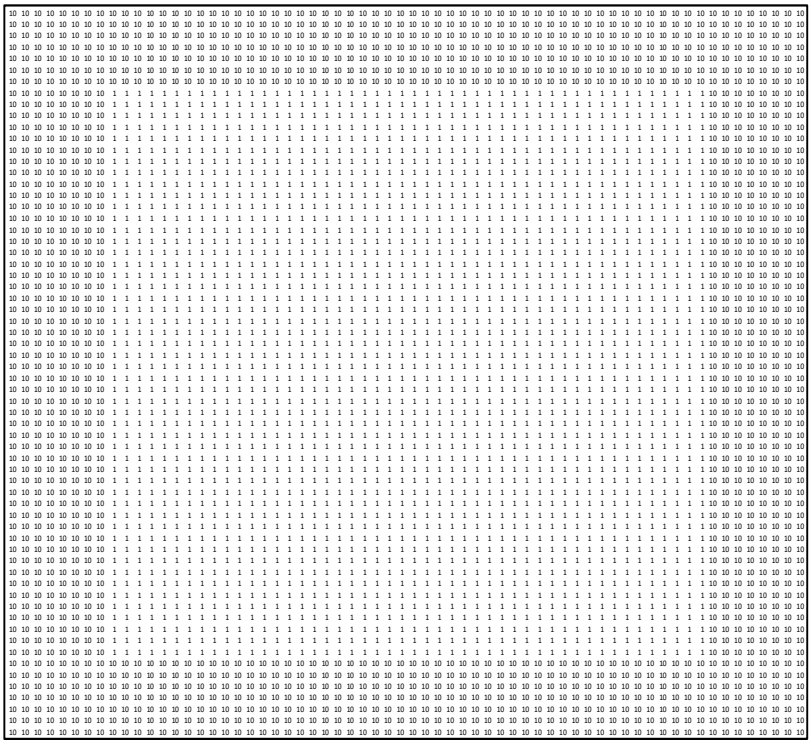


Decoder Block:



Task 2.2 and Task 2.3: Create the Reconstruction Loss and Overlapping Reconstruction Loss

Regular reconstruction loss is mean-squared-error of the predicted pixel values in the cropped 64 x 64 x 3 region. Overlapping reconstruction loss is again mean-squared-error but it is implemented using loss weighting. For this purpose, a weight mask with size 64 x 64 x 3 is created. The weight mask that is applied to one channel is as follows and each RGB channel has the same mask.



M. Arda Düzceker
18-950-485
ardad@student.ethz.ch

Task 2.4: What is the purpose of these different losses?

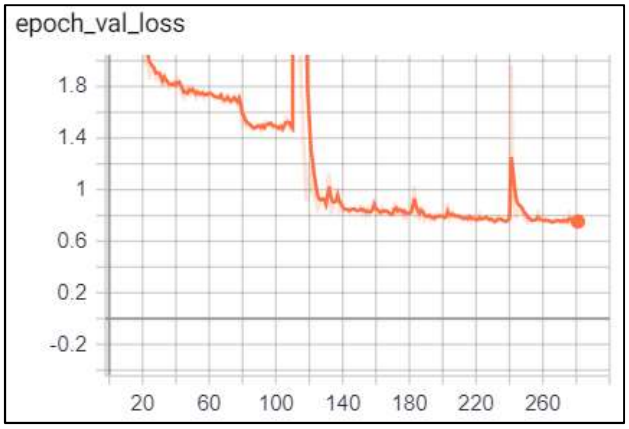
Regular L2 (mean-squared-error) loss encourages the network to produce a rough outline of the texture in the cropped area. However, it fails to learn high frequency details. It results in predicting the mean of the distribution since such prediction can actually minimize the mean pixel-wise error.

The idea behind the overlapping loss is to emphasize the consistency of prediction with the context by discouraging the network to make mistakes around the edges of the cropped area.

Task 3.2 and Task 3.3

Early stopping (monitoring the validation loss) is added to the training procedure to prevent ending up with an overfitting model.

Also, learning rate is halved whenever training loss experiences a plateau. The effect of this strategy is clearly observable in training loss graph below (left).



For this training, optional dropout and spatial dropout layers that I added are disabled (I observed that they often prevent training loss to converge in inpainting context). Input images are randomly augmented at each iteration according to the discussion in Task 1.1 and Task 1.2. The loss function is the regular L2 loss without mask as assignment requires and the hyperparameters are,

Batch Size: 32

Initial Learning Rate: 0.0005

L1 Penalty Lambda = 0.0

L2 Penalty Lambda = 0.003

Best validation loss is acquired at 231st iteration with 0.7399 which caused early stopping 50 iteration later. The model at 231st iteration is recovered and used for prediction of test images. The training loss at this point is very low compared to the validation and test loss which is a clear sign of overfitting that is also observable in qualitative results from almost "memorizing" behavior. However, I could not find the correct hyperparamaters that provides the medium between underfitting and overfitting.

Test loss is of this model is 0.7990809306502342 (can be found at the end of the console output)

M. Arda Düzceker
18-950-485
ardad@student.ethz.ch

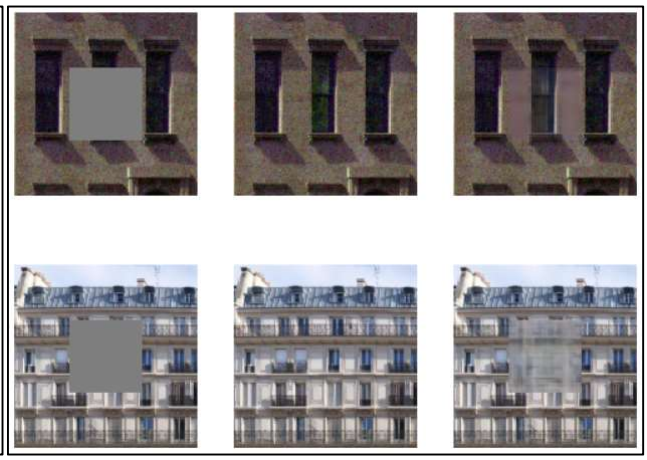
Training and Validation Examples:

0th iteration

70th iteration

150th iteration

230th iteration



Test Examples (there are better/worse predictions that can be found in “middle-crop-run/Results” folder):

M. Arda Düzceker
18-950-485
ardad@student.ethz.ch

Task 3.4

Like Task 3.2, early stopping is used. The learning rate is halved whenever training loss experiences a plateau. Optional dropout and spatial dropout are disabled, also no kernel regularization is applied this time. Because, as one of the main differences from fixed cropping, the network becomes more robust and less likely to overfit when random crops are utilized. Finally, input images are randomly augmented at each iteration.

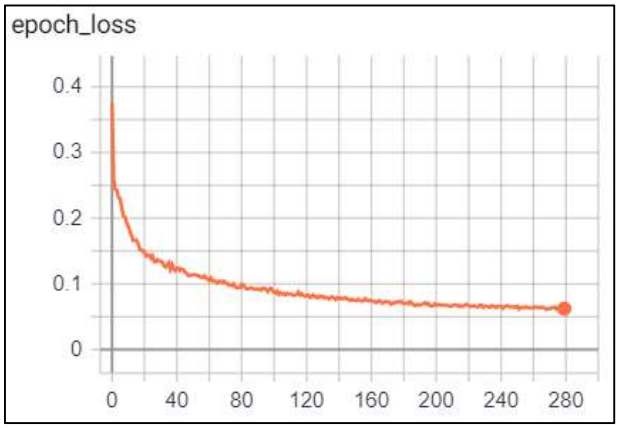
The loss function is the overlapping L2 loss with weight mask as assignment requires and the hyperparameters are,

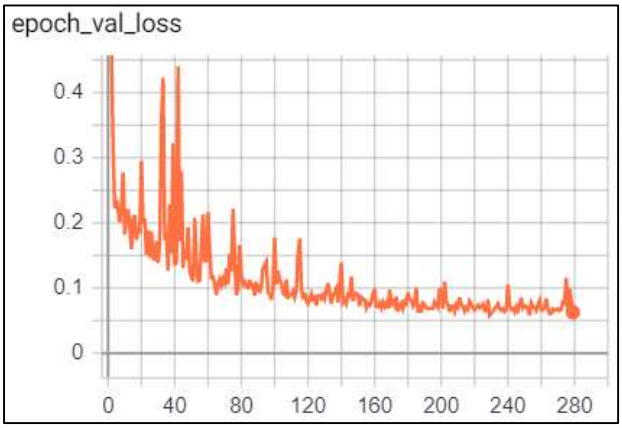
Batch Size: 48

Initial Learning Rate: 0.001

L1 Penalty Lambda = 0.0

L2 Penalty Lambda = 0.0





Best validation loss is acquired at 229th iteration with 0.05803 which caused early stopping 50 iteration later. The model at 229th iteration is recovered and used for prediction of test images.

Test loss is of this model is 0.07165404409170151 (can be found at the end of the console output). Note that, Task 3.2 and Task 3.4 test losses are not comparable quantitatively since different losses functions are used.

There are several significant differences between Task 3.2 and Task 3.4 results:

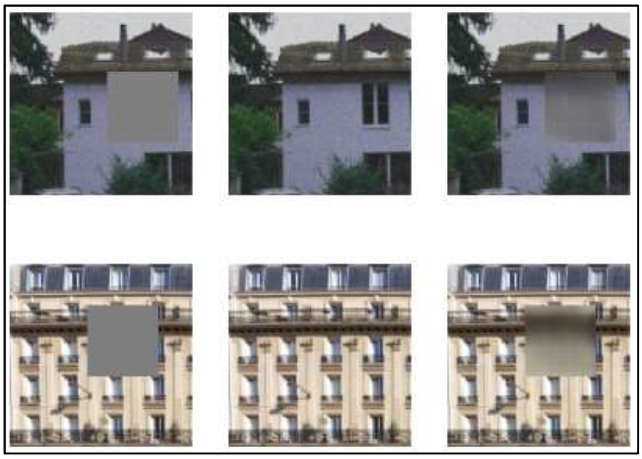
1. Training procedure is more stable with 3.4 (no exploding gradients observed like the ones in 3.2 output graphs)
2. Initial predictions do not produce strange red-green-blue pixels in 3.4. I think, the reason is the overlapping loss which ensures consistency.
3. As stated before, the network becomes more robust and less likely to overfit when random crops are utilized in 3.4.

M. Arda Düzceker
18-950-485
ardad@student.ethz.ch

Training and Validation Examples:

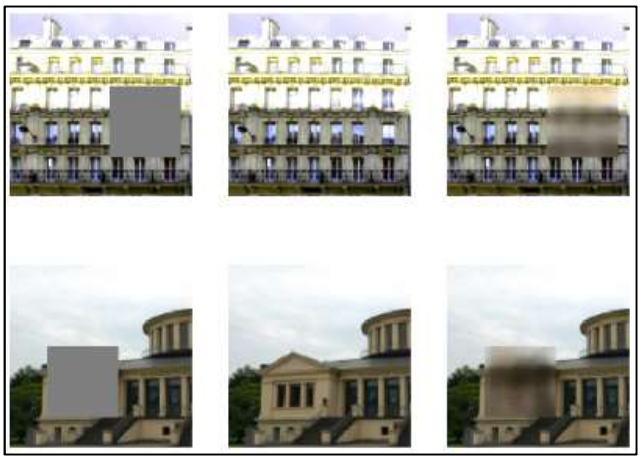
0th iteration

70th iteration

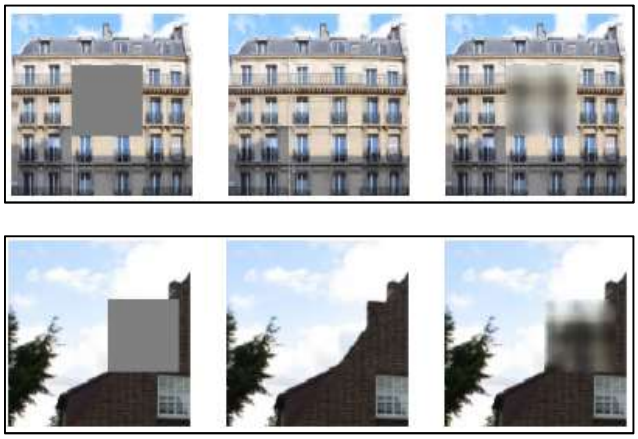


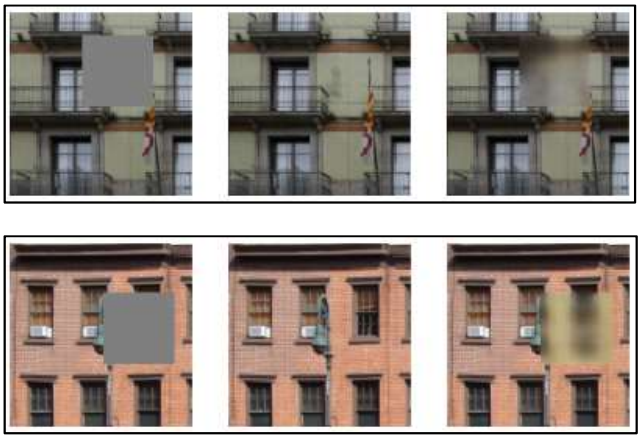
150th iteration

230th iteration



Test Examples (there are better/worse predictions that can be found in “random-crop-run/Results” folder):





M. Arda Düzceker
18-950-485
ardad@student.ethz.ch

Task 3.5

- What other method can you think of to prevent overfitting?
  - Although they are enabled during the runs that provided the results above, dropout layers (spatial dropout for 2D feature maps, standard dropout for 1D (1 x 1 x 400) feature map) are already added to the network as another method to prevent overfitting.
  - Another method can be random augmentation of the input data which is again implemented and actually utilized during the trainings. This method increases the diversity in the dataset which makes it harder for the network to overfit.

- What is the purpose of kernel regularization?
  - In very general machine learning sense, regularization penalizes the coefficients of the estimator by adding terms (e.g. L1-norm or L2-norm of the coefficients) to the cost function which forces it to produce a simpler model. In neural networks and deep learning, similarly, kernel regularization penalizes the weight matrices of the nodes. It is one of the most common methods used in order to prevent overfitting.

- What other form of preprocessing could we perform on the data to make training more stable?
  - Currently we are scaling the input images between 0.0 and 1.0 without considering the underlying true distribution of the data. Another commonly used method is zero mean, unit variance scaling (normalization) since it is known that machine learning algorithms may greatly benefit from working on the normalized data.

References

[1] Deepak Pathak, Philipp Krahenbühl, Jeff Donahue, Trevor Darrell, and Alexei Efros. Context encoders: Feature learning by inpainting. 2016.