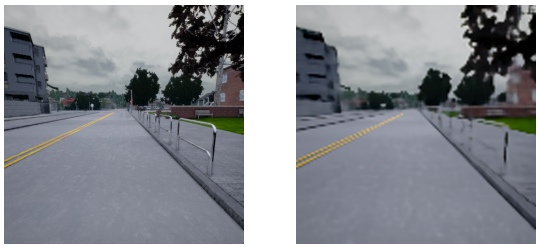
# Model Pruning: Weekly Report 8

Patricia Gschoßmann

## 1. Weekly Progress

In this week a new RGB autoencoder with a bottleneck of size $16 \times 16 \times 128$ was pruned to reduce 65% of its encoder's parameters. The original model's results can be seen in fig. 1b.



(a) Original image (b) Original model output

Figure 1: Original and reconstructed image data

## 2. Results

### 2.1. Addendum to last week

Last week's model reached a validation loss of 2.57 at $\alpha = 0$. The training was resumed repeatedly with different learning rates (0.01 and 0.1). Unfortunately, this did not improve the performance.

### 2.2. This week's results

In contrast to last week, only the encoder was pruned in this week, as this affects the latent space. The decoder then needs to be trained accordingly - this happens during the pruning process. Moreover, each convolutional layer in the encoder was pruned, regardless of whether a transposed convolution follows[1], leading to a reduced bottleneck.
The same setup as last week was used: Pruning with weight updates of the pretrained model and a learning rate of 0.001 were used, as well as a step scheduler, which reduces $\alpha$ by 0.1 at each iteration. The original model reached a minimum validation loss of 1.40.

For the validation development at each $\alpha$-decay the same behavior as last week could be observed: Except for $\alpha = 0$, the validation loss was the highest for the model with $\alpha = 0.9$ with $val\_loss = 2.12$. With each $\alpha$-decay, the performance improved.
The model reached a validation loss of 2.05 at $\alpha = 0.1$. At the first iteration for $\alpha = 0$ it only reached a validation loss of 2.53. The training of the model was resumed with a learning rate of 0.01, however, the performance improved only slightly (2.48). Corresponding validation loss curves can be seen in fig. 2. Intermediate outputs for $\alpha = 0.1$ and $\alpha = 0$ before pruning are shown in figure 3. The differences are clearly visible. However, the silhouettes of the objects in the image are somehow still visible.

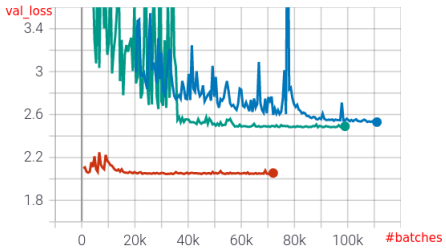



Figure 2: Validation loss development for $\alpha = 0.1$ (red), $\alpha = 0$ with $lr = 0.001$ (blue) and the resumed training for $\alpha = 0$ with $lr = 0.01$ (green).

(a) Output at $\alpha = 0.1$ (b) Output at $\alpha = 0$

Figure 3: Reconstructed image data during the pruning process. The original image is depicted in fig. 1a.

[1] This only applies to the last convolution anyway

## 3. Plan

- Improve performance for $\alpha = 0$ with learning rate approach (increase learning rate to 0.1) and smaller $\alpha$-decays
- Prune encoder correctly
- If successfull: Start pruning corresponding decoder