

Optional Assignment

Advanced Topics in Neural Networks

Students under 30 points will receive twice the points for this homework

Check Lab09/main.py. You must train a model which receives as input 3x32x32 images and outputs 1x28x28 grayscale horizontally and vertically flipped images (see the example). Your model is expected to be faster than doing the ground truth transformations sequentially on CPU. Your model is not expected to learn to do the transformation perfectly, but you are still required to minimize the loss by training.

1. Create a model that is able to do the transformations. The input of the model is represented by the first element of the tuple returned by the "CustomDataset", while the output is the second element of the tuple. Be creative with your model (and explain your creativity in the LaTeX document), but don't make the model too large or too complex. The simplest model might work the best. (3 points)
2. Use a suitable loss function for training. Motivate your choice in the LaTeX document. (2 points)
3. Use early stopping as a stopping criteria for the training. Motivate your early stopping choice in the LaTeX document. (2 points)
4. After training the model, include at least 5 images generated by your model in the LaTeX document and compare them with the ground truth. (1 point)
5. Benchmark the model using the "test_inference_time" with different parameters. Find on each device (CPU and MPS/GPU) when is the model faster than the real transformations. (2 points)

You don't need GPU and Google Colab for training, limit yourself to a small model that can be trained on CPU. You may still use GPU and Google Colab, especially for inference. Your training pipeline must run both on CPU and GPU, using a device parameter.

Send your homework on Discord before Christmas (New year is also accepted in special circumstances). In your submission, you must include:

- A README.md file in which you summarize what you did and the expected number of points.
- A small LaTeX document in which you detail your approach, the model you used, the loss function, and your inference results (comparing the model output with the ground truth). Include information about the runtime performance and pictures with the Tensorboard or Wandb logs.
- The weights of your model and an "inference.py" file in which you load the weights of the model and run the "test_inference_time" function with different parameters.