Deep Learning - Exercise 1

Implicit Neural Representations of Images

Overview:

Neural networks can be used for *signal representation*, in many domains. These approaches have several advantages over other traditional lossy methods. Examples of this concept are presented by Bricman et al. for representing single images, and similarly by Sitzmann et al. using sinusoidal activation functions for this task.

Goal:

The goal of this exercise is to extend these single image encoding paradigms, to *jointly* encode *N* images using a neural network (NN). To be clear, all the images should be encoded using *the same* NN. Based on these previous approaches, you will have to devise a new solution to make this happen.

Using this exercise, we would like to learn about how you think about computational problems, how you creatively solve novel problems using DL, and how you write code in a DL framework. In order to get to know you (and only you) better, *please complete this assignment by yourself.*

Data:

a. The data for this assignment can be found here: https://drive.google.com/drive/folders/1AD1GM6rVj60dxSb9qZAhHsoucSopNBKF?usp=sharing (you should have received access, if not please let us know).
b. It consists of 100 images of icons in (png format), in several different resolutions (24 - 256 pixels).
c. Start this exercise with the 48x48x3 resolution images.

General Instructions:

1. *Code:*
   a. You are highly encouraged to use the code published by Sitzmann et al. (make sure to download the *code* and *Google-colab notebook*). You may write your own version if you prefer, but please keep in mind that Google-Colab enables you to run the code on Google's GPU cloud service for free.
2. *Architecture:*
   a. Please select an appropriate network architecture.
   b. This architecture must converge during training, using your code and the data.
3. *Inputs:*
   a. Following the references (mentioned above), the network inputs must include the *image coordinates* and may include additional inputs.
4. *Performance:*
   a. This exercise is about demonstrating your ML methodology, and not about achieving the best results. Please do not spend too much time on performance tweaking!

Questions:

1. *Image representation* - Build a network that can represent *all* images of the dataset.
    a. Demonstrate the resulting network's generalization capability. You only need to test generalization using data from this dataset (i.e. generalization to unseen images is not required for this exercise):
        i. Define what *generalization* means in this specific task. Are there several kinds?
        ii. Track the network's generalization during training.

2. *Image interpolation* - *Generate new images.*
    a. First, demonstrate the networks ability to upsample the images.
        i. Upsample the images to a resolution of 256x256x3.
    b. Next, interpolate between selected pairs of images:
        i. Consider known similarity measure between representations, and *describe* one that you think is helpful for the following.
        ii. Select (three) pairs of images which will be "best" for interpolating between the images (in each pair).
        iii. Demonstrate interpolation between the two images in each pair selected. Repeat this for the three pairs of images.
        iv. Describe any shortcomings of your results, and hypothesize about why they occur.

3. *Improved image interpolation* - If your above result does not provide satisfactory interpolation results, design an improved solution.
    a. Design an improved solution to the image interpolation problem.
    b. Describe all the details of your solution, however no need to implement it.
    c. Explain why you hypothesize that this solution will provide a refined result.

Deliverables:

a. Your complete code, in a form we can run easily.
b. Instructions how to run your project.
c. Answers to all the questions (mentioned above), and a short summary of your results.
d. All deliverables should be uploaded to a Git repository.