Visual Computing & Artificial Intelligence
Department of Informatics
Technical University of Munich

Esolution
Place student sticker here

Note:

- During the attendance check a sticker containing a unique code will be put on this exam.
- This code contains a unique number that associates this exam with your registration number.
- This number is printed both next to the code and to the signature field in the attendance check list.

Introduction to Deep Learning

Exam: IN2346 / endterm
Date: Tuesday 1st August, 2023
Examiner: Prof. Dr. Matthias Nießner
Time: 08:00 – 09:30

	P 1	P 2	P 3	P 4	P 5	P 6	P 7
I							

Working instructions

- This exam consists of **24 pages** with a total of **7 problems**.
  Please make sure now that you received a complete copy of the exam.
- The total amount of achievable credits in this exam is 90 credits.
- Detaching pages from the exam is prohibited.
- **Answers are only accepted if the solution approach is documented.** Give a reason for each answer unless explicitly stated otherwise in the respective subproblem.
- Do not write with red or green colors nor use pencils.
- Do not write outside of the answer boxes, because this area might get cut off during scanning.
- In case you are running out of space for a question, use the additional pages at the end of the exam. Make sure to indicate that you used the additional pages and to which question it belongs.
- Physically turn off all electronic devices, put them into your bag and close the bag.

Left room from __________ to __________ / Early submission at __________

Problem 1 Multiple Choice (18 credits)

Mark correct answers with a cross ☒
To undo a cross, completely fill out the answer option ■
To re-mark an option, use a human-readable marking ×■

Please note:

- For all multiple choice questions any number of answers, i.e. either zero (!), one or multiple answers can be correct.
- **For each question, you'll receive 2 points if all boxes are answered correctly (i.e. correct answers are checked, wrong answers are not checked) and 0 otherwise.**

1.1 Which of the following statements are true about Autoencoders?

□ Autoencoders are only used for image-based applications and cannot be applied to other data types.

□ Autoencoders can be used for dimensionality reduction.

□ Autoencoders require labeled training data to learn meaningful representations.

□ Autoencoders can be trained without a loss function.

Important!

This is NOT a vector, but the samples in the datasets are scalars!

1.2 In the context of recurrent neural networks (RNNs), which of the following statements are true? Select all that apply.

□ RNNs can process sequential data of variable length.

□ RNNs have a hidden state that allows information to persist across time steps.

□ Long Short-Term Memory (LSTM) is an RNN variant that addresses the vanishing gradient problem.

□ RNNs use different weight matrices for each cell.

1.3 In the context of Transformers, which of the following statements are true? Select all that apply.

□ Transformers utilize self-attention mechanisms to capture relationships between different tokens in the input sequence.

□ Transformers rely on recurrent neural networks (RNNs) to model sequential dependencies.

□ The encoder-decoder architecture in Transformers is commonly used for machine translation tasks.

□ Transformers do not require any positional encoding as they inherently understand the order of tokens in a sequence.

1.4 Which of the following statements are true about Batch Normalization?

□ Batch Normalization speeds up the processing time of a single batch.

□ Batch Normalization accelerates the training process in most cases.

□ Batch Normalization layers are always skipped at test time.

□ Batch Normalization reduces the impact of poor weight initialization.

1.5 Which of the following statements are true about loss function and activation in a neural network?

□ The activation function is responsible for handling class imbalances in the dataset.

□ Activations introduce non-linearity to the neural network, enabling it to learn complex patterns.

□ The primary purpose of a loss function is to calculate the accuracy of the model's predictions.

It's to guide the weights through backprop

□ Removed option

1.6 Which of the following statements are true about Semantic Segmentation?

- □ Semantic segmentation is an unsupervised learning technique that does not require labeled training data.
- □ Semantic segmentation is the process of classifying images into broad categories, such as "dog," "cat," or "car," without pixel-level precision.
- □ A single semantic segmentation architecture can be trained for different multi-class segmentation tasks, each with an arbitrary number of classes, without any change to the architecture.
- □ The precision of an image semantic segmentation model can be influenced by variations in lighting conditions and image quality.

1.7 In the context of the input data, check all that is true:

- □ Random rotations with an angle in $[-30^\circ, 30^\circ]$ are appropriate data augmentation techniques for the MNIST digit dataset.
- □ CNNs are not suitable for inputs with high-dimensional input channels.
- □ Training on image inputs with values in the range of $[0, 255]$, compared to $[0, 1]$, enhances classification performance as the input data spans a higher value range.
- □ Re-scaling the input images from the range of $[0, 255]$ to $[0, 1]$ could mitigate overfitting issues.

1.8 Which of the following statements are true about data augmentation in a supervised learning setup?

- □ Data augmentation is a technique used to generate additional data for training.
- □ Data augmentation can help reduce overfitting by introducing variations in the training data.
- □ Data augmentation is only applicable to image data and not relevant for other types of data.
- □ Data augmentation should only be applied to the testing set to evaluate model generalization.

1.9 Which of the following statements are true about dropout regularization in neural networks?

- □ Dropout regularization is a technique used to add noise to the input data during training.
- □ Dropout regularization helps prevent overfitting by randomly setting a fraction of the neurons in the output of a layer to zero during training.
- □ Dropout regularization is only applicable to convolutional neural networks (CNNs) and not relevant to other types of neural networks.
- □ Dropout regularization helps prevent neural networks from memorizing specific sample(s) and promotes the learning of more generic features.

Problem 2 **Short Questions (15 credits)**

0 □ □ 1 □ □ 2 □ □ 3 □

2.1 You are training a network to classify between [dog, cat, monkey] with a multiclass cross-entropy loss. Consider the two proposed outputs of the network, after the application of the softmax function:

$$\hat{y}_1 = [0.4, 0.3, 0.3]^\top, \hat{y}_2 = [0.4, 0.5, 0.1]^\top$$

- What is the predicted animal class for $\hat{y}_1$ (0.5p) and for $\hat{y}_2$ (0.5p)?
- Assume both inputs are assigned a ground-truth label 'dog', i.e., $y = [1, 0, 0]$. How would the value of the loss function differ between $CE(\hat{y}_1, y)$ and $CE(\hat{y}_2, y)$? Explain your answer. (2p)

0 □ □ 1 □ □ 2 □

2.2 In the context of the previous question, you decide to drop the softmax activation before you apply the Cross-Entropy loss, to save some calculations. The resulting logits and ground truth one-hot-encoding vector are then:

$$\begin{aligned} \hat{y}_1 &= [-1.0, -1.3, -1.3] \\ y &= [1, 0, 0] \end{aligned} \tag{2.1}$$

Explain why we must apply a softmax-like activation before the usage of the Cross-Entropy loss function, and what problem will occur in this specific case.

0 □ □ 1 □ □ 2 □

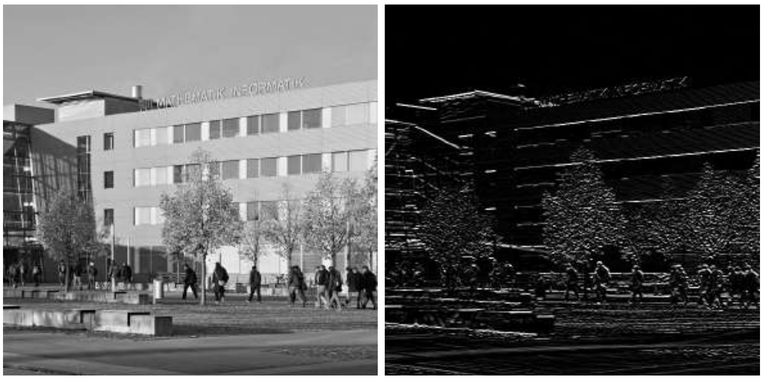
2.3 Explain the purpose and benefits of performing the "bias correction" step in the Adam optimizer algorithm. Consider the equations:

$$m_{k+1} = \beta_1 \cdot m_k + (1 - \beta_1) \cdot \nabla \mathrm{L}(x, \theta)$$

$$v_{k+1} = \beta_2 \cdot v_k + (1 - \beta_2) \cdot [\nabla \mathrm{L}(x, \theta) \circ \nabla \mathrm{L}(x, \theta)]$$

2.4 Given the following grayscale image (left), a 3x3 convolutional filter produces the right output image. Write down the 3x3 filter values and precisely name the operation the filter performs.

□ 0
□
□ 1
□
□ 2



0 □
□
1 □
□
2 □

2.5 Deep Learning has huge potential. However, there are also potential risks. Name and explain 2 risks associated with deep learning specifically **in the context of autonomous driving** and propose solutions to mitigate these risks.

2.6 During training, we usually feed the network with batches of inputs rather than a single input at a time. Give two advantages of using a batch of inputs during training over a single input at a time. Explain your statements.

□ 0
□
□ 1
□
□ 2

2.7 Find the derivative of the $tanh(x)$ function and express its derivative with the $tanh(x)$ itself:

$$tanh(x) = 2\sigma(2x) - 1$$

where σ is the sigmoid function:

$$\sigma(x) = \frac{1}{1+e^{-x}}$$

□ 0
□
□ 1
□
□ 2

Problem 3 Backpropagation (15 credits)

You are training a 1-layer neural network to fit the following dataset:

Input $x \in \mathbb{R}$	2	3	4
GT Value $y \in \mathbb{R}$	0.5	0.75	1.0

Important!

This is NOT a vector, but the samples in

the datasets are scalars!

You have chosen the following network architecture:

$$\hat{y} = ReLU\left(Wx + b\right)$$

$\hat{y}$ is the output of the network, x is the input, W and b are the trainable network parameters.
In the following, for W and b '=' denotes that all entries of the tensors are assigned the same value.

0 □ □ 1 □ □ 2 □

3.1 What is the dimensionality of the tensor W? How many trainable parameters do we have in total?

0 □ □ 1 □ □ 2 □

3.2 You plan to minimize the following Loss function:

$$\mathcal{L}\left(y, \hat{y}\right) = \left(y - \hat{y}\right)^2$$

Can a loss value of zero be achieved on your dataset? Explain why not, or provide parameters that would achieve that.

0 □ □ 1 □

3.3 You go ahead with the above loss function $\mathcal{L}$, and start training using SGD, a mini-batch size of 1. Consider the following initialization:

$$W_0 = 1, b_0 = 1$$

Perform a forward pass for the third datapoint, $x = 4$. What is the model's predicted value?

0 □ □ 1 □

3.4 What is the value of the loss function for this datapoint?

3.5 We will now perform the backward step. Use the chain rule and the above calculations to calculate the value of the derivatives $\frac{\partial \mathcal{L}}{\partial W}$ and $\frac{\partial \mathcal{L}}{\partial b}$.
(**NOTE**: If you are unsure of your calculations up to this point, you may use $\hat{y} = 2$ to avoid chain errors.)

□ 0 □ □ 1 □ □ 2 □ □ 3 □ □ 4

3.6 We will now perform an SGD update steps to W and b, using a learning rate of $lr = 0.5$. Write the update equation (1p) and calculate the updated values for W_1 and b_1.
(**NOTE**: If you are unsure of your calculations up to this point, you may use $\frac{\partial \mathcal{L}}{\partial W} = 50$ and $\frac{\partial \mathcal{L}}{\partial b} = -10$ to avoid chain errors.)

□ 0 □ □ 1 □ □ 2

3.7 In the next iteration of the training loop, the first datapoint $x = 2$ is chosen. What is the value of the gradient? Using the above values for W_1, b_1, perform the forward pass and calculate the model's **prediction** and the value of the **loss** function. (**NOTE**: If you are unsure of your calculations up to this point, you may use $W_1 = -5, b = 3$ to avoid chain errors.)

□ 0 □ □ 1

0 □
□
1 □

3.8 You let the training loop keep running from this point on for 3 more iterations. You can assume training samples will be selected by their order of appearance in the table (i.e, 2,3,4,2,3,4,...). After the last gradient descent step, write down W_4?

0 □
□
1 □

3.9 What minimal modification could you make to your **network** to improve the training, and what issue would this change address and how? (Assume the same learning rate, initialization and training scheme as previously).

Problem 4 **Optimization (14 credits)**

You work for a gambling company, and with the Boxing World Cup coming up, your job is to create a deep learning model that predicts the outcome of a match between two boxers. You have access to a lot of data collected over the past 10 years from different boxing matches around the world. The model needs to take two vectors as input, one for each boxer, containing information like their age, number of matches, number of wins, weight, height, and more. The goal is to predict which boxer is going to win. You have decided to use the Kaiming initialization for the weights, the ReLU activation, and the SGD optimizer with a learning rate of $1e^{-4}$. Additionally, you use L_2 regularization with $\lambda = 1$.

4.1 How would you construct the input to the model (1p)? What type of architecture fits the task more appropriately - An Autoencoder, a Fully Connected network, or the VGG16 architecture (0.5p)? Explain (1p).

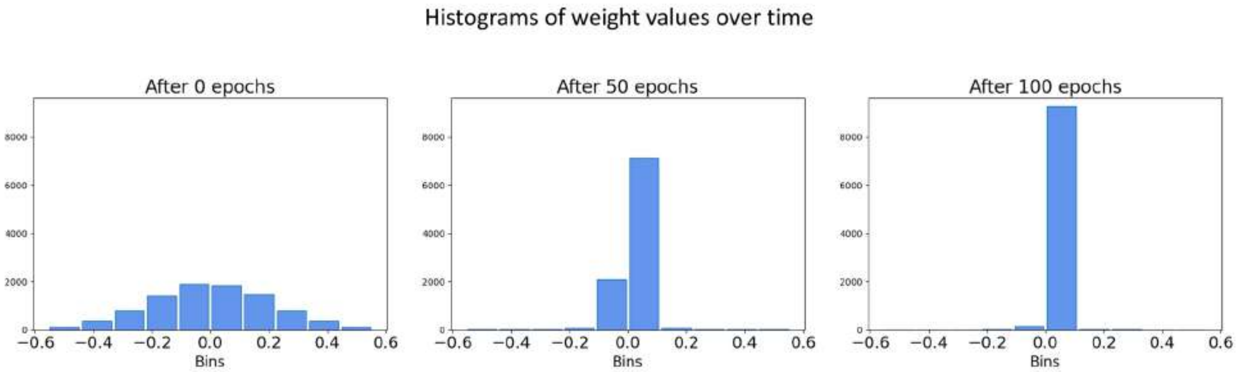
□ 0 □ □ 1 □ □ 2 □

4.2 What is the definition of L_2 regularization and why is it used in general?

□ 0 □ □ 1 □ □ 2

4.3 After a few epochs of training, you decide to visualize the values of the weight matrices as a histogram. It seems the values of your weights are not very well distributed and all close to 0. Explain why?

□ 0 □ □ 1



0 □ □ 1 □ □

4.4 Since it is a binary classification task, you remember that logistic regression utilizes a Sigmoid function as activation. You decide to change all the ReLU activations with Sigmoid activation layers. Which problem does this change to the nextwork introduce, given that nothing else has been changed? How can the problem be mitigated?

0 □ □ 1 □ □ 2 □ □ 3 □

4.5 Explain the vanishing gradient problem (2p). Name 2 activation functions that are known to suffer from this problem (0.5p each).

0 □ □ 1 □ □ 2 □

4.6 What is the main difference between the vanishing gradient and the "Dead ReLU"? Explain why the latter happens, and state the core difference between the two.

4.7 Out of a large dataset containing hundreds of thousands of matches, you have selected the last 50,000 matches as your training set. You realize that the entire training set can be loaded into (V-)RAM and that you can perform mathematical operations with it. Explain how to leverage this fact, to optimize the training process.

□ 0
□
□ 1
□
□ 2

Problem 5 Autoencoder (9.5 credits)

With the knowledge of this course, you are planning to found an AI startup to revolutionize data encryption and compression focusing on company-specific image data. Your idea is to use an Autoencoder-like network architecture, which is trained on the company's data. After the model is converged, only the decoder is distributed once among its clients, and the company can send the encoded, low-dimensional latent vector of the respective data to its clients.

0 □ □ 1 □ □

5.1 Which type of Autoencoder-like architecture is better suited for the task: a vanilla fully-convolutional Autoencoder or a fully-convolutional U-Net architecture? Explain.

0 □ □ 1 □ □ 2 □ □ 3 □

5.2 A company is interested in your technology and wants to give it a try. You train a model on their data, i.e. images of their products. However, they notice that the decoded images are blurry. Name and explain two possible reasons that could cause this problem.

0 □ □ 1 □ □ 2 □ □ 3 □

5.3 After several improvements, the company is convinced of your idea. Also, the company is developing a new product. To make sure employees do not accidentally send encoded images of their new prototype via email to clients, they want to implement an extra feature to detect and prevent these mistakes.

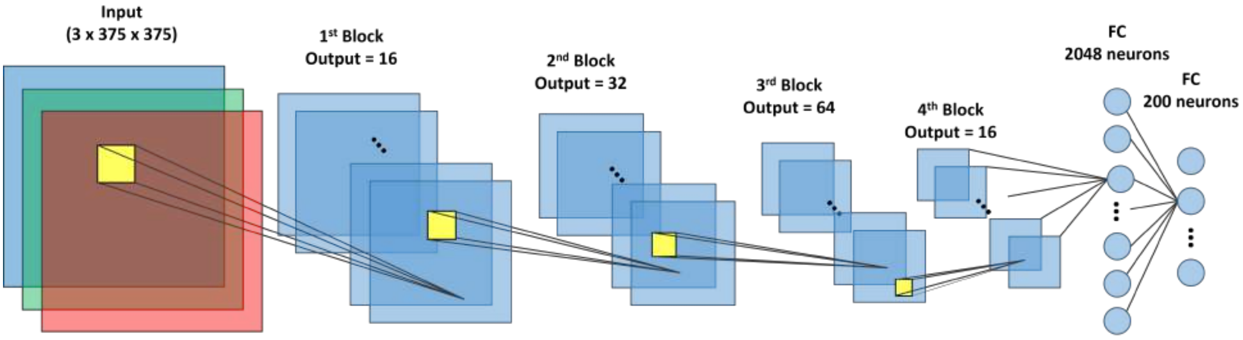
Since there are not yet a lot of images of their prototype to train an image classifier, explain how you might use your previous model to get a usable model given the huge amount of other product images available. Also, include in your answer the loss function and the number of neurons in the output.

5.4 The Lead Product Designer of the company approaches you and would like to know if the approach can also be used to produce images of new product ideas which are similar to the company's previous products. She heard of an architecture called "Variational Autoencoder". Explain why a VAE is better suited for such task (1p). What changes to the architecture are needed (0.5p), and what is the loss function (0.5p)?

□ 0
□
□ 1
□
□ 2

Problem 6 CNNs (8.5 credits)

In the following, we assume that the input of our network is a (3 × 375 × 375) RGB image. The task is to perform image classification on 200 classes. You design a network with the following structure [CONV - ReLU - MaxPool] x 4 - FC - ReLU - FC. Each convolutional layer has a fixed kernel size 3 × 3, stride of 1 and no padding. The output channels of these four convolutional layers are 16, 32, 64, and 16. The max-pooling layers have a window size of 2 × 2 and a stride of 2.



Block = [Conv – ReLU – MaxPool]

0 □ □ 1 □ □ 2 □

6.1 What is the number of parameters in the **first** Convolutional layer of the network (Consider bias term)?

0 □ □ 1 □ □ 2 □ □

6.2 Write down the receptive field of a neuron after the **3rd** convolutional layer with respect to the input image. Explain the individual steps graphically or by calculation.

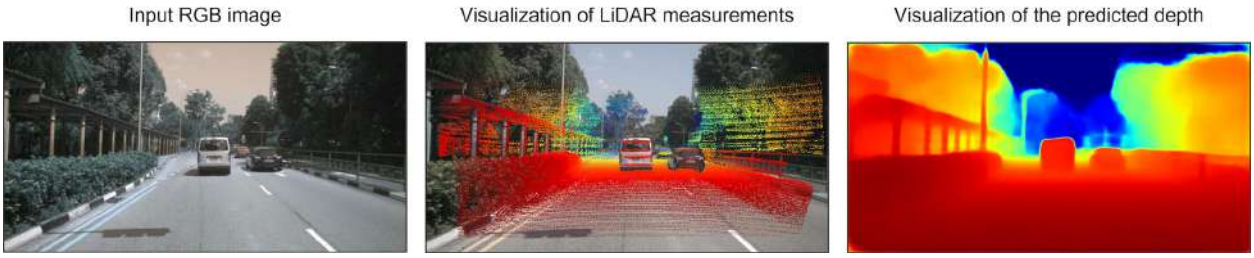
6.3 What is the output size of the network after the **4th** Pooling layer? Write down all steps.

□ 0
□
□ 1
□
□ 2

6.4 We want to modify the network to be fully convolutional. State the shapes of the new weight matrices.

□ 0
□
□ 1
□
□ 2

Problem 7 Data Loading (10 credits)


Input RGB image

Visualization of LiDAR measurements

Visualization of the predicted depth

For autonomous driving, cars are often equipped with expensive LiDAR sensors which capture the distance to its surrounding. One essential task in computer vision is depth prediction from a single RGB image. The ground truth for this supervised task can be acquired from an RGB camera and a LiDAR sensor attached to the car, which is driving around streets.
The recorded data consists of scenes captured in both daylight and nighttime. The dataset is sorted in such a way that all the daytime scenes are stored first, and so the nighttime ones are stored last. Consecutive data samples (⟨RGB image, depth map⟩ pair) may come from the same scene, captured with a gap of 20 milliseconds.

0 □
□
1 □
□

7.1 While loading your training data, you notice the following batch:

$$\begin{bmatrix} \langle \text{sun-0001.jpg}, \text{sun-0001-gt.png}, \rangle \\ \langle \text{sun-0002.jpg}, \text{sun-0002-gt.png}, \rangle \\ \langle \text{sun-0003.jpg}, \text{sun-0003-gt.png}, \rangle \\ \langle \text{sun-0004.jpg}, \text{sun-0004-gt.png} \rangle \end{bmatrix}$$

Where "sun" indicates that the images were taken during daytime. State one problem with this batch, given the proposed dataset, and explain it.

0 □
□
1 □
□

7.2 Explain why it is important to shuffle the dataset before splitting it into the splits (train / val / test)?

Note that before shuffling, everything that is in your dataset - that is the distribution. When splitting it - there might pop up problems of mistmach distributions like day and night time images.

7.3 Let’s assume the dataset consists of very high-resolution images. You set the batch size to 16, but after starting your training script, you get the following error message "RuntimeError: CUDA Out of memory". Name two possible solutions to mitigate memory usage.

□ 0 □ □ 1 □ □ 2

7.4 While loading the data, you desire to apply various transformations: data augmentation and input normalization operation. To which of the splits (train, val, test) do we apply the mentioned transformations? Explain the difference.

□ 0 □ □ 1 □ □ 2

0 □ □ 1 □ □ 2 □

7.5 In common deep learning libraries (e.g. pytorch), the Dataset and Dataloader classes are commonly used for efficient data handling. One of the essential methods of the Dataset class is getitem(). State the input and output of the getitem() method for a supervised learning scenario. Explain the role of the Dataloader class.

0 □ □ 1 □

7.6 You are adding random horizontal flip as data augmentation to your input images. You observe that your model trains much worse than before, i.e. both the training and validation loss remain high. What could be the reason for this behavior?

Additional space for solutions–clearly mark the (sub)problem your answers are related to and strike out invalid solutions.