In the name of God

University of Tehran

Faculty of Electrical and Computer Engineering

Neural Networks and Deep Learning Course

Assignment 3

Question 1	
Designer Assistant Name	Parmis Bathaeian
Email	parmisbathaeiyan@gmail.com
Question 2	
Designer Assistant Name	Reza Barati
Email	Miladbarati2000@gmail.com

Contents

1 Rules **2**

2 Question 1: Urban Scene Segmentation **4**
2.1 Description of the Proposed Model (15 points) 4
2.2 Dataset Preparation (5 points) 4
2.3 Optimizer, Metrics, and Loss Function (15 points) 4
2.4 Model Implementation (20 points) 4
2.5 Model Training (30 points) 4
2.6 Model Evaluation (15 points) 4

3 Question 2: Oriented R-CNN for Object Detection **5**
3.1 Part One: Theoretical Questions (50 points) 5
3.1.1 Conceptual Understanding (10 points) 5
3.1.2 Model Components (15 points) 5
3.1.3 Rotated RoI Align (10 points) 5
3.1.4 Performance and Efficiency (15 points) 5
3.2 Part Two: Practical Implementation (60 points) 6
3.2.1 Environment Setup and Dataset Preparation (15 points) 6
3.2.2 Training the Oriented R-CNN Model (20 points) 6
3.2.3 Evaluation and Result Analysis (15 points) 6
3.2.4 Comparative Analysis and Improvement Suggestions (10 points) . 6

1 Rules

Before answering the questions, please read the following carefully:

- Submit your answers in a report using the format provided on the course page on the Elearn system, named **REPORTS_TEMPLATE.docx**.
- It is recommended to do the assignments in groups of two. (More than two people are not allowed, and individual submissions will not receive extra credit). Note that there is no requirement for group members to remain the same throughout the semester. (i.e., you can do the first assignment with person A and the second assignment with person B, etc.).
- The quality of your report is very important in the grading process; therefore, please include all the points and assumptions you made in your implementations and calculations in the report.
- In your report, according to what is provided in the sample template, include captions for figures and tables.
- You are not required to provide detailed explanations of the code in the report, but you must report and analyze the results obtained from it.
- Analysis of the results is mandatory, even if not explicitly asked for in the question.
- The teaching assistants are not obliged to run your codes; therefore, any results or analysis requested in the questions should be clearly and completely presented in the report. Failure to comply with this will result in a deduction from your assignment grade.
- Codes must be prepared in a notebook with the extension **.ipynb**. At the end of the work, all the code must be run, and the output of each cell must be saved in this submitted file. For example, if the output of a cell is a plot that you have included in your report, this plot must also be present in the notebook of the codes.
- In case of cheating, all involved individuals will receive a score of -100.
- The only authorized programming language is **Python**.
- Using pre-written codes for the assignments is strictly forbidden. If two groups use a common source and submit similar codes, it will be considered cheating.
- The late submission policy is as follows: after the submission deadline, you have a maximum of one week to submit with a penalty. After this one week, your score for that assignment will be zero.
  - First three days: no penalty
  - Fourth day: 5% penalty
  - Fifth day: 10% penalty
  - Sixth day: 15% penalty
  - Seventh day: 20% penalty

- The maximum score for each question is 100. If the total marks for a question exceed 100, and a student scores more than 100, the score will be capped at 100.
  - For example, if the score obtained from question 1 is 105 and the score from question 2 is 95, the final score for the assignment will be 97.5 and not 100.
- Please put the report, codes, and other attachments in a folder with the following name, compress it, and then upload it to the Elearn system:

  HW[Number]_[Lastname]_[StudentNumber]_[Lastname]_[StudentNumber].zip

  (Example: HW1_Ahmadi_810199101_Bagheri_810199102.zip)
- For groups of two, submission by one of the members is sufficient, but it is recommended that both members upload the submission.

2 Question 1: Urban Scene Segmentation

In this exercise, you will work on implementing the Fast SCNN model for the urban scene segmentation problem. The goal of this question is to understand the principles of image segmentation and to use the model introduced in the attached paper.

2.1 Description of the Proposed Model (15 points)

Read the paper and explain the functionality of the proposed model, its architectural structure, and the role of each part in the segmentation process. Compare this model with encoder-decoder models like U-Net in terms of structure and performance.

2.2 Dataset Preparation (5 points)

The dataset selected for this exercise is CamVid. You can download this dataset from this GitHub page. Read the images and display a few samples of the original images along with their masks. Report the number of data in the training and validation sets. Note that the use of data augmentation methods is not mandatory in this exercise.

2.3 Optimizer, Metrics, and Loss Function (15 points)

Explain the two metrics, Dice Coefficient and IoU Score. Implement them yourself and use them along with Accuracy as metrics during the network training. Set the optimizer and loss function according to the paper or by your own choice.

2.4 Model Implementation (20 points)

Implement the Fast-SCNN model according to the paper for the image segmentation task. Also, report the total number of parameters of the created model. In your report, provide a brief explanation of the structure of the main blocks used in the model, namely **Depthwise Separable Convolution**, **Inverted Residual Block**, and **Pyramid Pooling Module**.

2.5 Model Training (30 points)

For training the model, choose the number of epochs and batch size as you wish and report their values along with other hyperparameters, including the loss function, optimizer, and learning rate. At the end, plot and analyze the loss function, accuracy, IoU Score, and Dice Coefficient graphs on the training and validation data. Note that reaching the results of the paper is not necessary, and achieving an IoU Score and Dice Coefficient of 0.5 on the validation data indicates good learning by your model.

2.6 Model Evaluation (15 points)

After training the model, display 10 samples of the evaluation images along with the ground truth mask and the predicted mask. Based on the results obtained on the images, explain the performance of the model.

3 Question 2: Oriented R-CNN for Object Detection

This exercise focuses on the Oriented R-CNN model for object detection, as presented in the paper "Oriented R-CNN for Object Detection". The purpose of this exercise is to enhance your theoretical understanding of advanced oriented object detection methods and provide practical experience in implementing and evaluating deep learning models. The theoretical questions will assess your understanding of the concepts, architectures, and techniques discussed in the paper, while the practical tasks will guide you through setting up the environment, training, evaluating, and analyzing the performance of the Oriented R-CNN framework.

3.1 Part One: Theoretical Questions (50 points)

3.1.1 Conceptual Understanding (10 points)

a. Explain the main motivation behind the development of Oriented R-CNN. What limitations of previous methods does this model address? Provide examples. (5 points)

b. Explain the advantages of using the "midpoint offset" representation compared to traditional bounding box representations. Provide examples to clarify your explanation. (5 points)

3.1.2 Model Components (15 points)

a. Describe the architecture of the Oriented RPN and explain its difference from the traditional RPN. Use a diagram or sketch to clarify your explanation. (7 points)

b. Explain how the loss function in the Oriented RPN is formulated. Clearly define each component of this function and describe their objectives. (8 points)

3.1.3 Rotated RoI Align (10 points)

a. What is the purpose of Rotated RoIAlign? Explain step-by-step how this operation is performed and provide a clear example. (5 points)

b. Explain the potential problems if Rotated RoIAlign is not used. Provide examples or theoretical justifications for your reasoning. (5 points)

3.1.4 Performance and Efficiency (15 points)

a. Explain how Oriented R-CNN achieves high accuracy and efficiency. Specifically refer to the experiments and results presented in the paper. (5 points)

b. Explain the factors affecting the computational efficiency of the Oriented R-CNN framework and specify the role of each of these factors. (5 points)

c. Provide a brief critical analysis comparing Oriented R-CNN with other two-stage oriented detectors mentioned in the paper and highlight its key strengths and weaknesses. (5 points)

3.2 Part Two: Practical Implementation (60 points)

3.2.1 Environment Setup and Dataset Preparation (15 points)

Download and preprocess the HRSC2016 dataset using the provided code. Implement a custom PyTorch Dataset class that loads the data and converts the annotations to the midpoint-offset representation. Provide code snippets and at least three examples of successfully loaded data. (15 points)

```
import kagglehub
dataset_path = kagglehub.dataset_download('weiming97/hrsc2016-ms-dataset')
```

3.2.2 Training the Oriented R-CNN Model (20 points)

a. Train the Oriented R-CNN model (without using MMDetection) with a ResNet-50-FPN backbone according to the paper for 36 epochs. Clearly document your training settings, including the learning rate schedule, optimizer settings, batch size, and any data augmentation used. (10 points)

b. Plot and analyze the loss function graphs during the training process. Provide an interpretation of these graphs and examine convergence patterns, signs of overfitting, and overall training stability. (10 points)

3.2.3 Evaluation and Result Analysis (15 points)

a. Evaluate your trained model by displaying the proposals and Ground Truth boxes on the test set images. Provide at least five images with the model's predictions compared to the Ground Truth. Provide a detailed analysis of each image, including the accuracy of the predictions, errors, and probable reasons for incorrect predictions. (15 points)

3.2.4 Comparative Analysis and Improvement Suggestions (10 points)

a. Compare the performance of your model with the metrics provided in the original paper. Clearly state your results and analyze the potential differences with the paper. Point out the probable reasons for these differences and provide specific and practical suggestions for improving your current implementation. (10 points)