

COMP9517: Computer Vision

2024 Term 2

Group Project Specification

Maximum Marks Achievable: 40

The group project is **worth 40% of the total course mark**.

Project work is in Weeks 6-10 with deliverables due in Week 10.

Deadline for submission is Friday 2 August 2024 18:00:00 AET.

Instructions for online submission will be posted closer to the deadline.

Refer to the separate marking criteria for detailed information on marking.

Introduction

The goal of the group project is to work together with peers in a team of 5 students to solve a computer vision problem and present the solution in both oral and written form.

Group members can meet with their assigned tutors once per week in Weeks 6-10 during the usual consultation session hours to discuss progress and get feedback.

The group project is to be completed by each group separately. Do not copy ideas or any materials from other groups. If you use publicly available methods or software for some of the tasks, these must be properly attributed/referenced. Failing to do so is plagiarism and will be penalised according to UNSW rules described in the Course Outline.

Note that high marks are given only to groups that developed methods not used before for the project task. We do not expect you to develop everything from scratch, but the more you use existing code (which will be checked), the lower the mark. We do expect you to show creativity and build on ideas taught in the course or from computer vision literature.

Description

In order for autonomous vehicles to navigate safely and accurately in natural environments, it is important that they are able to recognise the different types of scenarios and objects they may encounter along the way. For example, a vehicle may need to proceed more cautiously when travelling through sand or mud, or when there are many trees around, than when driving over gravel or asphalt in a clear area, while water must be avoided at all times. Compared to urban environments, perception in natural environments is more challenging, as these generally contain highly irregular and unstructured elements.

The first step toward comprehensive scene understanding is to perform fine-grained semantic segmentation of the images captured by the vehicle's cameras. That is, to assign a label to each and every pixel in the images, indicating to which class that pixel belongs.

Task

The goal of this group project is to develop and compare different computer vision methods for semantic segmentation of images from natural environments.

Dataset

The dataset to be used in the group project is called WildScenes (see links and references at the end of this document). This is a recently released multimodal dataset consisting of five sequences of 2D images recorded with a normal video camera during traversals through two forests: Venman National Park and Karawatha Forest Park, Brisbane, Australia. The dataset also contains 3D point cloud representations of the same scenes recorded using a lidar scanner, but in this group project we will ignore that part of the dataset and use only the 2D images. In total, the dataset has 9,306 images of size 2,016 x 1,512 pixels. Each and every one of the images has been manually annotated.

Methods

Many traditional, machine learning, and deep learning-based computer vision methods could be used for this task. You are challenged to use concepts taught in the course and other techniques from literature to develop your own methods and test their performance. At least two different methods must be developed and tested.

Although we do not expect you to develop everything from scratch, we do expect to see some new combination of methods, or modifications of existing methods, or the use of more state-of-the-art methods that have not been tried before for the given task.

As there are virtually infinitely many possibilities here, it is impossible to give detailed criteria, but as a general guideline, the more you develop things yourself rather than copy straight from elsewhere, the better. In any case, always do cite your sources.

Training

If your methods require training (that is, if you use supervised rather than unsupervised segmentation approaches), you can use the same procedure for splitting the dataset into training, validation, and test subsets, as the creators of the WildScenes dataset. In their paper (see references below) they describe the procedure in detail and provide code for this in their GitHub repository (see references below). The procedure ensures that the training, validation, and test subsets have a uniform class distribution.

Even if your methods do not require training, they may have hyperparameters that you need to fine-tune to get optimal performance. In that case, too, you must use the training set, not

the test set, because using (partly) the same data for both training/fine-tuning and testing leads to biased results that are not representative of actual performance.

Testing

To assess the performance of each of your methods, compare the segmented images quantitatively with the manually annotated (labelled) images by calculating the intersection over union (IoU), also known as the Jaccard similarity coefficient (JSC), for each class and then taking the mean over all classes in the whole test set. Notice that although the annotations contain more classes, only 15 classes are to be used for evaluation (see further details in the supplementary material of the paper referenced below).

Show these quantitative results in your video presentation and written report (see deliverables below). Also show representative examples of successful segmentations as well as examples where your methods failed (no method generally yields perfect results). Give some explanation why you believe your methods failed in these cases.

Furthermore, discuss whether and why your methods performed better or worse than the methods already evaluated by the creators of the WildScenes dataset (as reported in the paper referenced below). And, finally, discuss some potential directions for future research to further improve the segmentation performance for this dataset.

Practicalities

The WildScenes dataset is about 100 GB in total. However, only the 2D images and annotations are needed for this project, which amounts to about 50 GB. Still, this may be challenging in terms of memory usage and computation time if you are planning to use your own laptop or desktop computer for training and testing. To keep computations manageable, you are free to use only a subset of the data, for example 50%, 40%, 30% (again, use a correct splitting procedure to ensure uniform class distributions). Of course, you can expect the performance of your methods to go down accordingly, but as long as you clearly report your approach, this will not negatively impact your project mark.

Deliverables

The deliverables of the group project are 1) a video presentation, 2) a written report, and 3) the code. The deliverables are to be submitted by only one member of the group, on behalf of the whole group (we do not accept submissions from multiple group members). More detailed information on the deliverables:

Video

Each group must prepare a video presentation of at most 10 minutes showing their work. The presentation must start with an introduction of the problem and then explain the used

methods, show the obtained results, and discuss these results as well as ideas for future improvements. For this part of the presentation, use PowerPoint slides to support the narrative. Following this part, the presentation must include a demonstration of the methods/software in action. Of course, some methods may take a long time to compute, so you may record a live demo and then edit it to stay within time.

The entire presentation must be in the form of a video (720p or 1080p MP4 format) of at most 10 minutes (anything beyond that will be ignored). All group members must present (points may be deducted if this is not the case), but it is up to you to decide who presents which part (introduction, methods, results, discussion, demonstration). In order for us to verify that all group members are indeed presenting, each student presenting their part must be visible in a corner of the presentation (live recording, not a static head shot), and when they start presenting, they must mention their name.

Overlaying a webcam recording can be easily done using either the video recording functionality of PowerPoint itself (see for example [this YouTube tutorial](#)) or using other recording software such as [OBS Studio](#), [Camtasia](#), [Adobe Premiere](#), and many others. It is up to you (depending on your preference and experience) which software to use, as long as the final video satisfies the requirements mentioned above.

Also note that video files can easily become quite large (depending on the level of compression used). To avoid storage problems for this course, the video upload limit will be 100 MB per group, which should be more than enough for this type of presentation. If your video file is larger, use tools like [HandBrake](#) to re-encode with higher compression.

The video presentations will be marked offline (there will be no live presentations). If the markers have any concerns or questions about the presented work, they may contact the group members by email for clarification.

Report

Each group must also submit a written report (in [2-column IEEE format](#), max. 10 pages of main text, and any number of references).

The report must be submitted as a PDF file and include:

1. Introduction: Discuss your understanding of the task specification and dataset.
2. Literature Review: Review relevant techniques in literature, along with any necessary background to understand the methods you selected.
3. Methods: Motivate and explain the selection of the methods you implemented, using relevant references and theories where necessary.
4. Experimental Results: Explain the experimental setup you used to test the performance of the developed methods and the results you obtained.
5. Discussion: Provide a discussion of the results and method performance, in particular reasons for any failures of the method (if applicable).
6. Conclusion: Summarise what worked / did not work and recommend future work.

7. References: List the literature references and other resources used in your work. All external sources (including websites) used in the project must be referenced. The references section does not count toward the 10-page limit.

Code

The complete source code of the developed software must be submitted as a ZIP file and, together with the video and report, will be assessed by the markers. Therefore, the submission must include all necessary modules/information to easily run the code. Software that is hard to run or does not produce the demonstrated results will result in deduction of points. The upload limit for the source code (ZIP) plus report (PDF) together will be 100 MB. Note that this upload limit is separate from the video upload limit (also 100 MB).

Plagiarism detection software will be used to screen all submitted materials (reports and source codes). Comparisons will be made not only pairwise between submissions, but also with related assignments in previous years (if applicable) and publicly available materials. See the Course Outline for the UNSW Plagiarism Policy.

Student Contributions

As a group, you are free in how you divide the work among the group members, but all group members must contribute roughly equally to the method development, coding, making the video, and writing the report. For example, it is unacceptable if some group members only prepare the video and report without contributing to the methods and code.

An online survey will be held at the end of term allowing students to anonymously evaluate the relative contributions of their group members to the project. The results will be reported only to the LIC and the Course Administrators, who at their discretion may moderate the final project mark for individual students if there is sufficient evidence that they contributed substantially less than the other group members.

References

Webpage: WildScenes: A Benchmark for 2D and 3D Semantic Segmentation in Natural Environments. CSIRO 2023. <https://csiro-robotics.github.io/WildScenes/>

Paper: K. Vidanapathirana et al. (2023). WildScenes: A Benchmark for 2D and 3D Semantic Segmentation in Large-scale Natural Environments. arXiv:2312.15364. <https://arxiv.org/abs/2312.15364>

Dataset: CSIRO Data Access Portal: WildScenes Dataset. Version 2 (12 June 2024). <https://doi.org/10.25919/5hzc-5p73>

Copyright: UNSW CSE COMP9517 Team. Reproducing, publishing, posting, distributing, or translating this assignment is an infringement of copyright and will be referred to UNSW Student Conduct and Integrity for action.

Released: 28 June 2024