

COMP9517: Computer Vision

2023 Term 2

Group Project Specification

Maximum Marks Achievable: 40

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

Project work is in Weeks 6-10 with a demo and report due in Week 10.
Refer to the separate marking criteria for detailed information on marking.
Submission instructions and a demo schedule will be released later.

Introduction

The goal of the group project is to work together with peers in a team of 4-5 students to solve a computer vision problem and present the solution in both oral and written form.

Each group can meet with their assigned tutors once per week in Weeks 6-9 during the usual consultation session on Fridays 2-3pm 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 we give high marks only to groups who developed something new or tried more state-of-the-art methods not used before for the goal of this project. We do not expect you to develop everything from scratch, but the more you use or build on existing code (which will be checked), the lower the mark. We do expect you to show creativity and build on ideas you have learned in the course or from computer vision literature.

Description

Two important and challenging computer vision tasks are object detection and classification in real-world images or videos. Example applications include surveillance, traffic monitoring, robotics, medical diagnostics, and biology.

In many applications, the large volume and complexity of the data make it impossible for humans to perform accurate, complete, efficient, and reproducible recognition and analysis of the relevant image information, and thus full automation is needed.

The goal of this group project is to develop and evaluate methods for the detection and

classification of animals in wildlife images. Specifically, in this project, we will focus on two types of animals: penguins and turtles. The challenge is to develop methods that can analyse the images accurately and efficiently.

Tasks

Dataset

The dataset to be used in the group project is the Penguins versus Turtles dataset available from Kaggle (see reference at the end of this document). It consists of a training set of 500 images and a validation set of 72 images. Each image contains either a penguin or a turtle, in an arbitrary location, as indicated in the corresponding annotation files.

Detection

The first task is to detect and localize the animal in each image. Specifically, the task is to develop a method that can take any image from the dataset as input and produce a bounding box as output (x_min, y_min, width, height, all in pixels).

It is up to you whether you solve this as a stand-alone task, or whether you first solve the classification task (described next) and then use the predicted class label to inform the detection (as this allows to employ a more dedicated detector for each class), or even whether you somehow solve the two tasks jointly.

Classification

The second task is to classify the animal in each image. Specifically, this task is to develop a method that can take any image from the dataset as input and produce a class label as output (1 = penguin, 2 = turtle).

It is up to you whether you solve this as a stand-alone task, or whether you first solve the detection task (described above) and then use the predicted bounding box to inform the classification (as this allows to focus on the animal and ignore the larger background), or even whether you somehow solve the two tasks jointly.

Methods

Many traditional and/or machine/deep learning-based computer vision methods could be used for these tasks. You are challenged to use concepts taught in the course and other methods from literature to develop your own method and evaluate its performance.

The codes of some popular detection and classification methods are publicly available. You can study them for inspiration, but you should not use them directly (we will check whether you used existing code or not, see the notes above and below).

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

As there are virtually infinitely many possibilities here, it is impossible to give detailed criteria, but as a general guideline, the more you develop 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 detection and classification approaches), you can use the training set (500 images) for this purpose. 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 validation 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

For the testing of your method, you must use the validation set (72 images). To assess the overall performance of the method, calculate and report the following metrics.

Detection performance: For each validation image, calculate the distance between the centre location of the predicted bounding box and the centre location of the corresponding true bounding box (available from the annotation file), and report the mean and standard deviation of the distances over all validation images. Also calculate the intersection over union (IoU) of the predicted bounding box and its corresponding true bounding box for each validation image and report the mean and standard deviation.

Classification performance: For each validation image, use the true class label (available from the annotation file) to determine whether the predicted class label is correct or not, and report the confusion matrix of the classification results. From this, calculate and report the accuracy, precision, recall, and the F1-score of your method.

Show these quantitative scores in your demo and written report (see deliverables below) and also show representative examples of successful detections and classifications as well as examples where your method failed (no method generally yields 100% perfect results). Give some explanation why you believe your method failed in these cases.

Visualisation

In addition to quantitative testing (described above) your method must also show the detection and classification result. That is, for each image, it should not only detect and classify the animal, but also draw its corresponding bounding box and class label onto the image.

Deliverables

The deliverables of the group project are 1) a group video demo and 2) a group report. Both are due in Week 10. More detailed information on the two deliverables:

Video Demo

Each group will 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. This part of the presentation should be in the form of a short PowerPoint slideshow. Following this part, the presentation should 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 cut off). 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 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 be easily 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 reencode with higher compression.

During the scheduled class hours in Week 10, that is Monday 31 July 2023 12-2pm and/or Friday 4 August 2023 1-3pm, the video demos will be shown to the tutors and lecturers, who will mark them and will ask questions about them to the group members. Other students may tune in and ask questions as well. Therefore, all members of each group must be present when their video is shown. A roster will be made and released closer to Week 10, showing when each group is scheduled to present.

Report & Code

Each group will also submit a report (in [2-column IEEE format](#), max. 10 pages of text, and any number of references) along with the source code, before 4 August 2023 18:00:00 AEST.

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 evaluate 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.

The complete source code of the developed software must be submitted as a ZIP file and, together with the 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 (each is 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 similar 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

Kaggle. Penguins Versus Turtles: Image Classification With Bounding Boxes. 2023.
<https://www.kaggle.com/datasets/abymorgan/penguins-vs-turtles>

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: 30 June 2023