COMP3007/COMP4106
Computer Vision Coursework Description

2023-24

1 Introduction

We have looked in class at different kinds of convolutional neural networks (CNNs) and the basics of how they work. We have also seen some simple examples in MATLAB for both classification and segmentation. In this coursework, you will have a chance to research this topic in more detail, and build your own **segmentation** network in MATLAB, to work on an existing dataset.

2 Key dates

Submission deadline of MATLAB code and report: 3pm **7th May 2024**.

More details are given in the *Module Assessment Sheet* in Moodle. *Note* that COMP3007 and COMP4106 have different assessment sheets (COMP4106 has an additional presentation component).

3 Detailed requirements

We will work with the **Oxford Flower Dataset**, 17 class version (see right). This dataset consists of photos of flowers, grouped into categories depending on flower type. It also comes with some segmentation maps of the flower regions.

The dataset for our coursework tasks will be provided locally on Moodle. You are expected to use the provided version of the dataset to complete this coursework.

More details on the background of the dataset is available here:https://www.robots.ox.ac.uk/~vgg/data/flowers/17/index.html

The dataset was first presented in this paper

An early segmentation paper (pre-deep learning!) from the dataset is here.

4 Method

You have two main tasks.

Coding. You must determine how best to arrange, load and, if needed, pre-process the image and segmentation data. Using this data, you will use two approaches to using a CNN:

(1) Re-using an existing segmentation model, re-training as necessary, on the provided dataset.
(2) Building your own CNN model, and training from scratch on the provided data.

Report. You must write up both your methods as a **research paper**, according to a fixed template and set of subheadings (available on Moodle).

Both tasks are detailed below.

Coding

You must complete two tasks involving CNNs.

(1) Use an existing (even pre-trained) CNN.
(2) Develop your own CNN architecture (your own code).

In both cases the task is to **segment** a flower dataset.

Specifically, your CNNs are required to segment the images in **our version** of the dataset on Moodle.

IMPORTANT: In this data, not all classes or images have associated segmentations (more info below)

You will need to consider:

- How to organise the data files
    a. As noted, not all images have annotation labels. Your first task will be to handle this somehow (note, you are not expected to provide new annotations as part of the coursework)
- How to load and preprocess the data
- How to load and use the labelled data
- How to build and train a CNN model, and use/retrain an existing model
- How to evaluate how well it is working

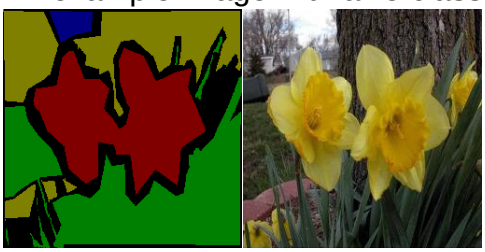
You MUST

1. Implement and train a segmentation network which takes in colour images of size 256x256x3 (ie. RGB)
2. Save the trained network and submit as part of your submission (see submission details below).

Please note:

- The **segmentation groundtruth** exists as colour maps. If you inspect the images you will see that value 0 is null/boundaries, 1 is flower, 3 is background. Some images also have class 2 as leaves and 4 as sky, but these aren't very reliable. **There are only 2 of these classes we care about: flower (1) and background (3)**. So, you can if you like assume any class, except flower, is in a background class; it is flower segmentation we are concerned with here. Your model should produce labels for class 1 and 3.

  - An example image with all 5 classes looks like:



  - There are 1360 (17 classes*80 examples) RGB images in total, but only 846 of them have labelled images for the segmentation task; some classes have no labels at all. So, not every image has a corresponding label file. However, they do have paired names, ie. RGB image *image_1288.jpg* pairs with annotation *image_1288.png*. We recommend you discard or just do not use the unannotated images – how you do this is up to you and should be described in the paper.
  - Note images have already been resized for you, to 256x256
  - If you need to resize images again, it is most desirable to do this in MATLAB in code, but you can use an external tool/pipeline as an alternative. Make sure you describe all your processing in the paper.

Report:

You also need to submit a report that describes your work, written in the style of a scientific paper. Word and Latex templates are provided on Moodle, which is an IEEE standard conference paper format. You need to **follow the template** format in terms of font size and layout (double column).

In the report, you must include the following sections:

- Abstract
- Introduction
- Method
- Evaluation (including a discussion of results)
- Conclusion
- References

The length of the report needs to be a **minimum of 3 pages but no more than 4 pages (References can be in the 5th page)**. Scientific writing tips were introduced in one of the tutorials.

The report needs to be submitted in .pdf format in Moodle.

What to submit

Precise naming is important!

Segmentation – existing model:

One MATLAB file called **`segmentationExist.m`**;

One trained network model file called **`segmentexistnet`** (please name your network variable 'net')

Segmentation – own model:

One MATLAB file called **`segmentationOwn.m`**;

One trained network model file called **`segmentownnet`** (please name your network variable 'net')

Report: One PDF matching the template and page requirements. Name the report file **`report.pdf`**

You will be shown how to save a network model file. When use an existing network for one of the task, you must submit your retrained/adapted version of the model.

Create a zip file where you will include the MATLAB code files, saved model files, and report with the following name: **`YourStudentID_ModuleCode.zip`.** For example, if your ID is 012345678 and you are an undergraduate student, please name the zip file as `012345678_COMP3007.zip`.

There is **no need** to include the dataset in the submitted zip file **IF you haven't changed the structure** of it – if you have (e.g. deleted/renamed files) please include your new data as a ZIP called **`data.zip`** (it will only be about 50mb). Explain the organisation of your new data if you have changed it.

Your .m code must train when run (ie. it must load the datasets provided from subfolders within the working folder: \images_256, \labels_256). Each task must save a trained network file in the working folder, and this must also be submitted as noted earlier.

Don't know where to start?

1. Try the HoG/SVM classification lab and the deep learning labs again
2. Work through some relevant examples in the coursework lab
3. We are asking for you to do two tasks: using and/or fine tuning an existing model, and creating your own network from scratch. If you really can't do one of these, just submit the one approach you can, for partial marks.
4. Remember: most marks are for the write-up
5. Use the final coursework lab before Easter to ask any questions about the coursework (to me, not the lab helpers please). I can't tell you how to do it of course, but can clarify any points you might have.
6. There will be a chance to ask questions again after the Spring break.

Marking

We will use the saved networks to test against some new, unseen images not in the training set.

A note on accuracy. We will take into account network performance on these unseen test images. However, *most* performance marks come from how you evaluate the code in the report. You do not need to push for incredible accuracy levels! An output which visually looks decent will score well, with some marks reserved for notable exceptional performance or more advanced implementation.

5 Marking Criteria

Matlab code – 40%	
Segmentation code functionality – 10%	Does the code run, is it correctly specified according to the formats required (e.g. resolution of image, filenames), does it take in and produce the right kind of data?
Segmentation approach and results (own) – 15%	Mark will consider implementation, pre-preprocessing, complexity of the approach, effectiveness of model on unseen images
Segmentation approach and results (existing) – 15%	Mark will consider implementation, pre-preprocessing, complexity of the approach, effectiveness of model on unseen images
Report – 60%	
Methodology – 20%	Description of **methodology** for both implemented networks.
Evaluation – 15%	**Explanation and presentation** of the results obtained.
Discussion – 15%	**Discussion** of the strengths and weaknesses of the chosen approach and methods; conclusions; future work
Writing style – 10%	**Scientific writing** and clarity, including writing of the introduction part of the paper and consideration of existing approaches.

6 Plagiarism

Copying code or report from other students, from previous students, from any other source, or soliciting code or report from online sources and submitting it as your own is plagiarism and will be penalized as such. FAILING TO ATTRIBUTE a source will result in a mark of zero – and can potentially result in failure of coursework, module or degree. All submissions are checked using both plagiarism detection software and manually for signs of cheating. If you have any doubts, then please ask.

Use of LLMs to assist you in coding and writing the report is forbidden in this task. Hence, comments must adequately explain your code.

We reserve the right to ask students to explain code and/or comment on their report in exceptional circumstances, to demonstrate understanding.