COMP61342 Computer Vision
Module Handbook

Contents

1 Introduction

1.1 Lecture Staff

Terence Morley	terence.morley@manchester.ac.uk
Dr Aphrodite Galata	A.Galata@manchester.ac.uk

1.2 Aim of the Module

The aim of COMP61342 is to give you a grounding in the theoretical and practical aspects of computer vision. Computer vision is a very large field so we cannot cover all topics that you may be aware of. The topics that we will cover are listed in Section 1.3.

Computer vision algorithms rely on the manipulation of images so we will cover some basic image processing topics. Those students with little or no experience of image processing may like to find a small image processing book to read or watch some videos on the internet. We will, however, quickly introduce image processing from the very basics and you can then put the necessary effort into the self-assessed practicals.

The key learning outcome is the ability to apply computer vision techniques, not only to be aware of them. So emphasis is placed on writing software to perform different aspects of computer vision.

At the end of the module, we hope that you will be able recall a number of image processing and computer vision techniques; understand how they can be applied to a particular problem; and to be able to write software to solve the problem.

As mentioned, computer vision is a very large field and this module is just the start of your journey. Section 5.2 lists books, some of which are free to download, that you may like to use during your studies for the module or if you decide to study the subject more deeply in the future.

We hope that you will find the module interesting and enjoyable.

1.3 Content

The unit will cover the following topics:

- Image Processing Refresher
- Segmentation and Grouping
- The Harris Corner Detector
- Local Feature Detection
- The SIFT Feature Detector
- Object Recognition
- Stereo Vision, Epipolar Geometry and Calibration
- Motion Analysis
- Visual Tracking
- Face Detection
- Model-based Vision
- Image Registration

We would like to help you learn the theory of computer vision and also give you the mental tools to be able to solve practical computer vision problems in your future careers. We say 'help you to learn' because we cannot plant information into your head, we can only assist you in your own learning. So please attend the lectures, study the materials, do the practicals and ask for help when you need it.

1.4 Module Presentation

The strange times in which we currently live have required changes to the delivery of the unit. However, policies may change before the unit starts and this description may need to be updated. But at the moment, this is the plan:

Lectures These will be recorded by the lecturers beforehand and be uploaded to the video portal. A link to the videos will be placed on the Blackboard page. They can be viewed at a time to suit to you and they can also be downloaded to your own devices.

Practical Work You will hopefully have downloaded the virtual machine (VM) to your own computer and it should already contain any software that you will need. If not, you are able to install software of your choice on it. You can find the VM and instructions here:

`https://wiki.cs.manchester.ac.uk/index.php/CSImage_VM`

The course software is Python and OpenCV (see below) and this should be used for the practical work. Make sure that you set up an appropriate directory structure to keep your work organised and remember to perform frequent backups.

If you choose not to use the supplied VM, there is some information in Section 4.4 to help you install it on your own computer. Bear in mind that we may not be able to fully support you with this.

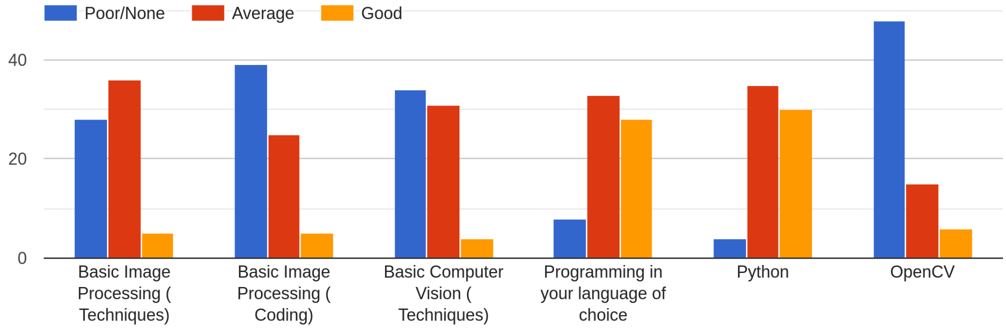
1.5 Participation

Read, write code, experiment and understand.

Helping each other is strongly encouraged. This means explaining things to each other, bouncing ideas off each other and giving pointers to get someone started or giving links to code snippets, papers and information. It does not mean sharing code and passing off somebody's results as your own. You will really learn by working through the concepts and writing, debugging and testing your own code.

1.6 Student Ability

Below are the results of the questionnaire that was performed on 15 January 2021 (69 respondents out of 130 students).



From this you can see the following.

Half of the group have average knowledge of basic image processing and computer vision, while the other half has little knowledge. Therefore, we will make sure that we cover the very basics and allow time to get familiar with image processing and writing software for it. **NOTE** If you are struggling with the programming side at the start of the second week (or before, if you are really having trouble), please ask for help.

Almost all of the group have average or good programming ability in Python. The few that are not so good at Python will be able to get familiar with the basics - we do not need very in-depth knowledge of Python for our purposes.

Few people have got knowledge of OpenCV and things will be introduced gradually. Once you are familiar with Python, it won't be difficult to get to grips with OpenCV.

1.7 Possible Changes

The information in this document and elsewhere is meant as a general guide. As the unit progresses, things may need to be changed. We hope that this will not cause any inconvenience or disappointment and we will strive to keep any changes to a minimum.

2 Schedule – Lecture Timetable

Week 1 – Tuesday 23rd March 2021

Time	Topic	Staff
11.00	Course introduction	TM
11.15	Basic Image Analysis (Part I)	TM
12.15	*Short break*	
12.30	Basic Image Analysis (Part II)	TM
13.30	*Lunch*	
14.00	Region-based Vision	TM
14.50	*Short break*	
15.00	Edge-based Vision	TM
15.50	*Short break*	
16.00	Practical Lab	TAs, TM

Week 2 – Tuesday 13th April 2021

Time	Topic	Staff
09.30	Segmentation and grouping	AG
10.30	*Short break*	
10.45	Harris Corner detector and Local Features (part 1)	AG
12.00	*Lunch*	
13.00	Local Features (part 2) and SIFT	AG
13.50	*Short break*	
14.00	Object Recognition (part 1)	AG
14.45	*Short Break*	
14.55	Object Recognition (part 2)	AG
15.40	*Short Break*	
15.50	Practical Lab	TAs, TM

Week 3 – Tuesday 20th April 2021

Time	Topic	Staff
09.30	Stereo, Epipolar Geometry and Calibration (part 1)	AG
10.30	*Short break*	
10.45	Stereo, Epipolar Geometry and Calibration (part 2)	AG
12.00	*Lunch*	
13.00	Stereo and applications (part 3)	AG
14.00	*Short Break*	
14.10	Practical Lab	TAs, TM
15.15	*Short Break*	
15.30	Practical Lab	TAs, TM

Week 4 – Tuesday 27th April 2021

Time	Topic	Staff
09.30	Motion analysis	AG
10.20	*Short break*	
10.30	Visual Tracking (part 1)	AG
11.20	*Short break*	
11.30	Visual Tracking (part 2)	AG
12.30	*Lunch*	
13.30	Short Lecture: Face detection	TM
14.05	*Short Break*	
14.15	Model-based Vision Part I: From Active Contours to Statistical Shapes	TM
15.10	*Short break*	
15.20	Model-based Vision including AAMs Part II: SSM, ASM and AAM	TM
16.00	Practical Lab	TAs, TM

Week 5 – Tuesday 4th May 2021

Time	Topic	Staff
09.30	Image Registration I: Basics and Pairwise Registration	TM
10.25	*Short Break*	
10.35	Image Registration II: Warps and Regularisation	TM
11.30	*Short break*	
11.40	Image Registration III: Fluids and Groupwise Registration	TM
12.30	*Lunch*	
13.30	Practical Lab	TAs, TM

3 Assessment

For the assessment on the module, there is a practical assignment, a research assignment and an examination.

The practical assignment covers material throughout the course so it will need to be completed in parts as you study the materials. But it will be submitted as a single report.

The research report will give you the opportunity to show your ability to extract and summarise important information by examining a number of published papers that will be provided on Blackboard. **NOTE** You can start reading the papers and taking notes after the first week when you have revised some basic techniques.

Finally, an examination will allow you to demonstrate your understanding of the whole unit.

The assignments will be available in separate documents on Blackboard.

The deadlines and proportion of the final mark for the different assessments is shown below.

Deadline	Assessment	Percentage of Overall Mark
Tuesday 4th May 2021	Assessed practical	25%
Monday 11th May 2021	Assessed research report	25%
TBA (19 May – 9 June 2021)	Final examination	50%

There are also self-assessed, self-study assignments which allow you to experiment with the computer vision techniques covered in the lectures. This gives you the opportunity to really think about how to solve computer vision problems and how you can get the described techniques to work for you.

For the theoretical side, there are weekly self-assessed Blackboard quizzes for you to check your understanding.

NOTE Graduate Teaching Assistants will be available for you to contact during the allocated lab sessions. These will take the form of a Zoom meeting where you will be able to share your screen to discuss your work with the GTAs. If a meeting is already taking place when you need help, you can try again later.

You can also put a query on the Blackboard discussion board, which can be answered by fellow students, GTAs and lecturers.

3.1 Self-Study Assignments

A selection of Python/OpenCV code snippets will be available on Blackboard. These are standalone scripts that perform a particular function. You should examine the source code in these snippets and try them with different data or input images. You may find that they don't work with your choice of image so you can work out how to improve them. You can use the code snippets to build up larger, more sophisticated programs.

Assignments will give you some direction on the sort of things to try but you are also encouraged to try out any other techniques that interest you.

3.2 Assessed Practicals

The assessed practical will be more involved than the self-assessed ones. It will show you that it is not sufficient simply to be aware of computer vision techniques, you also need to know how to apply them.

You are encouraged to discuss how to do things with your friends, but the work should be all yours.

3.3 Examination

For revision, you are advised to attempt these past exam papers. But obviously you can be tested on any of the unit topics so it is not advisable to do selective revision. You will only be assessed on the material covered in the lectures, so don't panic if you see things in these past exam papers that haven't been covered.

- 2018/19: `http://studentnet.cs.manchester.ac.uk/assessment/exam_papers/PG_sem2_2018/COMP61342.pdf`
- 2017/18: `http://studentnet.cs.manchester.ac.uk/assessment/exam_papers/PG_sem2_2017/COMP61342.pdf`
- 2016/17: `http://studentnet.cs.manchester.ac.uk/assessment/exam_papers/PG_sem2_2016/COMP61342.pdf`
- 2015/16: `http://studentnet.cs.manchester.ac.uk/assessment/exam_papers/PG_sem2_2015/COMP61342.pdf`

4 Course Software

4.1 Python v3

Python has become a very popular language and, as a result, it is very well supported with libraries for almost anything you can think of, as well as lots of tutorials, code example and books. It is relatively easy to learn the basics and is an excellent choice for experimentation.

This unit will use the OpenCV (Open Computer Vision) library for Python which is a powerful, extensive library with everything you will need for this unit and much more.

The OpenCV library is written in C++, so operations such as applying an image filter are extremely fast. However, when you write scripts for your custom operations using pixel-level access, you will notice the reduction in speed caused by the overhead of Python making many function calls into OpenCV. In a commercial system, you would be more likely to use C++.

But the good news is that the OpenCV functions are almost identical in Python and C++ so you can easily make the transition to OpenCV with C++. The benefit to us of using Python is the simplicity of the code for rapid development.

If you have C++ experience, there will be an optional self-assessed assignment that will help you to transition from Python to C++ with OpenCV. You will see that the function calls are basically the same, so the assignment will be just to get you started.

4.2 Coding

Please make your code easy to read and understand.

Use a blank line between blocks of code (a block being lines of code that are, together, performing a single purpose).

Consider leaving 2 or 3 blank lines between functions or add a line of 40 or so hashes, for example, as a dividing line. Remember that not all people who read your code will have an IDE that allows for collapsing blocks, and also that you will be submitting your code in a written up document (not as a source file).

Avoid the use of obscure programming constructs. Nobody will thank you for it, neither those marking your work nor people modifying the code you create in your future professional career. If it cannot be avoided easily, add a comment explaining the purpose.

4.3 Documenting Your Code

Add documentation to your code. This will help the person checking your code, as well as helping yourself when you use that code in the future.

Only add as much commenting as you think is necessary. Too many comments don't get read and cause a maintenance issue if you change the section of code that they relate to.

For example, you may have 5 or 6 lines of code to load an image, check if it was read successfully and convert it to greyscale. A comment at the start of the block saying *Load source image as greyscale* is sufficient. It does not require a comment for each line of code.

4.4 Installing Python an OpenCV

If you decide not to use the supplied Student VM, this section gives you some pointers on how to install the appropriate software and libraries. Bear in mind that we may not be able to fully support you with this. This is because we cannot be fully conversant with every operating system, distribution and individual setup.

We will be using **Python Version 3** or above and if you are using a recent Linux distribution that is likely to be installed already.

For other operating systems, you can download and install it from `https://www.python.org/downloads/`.

Open a terminal (or command window) and type

```
python3 --version
```

to check that it installed correctly (if you have got only one Python version installed, you might just need to type `python` rather than `python3`).

Now we need to install the OpenCV libraries for Python. You might only need the first of these, but you might as well install all three:

```
pip3 install opencv-python
pip3 install opencv-contrib-python
pip3 install opencv-contrib-python-nonfree
```

Depending on your Python installation, you may need to use `pip` or `conda` rather than `pip3`.

You can then check that OpenCV is installed properly by entering Python and printing out the version. Type the following in a terminal:

```
python3
import cv2
print(cv2.__version__)
```

Then press `Ctrl-D` to exit Python.

5 Extra Resources

In addition to the materials for the unit that can be downloaded from Blackboard, extra reading is highly recommended for proper understanding.

5.1 Online Resources

The Internet is full of information on Computer Vision, Python and OpenCV: tutorials, code examples, and tips. Some kind authors have even made their books freely available.

Published papers can be downloaded from the University library website but keep in mind that Google Scholar (`https://scholar.google.co.uk/`) gives free access to lots of papers which you can access after you leave the university.

The OpenCV website is `https://opencv.org/`. For documentation go to `https://docs.opencv.org/` and choose your OpenCV version under Doxygen HTML.

You can find the version of OpenCV that you have got installed by typing the following into a Python interactive terminal or a script file:

```
import cv2
print(cv2.__version__)
```

Try these websites for lots of information:

- https://www.learnopencv.com/
- http://homepages.inf.ed.ac.uk/rbf/CVonline/
- https://britishmachinevisionassociation.github.io/

5.2 Books

Recommended books

- Sonka, Hlavac and Boyle, Image Processing, Analysis and Machine Vision, Chapman and Hall, London, 1993.
- Davies, Machine Vision, Theory Algorithms and Practicalities, 2nd Ed. Academic Press, 1997.
- Schalkoff, Digital Image Processing and Computer Vision,John Wiley and Sons Inc. 1989.
- Jain, Kasturi and Schunk, Machine Vision, McGraw and Hill International Editions, 1995.
- Gonzalez and Woods, Digital Image Processing
- C.W.Therrien, Decision Estimation and Classification, John Wiley and Sons, 1989.
- C.M.Bishop, Neural Networks for Pattern Recognition, Oxford University Press, 1995.
- R.J.Barlow, Statistics: A Guide to the Use of Statistical Methods in the Physical Sciences, John Wiley and Sons, 1989.
- Forsythe and Ponce, Computer Vision: a Modern Approach, Prentice Hall, 2003.
- C.M.Bishop, Pattern Recognition and Machine Learning, Springer, 2006.

Free to download books (but please check for any licensing restrictions)

- Richard Szeliski, Computer Vision: Algorithms and Applications

  `http://szeliski.org/Book/2ndEdition.htm`

  `http://szeliski.org/Book/drafts/SzeliskiBook_20100903_draft.pdf`

- Simon J.D. Prince, Computer vision: models, learning and inference

  `http://web4.cs.ucl.ac.uk/staff/s.prince/book/book.pdf`

- Ballard and Brown, Computer Vision

  `http://homepages.inf.ed.ac.uk/rbf/BOOKS/BANDB/bandb.htm`

  `http://homepages.inf.ed.ac.uk/rbf/BOOKS/BANDB/Ballard__D._and_Brown__C._M.__1982__Computer_Vision.pdf`

- Brian Heinold, A Practical Introduction to Python Programming

  `https://www.brianheinold.net/python/A_Practical_Introduction_to_Python_Programming_Heinold.pdf`

- Allen Downey, Think Python – How to Think Like a Computer Scientist

  `http://greenteapress.com/thinkpython2/thinkpython2.pdf`

- Rafael Gonzalez and Richard Woods, Digital Image Processing 2nd Edition

  `http://web.ipac.caltech.edu/staff/fmasci/home/astro_refs/Digital_Image_Processing_2ndEd.pdf`

If you prefer physical books to Acrobat files, keep an eye on the following second-hand book websites where image processing and computer vision books can often be found for as little as three or four pounds (which includes delivery):

- `https://www.awesomebooks.com`
- `https://www.worldofbooks.com`