Topic	Description
Module	Digital Image Processing
Module Code	MECH-B-5-MRV-IMP-ILV
Semester	5
Lecturer	Daniel T. McGuiness, Ph.D
ECTS	5
SWS	4
Lecture Type	ILV
Coursework Name	Group Assignment
Work	Individual
Suggested Private Study	10 hours
Submission Format	Submission via SAKAI
Submission Deadline	26.01.26 17:00
Late Submission	Not accepted
Resubmitting Opportunity	No re submission opportunity

No lecture time is exclusively devoted to the aforementioned assignment.

This is a group assignment which you are to come up with your own project ideas with suggestions given in the document.

Project ideas along with group members should be sent to the lecturer within two weeks after the first lecture. If **NOT** submitted, students will be randomly allocated a project. For any questions, please send your emails to:

Daniel.McGuiness@mci.edu

A portion of the mark for every assignment will be, where applicable, based on style. Style, in this context, refers to organisation, flow, sentence and paragraph structure, typographical accuracy, grammar, spelling, clarity of expression and use of correct IEEE style for citations and references. Students will find *The Elements of Style (3rd ed.)* (1979) by Strunk & White, published by Macmillan, useful with an alternative recommendation being *Economist Style Guide (12th ed.)* by Ann Wroe.

Submission Requirements with Python

Before submitting your work for assessment please read the following requirements.

1. Work submitted using iPython notebooks (`.ipynb`) will be **rejected**.
2. Only work submitted to SAKAI will be considered. Links to private repos and email will **NOT** be considered for grading.
3. Code should be single spaced and use readable font sizes. Avoid using Tabbing.
4. Add spacing around operators if it improves clarity.

> To have a good overview of industrial standards of python programming, please have a look at PEP 8 - Style Guide for Python Code and NumPy Documentation Style Guide.

Commenting

1. The goal of commenting is to make your code easily understandable by someone else.
2. There should be a block comment at the start of your program explaining what it does and including: your name, the assignment and the date.
3. Every function/method should have a block comment explaining what is does.
4. Add inline comments where needed to explain subtle or tricky code.
5. Add blank lines and comments to separate sections of large methods that perform several different tasks.

Naming Variables, Methods, and Classes

1. Take the time to choose meaningful names.
2. Follow naming conventions (i.e., PEP, NumPy Docs).
3. You can use abbreviations, but add comments explaining them.

> Many people comment out old code that wasn't working correctly. Clean up and remove any old code before turning it in.
>
> **If there is any code which is left that contributes no functionality to the operation or the documentation of the code, points will be deducted from the final grade.**

> If there is any suspicion of the submitted code being written using external aids (i.e., AI, web-references) without explicit declaration of exactly where it was used, the code shall be deemed **plagiarised**.

An Example Code with Documentation

An example documentation which is the required standard for the submitted work is as follows:

```python
class ExampleClass(object):
    """The summary line for a class docstring should fit on one line.

    If the class has public attributes, they may be documented here
    in an ``Attributes`` section and follow the same formatting as a
    function's ``Args`` section. Alternatively, attributes may be documented
    inline with the attribute's declaration (see __init__ method below).

    Attributes
    ----------
    attr1 : str
        Description of `attr1`.
    attr2 : :obj:`int`, optional
        Description of `attr2`.
    """

    def __init__(self, param1, param2, param3):
        """Example of docstring on the __init__ method.

        The __init__ method may be documented in either the class level
        docstring, or as a docstring on the __init__ method itself.

        Either form is acceptable, but the two should not be mixed. Choose one
        convention to document the __init__ method and be consistent with it.

        Note
        ----
        Do not include the `self` parameter in the ``Parameters`` section.

        Parameters
        ----------
        param1 : str
            Description of `param1`.
        param2 : :obj:`list` of :obj:`str`
            Description of `param2`. Multiple
            lines are supported.
        param3 : :obj:`int`, optional
            Description of `param3`.
        """
        self.attr1 = param1
        self.attr2 = param2
        self.attr3 = param3  #: Doc comment *inline* with attribute

        #: list of str: Doc comment *before* attribute, with type specified
        self.attr4 = ["attr4"]

        self.attr5 = None
        """str: Docstring *after* attribute, with type specified."""
    def example_method(self, param1, param2):
        """Class methods are similar to regular functions and documented the same.

        Note
        ----
        Do not include the `self` parameter in the ``Parameters`` section.

        Parameters
        ----------
        param1
            The first parameter.
        param2
            The second parameter.

        Returns
        -------
        bool
            True if successful, False otherwise.

        """
        return True
```

Submission Format and Requirements with LaTeX

Before submitting your work for assessment please follow the following requirements.

1. **The work must be done using LaTeX and no other format will be accepted.** You are allowed to use whatever software you wish to write (i.e., Overleaf, TeXStudio, LyX, ...). The work must also conform to the **MCI** Documentation guidelines (which is in SAKAI). A template is provided to you here which was written by the author of this document.

   > If you are to use **mcidoc** template, set the `document-state` as `Report`.

   > To learn LaTeX, the author has tutorials which can be accessed here.

   Please treat this assignment as a great practice to learn LaTeX as you will need it when you are writing your thesis.

2. The report (and all its content) **must** be submitted as a `.pdf` along with all the necessary files (`.tex`, `.sty`) to compile it. Submit one (1) `.zip` file containing everything with the following name pattern.

   `[PROJECT-NAME].zip`

   For example, a group with a project named **Sudoku Detection A** with a would send their submission as:

   `Sudoku_Detection_A.zip`

   and their submission would be in arranged as the following directory:

   ```
   2710704051_Jason_Brigham/
   ├─ tex/
   │   ├─ 2710704051_Jason_Brigham.tex, *.pdf, mcidoc.cls, custom.sty, ...
   │   └─ figures/
   │        └─ .jpg, .png, ...
   └─ HW_Content/
       └─ src/, include/, .cpp, .bash, .py, ...
   ```

   where `...` are any additional requirement (i.e., `.bib`, `.glo`) you may need for TeX and the `HW_Content` is any content which is relevant to the Homework (i.e., if your HW is about python, this is where all the python scripts would be kept, or where your ROS 2 source files would be, or where your C++ codes would be for a project.)

3. The work must be submitted to SAKAI and **no link to personal repo will be accepted.**

> To make sure your submission conform to given requirements, please test out zipping/unzipping and running any code before submission. Any deviations from the requirements will incur penalties to the final grade.

Question	Maximum Point	Received Point
Group Assignment	100	
Sum	100	

[Q1] Group Assignment 100

Introduction For this group assignment, you are to come up with a project, and apply image processing techniques to real-life applications. Some examples are:

Object Measurement
Measuring the length of an object based on a reference object within a given frame.

License Plate Detection
Detection of European license plates and retrieving the text.

Sudoku Recognition
Detection of a sudoku grid and recreating it digitally.

Car Brand Recognition
Detecting the brand of a car based on a given photo.

QR Code Recognition
Detecting a QR code and deciphering the embedded URL.

Barcode Recognition
Detecting bar-code from wider variety of angles.

Chess Board Detection
When given a photo of a chess board, the algorithm should be able to decipher the chess pieces and recreate the output.

As some ideas require non-deterministic methods (i.e., machine learning) you are allowed to use machine learning methods in your group project. However, if used, you have to explain how it works in principle and you may be asked in your Presentation QnA.

> You are allowed to come up with your own project idea, however, prior discussion is needed prior to its acceptance.
>
> Your group should either be 2 or 3 people.
>
> All programming work must be done in Python.

Presentation You are to present your work to the class as a presentation which should **NOT** last longer than 20 minutes. The presentation should be used as an opportunity to showcase your work and should answer the following questions:

- What is the topic?
- What methods are there in industry for tackling the problem?
- Why is it important to tackle?
- What method did you use to tackle the problem?
- What worked and/or what didn't work?
- What can be improved?

In addition to the showcase of your work. For example, if your work is about Sudoku detection, a showcase of real-life detection of a Sudoku grid from a newspaper article should be shown. The last two (2) sessions of the lecture will be allocated for presentations. A group will have 30 minutes of presentation slot which 10 minutes will be left for Questions and Answers (Q&A) related to work and an individual question pertaining to the topics taught at the lecture.

Report You are to write a report in LaTeX on your term-long work which should explain the idea and the execution of your work including:

- An introduction to your work, explaining what it is and why should it be tackled,
- A light literature review on what methods are used in tackling the chosen problem.
- Explanation of the methods used in the tackling of the problem, such as the code used along with proper documentation. For documentation of a function or class, use numpy style documentation.
- Presenting of its results and discussion.

In your report, you are to write who did which part of the work (i.e., Student A did the literature work, Student B wrote the `DetectObject` Class, . . .)

For guidelines for how to submit your work to SAKAI please look at the LaTeX guidelines.

Table 1: Grading of the assignment

Report			Presentation			Sum
Content	Documentation	Style	Presentation	QnA	Content	
30	20	10	20	10	10	100