ENGN 8501 2021 S2

Research Project Guidelines

(Draft version V 0.05 🡪 Will be updated before Week-5.).

About Research Project

One of ENGN8501’s objectives is to provide you with hands-on experiences in solving practical real world problems by using advanced computer vision techniques.

The projects are to be completed in teams (i.e. for training teamworkship), with each team consists of between 2~3 students. However, due to the current covid situation, we do accept sole-member “team” with only 1 student. Tutors will take this into account appropriately when they mark your project outcomes.

If you are work in a real team in doing your project, your teammates will assess your individual contribution to the project via the form of confidential peer-reviews.

1.  Project Instructions

**Steps to take:**

1. Please read this document very carefully.
2. Form a team with 1~3 students per team.   Tutors will shortly explain some possible way to help you to find a team to join in, if you need one.
3. Quickly skim through, or read in detail, all the papers posted on the “WeeklyStudy Plan”.
4. Call for a zoom meeting with all your team members, and discuss actively, and select one or two papers that you as a team agree to work on as your research project.
5. Discuss, select and determine on the project topic that your team will be working on.
6. Write your project proposal (one page PDF in CVPR format) with your team information and project information, and submit it before due.
7. Work on your project (have weekly discussions with your teammates. Attend project Q&A sessions, coding and testing.
8. Submit your final Project Report as a PDF, and source-codes as a single ZIP file, to Wattle.

Note: **Python is the only acceptable main programming language for doing your project.** You may call some basic image I/O functions written in C/C++ or in OpenCV. However, the major part must be in Python. If you are doing deep-learning, then PyTorch is also allowed.

**Total project marks = 50%, which consists of**

* Project proposal (5%) (group submission)
* Project Report with a “confidential peer review report “ (i.e., work-load distribution: for example, who did what, and relative contribution ratio. (20%)  ( individual submission)
* Source code (5%) (group submission).
* 10 minutes seminar presentation (20%) (video recording, group submission) (20%).

2.  Project Proposal Requirement

**Your project proposal** must be no more than 1 page A4 in PDF (with references inclusive), and must contain the following contents:

* Team members (student names, uni IDs)
* Paper Title.
* Project aims and main method.
* Project timeline and tentative work-load plan (i.e., who will be mainly working on which parts/aspects of the project).
* Relevant key bibliographic references.

**Please follow the due date for the project proposal.  Late submission or a late change of project topic will incur a late penalty.**

**Note: You have about 8 weeks to complete your Project (including report writing time).   On average, every week each student is expected to work at least 5 hours on your project (e.g., discussion, studying, and coding, testing, reporting).  For a 2-student team, this amounts to 10-hour worth of effort per week per team.**

3.  Project Report requirement

Submission guide:

* **NOTE, VERY IMPORTANT:  although the term-project is a team project,  and your final project report is largely identical,  your final project report must be submitted individually to Wattle by yourself.**
* **This is because (1) Wattle’s marking system only allows individual marking and (2) More importantly, your report will be slightly different from your teammates’, at least in two sections (a: personal reflection, and b: confidential peer review report.)**

Project Report format:

* Up to 6 pages in CVPR format (excluding references).
* Longer than 6 pages main text will result in a 5-mark deduction.
* You must typeset your Project Report in the provided CVPR Latex template.
* Your report must have a clear, logical technical report structure, and must contain the following sections, just like the paper your project is based on.
* However, you must write your report in your own sentences, and all the words must be your own words.
  + Project Title, Team members (names, stud-ID) , please use an underline to indicate the submitting author, i.e. yourself.
  1. Abstract: a summary of the project
  2. Introduction (include background, motivation, problem statement, and contribution summary)
  3. Problem Statement, definition, and formulation
  4. Method description
  5. Experiments
  6. Conclusion
  7. Reference (papers, and http webpages).
  8. Confidential peer review (on the same page where your personal reflection is.   You may write for example:  “*In doing this project, to the best of my judgement,  I confirm that TeamMember#1 mainly contributed to …, and his/her overall contribution is about 35%, TeamMember#2 mainly worked on …, and his/her contribution is about 40%, and TeamMember#3 was responsible for …, and his/her contribution counts about 25% of the total project workload.”* )
* Your report must be free of errors in pagination, grammar and spelling as much as you can. This will affect your final marks.

4. Source codes requirement

Please read the following sections to appreciate general requirement for your source code/software.

Software (source code)

**1. Code presentation (structures)**

1.1 Programs should be well and thoroughly commented.  Clear, logical module structure [function, comments, references, Appendices, as appropriate]. YES/NO

1.2 Internal layout of the coding is sound - appropriate use of headings and numbering, uncluttered text, tidy coding. YES/NO

1.3 Proper acknowledgements: which parts (modules) are the input from elsewhere, which parts (modules) are the author’s own work, and which parts (modules) are the modifications based on existing modules. YES/NO

1.4. Whether there is **a README file**, the README file should contain a high-level description of each program/module/class developed by a student, as well as a description of how to use the program including calling sequences (e.g., command line arguments), instructions for installing and running the code, shell scripts, acknowledgement of the third party and/or licensed software, datasets, etc. YES/NO

**2 Software Contents**

2.1 Complete: the code can be run correctly and smoothly without broken. YES/NO

2.2 Appropriate level of difficulty evidence of a challenge in the work undertaken; Technical/non- trivial i.e., easy or hard implementations, naive/sophisticated algorithms and/or data structures used, modified algorithm implementation or existing algorithm implementation. YES/NO

2.3 Accuracy/Validity analysis is consistent and thorough; results are verifiable; conclusions are valid reflections of the analysis and results presented. YES/NO

2.4 Whether there is a subset of testing data for the program, whether the testing results are consistent with the project objectives? YES/NO

**Special Note:**

**In some cases, you may be able to Google the internet and find out some sample implementations related to your project. In some cases, the authors themselves may also provide their own implementation. However, you must not simply copy those codes. You must provide clear and complete reference to the source implementation. You must read, understand, and rewrite them, and add your novel contributions.**

**Tutors may run TurnitIn or other text-matching software to automatically detect duplications (potential plagiarism) between your submitted codes and other students’ submitted codes, as well as those available on the Internet.**

5.  Marking Criteria (for your reference)

 Project Report:

* Seems not to have understood the problem. Numerous glaring mistakes. No structure. A token effort. Unclear whether the author knew what this project was about. Painful to read →  FAIL (<50%).
* Competently communicates what was done in the project. Reasonably achieved the basic goals of a basic project. No major conceptual flaws, but lots of room for improvement. Generally a poorly structured, poorly balanced (in terms of content) piece of work → PASS (50%--59%)
* Competently carried out a decent amount of technical work on a non-trivial project. Demonstrated some engineering nous in how things were done. Clearly structured report that conveys the work competently → CREDIT (60%--69%)
* A worthy piece of work. The student has demonstrated an ability to manage, execute and document a significant piece of individual work. A good example of term-project report.   Only flaws in this work are at the detail level. An easy read. → DISTINCTION (70%--79%)
* Something special; A real pleasure to read. A professional piece of work demonstrating obvious mastery of techniques. Almost flawless structure and implementation. [Apparent that the supervisor would have gained a lot from the project.] → HIGH DISTINCTION (80%~90%).
* Award level: already at the level of High Distinction, and more than that.  The proposed method/algorithm is  innovative, original, novel, or creative.  The project report itself is almost ready to be published at a major national or international conference.  The report formatting is already at NeurIPs level. →  OUTSTANDING (>=95%).

Software  (Source code):

How would you grade the overall quality of this artifact/software?

1. Seems not to have understood the problem or implemented the code incorrectly. Numerous glaring mistakes. No structure. A token effort. Unclear whether the author knew what this project was about. Painful to read the coding. FAIL

2. Competently communicates what was done in the project. Reasonably achieved the basic goals of a basic project. No major conceptual flaws, but lots of room for improvement. Generally a poorly structured, poorly balanced (in terms of content) piece of work, without modulation and without any comments. PASS

3. Competently carried out a decent amount of technical work on a non-trivial project. Demonstrated some engineering nous in how things were done. Clearly structured report that conveys the work competently, and the code runs correctly. CREDIT

4. A worthy piece of work. The student has demonstrated an ability to manage, execute and document a significant piece of individual work. A good example of term-projects. Only flaws in this work are at the detail level. An easy read. DISTINCTION

5. Something special; out of the ordinary. A real pleasure to read. A professional piece of work demonstrating obvious mastery of techniques. Flawless structure and almost flawless efficient implementation. All components are properly documented and commented. HIGH DISTINCTION

Seminar Presentation (10 minute video recording):

The following factors could be taken into account in determining the effectiveness of seminar presentations.

These factors are presented as suggestions only, and should be modified according to the particular situation.

**1. Presentation**

• clarity of speech  
• pacing (too fast? too slow? too static? varied?)  
• use of visual aids (images, drawing, animation, videos and other aids)  
• enthusiasm  
• suitability of overall presentation for the size of the group and the topic

**2. Content**

• appropriateness of the level of material  
• familiarity with the topic, including current research findings  
• use of relevant examples and illustrations (where appropriate)  
• appropriateness of the content of visual aids

**3. Organisation and Integration**

• logic and coherence of material  
• balance between different parts of the presentation   
• relevance of all the parts to the whole

Example marking sheets:

Graphical user interface, text, application

Description automatically generated

Another example marking sheet:

Table

Description automatically generated with medium confidence