

---

# **GRP Project Catalogue 2021-2022**

Prof. Dave Towey

School of Computer Science

14th September 2021

## Contents

<b>Introduction</b>	<b>5</b>
[P01] Health analytics – toolset for analysis of vital signs data . . . . .	6
Requirements . . . . .	6
Background . . . . .	6
Outline . . . . .	6
Tasks & Deliverables . . . . .	7
[P02] A Framework for the Competition of Searching Algorithms . . . . .	8
Requirements . . . . .	8
Background . . . . .	8
Outline . . . . .	8
Tasks & Deliverables . . . . .	8
Additional Information . . . . .	9
[P03] Machine Learning Development Checklist Software . . . . .	10
Requirements . . . . .	10
Background . . . . .	10
Outline . . . . .	10
Tasks & Deliverables . . . . .	11
Additional Information . . . . .	11
[P04] A Software Tool for Visualizing and Repairing Logical Contradictions . . . . .	13
Requirements . . . . .	13
Background . . . . .	13
Outline . . . . .	13
Tasks & Deliverables . . . . .	13
Additional Information . . . . .	14
[P05] RoboMaster University AI Challenge Strategy Simulator . . . . .	15
Requirements . . . . .	15
Background . . . . .	15
Outline . . . . .	15
Tasks & Deliverables . . . . .	15
Additional Information . . . . .	16
[P06] A Portable Firefighting On-Site Command Centre System (FOSC) . . . . .	17
Requirements . . . . .	17
Background . . . . .	17
Outline . . . . .	18
Tasks & Deliverables . . . . .	18
Additional Information . . . . .	18

[P07] Cultural Heritage Digitisation: Soul Jars . . . . .	19
Requirements . . . . .	19
Background . . . . .	19
Outline . . . . .	19
Tasks & Deliverables . . . . .	21
Additional Information . . . . .	21
References . . . . .	21
[P08] Unification of University Information Services . . . . .	22
Requirements . . . . .	22
Background . . . . .	22
Outline . . . . .	22
Tasks & Deliverables . . . . .	22
References . . . . .	23
[P09] An external personnel recording system for office building management . . . . .	24
Requirements . . . . .	24
Background . . . . .	24
Outline . . . . .	24
Tasks & Deliverables . . . . .	25
Additional Information . . . . .	25
[P10] iSense: An Interactive Sensory System to Monitor Living Conditions . . . . .	26
Requirements . . . . .	26
Background . . . . .	26
Outline . . . . .	26
Tasks & Deliverables . . . . .	27
Additional Information . . . . .	27
Requirements . . . . .	28
Background . . . . .	28
Outline . . . . .	28
Tasks & Deliverables . . . . .	29
[P12] AI-based Magnetic Nano-grain Images Automatic Detection and Evaluation . . . . .	30
Requirements . . . . .	30
Background . . . . .	30
Outline . . . . .	30
Tasks & Deliverables . . . . .	30
[P13] Quiz App . . . . .	31
Requirements . . . . .	31
Background . . . . .	31
Outline . . . . .	31

Tasks & Deliverables . . . . .	31
[P14] Vision-Based Project Management System for Home Decoration Companies . . . . .	32
Requirements . . . . .	32
Background . . . . .	32
Outline . . . . .	32
Tasks & Deliverables . . . . .	33
[P15] Building a Software to Visualise Compressed Graphs with Possibly Overlapping Super-nodes . . . . .	34
Introduction . . . . .	34
Requirements . . . . .	35
Background . . . . .	35
Outline . . . . .	36
Tasks & Deliverables . . . . .	36
[P16] Production Job Scheduling Software System . . . . .	37
Background . . . . .	37
Requirements . . . . .	37
Tasks & Deliverables . . . . .	37
[P17] Feedback Analysis Based Product Rating . . . . .	38
Requirements . . . . .	38
Background . . . . .	38
Outline . . . . .	38
Tasks & Deliverables . . . . .	38
[P18] Flexible Data Annotation Tool for Machine Learning Techniques . . . . .	39
Requirements . . . . .	39
Background . . . . .	39
Outline . . . . .	39
Tasks & Deliverables . . . . .	39
Additional Information . . . . .	40
[P19] Surveillance Video Analysis Toolbox . . . . .	41
Requirements . . . . .	41
Background . . . . .	41
Outline . . . . .	41
Tasks & Deliverables . . . . .	42
Readings and Resources: . . . . .	42

## **Introduction**

This catalogue of project briefs should be read in conjunction with the GRP handbook, available through Moodle.

**[P01] Health analytics – toolset for analysis of vital signs data****Requirements**

General goals for this project includes:

- Understand and deploy software engineering principles
- Review and implement appropriate software libraries and algorithms
- Develop and implement a software solution
- Write a report

Activities for this project includes:

- Project planning, management, and reporting, including final report
- Implementation of data analysis algorithms
- Implementation and testing of statistical and ML algorithms
- Software development and implementation
- Presentation of work progress

Interactions and supervision

- Regular weekly workshops will be organized for discussion of research topics
- Bi-weekly group/supervisor meeting will be conducted

**Background**

The main theme of this GRP project for 2021/22 is building analytical toolsets for Data Analytics of human heart data sets. Rapid development of mobile sensors and microelectronics have enabled new modalities of heart health care. With the increase of the number of devices and quantities of health monitoring data new solutions are needed for capture, management, and use of these data.

**Outline**

The issues to be addressed include device calibrations and cross calibration, multi sensor data fusion, dealing with Big Data, data heterogeneity, and data safety and privacy. This project focuses on calibration and data fusion of multiple streams of heart related data from various devices (including ECG,

chest strap, smart band, optical finger sensor, and images/videos). The topic specific project goals include:

- Design and implement software for the analysis of calibration experiments of various healthcare devices.
- Implement visualization algorithms (static and dynamic graphing).
- Develop and implement algorithms for analysis of multi-sensor health > data including synchronization and denoising.
- Identify and analyze characteristic patterns in heart health data.

### **Tasks & Deliverables**

Project goals include:

- Data cleaning and quality control of raw data files
- Analysis tools for calibration experiments (pre-processing step 1)
- Analysis tools for time series synchronization (pre-processing step 2)
- Visualization tools for presentation of pre-processed data
- Identification of patterns in pre-processed data

Python is preferred programming language

## **[P02] A Framework for the Competition of Searching Algorithms**

### **Requirements**

This project will require a team who loves Java programming and AI algorithms design.

### **Background**

This project aims to develop a framework for the competition of AI algorithms, with which researchers will be able to compare their algorithms with others. An example of such a framework is HyFlex (Hyper-heuristics Flexible framework), a Java object oriented framework for the implementation and comparison of different iterative general-purpose heuristic search algorithms (also called hyper-heuristics) developed by a group of researchers in the School of Computer Science, UNUK. It can be used to measure the performance of algorithms over several problem domains, rather than just one problem domain. See <http://www.asap.cs.nott.ac.uk/external/chesc2011/index.html> for more details of HyFlex.

### **Outline**

The software developed in this project is expected to contain a user interface, a number of standard problem instances, and an evaluation of algorithms inputted by the users. The structure of the framework can be same as HyFlex. HyFlex contains only 4 test problems. The framework will contain some new problems.

### **Tasks & Deliverables**

The prospective team will work on a framework for the competition of AI algorithms which contains a user interface, a number of standard problem instances, and an evaluation of algorithms inputted by the users.

**The user interface** will allow multiple users to submit their algorithms simultaneously, to choose what problems to solve, and to display the results of evaluations and competitions.

**Extensible problem domains** will contain one or some search problems defined by the stakeholders of this project (some academics and research students at the school of CS, UNNC). It should be easy for future developers to add in more problems.

**Problem instances** are a set of instances of chosen problems for the comparison and competition of algorithms. There are some benchmarks of specific problems available on Internet that the team can



use directly. The team will need to generate new instances under the guide of the supervisor if there is not enough benchmark for the chosen problem domains.

**Structure of Evaluation** will be similar to HyFlex while the evaluation of algorithms depends on what problems are chosen.

### **Additional Information**

This project is 50% implementation and 50% research. After doing the project the students involved will gain not only the experience of software engineering but also the knowledge on solving online optimization problems.

## **[P03] Machine Learning Development Checklist Software**

### **Requirements**

This project requires the development and delivery of a user-friendly application that implements a machine learning development checklist that will assist practitioners and academics implement high-quality and validated machine learning systems.

### **Background**

In many professions there are clear steps to achieve successful results. We know what needs to be done to complete tasks correctly. However, in all professions, errors still occur. A large source of these errors is due to professionals forgetting, overlooking or under-estimating standard procedures. It has been proposed that adopting a checklist procedure is an excellent way to avoid these errors. In his book *Checklist Manifesto*, surgeon Atul Gawande argues about how checklists can be used in many professions. With a special focus on the medical profession, he shows how using checklists can save lives by addressing human fallibility and avoiding simple errors.

In machine learning development there are also standard steps required to get a machine learning system working correctly and to ensure it is validated correctly. Development methodologies such as CRISP-DM have been around for a long time and several authors have proposed systematic approaches to machine learning development. However, these steps are routinely ignored, overlooked or implemented incorrectly. There is room for development of checklist software that will aid academics and professionals in developing good quality machine learning software.

### **Outline**

This project aims to develop a software package to implement a checklist for use as part of machine learning development. Although a checklist could be a simple set of actions that need to be ticked off one by one, in this project more complex actions will need to be expressed in the software:-

- (1) Branches when different checklist actions are required for different types of problem; e.g. the difference between supervised and unsupervised machine learning will require some different validation actions.
- (2) Measurements will need to be entered into the system, related to the machine learning algorithm performance.
- (3) Some actions further along the list may be dependent on responses at previous steps (e.g. matching measured performance against objectives).

(4) Customizable options will need to be accepted to allow for special requirements for different machine learning applications.

(5) Generation of a validation certificate at the final completion stage.

The development of machine learning checklist software will greatly help practitioners and academics working in this area.

### Tasks & Deliverables

***This is a software development task, not a machine learning research project!*** The project can be broken down into several tasks as follows.

<i>Task</i>	<i>Description</i>	<i>Deliverable</i>
Investigation	A great deal of the project is to analyse and decompose what the stages of a successful machine learning project are. For this, the team will need to follow the few standard guidelines available, and interview practitioners and academics.	Record of interview and review of standard practices.  Functional specification.
System design	It is expected that the system will be designed at two levels: (1) a general-purpose checklist system; (2) an instance specifically for machine learning development.	System design document.
Application	The checklist software needs to be implemented as a stand-alone robust application on a Windows PC.	Executable file(s) with test documentation.
Documentation		User manual and supporting documentation.

### Additional Information

For background information, please see:-

- *Checklist Manifesto* by Atul Gawande
- *Data Science for Business* by Tom Fawcett and Foster Provost (2013)

- How to avoid machine learning pitfalls: a guide for academic researchers (2021), Michael A. Lones; [arxiv.org/abs/2108.02497](https://arxiv.org/abs/2108.02497)
- Validate AI; [validateai.org](https://validateai.org)

**[P04] A Software Tool for Visualizing and Repairing Logical Contradictions****Requirements**

This project will require a good understanding of propositional logic and predicate logic.

**Background**

This project is motivated by the development of a reasoner for reasoning about direction relations using axioms in a logic of directions [1]. A reasoner has been implemented using POP-11, the core language of Poplog [2]. However, using the current command line version is difficult to interact with users. A software tool is needed to provide a nice visualisation of logical contradictions detected by the reasoner and help users repair logical contradictions to restore the consistency of input datasets. An existing similar tool is MatchMaps [3, 4].

**Outline**

This project aims to develop a software tool for visualizing and repairing logical contradictions detected by an existing reasoner based on a logic of directions.

**Tasks & Deliverables**

- Provide an expanded description of the problem to be solved.
- Review existing similar software tools and analyse their advantages and limitations, write this up in a chapter of literature review.
- Specify functional and non-functional requirements, and write this up in a chapter of requirement specification.
- Design a software making use of the knowledge and skills learnt in software engineering modules, and write this up in a chapter of software design.
- Implement the software using Java, Python or other programming languages and write this up in a chapter of software implementation.
- Evaluate and test the implemented software making use of the knowledge and skills learnt in software engineering modules, and write this up in a chapter of evaluation and testing.
- Provide a user manual of the software.
- Work in a team, collaborate and learn from each other. Write an individual reflection report.

For more details, please refer to the GRP module handbook.

### **Additional Information**

[1] [\*\* Anonymized by GRP convenor \*\*]

[2] Aaron Sloman (2011) *Teach Primer – An Overview of POP-11*. Available Online at: <https://www.cs.bham.ac.uk/research/projects/poplog/primer/> .

[3] [\*\* Anonymized by GRP convenor \*\*]

[4] [\*\* Anonymized by GRP convenor \*\*]

**[P05] RoboMaster University AI Challenge Strategy Simulator****Requirements**

This project requires the following knowledge and skills:

- General knowledge on simulation skills
- Programming skills (Python, C++)
- General knowledge on machine learning algorithms
- General knowledge on reinforcement learning
- General knowledge on visualization
- General knowledge on software engineering

**Background**

Since 2017, the RoboMaster University AI Challenge has been co-hosted by the DJI RoboMaster Organizing Committee and the IEEE International Conference on Robotics and Automation in Singapore, Australia, and Canada. This competition has attracted a large number of top universities and scientific research institutions around the world to participate in the competition and academic seminars, furthering its international impact in the field of robotics.

In RMUA 2021, automatic robots from both sides (2V2) will shoot against each other in the battlefield that is full of functional organs. Participating teams need to use the official robotics platform to conduct motion planning and control and autonomous decision making by sensing the environmental information of the battlefield. The full-automatic robots will launch projectiles against the enemy's robots. At the end of the game, the team with the highest robot HP wins.

**Outline**

This project aims to develop a simulator that uses AI to choose the right strategy to win the 2v2 game. The simulator will support different AI strategies and allow two sides fight with each other based on the pre-defined rules. The winning strategy can be also deployed on university's robots that compete in actual RM.

**Tasks & Deliverables**

- Specify functional and non-functional requirements.

- Design the simulator that makes use of the knowledge and skills learnt in software engineering modules.
- Implement the simulator using Python or C++ or other programming languages.
- Construct the simulation environment of the game scene based on rules, apply different ML/RL methods to obtain different strategies.
- Test different strategies and analyse their pros and cons.
- Review other universities' open-source strategy, analyse their pros and cons.
- Evaluate and test the simulator that makes use of the knowledge and skills learnt in software engineering modules.
- Provide a user manual of the simulator.
- Work in a team, collaborate and learn from each other.

**Additional Information**

[RoboMaster 2021 University AI Challenge Rules Manual](#)



## [P06] A Portable Firefighting On-Site Command Centre System (FOSC)

### Requirements

FOSC is an on-site system to provide the most comprehensive real-time fireground information in a visual manner for assisting incident commander (IC) who is responsible for managing incident response. The portable design of FOSC integrates on-site firefighters' info, indoor and outdoor video surveillances, maps and building structures to provide real-time updates on fire scene. This project will require developer team to work with firefighter(s) from Ningbo Fire Rescue Detachment for requirements gathering and software specification, along with a Final Year Project (FYP) student from Department of Electrical and Electronic Engineering (EEE) in hardware design and implementation.

### Background

Current on-site fire command center is commonly set up in a vehicle where ICs monitors the fireground situation via multiple screens remotely as illustrated in the left figure below. Unfortunately, the vehicle needs to be parked at least 200 meters away from the fire scene as the size of the vehicle is too large to enter the fire scene. On-site firefighter inspectors also have limited time to perform inspections around the on-fire building before the rescue team arrives. In most cases, the map info of an on-fire building needs to be requested from the command center and then delivers to the vehicle remotely. The printed copies are then delivered to on-site inspectors for rescue planning as depicted in the right figure below. Furthermore, the video surveillance can only be accessed from the vehicle, and info can only be transferred to on-site firefighters verbally. Conclusively, on-site firefighters might not have sufficient data to perform firefighting activities which could increase their safety risk. In worst case scenario, fatality and tragedy could happen.



**Outline**

This project aims to develop a portable on-site command centre system that enables remote connection with drones for outdoor surveillance, remote access to indoor building surveillance cameras, quick access to digital maps, and immediate access to info of all on-site firefighters. In existing fire-fighting framework, most info is inputted manually on handwritten forms by a designated firefighter. One of the key objectives is also to digitalize the forms and develop a flexible method to input the info in the most convenient way possible and to reduce input time. The developer team is also expected to design a user-friendly graphical user interface (GUI) that meets the software requirements and specifications as agreed with the stakeholders.

**Tasks & Deliverables**

Tasks to be accomplished in this project include:

1. Support real-time remote access to multiple drones and display of thermal imaging video cameras' images simultaneously.
2. Support of simulating real-time access and display of indoor building multiple surveillance cameras.
3. Allow keyboard inputs to the digital forms and quick retrieval of forms.
4. (Advanced) Allow smart inputs to the digital forms via voice input, and quick retrieval of information with search engine capability.
5. User-friendly GUI for information display.

The prospective team will be expected to present their final work to the stakeholders (firefighters) few times until completion. The team should also gather software requirements and specification in the beginning of the project and collect feedback from the stakeholders during the implementation for analysis and improvements. The team should also update their work regularly with the stakeholders following the principle of Software Engineering.

**Additional Information**

Team leader/members should consider a plan to assist those member(s) that are not able to return to campus especially during requirements gathering and specification designs, as well as meeting with stakeholders.

## **[P07] Cultural Heritage Digitisation: Soul Jars**

### **Requirements**

A client has asked for help digitising various objects. The selected project team will learn how to use the client's current equipment and software, examine the current system, and explore potential changes or enhancements. In addition to digitising objects, this project also offers an opportunity to engage with a possible oracle problem [1]: how can we be sure of the correctness of the digitised versions of some objects? Other related projects and initiatives will also be explored for potential inclusion or linking.

### **Background**

University of Nottingham Ningbo China (UNNC) has been identified as a fast-growing hub for excellence in research, innovation, and knowledge exchange. Sustainability and meaningful social impact (both domestically and internationally) are at the heart of UNNC's strategy, with some of the UNNC research projects aligned to the United Nations Sustainable Development Goals (SDGs) [2].

Partly due to UNNC's position as a leader in the development and use of technology to preserve cultural heritage, an exciting new opportunity for a team of excellent software engineering students has become available: A number of local stakeholders (including Ningbo Museum) have asked for help to digitise the complete collection of the Ningbo Museum. This includes not only the objects and artifacts in the exhibition, but also those in storage. A special focus will be on the collections of Celadon ceramics (from around Ningbo), especially on a type of funeral vessel called "soul jar" (*hunping*) [3].

Two methods of digitisation will be applied: laser scanning and photogrammetry. This part of the project will initially use the equipment and facilities of the Ningbo Museum and UNNC to laser scan some objects. This should lead the team to develop an algorithm that can be used to digitise all of the Ningbo Museum collections. Supported by the Chinese Government, this project will help make the rich cultural heritage of Ningbo available to a larger national (and international) audience.

Some other ongoing projects (including the development of a virtual UNNC campus) may offer interesting opportunities for this project to align and link with. The project team will study and explore such potential connections (and offer suggestions), as part of this project.

### **Outline**

The selected students will form a software engineering (SE) team, and work together to complete a full cycle (maybe several full cycles) of the SE process, resulting in delivery of a *system*, as requested by the client(s).

The SE team will need to go through a *requirements engineering* process to identify the exact SE project requirements. Part of this will involve applying an appropriate requirements elicitation methodology.

The team will need to make informed decisions about which SE process approaches or methodologies to apply to this project. (Experiences from previous SE teams may prove useful.)

Over the course of the project, the SE team will need to produce several deliverables, including (but not limited to):

- a report on the current situation
- a system requirements specification
- design documentation
- team management and planning documentation
- prototypes
- progress reports
- verification and validation plans
- code
- code documentation
- instructions manual(s)
- suggestions and future plans

The target goal of the SE project will be to deliver, on time, a system (or several systems) that meet the requirements of the various stakeholders.

Tentatively, this system will enable easy digitisation (and quality assurance) of various museum objects.

An informal vision of the system, from the perspective of one stakeholder, has been proposed as follows:

“I hope that the team can help set up a database with digitised Celadon funeral vessels (soul jars) to encourage scholarly debate and research on the topic. I also hope they can create a system that can make this digitisation process easier. Maybe some of these future digital campus things can also interface with the museum, and offer direct virtual museum visits from the virtual campus.”

**Tasks & Deliverables**

This unique project may have several groups of stakeholders, and will require significant project management skills to complete. In addition to the deliverables listed above, the team may also need to produce other resources, including recommended protocols or format specifications, or other advice for future evolution.

**Additional Information**

This project involves real clients, and a very real opportunity to innovate.

**References**

- [1] [https://en.wikipedia.org/wiki/Test\\_oracle](https://en.wikipedia.org/wiki/Test_oracle)
- [2] <https://www.nottingham.edu.cn/en/research-and-business/sustainable-development/index.aspx>
- [3] <https://en.wikipedia.org/wiki/Hunping>

## **[P08] Unification of University Information Services**

### **Requirements**

A recent survey of staff and student experience at University of Nottingham Ningbo China (UNNC) identified user-experience issues in some of the university provided systems. Particularly, respondents noted a lack of unity and cohesion between the various systems (e.g., Moodle, Timetabling, Module Catalogue, Reading Lists, etc) that are used daily by staff and students. This project will seek to produce a web-based system which better integrates these various systems as well as synthesising the information they provide.

### **Background**

The digitalization of university procedures has been a much-needed development at UNNC [1]. Whilst the underlying technological implementation of these systems works well, lesser focus has been given to the experience of using these systems as a part of a typical staff/student workflow. This project seeks to redress this balance.

One suggested area for improvement is the way course and module-level information are accessed. At present, a user is required to consult several disjoint systems, to comprehend the particulars of a module. By following the appropriate SE methodologies, the student team will produce a system which unifies this information in a user-friendly manner.

### **Outline**

Through the application of standard Software Engineering (SE) and Human-Computer Interaction (HCI) methodologies, the selected student team will produce a system which better supports staff/student workflows. To achieve this, the student team will perform a multi-cycle requirement engineering process to identify the requirements of this project.

### **Tasks & Deliverables**

Produce a web-based system, using suitable technologies to better support staff/student workflows at UNNC. The final system must work well on a variety of platforms of various form factors (e.g., laptop, tablet, and phones). It is likely that this project will require the following skillsets:

- Web Scraping
- Automatic Extraction of Hierarchical Relations

- Data Visualisation
- Automatic Summarisation Techniques
- User-Experience Evaluation
- Web-Technologies

In addition to the stated technological deliverables, the selected student team will also be expected to deliver:

- Supporting system documentation
- Code documentation
- Test documentation
- User evaluation report

## **References**

[1] Project Transform - <https://www.nottingham.ac.uk/project-transform/>

**[P09] An external personnel recording system for office building management****Requirements**

The target of this project is to create an AI-based recording and reporting system that collects the external personnel's appearance statistics for a certain office building, and deploy a complete web-based system to present the results to stakeholders, which is the office building management board.

**Background**

In the situation like Covid-19 pandemic, it becomes the top priority to quickly capture the physical occurrence of a certain personnel, in order to figure out the possible spreading trace and connections. In the view of office building management, recording external personnel may be inefficient and costly to fulfil the epidemic prevention requirements. The development of AI, especially neural network-based face recognition and object tracking techniques, makes it possible to resolve the above issues.

**Outline**

The selected students will form a software engineering team, and work together to complete a full cycle (maybe several full cycles) of the software engineering process, resulting in delivery of a prototype AI-based external personnel recording and reporting system and accompanying artefacts, as requested by the client.

The SE team will need to go through a complete requirement-engineering process to identify the exact SE project requirements. Part of this will involve applying an appropriate requirements elicitation methodology. The team will need to make informed decisions about which SE process approaches or methodologies to apply to this project.

Over the course of the project, the SE team will need to produce several artefacts, including (but not limited to):

- A survey on the current pain points of the external personnel recording practices and currently existing methods to resolve the issues
- a requirements specification for the considered system
- initial and iterative design documentation
- team management and planning documentation
- prototypes



- progress reports
- verification and validation plans
- code
- code documentation
- user's manual

### **Tasks & Deliverables**

The goal of this SE project will be to deliver, on time, a prototype system that can intelligently detect and collect external personnel appearance and other necessary statistics taken from the surveillance camera in an office building. Also, the collected data should be appropriately stored in the backend of the system, and the frontend of the system should be excellently built to display and interact with the users about the collected data. In other words, the deliverable should be a proof-of-concept prototype that links the UI, AI engine, and backend to form a working system. Note, this project encourages students to apply AI, while it is not a research topic that improves the efficacy of the AI system. All used dataset should be obtained from public resources.

### **Additional Information**

I. Masi, Y. Wu, T. Hassner and P. Natarajan, "Deep Face Recognition: A Survey," 2018 31st SIBGRAPI Conference on Graphics, Patterns and Images (SIBGRAPI), Parana, 2018, pp. 471-478, doi: 10.1109/SIBGRAPI.2018.00067. [<https://ieeexplore.ieee.org/document/8614364/>]

## **[P10] iSense: An Interactive Sensory System to Monitor Living Conditions**

### **Requirements**

This project requires:

1. Good programming skill
2. Sensory system implementation
3. Data analysis and manipulation

### **Background**

Living conditions have the capacity to impact mental and physical health status, and may result in serious diseases and health issues such as stress, depression, heart attack and headache if they are not properly managed. For example, people may experience depression if they live in dark areas/rooms for long time. For this, the correlation between ambient parameters and mental/physical health factors needs to be continuously monitored and analysed to study the impact of environmental/ambient factors on health conditions. This helps to manage/control environmental factors such as light, noise, temperature and humidity to improve living experience and health condition especially in crowded and large cities.

### **Outline**

This project aims to develop an interactive sensory system to monitor and visualise ambient and body data, and provide the users appropriate recommendations to manage/improve living conditions according to their health status and/or preference. This system has three key components including: a mobile sensory framework (e.g., smart phones or mobile sensors) to collect and report environmental/ambient data and psychological/physiological parameters, a data analysis module to process and visualise the collected data, and an interactive virtual agent (e.g., a game avatar) to provide users relevant recommendations and/or feedbacks. This project includes the following steps:

1. Design and implement a (mobile)sensory data collection framework to collect and report ambient (e.g. indoor temperature, noise and light), environmental (e.g. weather and humidity), biological (e.g. heartrate and blood pressure) and psychological (e.g. brain signals) data elements in real-time.
2. Design and implement an online data analysis module (e.g., web-based and/or android UI) for data storage, processing and visualisation.

3. Develop a (gamified)virtual agent to communicate with the users (i.e., via a Chatbot) and provide them feedback for living condition and health status.

**Tasks & Deliverables**

The project deliverables are outlined as below:

1. Interactive sensory system as described above.
2. User's feedback and evaluation results.
3. Project report according to GRP module requirements

**Additional Information**

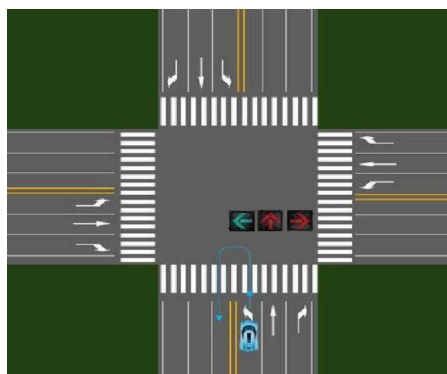
Required sensory devices will be provided.

**[P11] Traffic Control Simulation System****Requirements**

1. Knowledge for data generation and structured data handling
2. Experience for render and display animation framework.
3. Memory and concurrency handling in large scale scenario.

**Background**

Flow control systems are crucial for metropolitan areas to maintain a smooth flow of traffic. Numerous algorithms and strategies have been implemented in these systems to optimize the traffic control approach and alleviate congestion. Regardless of the controlling algorithms used, these systems all follow a similar routine, which includes data collection, data processing, strategic deployment, and evaluation. You will be responsible for developing a traffic light control simulation platform in this project. You will be required to generate real alike traffic data for numerous (four or nine) connected crossroads (as indicated in the figure). You will also be required to simulate controlling traffic lights signals and display various outputs such as traffic flow animation, vehicle, and traffic congestion statistics.



**Figure 1:** Graphical user interface, diagram Description automatically generated

**Outline**

1. Data Preparation: A data unit capable of generating real alike traffic data (potentially from pre-recorded or real-time transmitted data), such as vehicle, lane, and traffic light data, among oth-

ers.

2. Object Motion and Animation Handling: Different objects have their own motion animations and control handlers.
3. Simulation and Display: To demonstrate the results, the data visualization system may need to operate in large-scale traffic scenarios.
4. Evaluation: Unit for evaluating traffic conditions when various traffic signal control techniques are used.
5. Additional: Any function you find appropriate.

### **Tasks & Deliverables**

1. Functional PC-based software with all the functions list in the outline.
2. Report

**[P12] AI-based Magnetic Nano-grain Images Automatic Detection and Evaluation****Requirements**

This Software Engineering (SE) project will require AI-based automatic detection and evaluation for magnetic nano-grain images.

**Background**

The current situation of detection and evaluation for magnetic nano-grain images is manually, which is time consuming and costs manpower, and also depends on the expertise and experience. Accordingly, there is an ever increasing need to develop an automatic detection and evaluation software for magnetic nano-grain images.

**Outline**

This SE project aims to utilize Image Processing and Artificially Intelligence techniques to detect and evaluate magnetic nano-grain images automatically. The selected students will form a software engineering team, and work together to complete a full cycle of the software engineering process, resulting in delivery of a tool.

The SE team will go through a complete requirements of SE process to identify the exact SE project requirements with the Ningbo Magnetic Company. Part of this will involve applying an appropriate requirements elicitation methodology.

The SE team will need to make informed decisions about which SE process approaches or methodologies will apply to this project.

**Tasks & Deliverables**

The prospective SE team will have familiarity with SE process. Over the course of this project, the SE team will need to produce several artefacts, including (but not limited to): a report on the current situation, a system requirements specification, design documentation, team management and planning documentation, prototypes, progress reports, verification and validation plans, code, code documentation, and instructions manual.

**[P13] Quiz App****Requirements**

Good programming skills.

**Background**

There are many quiz software programs available that facilitate interactive teaching. Indeed, many of these programs have a rich set of functionalities. They are all, however, written for a certain set of formats and activities. As a result, they cannot address all possible scenarios.

**Outline**

The aim of this project is to design a software program that may be used easily for simple quizzes in the classroom. These quizzes are typically simple multiple-choice questions created spontaneously during the lecture so that the teacher may obtain a better feedback from students and manage the pace of the teaching based on how students catch up with the flow of the material.

**Tasks & Deliverables**

The intended deliverable is a quiz software program that is:

- Free and open source: This is crucial as there are proprietary programs available which provide rich features (to various degrees). The program should not only be free and open source, but also depend only on free and open source libraries.
- Lightweight: It should not be too demanding on computational resources of the platform.
- Portable: It may be run on all major platforms.
- Easy-to-use: It should be intuitive to use.

The minimal functionality that is required is support for simple multiple-choice types of questions for an interactive teaching environment.

The variety of features that may be incorporated into the final product depends on the productivity of the team members. For instance, one might enhance the application with support for free-response types of questions. This may be augmented with the ability to draw diagrams, or write formulae in LaTeX, although these are not necessary.

## **[P14] Vision-Based Project Management System for Home Decoration Companies**

### **Requirements**

The target of this project is to create a vision enabled project management system for a home decoration company. This system aims to show the project scheduling of a typical home decorating project. It collects daily project schedule and resource status and provides a WeChat-based interface to present the results to stakeholders. The client is the project supervisor who acts as a project manager who himself/herself strives to promote the user-experience (including the project manager, the owner of a house/apartment and other potential users of the system) under a mind of serious scheduling control, and who also would like to promote such system to a wider audience.

### **Background**

Project managers of a home decorate project face some challenges in order to monitor the progress of a project and minimize the completion time of a project. A project consists of a series of tasks, where the precedence relations between tasks are given. Resource required to fulfil each task is known.

In the system, a network fixed project plan (PP) is given. The system considers to give a schedule for all tasks in the PP considering the precedence constraints and resource volume constraints. Accurately recording and tracking each task's completion status is vital for the completion of the entire project. There will be different users in the system with different functionalities and permission. Vision-based techniques will be used to present the status of a project completion for different users.

### **Outline**

The selected students will form a software engineering team, and work together to complete a full cycle (may be several full cycles) of the software engineering process, resulting in delivery of a vision based project scheduling system, as requested by the client.

The system should focus on tracking the status of each task in a project, including the completion status and resources preparation for each task. It should present vividly the status of each project as well as each task in the project to the manager through a WeChat applet. Some statistic results should be exhibited in hierarchical detail levels. Other users, such as the resource provider of a task should be able to see and edit the status of the task. The applet should follow a freemium model for the project managers, but with limited access for other users who may register in the system.



**Tasks & Deliverables**

The team will need to produce several artefacts, including (but not limited to):

- A report on the current main challenges of the home decorating company
- A requirements specification for the targeted system
- Design document
- Prototypes
- Progress reports
- Verification and validation plans
- Code and code document
- Instruction manual book

## [P15] Building a Software to Visualise Compressed Graphs with Possibly Overlapping Supernodes

### Introduction

Graphs with highly connected nodes are widely used in various applications, such as ER-relationship in database structures, individuals' or groups' relationship in social network, and authors' or research groups' collaboration relationship in the research cooperation/collaboration community, the complexity of which makes it difficult to read and digest. Edge compression offers a tool to re-organize the structure of the graphs representing such kinds of relationships through clustering the nodes which shares the similar neighbors in one supernode. See the following figure for an example.

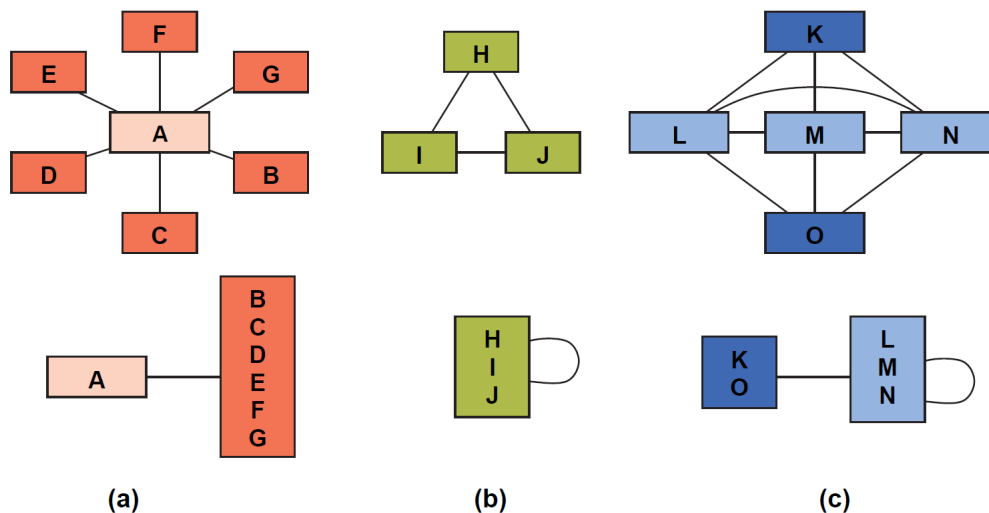
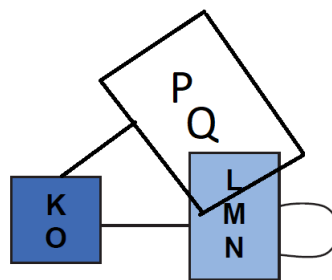


Fig 1. **Edge compression:** Collapsing networks based on identical network neighborhoods. The darker nodes in (a) all have identical neighbor sets  $\{a\}$  and can be collapsed into a single clustered node. Although the nodes in (b) are all structurally interchangeable they have different neighbor sets  $\{i,j\}$ ,  $\{h,j\}$  and  $\{h,i\}$  respectively; we can still merge them if we consider  $N_{self}$  set  $\{h,i,j\}$  instead. The graph in (c) has neighbor sets  $\{l,m,n\}$  for both K and O (dark blue) and  $N_{self}$  set  $\{k,l,m,n,o\}$  (light blue) for L, M and N.

In the figure above, we can see that when edges are compressed, not only the number of edges are reduced, we also obtain some supernodes, which are formed by the groups of nodes in the original graph. Thus the graph obtained by doing the edge compression also has less nodes. In such a way, we could reduce the number of edges and nodes while without losing any connection information in the original graph. In order to maximize the profit brought by the edge compression technique, the graph compression optimisation problems seek for the reduction of a general graph to have minimum or small number of edges or nodes, when it is possible.

As observed, in the subfigure (c) of the above figure, we can see that each supernode is isolated from other supernode. However, in the general situation of the compression results of the input graph, the supernodes are permitted to have overlaps. See the following figure as an example.



**Figure 2:** Overlapping supernodes.

The main target of this project is to create a visualisation software to visualise such resulting compressed graphs with possibly overlapping supernodes.

## Requirements

Develop a software application that visualizes the compressed graphs with possibly overlapping supernodes as introduced in the Introduction section. The group members should use C++ programming language with OpenGL computer graphic library to do the implementation if there is no other special reason mentioned. The software should be able to visualize the compressed graphs such that

- Different supernodes should not have extra overlap except for the originally overlapped parts of the input graphs.
- The intersections of supernodes and edges should be reduced as much as possible.
- The crossing number between edges should be reduced as much as possible.
- The display of the whole graph is a clear visualisation of the content of all the supernodes, all the edges, and the whole structures of the connections of the supernodes.
- Moreover, if possible, the corresponding group could also come up with different visualisation outputs according to different visualisation aesthetic criteria or parameters.

The aim of this project is to develop a visualisation software to visualise the compressed graphs with possibly overlapping supernodes as mentioned in the Introduction section.

## Background

Compressed graphs/diagrams find wide applications in visualization field, social network, biological network, and relationship diagrams in databases field.

**Outline**

A software engineering team will be formed. It will go through one or more full SE cycles regarding the methodology the team chooses.

The team will have to:

1. Gather system requirements
2. Propose a proper code design
3. Implement it & try to always have a working demo
4. Goto 1

More specifically:

- Come up with some nice and feasible scheme to visualise supernodes, edges, and the whole structure of the connections between the supernodes in the input graph.
- Think of how to implement it
- Make some prototypes
- Choose the best prototype & improve it into a demo
- Improve the demo into a mature working application

**Tasks & Deliverables**

- A software application that visualise the compressed graphs with possibly overlapping supernodes.
- Documentations on the program design
- Software engineering related documents:
  - A report on the status of the project
  - A system requirement specification
  - Design documentation
  - Others required by the GRP module

## **[P16] Production Job Scheduling Software System**

### **Background**

In manufacturing, scheduling the production jobs in a dynamic and uncertain business environment is the one of the most important and yet challenging problems. On one hand, the optimisation of the production job schedules can maximise the efficiency of the key production resources (machine, labor, etc.) and reduce the operational costs and lead time. On the other hand, however, the problem is naturally intractable computationally (NP-Hard) and it becomes even harder with inclusion of dynamic and uncertain factors. A description of the basic job shop scheduling problem can be found from [Wikipedia](#). One implementation of some classic optimisation algorithms for the job shop problem can be found from [Google OR-Tools](#).

### **Requirements**

In this project, you are asked to implement a web-based software system to support the production job scheduling of a local company in Ningbo. The system should support the followings:

1. A user-friendly interface that gives straightforward visualisations of the scheduling solutions (both in table and Gantt chart formats). It should also support users to make modifications of the solution using drag-and-drop types of operations.
2. Interfaces to import predefined problem data and to export the solutions to pre-defined formats for visualizations.
3. A problem definition parser for user-defined problems through simple scripts. The parser should support the definitions of most of the common job shop problems.
4. Integration of one of algorithms from OR-Tools.

### **Tasks & Deliverables**

1. Web-based software with functions mentioned in the previous section.
2. A user manual for how to use the software.

## **[P17] Feedback Analysis Based Product Rating**

### **Requirements**

This project will require big data analytic tools and techniques as well as the knowledge of sentiment analysis.

### **Background**

Product rating is an important part of online shopping websites. Customers provide feedback to the product(s) based on their level of satisfaction. Analysis of sentiment within the feedback helps to rate the product(s), which in turn helps the future customers to choose the right product(s).

### **Outline**

Here we propose an advanced Feedback Analysis Based Product Rating system that detects hidden sentiments in comments and rates the product(s) accordingly. The system uses Big Data analytics and sentiment analysis methodology in order to achieve desired functionality. This project will act as an add-on to the shopping websites where the registered user will view the product and product features and will comment about the product. System will analyse the comments of various users and will rank product. We use a database of sentiment based keywords along with positivity or negativity weight in database and then based on these sentiment keywords mined in user comment is ranked. Comment will be analysed by comparing the comment with the keywords stored in database. The System takes comments of various users, based on the comment, system will specify whether the product is good, bad, or worst. Once user login the system he can view the product and product features. After viewing product user can comment about the product. User can also view comment of other user's. The role of this application will be collecting user feedback of the products, analyse sentiment in the feedback, and rate the products. This application also works as an advertisement which makes many people aware about the product. This system is also useful for the user's who need review about a product.

### **Tasks & Deliverables**

The prospective team needs to develop a software tool which will act as a background service to the shopping websites. This tool will read and analyse customer feedbacks and generate rating of the products fairly.

## **[P18] Flexible Data Annotation Tool for Machine Learning Techniques**

### **Requirements**

Students who are interested in this project are required to be familiar with or willing to quickly learn GUI programming or web UI programming.

### **Background**

Nowadays, machine learning techniques have become very popular in Artificial Intelligent applications. Usually machine learning techniques make use of training data to teach the machine to do jobs as human does. The training data are those data which have annotation or label given by human. For example, in the application of document classification, given a text document, the machine learning algorithm is able to tell whether the document belongs to certain category, e.g. sports, politics, economics, etc. The training data for this application contain text documents whose categories are already known. In other words, the text documents in the training data have been annotated or labelled with their categories by human. This process is called data annotation or labelling. Being able to annotate a large amount of training data are critical to modern deep learning based techniques.

### **Outline**

This project aims to develop a software that facilitates annotation process for machine learning techniques. The functionality of the software should include but not limited to : 1) load the input data (could be text documents or images) and display to the human; 2) let user annotate the data; 3) load and save existing annotations; 4) allow user to modify existing annotations; 5) display the statistics of existing annotations (e.g. how many data have be annotated with certain labels, etc.); 5) allow multiple users to annotate the data at the same time; etc. More advanced functions could be added later depending on the progress.

### **Tasks & Deliverables**

Tasks to be accomplished include: 1) overall system design; 2) GUI design and development; 3) data related design and development; 4) testing at different levels. The deliverable will be a flexible data annotation tool for machine learning techniques.

**Additional Information**

The student should not worry about the detailed annotation process and what kind of annotation needs to be dealt with in the beginning of the project. These implementation details will be discussed during overall system design phase. The web based UI system is preferred in this project.



## **[P19] Surveillance Video Analysis Toolbox**

### **Requirements**

This project requires the following knowledge and skills:

- Some knowledge on image analysis and computer vision techniques, especially on object detection, face recognition, action recognition. Hands-on experience on openCV is preferred.
- Some knowledge on machine learning, pattern recognition and deep learning. Hands-on experience on building a convolutional neural network is preferred.
- General knowledge on software development. Some teamwork experience is preferred.

### **Background**

Ningbo-Zhoushan Port is the world-leading port in Ningbo. To surveillant the operations in the port and monitor the daily operations, a surveillance video analysis toolkit will be useful. With the recent developments in computer vision and machine learning, analyzing surveillance videos and understanding the content of the video becomes feasible. In this project, you are expected to develop and implement a surveillance video analysis toolbox, including functions like detecting and tracking people and vehicles, recognizing faces and gaits and detecting abnormal events such as fires and smokes, using existing techniques.

### **Outline**

More specifically, you are expected to construct a video surveillance toolkit. The toolkit should consist of a well-designed GUI and some APIs to facilitate the analysis of videos. Moving targets such as people vehicles should be recognized and tracked. You may detect and analyze other objects in the scene if possible. You may identify peoples through face recognition or gait recognition. You may detect abnormal events in the port.

You are advised that this project requires knowledge in more than one area, and it requires a good teamwork spirit to make the project succeed. At the end of the project, you will be able to learn knowledge and have hands-on experience in computer vision, image/video processing and software engineering.

You may use the existing techniques to implement the system.

**Tasks & Deliverables**

The following functionalities are expected for the toolkit:

- Load the saved videos for analysis.
- Analyze the online video streaming (real-time performance).
- Recognize common objects, people and vehicles.
- Track the objects, and if possible across cameras.
- Align the camera scene with the map of the port, and report the location of people or vehicles.
- Recognize people through faces or gaits.
- Detect abnormal events such as fires or smoke.

At the end of the project, you are expected to deliver a live demo for the toolkit.

**Readings and Resources:**

Algorithms, codes and papers on machine learning/computer vision:

<https://paperswithcode.com/>

<https://github.com/>