CECS 2019-20 Summer Research Projects

Contents

Atomically thin Nano-materials and devices

Supervisor: Yuerui Lu [yuerui.lu@anu.edu.au]

Two-dimensional (2D) nano-materials, such as molybdenum disulfide (MoS_2) and graphene, have atomic or molecular thickness, exhibiting promising applications in nano-electro-mechanical systems. Graphene is a one-atom thick carbon sheet, with atoms arranged in a regular hexagonal pattern. Molybdenum disulfide (MoS_2) belongs to transition metal dichalcogenides (TMD) semiconductor family YX_2 (Y=Mo, W; X=S, Se, Te), with a layered structure. These 2D nano-materials can be integrated into nano-electro-mechanical systems, enabling ultra-sensitive mechanical mass sensors, with single molecule or even single atom sensitivities.
Moreover, the mechanical resonators based on these 2D nano-materials would be a perfect platform to investigate quantum mechanics, opto-mechanics, material internal friction force, nonlinear physics, etc.

Requirements/Prerequisites:
Specific requirements and /or prerequisites to be determined by the Supervisor

Nano Biomedical Devices

Supervisor: Yuerui Lu [yuerui.lu@anu.edu.au]

The ability to detect bio-molecule at ultra-low concentrations (e.g. atto-molar) will enable the possibility of detecting diseases earlier than ever before. A critical challenge for any new bio-sensing technology is to optimize two metrics --- shorter analysis time, and higher concentration sensitivity in clinically relevant small volumes. Moreover, practical considerations are equally important:

- simplicity of use
- mass producible (low cost), and
- ease of integration within the clinical structure.

Compared with other methods, nano-electro-mechanical system (NEMS) based bio-sensors are promising in clinical diagnostics because of their extremely high mass sensitivity, fast response time and the capability of integration on chip. We have demonstrated a low concentration DNA (atto-molar sensitivity) optically interrogated ultrasonic mechanical mass sensor, which has ordered nanowire (NW) array on top of a bilayer membrane. This method represents a mass-based platform technology that can sense molecules at low concentrations, which could be useful for early-stage disease detection. We can develop this sensor further to measure an array of biomarkers (e.g. DNA or proteins), by providing both the needed specificity and sensitivity in physiological disease (e.g. cancer) detection.

Requirements/Prerequisites:

- Specific requirements and /or prerequisites to be determined by the Supervisor

Systematic Boolean SAT(isfiability)

Supervisors: Charles Gretton, CECS [charles.gretton@anu.edu.au] and Josh Milthorpe, CECS [josh.milthorpe@anu.edu.au]

Systematic Boolean SAT(isfiability) solvers are notoriously good at not solving problems with symmetries. Even the measly pigeon hole problem confounds them to the edge of time. Of course, folks have for decades shown that we can do better than this! This project is to implement and experimentally evaluate a bespoke combination of techniques for tackling symmetry in Boolean constraints programming. If you do not have them, you will learn C++ and MPI skills. If you already have those, then you shall practice them in this project.

Requirements/Prerequisites:

- Specific requirements and /or prerequisites to be determined by the Supervisor

Students will gain:

- Skills and/or practice in C++ and MPI skills.

Facial Expression Recognition in Video Games

Supervisors: Penny Kyburz, CECS (Penny.Kyburz@anu.edu.au) and Tom Gedeon (tom@cs.anu.edu.au]

Biometrics are being increasingly used to measure and analyse the player experience in video games. Measures commonly include HR, EEG, GSR, and eye-tracking. Facial expression analysis is becoming more prevalent in analysis of user experience, but has had limited applications to video games. This project involves designing, developing, and testing facial expression analysis software for analysis of video game players. Applications include understanding and measuring the player experience, but also potentially reacting to and creating the play experience.

Background Reading:

- https://dl.acm.org/citation.cfm?id=1124926.
- https://link.springer.com/chapter/10.1007/978-3-319-60582-1_37.

Virtual Reality Video Game Player Experience

Supervisors: Penny Kyburz, CECS (Penny.Kyburz@anu.edu.au) and Tom Gedeon (tom@cs.anu.edu.au]

This project involves surveying different tools and methods for collecting, analysing, and understanding player experience in VR games. Methods could include heuristic evaluations, user research, and biometrics. The goal is to grow our understanding of experience and enjoyment in VR games and to develop and refine our tools for measuring and understanding this experience.

Requirements/Prerequisites:

- Would suit students with passion for video games and an interest in HCI and user research
- Experience with game design or user evaluation would be a plus.

Brigitte's Summer

Supervisors: Alban Grastian, (Alban.Grastien@anu.edu.au)

Path-finding is the problem of finding a shortest path in a map. We are interested here in grids with 8 possible moves (north, south, east, west, north-east, north-west, south-east, south west). While this problem is very simple and can be solved in microseconds, there is still a lot of interest in solving it even faster.

The goal of this work is to improve the recently proposed approach named Brigitte. Described here (http://grastien.net/ban/brigitte), Brigitte partitions the map into "islands" and computes between every pair of islands a set of "bridges", i.e., blueprints of paths between cells of the islands. Given a path query, Brigitte retrieves the bridges of the islands that the source and target cells belong to, determines which bridge yields the shortest path, and constructs the path.

There are many potential improvements to Brigitte. For instance preprocessing is currently prohibitive. The goal of this project is to implement and test some of these improvements.

Requirements/Prerequisites:

- Students will need very good programming skills in C
- Some basic knowledge of AI, e.g., A* is also required

FPGA-based Hexapod Robot Spider With Voice Control via Smartphone

Supervisor: Dr. Aline I. Maalouf (Aline.Maalouf@anu.edu.au)

The aim of this project is to design a FPGA-based hexapod robot spider, which is used for student education purposes. The robot spider consists of 6 legs, which are located on both sides of the body. Each leg has 3 joints, i.e. each leg includes 3 freedoms. One freedom is rotating around lengthways axis. This is defined as the small leg joint. Another freedom is rotating around the crosswise axis and it is defined as the big-leg joint. A robot spider with six legs could have a lot of gait modes. The gait mode of the spider robot in this project is similar to the mode of coxswainlessix-oar rowboat. With the rowboat mode the movement of all six paddles are same at the same time, but for this robot there are some differences. All six legs of the robot are divided into two groups: Group A and Group B. Group A includes two left legs, L1,L3 and one right leg, R2. Group B includes two right legs, R1,R3 and one left leg, L2. Each group forms a triangle. All three legs which are in one group will make the same movement at the same time. The main consideration is that, when the legs of one group leave from the ground, then the other three legs of the other group keep standing on the ground. This can keep the robot movement stable. In term of kinematics design, the movement of each leg includes four beats: leg rising→return stroke→leg descending →paddling. The movement is like as human arm in free style swim, or like as paddle in a rowboat. The positions of one right leg of the spider during 4 beats, where in states 3 and 4, the leg contacts the ground, in states 1 and 2, the leg leaves the ground. In state 4, the leg carries the spider forwards. The aim is to have the three joints of the three big legs in one group output the maximum power. The spider takes 250ms for one beat. Four beasts form a loop, which takes one second. A stride length of a loop is 50mm.

The design should include: FPGA board, servo motor kit, sonar module, Bluetooth in addition to the physical body of the spider robot. For voice control via smartphone, the RFS card should be used and a speech to text conversion should be performed using some available applications. The RFS card is connected to the FPGA board and the connection between the HC-05 bluetooth and the FPGA board is established through the RFS card by using UART or wifi connection.

Requirements/Prerequisites:

- Knowledge of FPGA/Verilog language, robotics background

Touchscreen Synthesizer

Supervisor: Dr. Aline I. Maalouf (Aline.Maalouf@anu.edu.au)

This project introduces a new alternative in synthesizers – a touchscreen synthesizer. A synthesizer is a sound producing electronic instrument that is used to generate a broad range of sounds and tones, as well as apply different effects, such as filtering and resonance. Although it is usually controlled by an analog interface consisting of buttons, switches, sliders, knobs, and piano keys, this project aims to employ these features via a touchscreen, relieving the user from any physical stress. The objectives are:

1. Read touch input from the Terasic Multi-Touch module
2. Generate basic waveforms (up to 3 octaves)
3. Read in additional waveforms from memory
4. Generate major and minor chords
5. Generate and apply filters to waveforms (equalizer)
6. Output graphical user interface to LCD screen
7. Output audio signals

Requirements/Prerequisites:

- Knowledge of FPGA/Verilog language, good knowledge of signal processing.

Voice Controlled Servo Motor By Android App

Supervisor: Dr. Aline I. Maalouf (Aline.Maalouf@anu.edu.au)

This project aims to control a servo motor by voice commands for remote operation. A FPGA is used together with a Bluetooth device interfaced to the control unit for sensing the signals transmitted by any Android application running cell phone. Remote operation is achieved by any smart-phone/Tablet having Android OS upon a GUI (Graphical User Interface) based voice operation. The transmitting end uses an Android application through which the voice commands are transmitted to digital bits. At the receiver end, these commands are used for controlling the speed and position of the servo motor. Serial communication data sent from the Android application is received by the Bluetooth receiver interfaced to the FPGA.

Requirements/Prerequisites:

- Knowledge of FPGA/Verilog language

Real Time Face Detection and Tracking

Supervisor: Dr. Aline I. Maalouf (Aline.Maalouf@anu.edu.au)

The area of face detection and tracking plays an important role in many applications such as video surveillance, biometrics, or video coding. Many different methodologies have been proposed in literature and can mostly be categorized as featured-based, appearance-based, or colour-based. The goal of this project is to create an FPGA system to detect and track a human's face in real time. The overall setup should include the Verilog program, an Altera DE2- 115 board, a camera, and a VGA monitor. The face detection algorithm should be based on skin detection and image filtering. After the face region is detected, its location should be determined by calculating the centroid of neighbouring skin pixels.

Requirements/Prerequisites:

- Knowledge of FPGA/Verilog language, computer vision, image processing

Cognitive Load Assessment

Supervisor: Md Zakir Hossain [zakir.hossain@anu.edu.au] and Tom Gedeon [tom@cs.anu.edu.au]

Cognitive load assessment is a process to measure learners' stress levels while doing various learning tasks. It refers to available working memory resources during the task that is reflected through human physiological signals. This project is to construct a machine learning and statistical model to effectively measure learners' cognitive load while doing three tasks in three different difficulty levels - reading comprehension, solving mathematical problems, and solving Sudoku puzzles. The relevant physiological data is available and ready to analyse.

Requirements/Prerequisites:

- Knowledge of basic machine learning classifiers and basic statistics.
- Experience in the use of Matlab, Python or similar tool.
- Basic knowledge about physiological signals (especially galvanic skin response, temperature, and blood volume pulse).

Students will gain:

- Skills in conducting human computing based research.
- A solid understanding on various physiological data.
- Experience in advance data analysis and prediction.

Australian Signals Directorate (ASD) – Various Projects

Supervisor: To be determined

The Australian Signals Directorate (ASD) is a vital member of Australia's national security community, working across the full spectrum of operations required of contemporary signals intelligence and security agencies: intelligence, cyber security and offensive operations in support of the Australian Government and Australian Defence Forces (ADF). ASD has entered into a partnership with the ANU, the goals of which include research collaboration on topics related to ASD's mission and building national capability in Science, Technology, Engineering and Mathematics (STEM), including training future generations of signals intelligence and cyber security professionals.

ASD values strong critical thinking and research skills, with an emphasis on an ability to present and reason analysis in diverse forums. ASD analysts work on a broad range of tasks which include: investigating large and complex data sets, developing new methods of analysing data for intelligence or information security purposes and solving cryptological problems using advanced mathematical concepts.

As part of this partnership, ASD is pleased to offer funding for research projects at the Summer Research Scholar level. These research projects may come from anywhere in the university (computer science, mathematics, physics, statistics and linguistics are all represented in ASD's business); however, they will typically align with these areas:

- number theory
- cryptography
- statistics
- data science
- secure systems,
- vulnerability research

To apply, submit a short paper describing your work. There will also be an opportunity to deliver a short presentation about your work to ASD and ANU

Requirements/Prerequisites:

- Australian Citizens only
- Minimum GPA of 5.5/7.0 (preference will be given to applicants with strong academic performance in a cognate background)
- Nominate your area/s of interest, based on the list above.

Accelerating Fuzzers with Performance Management Units (PMU)

Supervisor: Tony Hosking (Anthony.hosking@anu.edu.au) and Michael Norrish (Michael.Norrish@data61.csiro.au)

Fuzzing is an automated technique for discovering bugs and vulnerabilities in software. Coverage-guided greybox fuzzers use code coverage to drive themselves towards exploring new and interesting code within a program. This code coverage information is typically generated by instrumentation injected into the target program at compile time. Modern CPUs contain Performance Management Units (PMU) that provide mechanisms for tracing code coverage information in hardware. This has the advantage of:
(a) higher accuracy, and
(b) lower overheads.

This project will investigate incorporating PMUs (e.g., Intel PT) into popular coverage-guided greybox fuzzers and evaluating the accuracy/performance benefits. The student will have a strong background in low-level systems programming in C and an interest in automated vulnerability discovery.

Requirements/Prerequisites:

- Australian Citizens only

- Minimum GPA of 5.5/7.0 (preference will be given to applicants with strong academic performance in a cognate background)

Fuzzing with More Precise Path Context

Supervisor: Tony Hosking (Anthony.hosking@anu.edu.au) and Michael Norrish (Michael.Norrish@data61.csiro.au)

This project is about exploring alternative approaches to capturing program profiles for guiding fuzzing. The current state of the art assumes that block coverage is insufficiently precise and that edge coverage is cheap to maintain and gives adequate precision, but increasing the path context is too expensive even though it improves precision. We wish to understand better the impact of these decisions, and hope to devise cheaper path coverage techniques for improved precision without high overhead.

Requirements/Prerequisites:
- Australian Citizens only
- Minimum GPA of 5.5/7.0 (preference will be given to applicants with strong academic performance in a cognate background)

Targeting Suspect Code with Static Analysis for Fuzzing

Supervisor: Tony Hosking (Anthony.hosking@anu.edu.au) and Michael Norrish (Michael.Norrish@data61.csiro.au)

Fuzzing is an automated technique for discovering bugs and vulnerabilities in software. Coverage-guided greybox fuzzers use code coverage to drive themselves towards exploring new and interesting code within a program, with the ultimate goal of uncovering security-interesting bugs. However, not all code is considered equal when it comes to vulnerabilities!

In this project, the student will explore using static analyses (using the LLVM compiler framework) to score blocks of code depending on how likely they are to contain bugs. This will then prioritise code for the fuzzer to explore. For example, if a region of code calls functions that are prone to memory errors (e.g., strcpy, etc.), or performs complex pointer arithmetic, then there may be a higher probability of that code region containing a bug.

The student will learn about static analysis techniques (with a focus on security-related analyses), and work with the LLVM IR to implement these analyses. The ideal student will have a solid background in C++ programming and an interest in software security, static analysis, and compiler techniques.

Requirements/Prerequisites:
- Australian Citizens only
- Minimum GPA of 5.5/7.0 (preference will be given to applicants with strong academic performance in a cognate background)

CakeML parsing with error messages

Supervisor: Michael Norrish (Michael.Norrish@data61.csiro.au)

The CakeML compiler (see cakeml.org) is the world's first verified compiler for a functional language that goes from concrete syntax all the way to machine code. Unfortunately, users who write concrete syntax currently get terrible error messages ("There was a parse error somewhere in your program.") The aim of this project is to figure out how to extend the parsing technology so that it is behaviourally the same on correct programs (critically important for verification purposes), but gives useful error messages (with line numbers, for a start!) Students will need a strong logic/verification background, and an interest in writing code in a functional style.

CakeML streams

Supervisor: Michael Norrish (Michael.Norrish@data61.csiro.au)

The CakeML compiler (see cakeml.org) is the world's first verified compiler for a functional language that goes from concrete syntax all the way to machine code. Unfortunately, input programs are consumed all at once rather than read a character at a time. This makes the correctness statement simple to write and understand ("if there is a parse of the input string, type check it and compile it"). Sadly, it also makes the front-end of the compiler rather inefficient, because all of the input needs to be read before anything can happen. It would be much better to be able to read the input file a character at a time, and have it lexed and parsed incrementally. There is existing support for writing code to access files in CakeML's library, but the compiler itself can't currently use it. Students will need a strong logic/verification background, and an interest in writing code in a functional style.

Micro-synchrophasor: sense, observe, and understand solar PV grid integration challenges

Supervisor: Elizabeth Ratnam (Elizabeth.Ratnam@anu.edu.au)

In recent years, a dramatic increase in electrical power generation from renewable energy sources including 'rooftop' solar photovoltaics (PV), has been observed in many countries. As a consequence of the daily and seasonal variability of solar and wind energy resources, more advanced sensors are needed to observe and monitor the safe and stable operation of the power grid – particularly at voltage levels closer to both residential homes and 'rooftop' PV generators i.e., the distribution grid.

The 2017 CSIRO / ENA Electricity Network Transformation Roadmap highlighted that approximately $16 billion in network infrastructure investment could be avoided by the orchestration of DER, including solar PV and battery storage. Furthermore, there could be a market for grid support services delivered from DER customers of over $2.5 billion per annum by 2050. A key capability required to unlock both sources of value is the real-time monitoring and analysis of potentially millions of DER.

The project will directly contribute to solutions for de-carbonizing the electricity grid – of which the first step is to sense, observe, and understand the present-day operation of low and medium voltage networks. This project will be hosted at ANU with the support of the newly formed Battery Storage and Grid Integration Program. The ANU team brings a deep understanding of Australian distribution grids based on extensive relationships with network service providers.

Requirements/Prerequisites:

- Students should have taken a course on machine learning and/or control systems
- Students should have strong mathematical and programming skills (in Python).
- Independent worker willing to tackle a challenging research problem with potential follow through beyond the summer scholar program

Students will gain:

- Experience working on cutting edge problems in power systems
- Skills in conducting and presenting research
- Be involved with state-of-the-art researchers with the goal of de-carbonizing the grid.

Accurate Performance Evaluation of Short-Packet Communications in Beyond 5G and 6G Eras

Supervisor: Dr. Nan Yang [nan.yang@anu.edu.au)

Using short packets to communicate has been identified as a key enabler for ultra-reliable and low-latency communications (URLLC), a hot direction in the beyond 5G era and 6G era. Despite such importance, the accurate performance evaluation of this communication paradigm, especially over millimetre-wave and terahertz bands, has not been revealed. In this project, the student is expected to revisit the models used for approximating the Q function and then apply appropriate model(s) to analyse the reliability and latency performance achieved by short-packet communications. The anticipated result would establish the theoretical foundation for designing and implementing URLLC for future wireless data applications, such as the Internet of Vehicles and Factory Automation.

Requirements/Prerequisites:

- Strong mathematical skills.
- Experience in the use of Matlab.
- Basic knowledge of wireless communications theory.

Simulator Development for Ultra-Reliable and Low-Latency Communications in Beyond 5G and 6G Eras

Supervisor: Dr. Nan Yang [nan.yang@anu.edu.au)

Ultra-reliable and low-latency communications (URLLC) is a pivotal and challenging scenario in the beyond 5G and 6G eras. It has been acknowledged within the community that the use of short codes is a must to realise URLLC. However, the performance of potential short codes candidates, e.g. Bose–Chaudhuri–Hocquenghem (BCH) code, in practical URLLC systems has remained unknown. In this project, the student is expected to develop a simulator to examine the reliability and latency performance of short codes (such as BCH codes) in practical scenarios and reveal the impact of environmental parameters (e.g. transmission distance and mobile user mobility) on such performance. The anticipated result would serve as a highly useful platform for evaluating and applying short codes into future URLLC applications, such as the Internet of Vehicles.

Requirements/Prerequisites:

- Rich experience in Matlab/C programming.
- Good mathematical skills.
- Basic knowledge of coding theory and communications theory (or information theory).

Synthesizing to Learn and Learning to Synthesize

Supervisor: Liang Zheng (liang.zheng@anu.edu.au)

Computer vision is at the bottleneck when large-scale datasets are lacking due to ethics, extreme environment, etc. To still be able to work on such problems, a good way is to resort to data synthesis. Through 3D simulation engines powered by computer graphics, we can obtain large-scale datasets under various environments like raining, sunset, and low resolution. This, together with state-of-the-art domain adaptation technologies, will bring computer vision to solve challenging real-world problems, such as detecting a car under hazy environment. Therefore, learning can benefit significantly from synthesis.

On the other hand, in order to synthesize scenes that look real, it is critical to let the engine look at what the real-world looks like, and learn to synthesize realistically. Basically this has a two-fold meaning. First, the engine needs to be able to synthesize the geometry among the objects, e.g., trees should be alongside the road. Second, synthesized objects should have a realistic appearance. Those objectives could be realized by the usage of re-inforcement learning, graph convolutional networks and generative adversarial networks. Thus, learning can also benefit data synthesis.

Background literature:

- Kar et al., Meta-Sim: Learning to generate synthetic datasets. ICCV 2019.
- Sun et al., Dissecting person re-identification from the viewpoint of viewpoint. CVPR 2019.
- Ruiz et al., Learning to simulate. NIPS 2018.

Requirement/Prerequisites:

- A strong motivation in conducting cutting-edge computer vision research.
- Strong coding abilities with Python, C++.
- Experience and knowledge in deep learning, such as convolutional neural networks, deep re-inforcement learning, graph conv nets and generative nets.
- Experience in deep learning platforms like Pytorch and Tensorflow.
- Experience in using simulation engines like Unity.

Students will gain:

- Deep understanding about deep learning models to be used in this project.
- Practical skills in training/testing deep models.
- Experience in designing new models.
- Experience in working with active researchers towards impactful research.

Linked Data Mining for Contextual Electricity Demand Forecasting

Supervisor: Kerry Taylor (Kerry.Taylor@anu.edu.au) and Nathan Elazar (Nathan.Elazar@anu.edu.au)

Managing electricity demand for an entire city is a challenging prospect. Electricity suppliers need to understand how and when demand on the electricity grid changes so that they can plan their operations to accommodate all their customer's needs. Currently, basic models are used to forecast the demand of each household in the ACT, however they fail to take into account all of the factors that may impact electricity demand.

- Semantic web technologies provide a framework for collecting data from multiple heterogenous sources into a single ontology. In an ontology data is expressed as a graph, with relationships between objects and object attributes all expressed as edges of different types. At the same time, linked data mining techniques have been developed to mine knowledge directly from ontologies. The current state of the art in linked data mining by refinement is an algorithm called Owlminer, which searches for logical descriptions to separate classes of nodes.

The goal of this project is to incorporate data from various sources (including household characteristics, aerial imagery, weather data, and geographic locations) into a single ontology, and apply linked data mining methods to improve the forecasting of household electricity demand in the ACT.

Requirement/Prerequisites:

- Students should have taken a course on machine learning and/or data mining.
- Strong programming skills (in Python and Java).
- Willingness to learn to use new tools and methods.

Students will gain:

- Gain knowledge and expertise of semantic web technologies.
- Get hands on experience working on a real-world data analysis project.
- Be involved in helping to improve electricity delivery for the entire ACT.

Musical Instruments that Play Along: Machine Learning for Music Generation, Interaction, and Performance

Supervisor: Charles Martin (Charles.Martin@anu.edu.au)

The concept of using AI/ML models that can generate music and sound has expanded dramatically in recent years. However, despite media attention towards musical AI research, music involving AI is rarely heard in concerts apart from a few special research events. This is partly due to a lack of musical ML systems directed towards music performers.

In this project, you will help to change this by developing a new ML model that can interact with a human in live performance. This model could connect directly into existing music technology components such as digital audio workstation software (DAWs) or be a self-contained computer musical instrument, touchscreen app, or custom sensor-based device for musical expression.

Requirement/Prerequisites:

- Coursework or experience in machine learning, AI, or data science. Knowledge/experience in deep learning would be a plus.
- Experience with Python.
- Strong interest in music, performance, and creativity.

Links and references:

- Creative Prediction Project: https://creativeprediction.xyz
- Interactive Musical Prediction System: http://charlesmartin.com.au/imps/

Semantic-based relation extraction in documents

Supervisor: Pouya G. Omran (P.G.Omran@anu.edu.au) and Kerry Taylor (Kerry.Taylor@anu.edu.au)

Document management is a vital task in any enterprise. In many domains, a massive number of documents written in natural language are available although they are not well configured, and the relations between them are not determined. Organising this amount of information manually is not feasible in many domains. A method that can assist us to extract the relations between documents is the first step toward an autonomous framework for managing documents.

In NLP, distributed representation (or word embeddings) for text has been widely studied. In this approach, a vector represents the elements of natural language, including word, phrase, paragraph, or even whole document. The vector representation captures the semantics of NL elements. In this project, we seek to address the issue of automatically extracting the relation between two documents based on their semantics.

Requirements/Prerequisites:

- Programming skill in Python
- Solid Background in Machine Learning or Natural Language Processing

Students will gain:

- Experience on developing solution for an open research question
- Skills in conducting research in ML and NLP

Verified Verifiers for Verifying Elections

Supervisor: Rajeev Gore (rajeev.gore@anu.edu.au) and Thomas Haines (thomas.haines@ntnu.no)

Electronic voting (e-voting) is now gaining a lot of attention as countries such as Estonia move to fully on-line national elections. Alas, the internet is a dangerous place with multiple mean-beans and nasty-pasties just waiting to interfere in national elections, to elect buffoons such as ... you know who.

In response, various researchers have designed e-voting schemes, such as Helios, which produce cryptographic evidence for the integrity of their results, making them software independent. That is, they can be run on totally insecure and compromised computers yet they still guarantee the following: if the evidence they produce for each phase of the election is correct with respect to its specification, then the result for that phase is correct with respect to its own specification.

So, to verify that the results of each phase are correct, we need to write a computer program to check that the evidence produced is correct with respect to its specification: a so-called verifier.

But the evidence that is produced involves all sorts of mathematical mumbo-jumbo such as zero-knowledge proofs, sigma protocols, group theory, integer factorisation, and discrete logarithm algorithms, so how can we be sure that the verifier we write is itself correct?

We have shown that we can formalise all of the previously mentioned mumbo-jumbo into logic, and then use an interactive theorem prover to not only check that the maths is done correctly, but to also extract the code for the verifier itself. That is, we obtain the code via logic and proofs rather than explicit coding. We want to push this idea further to encapsulate the gamut of end-to-end voter verifiable e-voting.

Requirement/Prerequisites:

You will need:

- a good undergraduate maths background
- an interest in mathematical logic

You will NOT need:

- deep knowledge about cryptology
- ability to code in low level languages such as C, Python, Java etc
- hacking skills.

Students will gain:

- this project may well lead to a publication in an international conference

Australian citizens may be eligible for further scholarships to work in this area for their honours thesis, and possibly their PhD, but Australian citizenship is *NOT* a pre-requisite.

Human Centred Computing / Bioinspired Computing projects

Supervisor: Tom Gedeon [tom@cs.anu.edu.au]

Please see my own student projects page at http://cs.anu.edu.au/people/Tom.Gedeon/projects.html for the current list of my projects. They range from human centred projects using eye gaze, physiological signals, EEG, fNIRS and other sensors, and effectuators such as electrical body and brain stimulation; bioinspired projects using neural networks, deep learning or evolutionary algorithms; or a combination of both human centred and bio-inspired. There is scope for multiple students in these areas.

Requirements/Prerequisites:

- An interest in human experiments particularly relating to human internal states such as emotion, stress, anxiety and so on;
- An interest in bio-inspired computing techniques;
- A willingness to have fun learning how computers can be made to be more useful and responsive to human beings.

Students will gain:

- Practical experience in design and conduct of human experiments
- Experience in advanced data analysis and prediction.

Benchmarking OpenCL for High-Performance Scientific Computing

Supervisor: Josh Milthorpe (josh.milthorpe@anu.edu.au)

High-performance computing (HPC) systems are becoming increasingly heterogeneous, with current nodes consisting of a mix of CPU and one or more GPUs, and it is expected that FPGAs will soon be thrown in the mix. The performance of HPC systems is often evaluated using the HPL benchmark [1], which measures the time to solve a dense system of linear equations. However, this benchmark is limited since linear algebra represents only one of many critical types of workloads for HPC systems.

The Extended OpenDwarfs (EOD) benchmark suite aims to encompass a wider range of patterns of computation and communication found in scientific computing workloads. [2] EOD comprises a set of realistic scientific codes written using OpenCL [3] so as to be portable to modern accelerator architectures including CPU, GPU and many-integrated-core (MIC). So far, EOD includes 13 benchmarks which have been evaluated on 15 different accelerator devices. It uses high resolution/low overhead timers to measure the performance of individual accelerators built in to each region of every EOD benchmark. The performance of certain accelerators are highly influenced by problem size, so a representative benchmark suite should be flexible with regard to problem size selection. To this end, a major focus of EOD was in enabling different problem sizes for each benchmark code, where problem size selection is based on the working memory footprint. So far, not all of the benchmark codes have been extended to support multiple problems sizes -- which is where you come in!.

For further information please visit this website: https://cecs.anu.edu.au/research/student-research-projects/benchmarking-opencl-high-performance-scientific-computing

Requirement/Prerequisites:

- Time;
- Motivation;
- Solid programming skills;
- Familiarity, or at least a passing acquaintance with GDB and valgrind. Also, most of the benchmarks are written in C++, and some in C. So knowing these languages couldn't hurt. That said, if you show an abundance of Requirements 1-3, we can help you learn you the tools and languages as you progress.

Students will gain:

- Demonstrate understanding of performance characteristics of scientific computing application codes on heterogeneous computing hardware
- Perform performance evaluations of benchmark codes
- Apply good benchmarking practice to enhance and extend an existing OpenCL benchmark suite
- Communicate performance results to a research audience

Can you hack it?

Supervisor: Dirk Pattinson (dirk.pattinson@anu.edu.au)

The goal of this project is to find the most effective, and efficient approaches to find both errors and vulnerabilities in binary code. Aside from quality assurance, there is interest from both industry and government to evaluate (and possibly certify) third party software products for their use.

The starting point for the project is the "Common Weakness Enumeration" available at https://cwe.mitre.org/ that lists, and catalogues, common programming errors, and their ramifications.

There are a number of both commercial and open-source tools already available, such as CWE Checker https://github.com/fkie-cad/cwe_checker and the utility in project goal is to:
evaluate their detecting software vulnerabilities
explore ways of combining these tools for finding vulnerabilities
suggest, and possibly implement, extensions.

Requirement/Prerequisites:

- strong programming skills
- interest in reverse engineering
- Australian Citizens only
- Minimum GPA of 5.5/7.0 (preference will be given to applicants with strong academic performance in a cognate background)

Students will gain:

- exposure to state of the art software tools
- working at the interface of academia / industry / governments
- practical knowledge in reverse engineering
- practical expertise in the discovery of vulnerabilities in real code

Algorithms for modal types

Supervisor: Ranald Clouston (Ranald.Clouston@anu.edu.au)

The extension of typed programming languages with type-formers called 'modalities' is an active current area of research. For example, modalities can give the programmer more fine-grained control over when code is compiled or executed. To be useful, typed programming languages need algorithms that can check whether a given program has the type claimed, or even try to infer the type of a given program. This project involves inventing and implementing algorithms to check or infer types for typed functional programming languages extended with a modality.

Requirement/Prerequisites:

- Competent programming, preferably with functional programming experience
- Interest in logic and/or principles of programming languages
- Familiarity with lambda-calculus or natural deduction would be helpful

Students will gain:

- Knowledge of about the foundations of typed functional programming languages
- An insight into the application of logic to programming languages research

A Verified Theorem Prover for Modal (Description) Logic K_n (ALC)

Supervisor: Rajeev Gore (rajeev.gore@anu.edu.au) and Michael Norrish (Michael.norrish@anu.edu.au)

Modal (description) logics are the basis of the main ontology language OWL. Many ontologies have been created, and are used in real-world applications: for example the GALEN ontology for medical diagnosis.

In applications of modal (description) logic, we require a theorem prover that is as efficient as possible, so most existing theorem provers for modal logic are written in languages such as C++. But they invariably contain bugs: that is, they sometimes claim that some formula is valid when it is unsatisfiable, and vice-versa. Clearly, the ideal is to produce a verified and efficient theorem prover for modal (description) logics.

Verified theorem provers for modal logics are currently a hot topic in research:
for example, the recent work of Minchao Wu. But it was done in Lean, which means that we don't actually have the code for a verified decision procedure in a real programming language: Minchao's code runs inside Lean but is not a stand-alone program in C say.

The aim is to use HOL4 and CakeML to produce a verified theorem prover for modal (description) logics, or to use Coq and code-extraction to do the same.

The modal logic K_n is the most basic normal (multi-)modal logic which is characterised by the multi-modal formulae that are valid in all Kripke models. It is known as the description logic ALC. The validity problem is decidable and is PSPACE-complete. There is a well-known sequent calculus which is sound, complete and terminating for deciding whether a given formula is valid.

Encode the syntax, the Kripke semantics of K_n/ALC and the well-known sequent calculus for K_n into HOL4/Coq and prove the soundness of the rules. Then prove the completeness by showing that backward proof-search terminates, hence extracting a verified decision procedure for this logic. Presumably this will require the use of the HOL4 to CakeML compiler but this is where Michael comes in. Alternatively we will use Coq for which we have local expertise.

Requirement/Prerequisites:
You will need:

- a good background in logic
- a good maths background

You will NOT need:

- programming skills in C++, or other low level languages

Students will gain:
There is a possibility that this work may be published in an international conference

Formalised Proof Theory of Substructural Logics

Supervisor: Rajeev Gore (rajeev.gore@anu.edu.au))

Since the seminal work of Gentzen and Lambek, logicians and philosophers have discovered a plethory of alternatives to classical propositional logic by altering the rules of Gentzen's LK.

We now have a very good proof-theoretic and semantic understanding of the substructural hierarchy, from the non-associative and non-commutative Lambek Calculus, via BCK, FLew, relevant logic, multiplicative linear logic, intuitionistic logic, to classical propositional logic.

Logicians have invented generalised proof systems to try to capture these logics in one setting: display logic; labelled calculi; hyper-sequent calculi and deep-inference calculi to name a few. However, we have no clear map of the expressive power of these formalisms.

The project is to investigate the relationships between these calculi and to formally prove these relationships inside the interactive theorem prover Coq. A typical result will be of the form: every logic which has a cut-free proof system in formalism F1 will also have a cut-free proof system in formalism F2.

By doing so, we will ensure that our results are formally verified and we shall be able to extra formally verified proof-systems for logic L in formalism F2 from the known ones in formalism F1.

Further work would allow us to investigate a general cut-elimination theorem in formalism F (say) which would cover a huge number of logics at once, and which might also give verified decision procedures for the decidable logics in this class from our formalisations.

Virtual Reality Audio

Supervisor: Thushara Abhayapala (Thushara.Abhayapala@anu.edu.au))

Virtual reality has becoming an important part of everyday life. Though there has been progress in 3D vision, 3D audio has yet to be matched to the requirement of VR applications. In this project, the student will undertake research in modifying measured head Related Transfer functions (HRTF) with missing data by spatial filtering with novel window functions. The student will have opportunity to embed their algorithms with Oculus Rifts to check their performance.

Requirement/Prerequisites:

- Basic Signal Processing background,
- Good with mathematical derivations
- Matlab/C
- willingness to implement and test with real signals,
- well developed personal attributes

Annotation-aware read alignment and variant detection in the human genome

Supervisor: Yu Lin (yu.lin@anu.edu.au)

Precision medicine is an emerging approach for disease treatment and prevention that considers individual variability in the genome, environment, and lifestyle for each person. Current genomic analysis for detecting individual variants aligns reads derived from an individual genome against the standard human reference genome. This standard human reference genome is composed of more than 50% repeat elements (including transposable elements and long duplicated regions. Ambiguous alignments due to repeat elements in the reference genome form a major source of errors in variant detection. This project aims to improve read alignment based on repeat annotations in the reference genome and detect accurate variant information in 1000 Genome Project Data.

Requirements/Prerequisites:

- Experience in programming.
- Basic knowledge of genomics is desirable.

Students will gain:

- Hands-on experience in genomics
- Skill in large-scale data analysis

Augmented Tangible Notations for Effective Workshops

Supervisor: Henry Gardner (Henry.Gardner@anu.edu.au)

This project will examine ways in which patterns of user experience can inform and improve diagrams constructed from tangible notations for requirements elicitation. The project will seek to validate modern theories of design and user experience to requirements-elicitation workshops as they are currently practiced by industry professionals. As an extension, the project will examine ways in which augmented reality might be used to improve and extend the conduct of these workshops to, for example, involve remote participants.

Requirements/Prerequisites:

- a solid background in human computer interaction.
- ideally have an understanding of modern design theories involving user experience.
- Previous design experience in augmented reality will also be an advantage

Interactive musical overlays of sculptures

Supervisor: Henry Gardner (Henry.Gardner@anu.edu.au)

This project will look at ways that sculptures in the urban environment might be augmented with holograms and interaction to steer music and related audio. This is a collaborative project with the CSIRO.

Requirements/Prerequisites:

- a solid background in human computer interaction and computer vision is necessary.
- Previous experience developing augmented reality applications for the Microsoft Hololens would be an advantage.