Realtime 3D scene understanding in year 2020

Role of computation in computer vision

Zeeshan Zia
zeeshan.zia@imperial.ac.uk

I recently co-chaired a British Machine Vision Association (BMVA) meeting with Prof. Andrew Davison in London, carrying the above title. We planned on having a 100% invited program and managed to get excellent speakers. The objective of the meeting was to look at software-hardware issues in real-time 3D scene understanding, in the context of future heterogeneous manycore computing architectures.

A lot of progress in computer vision over the past two decades has been driven by exponential increase in computational resources available to researchers. Up until 2005, programmers could rely on the next technology node to deliver twice as many transistors in the same chip area (Moore's law and Dennard scaling) which were arranged into increasingly complex superscalar processors, exploited automatically by modern compilers. As Dennard scaling broke down, the next generation of computing architectures is rapidly evolving towards heterogeneous multi-cores, including GPUs and application-specific hardware accelerators. Unfortunately with multi-core architectures, the burden is back on the shoulders of programmers to exploit the available computational resources. State-of-the-art compilers are not able to automatically extract the parallelism inherent in our programs and offload tasks automatically to the various types of computational cores that may be available on a system. In addition, platform portability is becoming a bigger challenge – a piece of code written for my big Nvidia GPU needs major changes before it can run on my smartphone which has a very different configuration of heterogeneous embedded cores and GPU. Yet another important issue, as computer vision moves towards practical consumer applications is that of power consumption. Even with the most sophisticated custom-designed hardware for handheld mobile vision applications, for instance on the Google Tango phones, the battery discharges completely within fifteen minutes. Since battery technology is not expected to improve by leaps and bounds anytime soon, there is an urgent need to think about power consumption when designing computer vision algorithms - if we want computer vision to become commercially relevant.

The speakers correspondingly included experts in computing architectures, computer vision, and application domains including robotics and augmented reality, as well as people who are working on software-hardware co-design for vision.

Andrew Davison opened the meeting by highlighting the exponential increase in computational capacity enabled by GPGPU in the last 7-8 years as opposed to the far slower improvement in CPUs. He admitted that early adoption of GPGPU technology by his group was responsible for its leading position in dense SLAM in the recent years. I particularly liked when he contrasted that a 1000 pound laptop from 2003 could run MonoSLAM, the first real-time monocular SLAM system, representing maps as 10s to 100s of sparse interest points; versus a 1000 pound laptop of today which can run ElasticFusion, the highest quality dense SLAM system which represents maps as 10s of millions of points (surfels). Its really exciting to think of this 6-7 orders of magnitude improvement in the fidelity of real-time 3D scene representation enabled almost entirely by GPUs! But does everybody in the vision community possess mastery of such unique programming models as CUDA, to keep pushing the boundaries?

Doug Watt, who is Multimedia Strategy Manager at Imagination Technologies talked about their PowerVR Imaging Framework. Imagination Technologies identifies applications of Computer Vision and Computational Photography as opportunities to differentiate from their competition in the domain of low-power embedded graphics cores. They have been exploiting GPUCompute and implementing a number of vision pipelines for image stabilization, face detection, background removal, face beautification etc on their heterogeneous CPU and GPU fabric. After the talk, I was wondering why such big SoCs do not incorporate specific hardware accelerators for low-level vision kernels; the sort that are available on the Myriad Vision Processing Units (VPUs) from Movidius or the EyeQ line of chips from MobilEye. Luckily, this thought was automatically addressed by **Gerhard Reitmayr**, Principal Engineer at Qualcomm Research when he gave his perspective on the issue. Explaining the business of big System-on-Chip (SoC) producers, he mentioned that his company sells 10s of millions of SoCs, and everything added to the SoC must be of value to the end OEMs. These SoCs have a multitude of heterogeneous computational cores, which can be reused at the OEM level if needed. On-chip real estate is very expensive and whenever you wish to add something new, you have to justify why it couldn’t be done with existing compute capability on the SoC, for example, why couldn't the modem DSP be exploited for a new computational module instead of integrating a new function-specific hardware accelerator? In general, when deciding on compute options, their first choice is to do it on the CPU (exploiting Neon SIMD engines), if their application has already fully exploited the CPU, they go for the DSP, whereas the embedded GPU is usually busy rendering for their Augmented Reality applications. Gerhard also showed videos for a number of 3D scene understanding applications (sparse and dense SLAM, segmentation, object recognition) running on their low-power mobile platforms.

While these speakers supported the pragmatic approach given available resources, the fact that GPUs are not well-suited to “branchy” code and the von Neumann bottleneck were not tackled until **Simon Knowles**, CTO of XMOS talked about a completely new computing device that he calls a “graph processor”. He claimed that computational vision is nothing like graphics, for which GPUs were originally designed. In particular, highlighting machine learning workloads he said that the associated data structures are often sparse, and arbitrary sparse structures are naturally represented as graphs. He further talked about the dynamic nature of machine learning research, citing “Dropout” in deep neural networks for regularization and Rectified Linear Units replacing sigmoid units which were the standard for a long time. Since the field is evolving rapidly, he does not believe in committing to fine-grained details of present algorithmic pipelines. Thus his team is working on a traditionally neglected computation model called the “Bulk Synchronous Model” employing large quantities of on-chip SRAM which will work as a “fat” communication model. The system will have multiple asynchronous cores such that the overall system is well-suited for sparse computation. The system will need a unique “graph-parallel programming abstraction” with a sequential outer control program, controlling “codelet” vertex functions. As far as I understood, they are evaluating this abstraction represented as an extension to the Python programming language. However, they do not propose this machine as a replacement for CPUs and GPUs, rather as a third component in a troika, with each component specializing in different kinds of workloads. My lab mates and I are really looking forward to playing with these devices soon!

Trying to act as a bridge between the architecture experts and the vision community, and as a representative of the **PAMELA** project, I talked about two of the projects that we in the PAMELA team are involved in. PAMELA stands for a Panoramic Approach to the Manycore Landscape, and is a project funded by a 5-year EPSRC programme grant. We have three partner universities involved in the project: Universities of Edinburgh and Manchester, and Imperial College London, with groups specialising in Computing Architectures, Compiler and Runtime design, Domain Specific optimisation and languages, and Robot Vision. The objective of the project is the same as the aims of this meeting, "to exploit future heterogeneous manycore architectures" with 3D scene understanding as the unifying challenge application. Specifically I talked about our SLAMBench framework, which is a publicly-available software framework for quantitative, comparable, and validatable experimental research to investigate trade-offs in performance, accuracy, and energy consumption of various SLAM systems (KinectFusion already available, LSD-SLAM and ORB-SLAM integration under way). SLAMBench provides implementations in C++, OpenMP, OpenCL, CUDA, ARM NEON and in our paper we perform experimentation on a variety of hardware platforms. I am particularly proud of the software level energy measurement instrumentation we have provided in the framework, which is able to read off fine-grained energy consumption at the level of individual kernels and hardware components. We have recently been performing design space exploration on top of SLAMBench, using active learning to jointly model algorithmic, compiler, and hardware parameter spaces. This machine learning enabled auto-tuning has been very beneficial in terms of providing us configuration points which allow significant speed-ups at much lesser energy consumption. Overall the work can allow compiler and architecture experts to optimise their designs for vision workloads - as well as computer vision researchers to compare and contrast various vision kernels in full pipeline context.

Simon Lynen, researcher at ETH-Zurich and Google Project Tango, presented a number of experiments done with Tango devices. To give some background: Google introduced its Project Tango last year, which comes in the form of two mobile platforms (a smartphone and a tablet) incorporating a depth camera as well as a high-performance computing device for vision processing. The project allows highly accurate real-time SLAM by tightly coupling inertial sensing with visual tracking. While the visual tracking appears to be based on sparse keypoints in selected keyframes, the map representation comprises of an occupancy grid populated from the depth sensor. This allows for a wide range of use cases specially in the area of augmented reality (some amazing videos are available on Youtube). The phone performs its vision processing on a Myriad vision processing unit from Movidius - having general-purpose vector processing capabilities together with a number of low-level hardware accelerators for vision tasks such as edge detection - all fitting within a tiny power envelope. The tablet on the other hand, comprises of an Nvidia Jetson TK1 SoC, which is a low-power version of Nvidia's standard GPU technology, which we thoroughly evaluate in our SLAMBench work also. The existence of these devices further emphasizes the importance of domain-specific optimization and architectures in enabling low-power high-performance vision. Simon mentioned how the platform has been maturing rapidly, with the odometry drift going from 1% (of the trajectory) just two years ago to

0.1% now. He has been testing the device under challenging conditions from fourteen consecutive roller coaster rides (resulting in only 3.5% drift) to Zero gravity flights to real-time obstacle avoidance on a quadrotor. The device has even been sent to the International Space Station. One interesting challenge he highlighted was that the integrated depth camera, due to eye-safety concerns, delivers less than 2.5% of the data as a standard Kinect camera.

Mike Aldred, Electronics lead at Dyson, gave the perspective of an integrator of these technologies, as he talked about their recent Dyson 360 Eye vacuum cleaner robot. As opposed to its predecessor, the famous Roomba robot, which randomly roams around the environment changing its direction when it hits an obstacle, the 360 Eye is a serious product with a powerful 120W vacuum cleaner. To avoid wasting energy, the device needs to do methodical motion and thus incorporates an omnidirectional camera (128 x 1024 resolution) used for sparse feature tracking and mapping (together with a few other sensors). He mentioned how even successful academic algorithms such as MonoSLAM or PTAM, still desired more than a decade of real-world testing and tuning, before they could reach a mature prototype. They have tested the robot in 100,000 homes already, and still keep finding new challenges - he described some funny and unexpected ways in which humans interact with these devices sometimes resulting in failure. He stressed that the vision community should explicitly report fallibilities together with capabilities of our algorithms, so the integrators can know what the challenges are. He shared his wisdom on other issues including pitfalls of premature optimisation and importance of keeping the architecture generic as far as possible.

Thomas Whelan, Dyson Research Fellow at Imperial College London talked about “the rise of dense methods” is SLAM. He started with a quick survey of dense SLAM algorithms and how these have been enabled primarily by recent advances in cheap consumer-grade hardware: GPUs and RGB-D cameras. He sees desktop/laptop GPUs reaching tens of teraflops and sensors moving towards higher resolution and frame rates by 2020, and stressed that these improvements will make things much easier for the algorithms. He also described his state-of-the-art large scale dense SLAM algorithms Kintinuous and the latest ElasticFusion (being presented at RSS 2015). The audience loved Tom’s live demo of ElasticFusion. He commented that in the domain of dense methods, we are moving towards 4D reconstruction methods (for dynamic scenes), where approaches will be going beyond the lambertian assumption and incorporating increasingly more sophisticated semantic concepts in real-time pipelines. Finally, he gave a sneak peak into his current research project, where he is able to recover the direction and position of lighting sources from his dense reconstructions!

Apart from speakers directly focusing on the intersection of computing and vision, we also had a few amazing speakers sharing their experiences in pure vision and robotics, while still citing the importance of computation.

Maurice Fallon, Lecturer at Edinburgh University and Perception lead on MIT’s entry to the recent DARPA Robotics Challenge, talked about his experiences designing and operating tele-operated humanoid robots working in disaster scenarios. Further, he showed some recent work involving autonomous walking with a passive camera, merging Kintinuous with a footstep planning algorithm. I wondered, seeing the recent deep learning revolution, if it might be possible to learn the manipulation and footstep planning from a combination of real and synthetic simulated data. Unfortunately, in Maurice’s experience, physical contacts and motion are still difficult to model faithfully - and thus he doesn't believe in simulated training for mid-level control problems. Infact, he is soon going to get a NASA Valkyrie robot in Edinburgh to advance his research with humanoids, where he plans to work on autonomous operation. Really exciting!!

While the vision component of the workshop was SLAM-heavy, we were lucky to also have serious machine learning muscle amongst ourselves. **Jamie Shotton**, Principal Researcher at Microsoft is well known for his contributions to random forests applied to computer vision problems. He showcased their latest research in real-time hand tracking from depth cameras, and how the technology is rapidly evolving into a robust and useable input device. Responding to a question about why there are never multiple hands tracked in his demo videos, he responded that the required computation to estimate a single hand pose takes up the whole GPU - again emphasizing the need for holistic thinking when it comes to high-performance vision!

Renato Salas-Moreno, co-founder of Surreal Vision, recently acquired by Oculus Research/Facebook talked about his PhD research at Imperial College London. He likes to think of computer vision as "inverse video game design", and is the developer of SLAM++ and Dense Planar SLAM (check out YouTube for some impressive videos, which have received 50,000+ views in total). These real-time SLAM pipelines incorporate semantic concepts into localisation and mapping, enabled by Renato's thorough understanding of Nvidia's GPUs and OpenGL. Renato demonstrated how performing 3D object detection and plane fitting inside the SLAM pipeline, allow for unique augmented reality applications.

The last half-hour of the programme was dedicated to a panel discussion with all of our speakers. Without giving away the "answers" we got, I will let you know the questions I scripted to ask our panelists! We will be happy to include your thoughts on these in future issues of this newsletter... :-)

1. What is the most impressive real-time 3D scene understanding demo that you expect to see at CVPR or ISMAR 2020?

2. Seeing that computer vision and machine learning are evolving rapidly, do you think it is prudent to already start optimizing them for software/hardware implementation?

3. What is the right level of abstraction to target for vision DSLs or SoCs? low-level kernels, generic parallel patterns, full pipelines, or something else.

A version of this report will be published in the IAPR and BMVC Newsletters.