THE UNIVERSITY *of* EDINBURGH

Edinburgh Research Explorer

Algorithmic Performance-Accuracy Trade-off in 3D Vision Applications

OPEN ACCESS

Algorithmic Performance-Accuracy Trade-off in 3D Vision Applications

Bruno Bodin[†], Luigi Nardi[¶], Harry Wagstaff[†], Paul H. J. Kelly[‡] & Michael O'Boyle[†]

Abstract— Simultaneous Localisation And Mapping (SLAM) is a key component of robotics and augmented reality (AR) systems. While a large number of SLAM algorithms have been presented, there has been little effort to unify the interface of such algorithms, or to perform a holistic comparison of their capabilities. This is particularly true when it comes to evaluate the potential trade-offs between computation speed, accuracy, and power consumption. SLAMBench is a benchmarking framework to evaluate existing and future SLAM systems, both open and closed source, over an extensible list of datasets, while using a comparable and clearly specified list of performance metrics. SLAMBench is a publicly-available software framework which represents a starting point for quantitative, comparable and validatable experimental research to investigate trade-offs in performance, accuracy and energy consumption across SLAM systems. In this poster we give an overview of SLAMBench and in particular we show how this framework can be used within Design Space Exploration and large-scale performance evaluation on mobile phones.

Fig. 1. The SLAMBench Graphical User Interface provides real-time status of the different performance metrics such as speed, power, and accuracy.

SLAMBench: Performance and accuracy benchmarking methodology for SLAM

Simultaneous Localisation And Mapping (SLAM) is a key component in robotics that constructs a map of an unknown environment while simultaneously keeping track of the robot's location within it. SLAMBench [1] is an open-source benchmark based on the SLAM system KinectFusion [2] that produces dense 3D model of an arbitrary scene using an RGB-D camera. SLAMBench provides implementations of KinectFusion using popular languages, such as CUDA, OpenCL, OpenMP and C++. Figure 1 shows the graphical interface of SLAMBench. The two frames on the top left of the interface are the RGB and Depth frame of the Kinect sensor. The bottom left is the tracking status of the algorithm and the right frame is the current map generated by the SLAM system. **SLAMBench supports research in hardware accelerators and software tools by enabling the comparison across algorithms, implementations, and datasets, of performance, energy-consumption, and accuracy of the generated 3D model in the context of a known ground-truth.**

HyperMapper: Co-design exploration of SLAMBench using machine learning

We examine how SLAMBench can be mapped to power constrained embedded systems [3], [4]. Key to our approach is the idea of incremental co-design exploration, where optimization choices that concern the domain layer are incrementally explored together with low-level compiler and architectu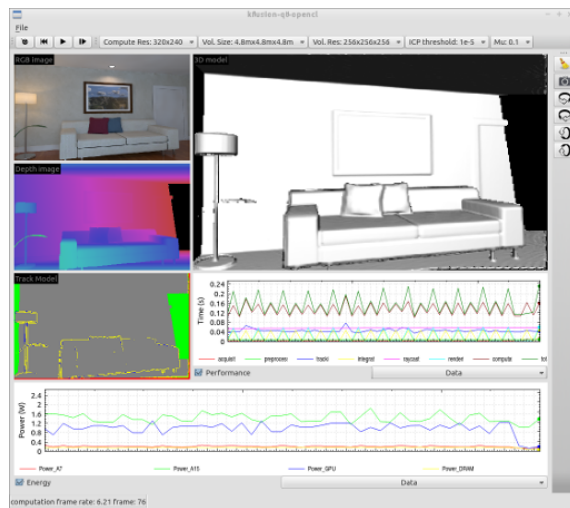re choices. The goal of this exploration is to reduce execution time while minimizing power and meeting our quality of result objective. As the design space is too large to exhaustively evaluate, we use active learning based on a random forest predictor to find good designs. Figure 2 shows an overview of this learning process. **We show that our approach can, for the first time, achieve dense 3D mapping and tracking in the real-time range within a 1W power budget on the Odroid XU3 embedded device. This is a 4.8x execution time improvement and a 2.8x power reduction compared to the state-of-the-art.**

Related work

Computer vision research has traditionally focused on optimising the accuracy of algorithms. In autonomous driving, for example, the KITTI benchmark suite [5] provides data and evaluation criteria for the stereo, optical flow, visual odometry and 3D object recognition. The ICL-NUIM dataset [6] and TUM RGB-D benchmark [7] aim to benchmark the accuracy of visual odometry and SLAM algorithms.

An important early benchmark suite for performance evaluation entirely dedicated to computer vision is SD-VBS [8]. SD-VBS provides single-threaded C and MATLAB implementations of 28 commonly used computer vision kernels that are combined to build 9 high-level vision applications. Another contribution at such performance evaluation is MEVBench [9], which focuses on a set of visual recognition applications including face detection, feature classification, object tracking and feature extraction. It provides single and multithreaded C++ implementations for some of the kernels with a special emphasis on low-power embedded systems. While such efforts are a step in the right direction, they do

[†] School of Informatics, University of Edinburgh, UK
[‡] Department of Computing, Imperial College London, UK
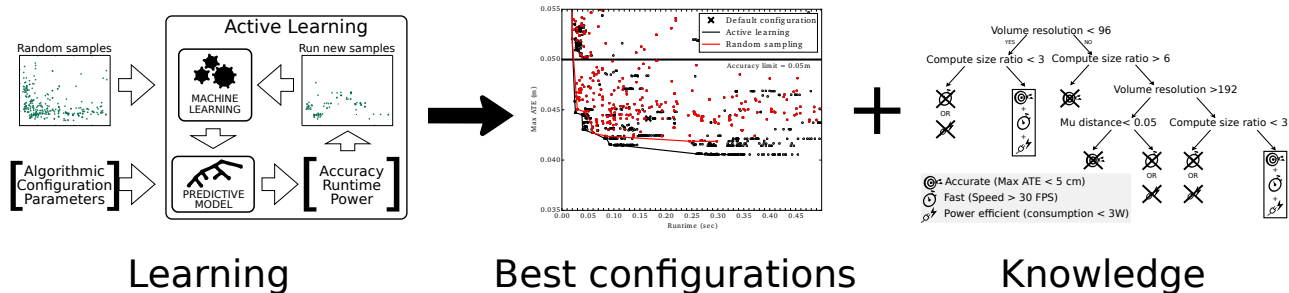[¶] Stanford University, USA

Fig. 2. Design space exploration methodology for algorithmic parameters. The first step performs random sampling of the space. Then a predictive model is built to guide the exploration. Finally this model can be used to understand the impact of parameters on the different performance metrics.
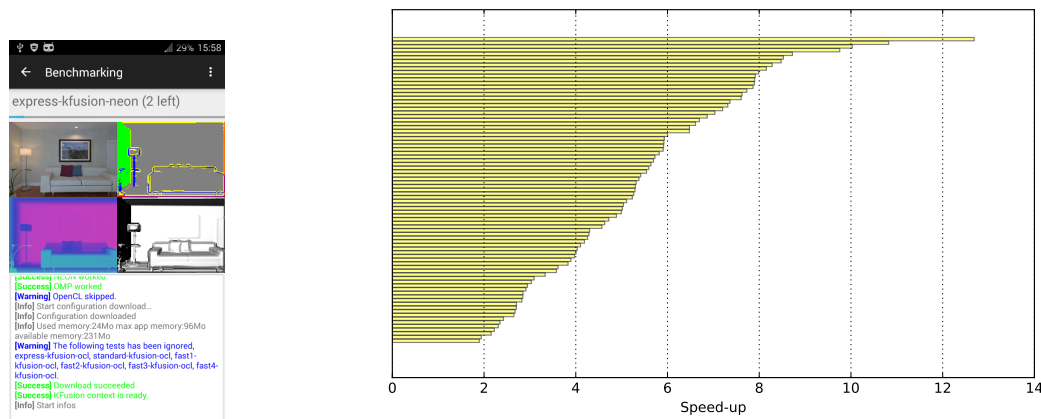

Fig. 3. The OpenCL KinectFusion has been run on 83 smart-phones and tablets from the market using an Android application. One the left can be seen a screenshot of the Android application running. On the right is the speed-up result. For each device, we computed the speedup of the configuration we found for the ODROID-XU3 with HyperMapper.

not provide the software tools for accuracy verification and exploitation of hardware accelerators or *graphics processor units* (GPUs). Nor do they enable investigation of energy consumption, performance and accuracy envelopes for 3D scene reconstruction algorithms across a range of hardware targets.

A key feature of SLAMBench is that it is designed on top of the recently-proposed ICL-NUIM accuracy benchmark [6], and thus supports wider research in hardware and software. The quantitative evaluation of solution accuracy into SLAMBench enables algorithmic research to be performed.

AN EVALUATION OF SLAMBENCH AND HYPERMAPPER ACROSS A WIDE RANGE OF MOBILE PHONES

The SLAMBench framework and more specifically its various KinectFusion [2] implementations has been ported to Android [10]. More than 1000 downloads have been made since its official release on the Google Play store. This success allowed us to crowdsource data from more than 100 different mobile phones. Figure 3 summarizes the performance results from the collected data. We now plan to use this data to analyse the performance of KinectFusion on those platforms, and to provide techniques to optimise KinectFusion performance depending of the targeted platform. We believe that by combining the potential of HyperMapper and the data collected on Android, we could train a decision machine for mobile phones.

REFERENCES

[1] L. Nardi, B. Bodin, M. Z. Zia, J. Mawer, A. Nisbet, P. H. J. Kelly, A. J. Davison, M. Luján, M. F. P. O'Boyle, G. Riley *et al.*, "Introducing SLAMBench, a performance and accuracy benchmarking methodology for SLAM," in *IEEE Intl. Conf. on Robotics and Automation (ICRA)*, May 2015.

[2] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohli, J. Shotton, S. Hodges, and A. Fitzgibbon, "KinectFusion: Real-time dense surface mapping and tracking," in *ISMAR*. IEEE, 2011.

[3] B. Bodin, L. Nardi, M. Z. Zia, H. Wagstaff, G. Sreekar Shenoy, M. Emani, J. Mawer, C. Kotselidis, A. Nisbet, M. Lujan *et al.*, "Integrating algorithmic parameters into benchmarking and design space exploration in 3d scene understanding," in *Proceedings of the 2016 International Conference on Parallel Architectures and Compilation*, ser. PACT '16. New York, NY, USA: ACM, 2016, pp. 57–69.

[4] L. Nardi, B. Bodin, S. Saeedi, E. Vespa, A. J. Davison, and P. H. Kelly, "Algorithmic performance-accuracy trade-off in 3d vision applications using hypermapper," in *iWAPT*. IEEE, 2017.

[5] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? The KITTI vision benchmark suite," in *CVPR*, 2012.

[6] A. Handa, T. Whelan, J. McDonald, and A. Davison, "A Benchmark for RGB-D Visual Odometry, 3D Reconstruction and SLAM," in *ICRA*, 2014.

[7] J. Sturm, N. Engelhard, F. Endres, W. Burgard, and D. Cremers, "A benchmark for the evaluation of RGB-D SLAM systems," in *IROS*, 2012.

[8] S. K. Venkata, I. Ahn, D. Jeon, A. Gupta, C. Louie, S. Garcia, S. Belongie, and M. B. Taylor, "SD-VBS: The San Diego vision benchmark suite," in *IISWC*, 2009.

[9] J. Clemons, H. Zhu, S. Savarese, and T. Austin, "MEVBench: A mobile computer vision benchmarking suite," in *IISWC*, 2011.

[10] "Slambench for android," https://play.google.com/store/apps/details?id=project.pamela.slambench.