ARBench: Augmented Reality Benchmark For Mobile Devices

Sofiane Chetoui, *Brown University*, Providence, USA
Rahul Shahi, *Brown University*, Providence, USA
Seif Abdelaziz, *Cupertino High School*, Cupertino, USA
Abhinav Golas, *Meta Platforms*, Burlingame, USA
Farrukh Hijaz, *Meta Platforms*, Redmond, USA
Sherief Reda, *Brown University*, Providence, USA

***Abstract*—This paper takes an important step towards the improvement of the AR mobile experience by designing and developing ARBench, the first Augmented Reality (AR) benchmark for mobile devices. ARBench incorporates different AR workloads that stress multiple hardware units of the SoC (CPU, GPU, DSP, etc), and measures the individual score for each AR workload. The proposed benchmark suite is then used to evaluate the AR performance of various commercial mobile devices, and their ability to support various functions of AR workloads.**

***Index Terms*—Augmented Reality, Mobile Devices, Benchmark, Performance, Mobile SoCs**

I. Introduction

In 2021, The number of Augmented Reality (AR) active users on mobile devices has increased to 810 million, because many consumers already own an AR capable smartphone. The release of software development kits for the design of AR Apps, like ARCore and ARkit, for Android and iOS facilitate the development of AR applications on mobile devices.

Unlike regular mobile Apps, AR Apps rely on the simultaneous usage of different hardware-units and the execution of multiple threads. For example, an AR app needs to consistently read the positional and GPS data, perform camera processing, run mapping and tracking algorithms, 3D rendering of the virtual object. These key differences, make the existing mobile devices less suitable to offer a seamless AR experience.

In previous work, frameworks and criteria to evaluate AR applications have been proposed [1]–[3], and characterizations of mobile AR Apps have been performed from a system and architecture perspective [4]. However, we still lack an AR benchmark suite dedicated for mobile devices, that encompasses various AR workloads and uses multiple hardware units.

In this paper we propose and release a new benchmark set for AR applications on mobile devices. To summarize, the contributions of this paper are as follows:

- We design and develop ARBench, the first AR benchmark suite that measures the AR performance of mobile devices. The benchmark incorporates different AR workloads that stress multiple hardware units of the SoC (CPU, GPU, DSP, etc), and measures the individual score for each AR workload. ARBench is publicably available at https://scale-lab.github.io/software/.
- We use ARBench to analyze the utilization of various SoC units, and to characterize the performance of multiple generations of SoCs. The analysis shows few insights, including the fact that the AR performance of existing mobile SoCs greatly depends on the type of AR workload.

S. Chetoui and S. Reda are with the School of Engineering. R. Shahi is with the department of Computer Science.
E-mail of the corresponding author: sherief_reda@brown.edu

II. Augmented Reality Benchmark

ARBench consists of six benchmarks, each of which evaluates the performance by stressing different units in the SoC. The description of the six Benchmarks of ARBench, their duration and objective are described in Table I. The ARBench benchmarks are based on the Google ARCore [5] framework, and include various categories of AR workloads with different user inputs to reflect the behavior of existing AR Apps.

ARBench measures and reports the frame per second of each particular AR workload by simulating an actual user session. To standardize the input data to ARBench, we performed two steps:

- **Input data generation:** In order to stress mobile devices with AR workloads, the data associated with the workload needs to be generated. The benchmark workload data includes recordings of the user input, such as the user screen touches coordinates, and includes recordings of camera's video stream and IMU data. This is achieved by using the recording API offered by ARCore. The benchmark workload data was generated by adding the record feature to 6 different AR Apps, and running each AR App with a particular pattern of user interaction.
- **Input data playback:** After generating the AR workload data, ARBench was designed by combining the 6 different apps, using the playback feature, which takes the the recordings generated in the previous step as input, and reconstruct the user behavior, while displaying the recorded data. Additionally, we included the computation of the frame rate for each benchmark, which is used as the performance metric.

III. AR Benchmarking of Commercial Mobile Devices

In this section, we show the per-unit hardware utilization of ARBench using a commercial mobile SoC. Then, ARBench is

TABLE I: The description and the objective of each benchmark of ARBench

Benchmark Name	Description	Objective
Object Generation	Inserts a single virtual object on a surface and views the object from a close distance, while moving the mobile device capture angle.	Tests basic functionality with low performance requirements on AR workloads that involve the insertion and rendering of single 3D virtual object.
Multi-Object Tracking	Maps out a large surface and inserts multiple objects. Spends most of the time moving around the space and viewing objects from different distances and angles.	Evaluates performance on AR workloads with intensive tracking and graphics, that involve the insertion and tracking of several 3D virtual objects.
Scene Overloading	Keeps inserting a very large number of objects in a densely packed fashion so that many objects are onscreen at the same time.	Evaluates performance on AR workloads where very intensive graphic workloads are the bottleneck for performance.
Augmented Faces	Applies a face filter to a human face visible in the frame. The human face keeps moving and changing capture angles throughout the benchmark.	Evaluates performance of face detection including identifying the different human face features.
Augmented Image	Inserts a virtual photo frame when viewing a specified target image. The virtual photo is viewed from different angles and distances.	Evaluates performance of 2D image detection and tracking.
Object Recognition	Scans real objects in a room and applies labels to identify them. The device keeps moving throughout the experiment, such that it captures new objects.	Evaluates performance on AR workloads that involve heavy computation for object detection and recognition.

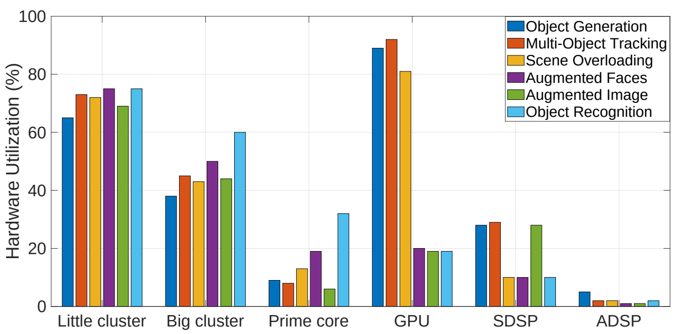

Fig. 1: The average per-unit hardware utilization of ARBench using a Snapdragon 865 SoC.

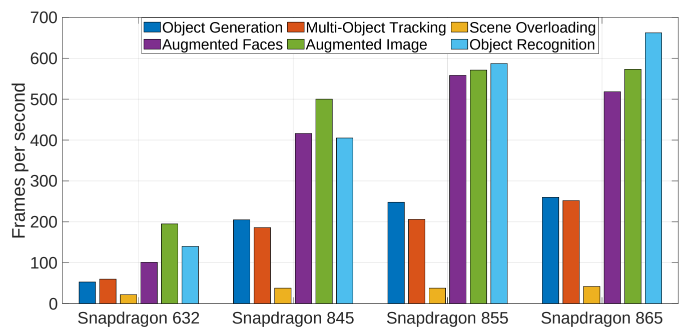

Fig. 2: The per-benchmark AR performance of different Snapdragon SoCs while running the proposed benchmark.

used to evaluate the AR performance of different commercial mobile SoCs, and their ability to support AR workloads is discussed. In order to evaluate the performance of different mobile devices, we use BrowserStack [6] to remotely access a variety of commercial mobile devices.

Per-unit hardware utilization: Figure 1 shows the per-unit hardware utilization of the ARBench benchmarks using the 865 Snapdragon SoC. The units include, the CPU, which is divided to a little and big clusters, and a prime core. In addition to the GPU, the image signal processor referred to as SDSP, the application DSP, referred to as ADSP. From Figure 1, we make the following observations:

- ARBench relies on the simultaneous usage of multiple hardware units. Especially, the little and big clusters, the GPU and the SDSP.
- The first three benchmarks of ARBench are highly compute and graphic intensive. On the other hand, the last three benchmarks are mainly compute intensive.
- The little cluster is highly and similarly used by all of the AR benchmarks.
- The different AR benchmarks use the Prime core, the GPU and the SDSP differently.

Evaluation of existing mobile SoCs: Figure 2 shows the per-benchmark AR performance of different Snapdragon SoCs, as evaluated by ARBench. It shows few insights:

- AR performance of existing mobile SoCs greatly depends on the type of AR workload.
- Even state-of-the-art SoCs are not able to meet the performance requirements for AR workloads that involve the insertion of multiple 3D virtual objects, as shown through the performance of the Scene Overloading benchmark. Additionally, the performance of AR workloads similar to the Scene Overloading benchmark is not scaling across the different SoC generations.
- The high variation of FPS across the different benchmarks, implies that runtime management could help in achieving important power savings, while meeting the AR performance requirements. This could be achieved by performing DVFS scaling of the CPU and GPU units.

IV. Conclusion

ARBench incorporates different AR workloads that stress multiple hardware units, and measures the individual score for each AR workload. The proposed benchmark suite can be used to evaluate the AR performance of commercial mobile devices, and their ability to support various AR workloads. The proposed benchmark is made publicly available.

REFERENCES

[1] M. Huzaifa, R. Desai, S. Grayson, X. Jiang, Y. Jing, J. Lee, F. Lu, Y. Pang, J. Ravichandran, F. Sinclair *et al.*, "Exploring extended reality with illixr: A new playground for architecture research," *arXiv preprint arXiv:2004.04643*, 2020.

[2] M. Ablyaev, A. Abliakimova, and Z. Seidametova, "Criteria of evaluating augmented reality applications," *Advanced Engineering Research*, vol. 20, no. 4, 2020.

[3] D. Munro, A. Calitz, and D. Vogts, "Architecture and architectural patterns for mobile augmented reality," *South African Computer Journal*, vol. 33, no. 1, pp. 59–78, 2021.

[4] H. Chen, Y. Dai, H. Meng, Y. Chen, and T. Li, "Understanding the characteristics of mobile augmented reality applications," in *2018 IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS)*. IEEE, 2018, pp. 128–138.

[5] M. Lanham, *Learn ARCore-Fundamentals of Google ARCore: Learn to build augmented reality apps for Android, Unity, and the web with Google ARCore 1.0*. Packt Publishing Ltd, 2018.

[6] "cross browser testing platform," Dec 2018. [Online]. Available: https://www.browserstack.com/