Video: Banner – An Image Sensor Reconfiguration Framework for Seamless Resolution-based Tradeoffs

Jinhan Hu, Alexander Shearer, Saranya Rajagopalan, Robert LiKamWa
Arizona State University

Tempe, Arizona

jinhanhu,acshear1,srajag25,likamwa@asu.edu

ABSTRACT

The high energy consumption of visual sensing continues to impede the future of mobile vision in which devices will continuously compute visual information from sensory data, e.g., for visual personal assistants or for augmented reality (AR). While vision algorithms continue to improve in task accuracy and speed, mobile and wearable vision systems fail to achieve sufficient battery life when vision tasks are continuously running. Continuous video capture drains the battery of Google Glass in 30 minutes [2].

It is well known that a common culprit is the energy-expensive traffic of image data [8, 9]. Transferring high resolutions at high frame rates draws substantial power consumption from the analog-digital conversion, the sensor interface transactions, and the memory usage. Simply capturing 1080p frames at 30 frames per second consumes more than 2.4 W of system power measured on a MOTO Z smartphone. However, capturing and displaying 480p frames only consumes 1.3 W of system power.

Image resolution can create an interesting tradeoff for visual tasks: low resolution promotes low energy consumption, while high resolution promotes high imaging fidelity for high visual task accuracy. For example, as we explore with our AR marker-based pose estimation case study, lower resolutions suffice when an AR marker is close, but high resolutions are needed when the AR marker is far away or small. This tradeoff has been explored by several visual computing system works including marker pose estimation, object detection, and face recognition [3–6, 9, 10, 13, 14, 17]. We too advocate that mobile vision systems should be able to benefit from the ability to situationally sacrifice image resolution to save system energy when imaging detail is unnecessary.

Unfortunately, any change in sensor resolution leads to a substantial pause in frame delivery. This is illustrated in Figure 1a. We measure that reconfiguring sensor resolution in the Android OS prevents the application from receiving frames for about 267 ms, the equivalent of dropping 9 frames (working at 30 FPS) from vision processing pipelines [6]. Consequently, computer vision applications don't change resolutions at runtime, despite the significant energy savings at lower resolutions. For example, Augmented Reality applications such as "Augment" and "UnifiedAR" constantly work at 1080p, drawing 2.7 W of system power.

(a) In legacy systems, any change in sensor resolution leads to a substantial pause in frame delivery.

(b) Banner completely removes frame-to-frame latency for reconfiguring sensor resolution.

Figure 1: Compared with current systems, Banner enables rapid and seamless sensor resolution reconfiguration.

Thus, in this paper, we target image sensor *resolution reconfiguration latency* as a chief impediment of energy-efficient visual systems. Referring to [6], we break the resolution reconfiguration latency into two types of latency. *End-to-end reconfiguration latency* is the time between an application's request to change resolution and the time the application receives a frame of the new resolution. *Frame-to-frame latency* is the interval between two frames provided to the application in which the latter frame is configured at the new resolution.

The problem of long resolution reconfiguration latency is common across all mobile platforms, as we measured on different devices. In the Android OS, there is a 400 ms end-to-end reconfiguration latency [6]. In the Linux V4L2 media framework, we observe a 260 ms end-to-end reconfiguration latency. End-to-end reconfiguration latency in iOS also takes around 400 ms, as measured by timestamping a simple video capturing application we built in Xcode [7]. Similarly, end-to-end reconfiguration latency in Gstreamer with NVIDIA Libargus occupies more than 300 ms.

Almost all of the resolution reconfiguration latency originates from the operating system; at the sensor level, hardware register values are effective by the next frame [1, 11, 12, 15, 16]. We proposed several alternatives at the Android framework and HAL level for discussion in our previous work [6]. However, from deeper understanding of the Android OS, we find that the problem stems from the lower level system, i.e., media frameworks in the kernel. The underlying issue is that the kernel's media frameworks require user

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

MobiSys '19, June 17–21, 2019, Seoul, Republic of Korea

© 2019 Copyright held by the owner/author(s).

ACM ISBN 978-1-4503-6661-8/19/06.

<https://doi.org/10.1145/3307334.3328594>

space applications to frequently invoke a sequence of expensive system calls in order to request a new sensor resolution.

We introduce Banner: a system solution for rapid sensor resolution reconfiguration. Banner revolves around two techniques, as shown in Figure 2b. *Parallel reconfiguration* maintains video capture streams and schedules sensor reconfiguration in parallel while the application is processing the frame. *Format-oblivious memory management* removes repeated memory allocation from the reconfiguration procedure, avoiding expensive system calls initiated by the application. With these techniques, after the application requests a new resolution, Banner will not go through all steps in the sequential procedure again. Instead, Banner maintains the stream without reallocating buffers and then asynchronously reconfigures the sensor in parallel through only one `ioctl` call from the application. As a result, Banner completely eliminates frame-to-frame latency, as illustrated in Figure 1b, allowing for seamless multi-resolution frame capture. Banner also achieves the minimum possible end-to-end reconfiguration latency, fundamentally bounded by the pipeline latency of frame readout (usually larger than two frames). In extreme cases, if the application requests only one buffer to be allocated (not allowed for video streaming) or if the system allows frames already captured to drop, the end-to-end reconfiguration latency can further be reduced in Banner.

Our Banner prototype is implemented in the Linux kernel. We evaluate Banner's efficacy within the Linux V4L2 framework by running three workloads on a Jetson TX2 board with the Ubuntu system, including display-only, cloud-based offloading, and marker-based pose estimation. Our evaluation results confirm that Banner completely eliminates frame-to-frame latency, even for workloads operating at 30 FPS. Furthermore, Banner creates a 54% reduction in end-to-end reconfiguration latency (from 226 ms to 105 ms). The reduction in reconfiguration latency results in a 49% power consumption reduction by reconfiguring the resolution from 1080p to 480p compared with computationally downsampling 1080p to 480p, measured on a Jetson TX2 board. Altogether, Banner will unlock new classes of vision algorithms that can balance the resolution-based energy efficiency and accuracy tradeoffs to maximize the performance in a variety of continuous mobile vision tasks.

Please check our demo video at <https://youtu.be/uarayL6X5Lw>.

CCS CONCEPTS

• **Computer systems organization** → *Sensors and actuators*.

KEYWORDS

Operating systems; Energy efficiency; Reconfiguration; Resolution-based tradeoff; Efficient visual computing; Device drivers

ACM Reference Format:

Jinhan Hu, Alexander Shearer, Saranya Rajagopalan, Robert LiKamWa. 2019. Video: Banner – An Image Sensor Reconfiguration Framework for Seamless Resolution-based Tradeoffs. In *The 17th Annual International Conference on Mobile Systems, Applications, and Services (MobiSys '19)*, June 17–21, 2019, Seoul, Republic of Korea. ACM, New York, NY, USA, 2 pages. <https://doi.org/10.1145/3307334.3328594>

REFERENCES

- [1] Apple. Use hdr on your iphone, ipad, and ipod touch. <https://support.apple.com/en-us/HT207470>, 2018.

Figure 2: In Banner, most of the sequential procedures are avoided for reconfiguring sensor resolution. (1) Banner avoids repeated memory allocation; (2) Banner sets sensor format in parallel with user application.

- [2] S. Berenbaum. Google glass explorer edition has a 30-minute battery life while shooting video. <https://www.digitaltrends.com/mobile/google-glass-30-minute-videobattery/>, 2013.
- [3] M. Buckler, P. Bedoukian, S. Jayasuriya, and A. Sampson. Eva2: Exploiting temporal redundancy in live computer vision. In *Proceedings of the 45th Annual International Symposium on Computer Architecture, ISCA '18*, pages 533–546, Piscataway, NJ, USA, 2018. IEEE Press.
- [4] K. Ha, Z. Chen, W. Hu, W. Richter, P. Pillai, and M. Satyanarayanan. Towards wearable cognitive assistance. In *Proceedings of the 12th Annual International Conference on Mobile Systems, Applications, and Services, MobiSys '14*. ACM, 2014.
- [5] M. Haris, G. Shakhnarovich, and N. Ukita. Task-Driven Super Resolution: Object Detection in Low-resolution Images. *ArXiv e-prints*, page arXiv:1803.11316, Mar. 2018.
- [6] J. Hu, J. Yang, V. Delhivala, and R. LiKamWa. Characterizing the reconfiguration latency of image sensor resolution on android devices. In *Proceedings of the 19th International Workshop on Mobile Computing Systems & #38; Applications, HotMobile '18*. ACM, 2018.
- [7] R. M. Ibrahim. Camera. <https://github.com/rizwankce/Camera/>, 2019.
- [8] V. Kodukula, S. B. Medapuram, B. Jones, and R. LiKamWa. A case for temperature-driven task migration to balance energy efficiency and image quality of vision processing workloads. In *Proceedings of the 19th International Workshop on Mobile Computing Systems & #38; Applications, HotMobile '18*. ACM, 2018.
- [9] R. LiKamWa, B. Priyantha, M. Philipose, L. Zhong, and P. Bahl. Energy characterization and optimization of image sensing toward continuous mobile vision. In *Proceeding of the 11th Annual International Conference on Mobile Systems, Applications, and Services, MobiSys '13*. ACM, 2013.
- [10] T. Lin, P. Goyal, R. B. Girshick, K. He, and P. Dollár. Focal loss for dense object detection. *CoRR*, abs/1708.02002, 2017.
- [11] ON Semiconductor. AR0330 1/3-inch CMOS Digital Image Sensor, 2017. Rev. 18.
- [12] ON Semiconductor. MT9P031 1/2.5-Inch 5 Mp CMOS Digital Image Sensor, 2017. Rev. 10.
- [13] J. Redmon and A. Farhadi. Yolov3: An incremental improvement. *CoRR*, abs/1804.02767, 2018.
- [14] V. Růžička and F. Franchetti. Fast and accurate object detection in high resolution 4K and 8K video using GPUs. *ArXiv e-prints*, page arXiv:1810.10551, 2018.
- [15] Samsung. [in-depth look] fast, fun and in-focus: The galaxy s8 camera. <https://news.samsung.com/global/in-depth-look-fast-fun-and-in-focus-the-galaxy-s8-camera>, 2017.
- [16] Wikipedia. High-dynamic-range imaging. https://en.wikipedia.org/wiki/High-dynamic-range_imaging, 2018.
- [17] B. Zhang, A. Davoodi, and Y.-H. Hu. Exploring energy and accuracy tradeoff in structure simplification of trained deep neural networks. In *Proceedings of the 23rd Asia and South Pacific Design Automation Conference, ASPDAC '18*.