ORBCam: In-Sensor ORB Feature Processing for Ultra-Low-Power Visual-Inertial Odometry

Yiwen Liang[1] Yuxiang Peng[2] Guoquan Huang[2] Weidong Cao[1] Chuchu Chen[1]
[1]George Washington University [2]University of Delaware
{yiwen.liang, weidong.cao, chuchu.chen}@gwu.edu {yxpeng, ghuang}@udel.edu

Abstract

In visual–inertial odometry (VIO) systems, image readout and data movement between sensor and processor are increasingly recognized as the dominant power bottleneck, overshadowing on-chip computation. To address this, we present ORBCam, a cross-layer sensor–estimator co-design that eliminates image readout and directly generates motion-required feature measurements within the sensor subsystem. Instead of exporting images or descriptors, ORBCam transmits only quantized pixel coordinates and flow measurements to the host. In system-level simulations at 752×480 *resolution and 100 FPS, ORBCam is compared against a conventional image sensor consuming 7.88 mW for full-frame acquisition and transmission. ORBCam reduces sensing power to 0.59 mW, achieving up to 13.3*$\times$ *energy efficiency improvement while maintaining comparable odometry accuracy.*

1. Introduction & Related Work

Continuous ego-motion tracking is a key capability for robots, AR/VR wearables, and always-on vision systems. Visual–inertial odometry (VIO), which tightly couples camera observations with high-rate IMU measurements, has become one of the most practical and widely adopted solutions [5]. However, deploying VIO on these platforms is challenging under strict size, weight, and power (SWaP) constraints while requiring motion estimation at tens to hundreds of frames per second, making energy efficiency a critical concern. Extensive research efforts, together with advances in hardware acceleration, have significantly reduced the arithmetic cost of visual–inertial estimation [1, 3, 12, 14]. However, in many resource-constrained systems, the dominant energy consumption no longer comes from computation, but from frequent image acquisition and the movement of large volumes of pixel data from camera sensors to processors.

A conventional VIO pipeline is illustrated in Fig. 1(a).

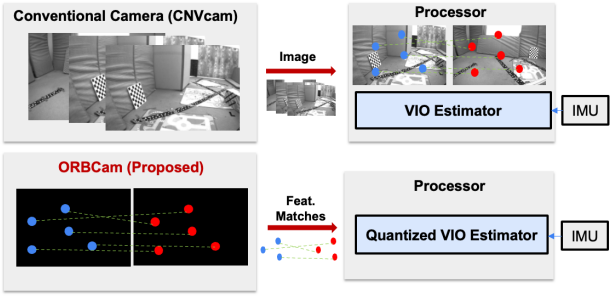


Figure 1. (a) Conventional VIO pipeline, where the camera acquires full-resolution images and transmits the massive raw pixels to the processor, which then performs original VIO estimation. (b) ORBCam pipeline, where the sensor directly produces quantized measurements without forming and transmitting images. Feature extraction, matching, motion validation, and quantization are performed within the sensor, and only compact quantized measurements are transmitted to the host estimator.

Image frames captured by the camera sensor are first transmitted to the host processor, where visual features (e.g., ORB keypoints and descriptors) are extracted and matched across frames to establish correspondences. These visual measurements are then fused with inertial data from the IMU to estimate ego-motion. In this pipeline, a CMOS image sensor (CIS) digitizes the entire pixel array through ADCs, followed by ISP processing and high-bandwidth off-chip transmission (e.g., MIPI CSI-2). As a result, all downstream processing relies on fully digitized images, leading to substantial energy overhead from ADC conversion, memory access, and data movement. Prior work has shown that data acquisition and transmission dominate sensor energy consumption [7].

However, this design reflects a fundamental mismatch between sensing and computation. While CIS is optimized to produce high-quality images for human perception, modern VIO algorithms ultimately rely on sparse feature correspondences across frames. Generating and transmitting full-resolution images is therefore inherently redundant. Although prior efforts have improved different stages of the

VIO pipeline, such as reducing computation [10], compressing data [6, 13], or moving processing closer to the sensor [3, 4], they still depend on full-frame acquisition. Consequently, the dominant cost of pixel-level digitization and transmission remains largely unchanged. *Can task-relevant features be generated directly within the sensor, eliminating the image as an intermediate representation?* This motivates us to a paradigm shift in sensor design.

Building on this insight, we envision a machine-centric camera for VIO, one that generates features (and their correspondences), not images, to address sensing bottlenecks. We present ORBCam, an algorithm-hardware co-design architecture that restructures the sensing pipeline from the ground up. Instead of digitizing full image frames, ORBCam performs feature detection, description, matching, and motion validation directly within a mixed-signal sensing architecture, exporting only compact and quantized keypoint correspondences. These measurements are then processed by a quantization-aware VIO estimator (Fig. 1(b)). By breaking the conventional full-frame sensing model, ORBCam reduces sensor energy and off-chip bandwidth while preserving the information required for accurate motion estimation.

2. ORBCam: A Machine-Centric Visual Sensor

We propose ORBCam, an ORB-based machine-centric camera designed for ultra-low-power VIO, as illustrated in Fig. 2(b)-(c).

2.1. ORBCam Architecture and Operation

Architecture overview. Fig. 2(c) illustrates the proposed in-sensor ORB processing architecture, which integrates three tightly coupled components: (i) a pixel array augmented with per-pixel processing elements (PEs), where each PE includes a sample-and-hold (S/H) buffer to latch the pixel intensity for reuse and a modified single-slope ADC (comparator) to perform in-pixel intensity evaluation. (ii) shared row/column analog interconnect buses that enable data exchange and access among pixels along the same row or column. (iii) a lightweight on-chip digital logic unit, local memory, and a microcontroller to coordinate PE/interconnect configuration, schedule comparisons, and manage intermediate feature data and final matched outputs. Compared to a conventional image sensor, newly added components are highlighted in blue.

Timing diagram. Fig. 2(a) illustrates the steady-state execution pipeline. Within each frame period, ORB processing consists of FAST detection, descriptor generation, matching, outlier rejection, and measurement quantization. These stages are triggered after exposure and precede data transmission. The quantized measurements are transmitted to the backend for motion estimation using a quantization-aware VIO estimator. Once the pipeline is filled, ORB processing of frame t is executed in parallel with the exposure of frame $t+1$. Since processing latency is shorter than exposure time, feature extraction and matching are effectively hidden behind exposure latency. As a result, the steady-state frame rate is primarily bounded by exposure rather than computation.

2.2. ORB Feature Extraction

2.2.1. FAST Detection and Orientation

For each candidate pixel, ORBCam performs FAST detection followed by orientation estimation using lightweight mixed-signal operations. The FAST engine (red box in Fig. 2(c)) scans as a sliding window across the pixel array and evaluates the standard 16-point FAST circle around each center pixel. For FAST evaluation, the intensities of the 16 surrounding pixels are first sampled onto their per-pixel comparator local capacitor (C_{az1}). The center pixel intensity (I_C), stored in the S/H buffer, is then broadcast through the row/column analog interconnect buses to enable capacitive differencing with each ring sample. This forms the analog intensity differences $(I_i - I_C)$, which are then evaluated by a programmable comparator against thresholds $\pm T$. The comparator generates two 16-bit binary masks corresponding to the brighter and darker tests. These masks are forwarded to the digital logic unit, which performs a circular contiguous run-length test with length 9. For pixels satisfying the FAST criterion, ORBCam estimates the keypoint orientation using a hardware-efficient approximation. Instead of full moment computation, we derive a pseudo-gradient from four axis-aligned samples: $g_x = I_5 - I_{13}$ and $g_y = I_1 - I_9$. The signs and relative magnitudes of (g_x, g_y) are quantized into one of eight discrete orientation bins, which are used to steer the subsequent rBRIEF descriptor generation. Finally, the digital logic performs score-based selection and enforces a minimum spatial separation to retain a stable set of keypoints for subsequent descriptor construction and matching.

2.2.2. Binary Descriptor Generation

For each detected keypoint, a 256-bit rBRIEF descriptor is constructed from pairwise intensity comparisons within its local neighborhood. rBRIEF represents a feature using binary comparisons between selected pixel pairs, which makes it particularly suitable for hardware implementation based on comparator operations. To achieve rotation invariance, the sampling pattern is rotated according to the quantized orientation estimated in the previous stage. Instead of performing rotation every time, ORBCam precomputes rotated sampling patterns for eight orientation bins and stores them in a look-up table (LUT). Given the keypoint's orientation bin, the corresponding set of pixel in-

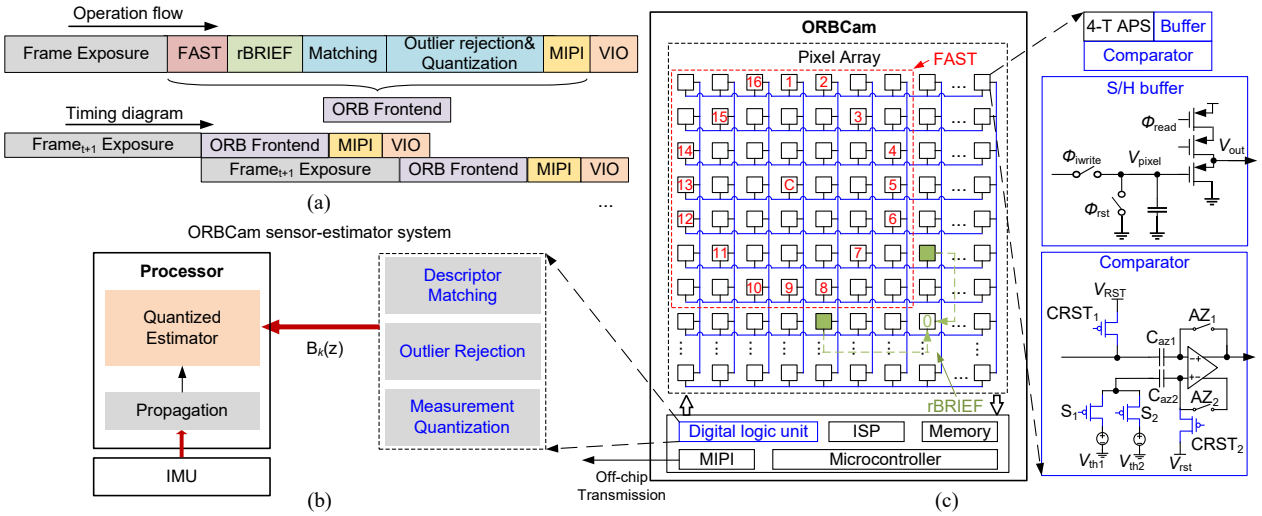


Figure 2. ORBCam architecture and execution flow. (a) Timing diagram of the ORBCam visual processing pipeline. (b) System overview of the ORBCam-based VIO architecture. ORBCam performs in-sensor feature extraction without full-frame image readout. Digital logic unit conducts descriptor matching, state-free outlier rejection, and measurement quantization before off-chip transmission. The processor receives only quantized visual measurements $B_k(z)$, which are fused with IMU propagation in a quantization-aware estimator. (c) In-sensor ORB processing architecture with augmented pixel array and digital logic unit. Newly added components are highlighted in blue. Circuit design of per-pixel processing element enabling local storage and intensity comparison.

dex pairs is fetched from the LUT. The descriptor bits are then generated through sequential pairwise comparisons. For each pair, the two pixel voltages are routed through the row/column interconnect buses to the shared PE comparator (green points in Fig. 2(c)), which performs a single binary comparison result. After generating all 256 bits of the descriptor, the resulting descriptors are written to local memory together with the keypoint coordinate and then forwarded to the matching engine.

2.2.3. Feature Matching and Motion Validation

After descriptor construction, ORBCam performs feature matching and motion validation to produce compact motion-consistent measurements for the backend estimator. These operations are implemented in the on-chip digital logic unit using fixed-point arithmetic and minimal control logic, introducing negligible area and latency overhead. Matching is based on 256-bit Hamming distance, computed via parallel XOR and a popcount adder tree, with symmetric cross-check to retain only mutually consistent correspondences. Outlier rejection employs a *state-free* flow-consistency gate: a chi-square test on per-match optical flow $\mathbf{f}_i$ against the frame-level mean and covariance $(\bar{\mathbf{f}}, \Sigma_f)$, combined with a maximum-displacement bound. Surviving matches are differentially quantized and transmitted as compact measurements to the quantization-aware VIO backend.

3. Experimental Results

3.1. Experimental Methodology

Hardware configuration and evaluation methodology. ORBCam is designed in a standard 65-nm Complementary Metal-Oxide-Semiconductor (CMOS) process, consistent with current commercial image sensor technology [11]. The digital logic unit is described in RTL and synthesized using a Synopsys/Cadence Electronic Design Automation (EDA) flow, while on-chip memory is emulated using synthesized SRAM blocks. The pixel array adopts a conventional 4-T active pixel sensor (APS) design with additional circuitry to support in-sensor feature extraction, and the Energy and latency are estimated using circuit-level analytical models derived from prior studies [8, 9]. These models account for pixel array readout, mixed-signal processing, ADC operations, SRAM accesses, and off-chip I/O transmission.

Baseline. We compare ORBCam against a conventional CMOS image sensor (CNV-CIS) baseline that captures and transmits full-resolution image frames. To ensure a fair comparison, both systems operate at a resolution of 752×480 and a frame rate of 100 FPS under the same technology assumptions. For off-chip communication, we assume a four-lane MIPI CSI-2 interface operating at 1.2 Gbps per lane.

3.2. ORBCam Results

We evaluate ORBCam across three feature budgets, while both ORBCam and the CNV-CIS baseline operate at 100

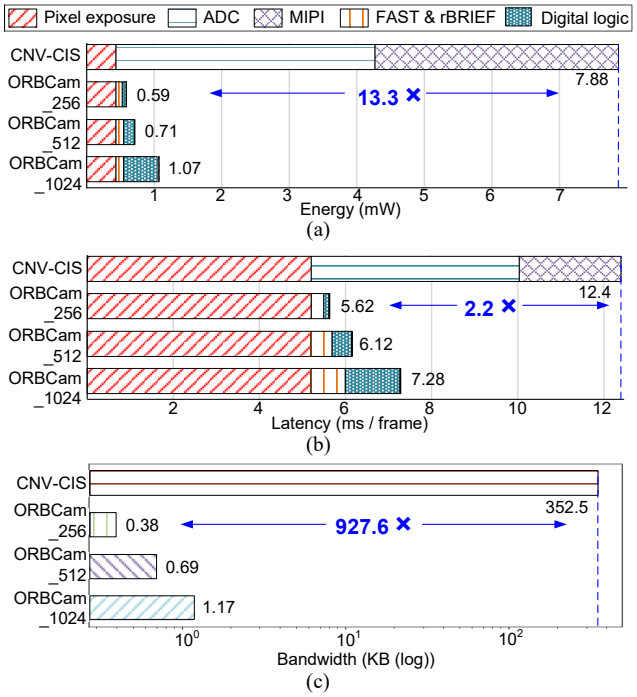
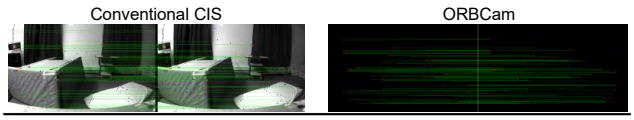


Figure 3. ORBCam evaluation results under 5-bit measurement quantization. (a) ORBCam energy breakdown. At a feature budget of 256, ORBCam achieves 13.3× energy efficiency compared with a conventional image sensor (CNV-CIS). (b) ORBCam latency breakdown. At the same feature budget, ORBCam achieves a 2.2× speedup compared with CNV-CIS. (c) Per-frame communication bandwidth comparison. ORBCam reduces per-frame communication by up to 927.6× compared with CNV-CIS.

FPS. Fig. 3(a) presents the detailed energy breakdown. The ORBCam energy savings primarily stem from two architectural factors: (i): The conventional pipeline digitizes (ADC) and transmits (MIPI) all pixels per frame. ORBCam removes full-frame ADC conversion and high-bandwidth pixel transmission, transmitting only compact quantized feature coordinates. (ii): Most feature detection and comparison operations are performed in the analog domain, avoiding energy-intensive digital computation and memory accesses. These results demonstrate that moving feature extraction into the sensor fundamentally shifts the energy bottleneck. At a budget of 256 features per frame, ORBCam consumes only 0.59 mW, achieving a 13.3× improvement in sensing energy efficiency compared with the CNV-CIS baseline. Fig. 3(b) shows the latency breakdown across different stages. Since ORBCam significantly reduces both processing latency and transmission time, the system is no longer limited by computation or I/O but instead by the exposure time of the image sensor. Fig.3(c) compares the per-frame data size under different visual front-end representations. For the conventional CIS pipeline, which transmits full-resolution images with 8-bit encoding, the communication cost reaches hundreds of kilobytes (KB) per frame. In contrast, ORBCam transmits only compact multi-bit motion measurements, resulting in a substantially smaller data size per feature. At a feature budget of 256, ORBCam achieves up to a 927.6× reduction in communication bandwidth compared with the CNV-CIS baseline.



	Energy (mW)	Latency (ms)	Bandwidth (KB)	ATE (deg / m)
Conventional	7.88	12.4	352.5	1.590 / 0.108
ORBCam	0.59	5.62	0.38	1.724 / 0.129
Improvement	**13.3 ×**	**2.2 ×**	**927.6 ×**	——

ORBCam adopts a 256-feature budget with 5-bit quantization.

Figure 4. Comparison between conventional image sensor and ORBCam.

3.3. VIO Performance

We compare our quantization-aware VIO system — which integrates ORBCam outputs with measurement quantization and a quantization-aware state estimator against a VIO Baseline that uses the original CNV-CIS front-end with the standard VIO pipeline. To evaluate whether the proposed sensor-side ORB front-end preserves estimation performance, we report VIO accuracy on the EuRoC MAV [2] benchmark under two representative back-ends: an MSCKF-style filter and a sliding-window SLAM estimator. For each sequence, we report the average Absolute Trajectory Error (ATE), expressed in degrees for orientation error and meters for position error. As shown in Fig. 4, the proposed system achieves accuracy on par with the baseline across both back-ends, demonstrating that the quantization-aware design incurs minimal estimation degradation while enabling significant reductions in sensor-side data throughput.

4. Conclusion

This paper presents ORBCam, a machine-centric sensing architecture that directly generates quantized sparse ORB feature measurements instead of full-resolution images. By tightly co-designing sensing and feature extraction, ORBCam eliminates redundant pixel-level acquisition and transmission, substantially reducing front-end energy consumption. Furthermore, the sensing architecture is jointly optimized with backend VIO algorithms to enable efficient end-to-end feature-based processing. Experimental results on the EuRoC dataset demonstrate that ORBCam achieves comparable motion estimation accuracy to conventional image-sensor-based pipelines.

References

[1] Reality labs chief scientist outlines a new compute architecture for true ar glasses. https://www.roadtovr.com/michael-abrash-iedm-2021-compute-architecture-for-ar-glasses/. 1

[2] Michael Burri, Janosch Nikolic, Pascal Gohl, Thomas Schneider, Joern Rehder, Sammy Omari, Markus W Achtelik, and Roland Siegwart. The euroc micro aerial vehicle datasets. *The International Journal of Robotics Research*, 35 (10):1157–1163, 2016. 4

[3] Chuchu Chen, Yuxiang Peng, and Guoquan Huang. Qvio2: Quantized map-based visual-inertial odometry. In *Proc. International Conference on Robotics and Automation*, Atlanta, GA, 2025. 1, 2

[4] Jorge Gomez, Saavan Patel, Syed Shakib Sarwar, Ziyun Li, Raffaele Capoccia, Zhao Wang, Reid Pinkham, Andrew Berkovich, Tsung-Hsun Tsai, Barbara De Salvo, and Chiao Liu. Distributed on-sensor compute system for ar/vr devices: A semi-analytical simulation framework for power estimation, 2022. 2

[5] Guoquan Huang. Visual-inertial navigation: A concise review. In *Proc. International Conference on Robotics and Automation*, Montreal, Canada, 2019. 1

[6] Simon Lynen, Torsten Sattler, Michael Bosse, Joel A Hesch, Marc Pollefeys, and Roland Siegwart. Get out of my lab: Large-scale, real-time visual-inertial localization. In *Robotics: Science and Systems*, page 1, 2015. 2

[7] Tianrui Ma, Adith Jagadish Boloor, Xiangxing Yang, Weidong Cao, Patrick Williams, Nan Sun, Ayan Chakrabarti, and Xuan Zhang. Leca: In-sensor learned compressive acquisition for efficient machine vision on the edge. In *Proceedings of the 50th Annual International Symposium on Computer Architecture*, pages 1–14, 2023. 1

[8] Tianrui Ma, Yu Feng, Xuan Zhang, and Yuhao Zhu. Camj: Enabling system-level energy modeling and architectural exploration for in-sensor visual computing. In *Proceedings of the 50th annual international symposium on computer architecture*, pages 1–14, 2023. 3

[9] Tianrui Ma, Zhe Gao, Zhe Chen, Ramakrishna Kakarala, Charles Shan, Weidong Cao, and Xuan Zhang. Systematic methodology of modeling and design space exploration for cmos image sensors. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 2025. 3

[10] Eric J Msechu, Stergios I Roumeliotis, Alejandro Ribeiro, and Georgios B Giannakis. Decentralized quantized kalman filtering with scalable communication cost. *IEEE Transactions on Signal Processing*, 56(8):3727–3741, 2008. 2

[11] Jun Ohta. *Smart CMOS image sensors and applications*. CRC press, 2020. 3

[12] Yuxiang Peng, Chuchu Chen, and Guoquan Huang. Quantized visual-inertial odometry. In *Proc. International Conference on Robotics and Automation*, Yokohama, Japan, 2024. 1

[13] Torsten Sattler, Michal Havlena, Filip Radenovic, Konrad Schindler, and Marc Pollefeys. Hyperpoints and fine vocabularies for large-scale location recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2102–2110, 2015. 2

[14] Amr Suleiman, Zhengdong Zhang, Luca Carlone, Sertac Karaman, and Vivienne Sze. Navion: A 2-mw fully integrated real-time visual-inertial odometry accelerator for autonomous navigation of nano drones. *IEEE Journal of Solid-State Circuits*, 54(4):1106–1119, 2019. 1