EntoBench: A Benchmark Suite and Evaluation Framework for Insect-Scale Robotics

Derin Ozturk, Nick Cebry, Angela Cui, Christopher Batten
School of Electrical and Computer Engineering, Cornell University, Ithaca, NY
{ddo26, nfc35, ayc62, cbatten}@cornell.edu

***Abstract*—Insect-scale robots face significant size, weight, power, and timing constraints that complicate system design, restrict demonstrations to controlled lab environments, and ultimately limit the achievable autonomy of these systems. This poster will present our ongoing work on EntoBench, a comprehensive benchmark suite and evaluation framework that addresses these challenges by evaluating latency, energy, and peak power on resource-constrained microcontrollers.**

I. Motivation

Insect-scale robots, typically characterized by lengths under 5cm and masses below 5g, are a rapidly growing area of robotics research. These platforms promise transformative capabilities in fields such as search-and-rescue and environmental monitoring. At these scales, familiar physical intuitions begin to break down: scaling laws introduce new constraints on actuation, sensing, and control, requiring roboticists to employ micro-intuition [22] and look towards biology for inspiration [14] in the robot design process. The effect is an explosion of diversity across demonstrated systems (e.g., flyers [10, 12, 29, 30, 50, 61], crawlers [2, 12, 24, 34, 47, 54, 62], jumpers [1, 8, 32, 54], swimmers [53, 60], gliders [18, 31, 48], and striders [21, 55, 57]) reflecting a wide range of form factors, actuation strategies, and control architectures tailored for operation at the insect scale.

A major trend in recent years is the push toward full autonomy [14, 25] in insect-scale robots, encompassing *sensing*, *control*, *compute*, and *power* autonomy. While most demonstrated systems currently rely on external position tracking, off-board computation, and tethered power sources, next-generation platforms aim to be self-sufficient: sensing and understanding their environment and internal state, making control decisions in real time, and doing so under tight size, weight, power, and timing constraints. Among these four pillars of autonomy, we argue that *compute autonomy* is the most critical to address first. Processor selection has recently been emphasized for its influence on algorithmic feasibility and efficiency in insect-scale robots [14]. The choice of onboard compute directly determines what sensing and control strategies are feasible and what power budget is sustainable, setting the stage for a virtuous robot-hardware-software co-design loop. Furthermore, optimized compute systems may unlock new capabilities for these robots, beyond enabling operation outside the lab.

The challenge of *compute autonomy* creates a natural opportunity for the RoboArch community to contribute low-level software optimizations, energy-aware system design, and custom compute architectures for insect-scale robots. However, to enable meaningful progress, we need benchmark suites and evaluation frameworks that reflect the realities of these insect-scale platforms. Existing robotics benchmark suites [5, 6, 9, 42] do not meet these needs for several reasons (see Table I). First, they do not reflect current insect-scale robotics algorithms or pipelines. Second, they assume an abundance of compute resources and software stacks that are impractical for insect-scale deployments. Third, their modularity and extensibility are limited in practice. Some suites simply aggregate open source projects and/or make it difficult to easily add new kernels. Fourth, they neglect energy as a first-class metric, measuring it only coarsely or only focusing on average power. Lastly, while some suites do not evaluate full end-to-end deployments, we view this as an important future direction. Since such deployments remain rare at the insect scale, we focus on individual kernels for this current work.

TABLE I. Comparison of Robotics Benchmark Suites

	MAV Bench	Robot Perf	RTR Bench	Ro Wild	Ento Bench
Insect Scale	✗	✗	✗	✗	✓
Resource Constrained	✗	✗	✗	✗	✓
Modular & Extensible	✓	✓	✓	✓	✓
Energy & Power Focused	✓	✓	✗	✗	✓
End-to-End	✓	✗	✗	✓	*

In this work, we introduce EntoBench, a new benchmark suite and evaluation framework tailored for insect-scale robotics. EntoBench provides a focused set of fundamental kernels representing key stages of the current insect-scale robot pipeline, enabling researchers to effectively evaluate performance and energy efficiency on resource-constrained microcontrollers in a reproducible manner. By doing so, EntoBench lays the groundwork for principled robot-hardware-software co-design at the insect scale.

II. EntoBench

EntoBench is a benchmark suite and evaluation framework purpose-built for insect-scale robotics. Unlike many existing robotics benchmark suites, EntoBench deliberately targets the tight constraints imposed by these ultra-small platforms. In this section we describe the evaluation framework design goals before providing a high level description of our catalog of kernels.

A. Benchmark Suite and Evaluation Framework Design Goals

Representative of the Insect-Scale Robot Pipeline – The suite aims to capture essential computational stages that current insect-scale robots are targeting—perception, state estimation, and control—while also acknowledging that additional stages (e.g., mapping, planning) practically exceed the capabilities of microcontrollers and are less relevant for insect-scale robot tasks. Kernels are curated to reflect algorithms more relevant for insect-scale robots, including direct and inspired implementations of those demonstrated in the context of insect-scale robots, and additionally others scaled down from slightly larger platforms, such as nanodrones.

Suitable for Resource Constrained Platforms – EntoBench does not require abundant external memory, double precision floating point hardware, sophisticated cache-based memory hierarchies, or external libraries and middleware (e.g., ROS, OpenCV). It is designed for microcontrollers with no external memory and limited SRAM and flash. We avoid dynamic memory allocation and virtual functions, and rely heavily on template meta-programming for compile-time parameterization, staying closer to code structures that are viable on real-time embedded systems at the insect scale.

Modular, Extensible, and Configurable Design – Each kernel is implemented as a small standalone module, with minimal dependencies, enabling easy integration, composition, and deployment across different ARM Cortex-M architectures and microarchitectural simulators such as gem5 [38]. Kernels are written against generic problem interfaces (i.e., task definitions), and evaluated via a reusable harness that handles I/O and orchestrates experiments. Our use of modern C++ and metaprogramming enables users to iterate on software optimizations, switch between single- and double-precision floating point, implement new kernels, or define entirely new problem types. The framework supports both validation (correctness), and evaluation (latency, energy, and accuracy), providing a structured methodology for benchmarking across implementations.

Energy as a First-Class Metric – EntoBench recognizes that insect-scale robots operate on extremely constrained energy budgets and thus treats energy as a first-class concern. Rather than relying on low-order models (e.g., FLOP counts) [20, 58, 63], EntoBench integrates direct energy measurement via a commercially available power measurement device, combined with a logic-analyzer for precise timing of kernels within a region of interest. This enables an apples-to-apples comparison across algorithms, not just in speed, but in energy feasibility for untethered insect-scale deployments. We also capture peak power consumption, which is critical for power electronics design, particularly under transient loads.

End-to-End Pipelines – End-to-end evaluation, from sensing to actuation, is desirable in robotics, as it reflects realistic workloads beyond isolated kernels. Although fully autonomous insect-scale robots remain out of reach, recent advances suggest this will soon become essential. EntoBench acknowledges this trajectory through the design of its modular benchmarks with deployment of end-to-end pipelines as future work.

B. Catalog of Kernels

Guided by our design goals, EntoBench implements kernels carefully selected for their relevance and applicability at the frontier of insect-scale robotics.

Perception – Our perception kernels reflect the growing importance of onboard feature extraction and visual motion estimation at the insect-scale. Current kernels include feature detectors and descriptors [7, 28, 37, 51, 52] and multiple optical flow methods [3, 27, 41, 56, 63] that span a range of complexity and computational demand.

TABLE II. LATENCY, ENERGY, AND POWER RESULTS

	Latency (10^3 cycles)			Energy (nJ)			Peak Power (mW)		
Kernel	M4	M33	M7	M4	M33	M7	M4	M33	M7
FAST	1626.5	1080.9	3113.4	1118.6	216.84	1016.3	117.8	38.6	106.2
ORB	9060.7	6559.6	9722.9	6108.8	1335.4	3302.7	126.9	38.2	106.6
LKOF	361.0	243.4	195.6	243.5	39.6	195.2	118	40.4	107.01
IIOF	232.2	195.6	210.1	158.7	8.2	85.34	118.8	24.3	116.96
Rel5Pt	129.7	105.8	92.1	92.9	11.51	40	127.7	40.7	132.3
Rel8Pt	51.5	38.7	32.9	36	4.34	13.64	125.3	39.5	136.5
TinyMPC	21	15.5	30.7	11.2	1.69	9.48	118.3	44.2	107.6

State Estimation – This stage includes attitude filters [19, 39, 40, 43], extended Kalman filters [17, 43, 59, 63], factor graph chains [46], and absolute and relative geometric pose estimators [16, 23, 33, 45]. We consider minimal solvers and non-minimal solvers for our pose estimators, including those that may assume extra information such as a known gravity vector as well as their deployment in a RANSAC framework [13, 49].

Control – EntoBench focuses on advanced control strategies beyond basic PID, including optimal controllers for linearized systems [15, 17], constrained formulations, such as TinyMPC [44], and more advanced strategies, such as geometric tracking control [36] and sliding window control [11], which have been demonstrated on flapping wing insect-scale robots.

III. PRELIMINARY RESULTS

We present preliminary results for seven representative kernels evaluated across Cortex-M4, M33, and M7 microcontrollers (see Table II). For perception, we benchmark FAST feature detection and ORB feature detection and description, and Lucas-Kanade and image interpolation optical flow, using sequences from the Middlebury datasets [4, 26]. For state estimation, we evaluate the 5- and 8-point relative pose algorithms using synthetic data as in [35]. In control, we evaluate TinyMPC on a quadrotor figure-eight trajectory. Experiments use data and kernel parameters fitting within the 128KB SRAM of the M4, enabling comparison across platforms.

To contextualize these results, Table III summarizes key architectural differences across the three Cortex-M architectures. These early results already highlight critical trade-offs. The M7 underperforms on several kernels due to suboptimal memory placement from a vendor-provided linker script that places the stack in AXI SRAM, bypassing faster tightly coupled memory. In contrast, the M33 demonstrates superior energy efficiency, primarily because microcontroller manufacturers imple-

TABLE III. CORTEX-M ARCHITECTURES

MCU	Key Features
Cortex-M4	3-stage pipeline (ARMv7E-M), up to ∼200 MHz, optional SP FPU, widely available even in ultra-compact packaging (e.g., WLCSP).
Cortex-M33	3-stage pipeline (ARMv8-M), up to ∼200 MHz, optional SP FPU, optional coprocessor interface, less commonly available in ultra-compact packaging (e.g., WLCSP).
Cortex-M7	6-stage superscalar pipeline with branch prediction (ARMv7E-M), up to ∼600 MHz, optional SP or DP FPU, optional I/D caches, optional tightly coupled memory (TCM), widely available even in ultra-compact packaging, typically larger than M4/M33

ment this newer architecture on more advanced semiconductor technology nodes. However, M33 microcontrollers are less commonly available in ultra-compact packages (e.g., WLCSP) required by insect-scale robotics. These findings underline an early key insight: better compute performance (M7) or energy efficiency (M33) is not cost-free, as each introduces challenges in managing memory allocation and available packaging.

ACKNOWLEDGMENTS

This work was supported in part by NSF CSSI Award #2311890 and a research gift from Intel and Xilinx.

REFERENCES

[1] C. A. Aubin, R. H. Heisser, O. Peretz, J. Timko, J. Lo, E. F. Helbling, S. Sobhani, A. D. Gat, and R. F. Shepherd. Powerful, Soft Combustion Actuators for Insect-Scale Robots. *Science*, 381, Sep 2023.

[2] A. T. Baisch and R. Wood. Design and Fabrication of the Harvard Ambulatory Micro-Robot. *Int'l Symp. on Robotics Research (ISRR)*, Aug 2011.

[3] S. Baker and I. Matthews. Lucas-Kanade 20 Years On: A Unifying Framework. *International Journal of Computer Vision*, 56(3):221–255, Feb 2004.

[4] S. Baker, S. Roth, D. Scharstein, M. J. Black, J. Lewis, and R. Szeliski. A Database and Evaluation Methodology for Optical Flow. In *2007 IEEE 11th International Conference on Computer Vision*, pages 1–8, Rio de Janeiro, Brazil, 2007. IEEE.

[5] M. Bakhshalipour and P. B. Gibbons. Agents of Autonomy: A Systematic Study of Robotics on Modern Hardware. *Measurements and Analysis of Computing Systems (MACS)*, Dec 2023.

[6] M. Bakhshalipour, M. Likhachev, and P. B. Gibbons. RTRBench: A Benchmark Suite for Real-Time Robotics. *Int'l Symp. on Performance Analysis of Systems and Software (ISPASS)*, May 2022.

[7] H. Bay, T. Tuytelaars, and L. Van Gool. SURF: Speeded Up Robust Features. In A. Leonardis, H. Bischof, and A. Pinz, editors, *Computer Vision – ECCV 2006*, volume 3951, pages 404–417. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006.

[8] S. Bergbreiter and K. S. Pister. Design of an Autonomous Jumping Microrobot. *Int'l Conf. on Robotics and Automation (ICRA)*, Apr 2007.

[9] B. Boroujerdian, H. Genc, S. Krishnan, W. Cui, A. Faust, and V. J. Reddi. MAVBench: Micro Aerial Vehicle Benchmarking. *Int'l Symp. on Microarchitecture (MICRO)*, Oct 2018.

[10] Y. Chen, E. F. Helbling, N. Gravish, K. Ma, and R. J. Wood. Hybrid Aerial and Acquatic Locomotion in an At-Scale Robotic Insect. *Int'l Conf. on Intelligent Robots and Systems (IROS)*, Aug 2015.

[11] P. Chirarattananon, K. Y. Ma, and R. J. Wood. Adaptive Control of a Millimeter-Scale Flapping-Wing Robot. *Bioinspiration & Biomimetics*, 9(2):025004, May 2014.

[12] Y. M. Chukewad, J. James, A. Singh, and S. Fuller. RoboFly: An Insect-Sized Robot With Simplified Fabrication That Is Capable of Flight, Ground, and Water Surface Locomotion. *IEEE Transactions on Robotics*, pages 2025–2040, May 2021.

[13] O. Chum, J. Matas, and J. Kittler. Locally Optimized RANSAC. In G. Goos, J. Hartmanis, J. Van Leeuwen, B. Michaelis, and G. Krell, editors, *Pattern Recognition*, volume 2781, pages 236–243. Springer Berlin Heidelberg, Berlin, Heidelberg, 2003.

[14] G. C. H. E. de Croon, J. J. G. Dupeyroux, S. B. Fuller, and J. A. R. Marshall. Insect-inspired AI for Autonomous Robots. *Science Robotics*, 7(67), Jun 2022.

[15] D. Dhingra, K. Kaheman, and S. B. Fuller. Modeling and LQR Control of Insect Sized Flapping Wing Robot. *Computing Research Repository (CoRR)*, arXiv:2406.20061, Jun 2024.

[16] Y. Ding, J. Yang, V. Larsson, C. Olsson, and K. Åström. Revisiting the P3P Problem. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4872–4880, Vancouver, BC, Canada, Jun 2023. IEEE.

[17] N. Doshi, K. Jayaram, S. Castellanos, S. Kuindersma, and R. J. Wood. Effective Locomotion at Multiple Stride Frequencies Using Proprioceptive Feedback on a Legged Microrobot. *Bioinspiration & Biomimetics*, 14(5):056001, Jul 2019.

[18] D. S. Drew, N. O. Lambert, C. B. Schindler, , and K. S. J. Pister. Toward Controlled Flight of the Ionocraft: A Flying Microrobot Using Electrohydrodynamic Thrust With Onboard Sensing and No Moving Parts. *IEEE Robotics and Automation Letters (RAL)*, 3(4):2807–2813, Oct 2018.

[19] H. Fourati, N. Manamanni, L. Afilal, and Y. Handrich. Nonlinear Attitude Estimation Based on Fusion of Inertial and Magnetic Sensors: Bio-logging Application. *IFAC Proceedings Volumes*, 42(19):349–354, 2009.

[20] S. Fuller, Z. Yu, and Y. P. Talwekar. A Gyroscope-Free Visual-Inertial Flight Control and Wind Sensing System for 10-mg Robots. *Science Robotics*, 7(72), Nov 2022.

[21] H. Gao, S. Jung, and E. F. Helbling. High-Speed Interfacial Flight of an Insect-Scale Robot. *Int'l Conf. on Robotics and Automation (ICRA)*, May 2024.

[22] A. Ghosh. Scaling Laws. *Mechanics Over Micro and Nano Scales*, May 2011.

[23] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, Cambridge, UK ; New York, 2nd ed edition, 2003.

[24] K. Hayaram, J. Shum, S. Castellanos, E. F. Helbling, and R. Wood. Scaling Down an Insect-Size Microrobot, HAMR-VI Into HAMR-Jr. *Int'l Conf. on Robotics and Automation (ICRA)*, May 2020.

[25] E. F. Helbling and R. J. Wood. A Review of Propulsion, Power, and Control Architectures for Insect-Scale Flapping-Wing Vehicles. *Applied Mechanics Reviews*, 70(1):010801, Jan 2018.

[26] H. Hirschmuller and D. Scharstein. Evaluation of Cost Functions for Stereo Matching. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, Minneapolis, MN, USA, Jun 2007. IEEE.

[27] D. Honegger, L. Meier, P. Tanskanen, and M. Pollefeys. An Open Source and Open Hardware Embedded Metric Optical Flow CMOS Camera for Indoor and Outdoor Applications. In *2013 IEEE International Conference on Robotics and Automation*, pages 1736–1741, Karlsruhe, Germany, May 2013. IEEE.

[28] D. Hutchison, T. Kanade, J. Kittler, J. M. Kleinberg, F. Mattern, J. C. Mitchell, M. Naor, O. Nierstrasz, C. Pandu Rangan, B. Steffen, M. Sudan, D. Terzopoulos, D. Tygar, M. Y. Vardi, G. Weikum, M. Calonder, V. Lepetit, C. Strecha, and P. Fua. BRIEF: Binary Robust Independent Elementary Features. In K. Daniilidis, P. Maragos, and N. Paragios, editors, *Computer Vision – ECCV 2010*, volume 6314, pages 778–792. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010.

[29] N. T. Jafferis, E. F. Helbling, M. Karpelson, and R. Wood. Untethered Flight of an Insect-Sized Flapping-Wing Microscale Aerial Vehicle. *Nature*, 57-:491–495, Jun 2019.

[30] J. James, V. Iyer, Y. Chukewad, S. Gollakota, and S. B. Fuller. Liftoff of a 190 Mg Laser-Powered Aerial Vehicle: The Lightest Wireless Robot to Fly. *Int'l Conf. on Robotics and Automation (ICRA)*, May 2018.

[31] K. Johnson, V. Arroyos, A. Ferran, R. Villanueva, D. Yin, T. Elberier, A. Aliseda, S. Fuller, V. Iyer, and S. Gollakota. Solar-Powered Shape-Changing Origami Microfliers. *Science Robotics*, 8(82), Sep 2023.

[32] M. Kovac, M. Fuchs, A. Guignard, J.-C. Zufferey, and D. Floreano. A Miniature 7g Jumping Robot. *Int'l Conf. on Robotics and Automation (ICRA)*, May 2008.

[33] Z. Kukelova, M. Bujnak, and T. Pajdla. Closed-Form Solutions to Minimal Absolute Pose Problems with Known Vertical Direction. In R. Kimmel, R. Klette, and A. Sugimoto, editors, *Computer Vision – ACCV 2010*, volume 6493, pages 216–229. Springer Berlin Heidelberg, Berlin, Heidelberg, 2011.

[34] Y. Lai, C. Zang, G. Luo, S. Xu, R. Bo, J. Zhao, Y. Yang, T. Jin, Y. Lan, Y. Wang, L. Wen, W. Pang, and Y. Zhang. An Agile Multimodal Microrobot with Architected Passively Morphing Wheels. *Science Advances*, Dec 2024.

[35] V. Larsson and contributors. PoseLib - Minimal Solvers for Camera Pose Estimation, 2020.

[36] T. Lee, M. Leok, and N. H. McClamroch. Geometric Tracking Control of a Quadrotor UAV on SE(3). In *49th IEEE Conference on Decision and Control (CDC)*, pages 5420–5425, Atlanta, GA, Dec 2010. IEEE.

[37] D. G. Lowe. Distinctive Image Features from Scale-Invariant Keypoints. *International Journal of Computer Vision*, 60(2):91–110, Nov 2004.

[38] J. Lowe-Power, A. M. Ahmad, A. Akram, M. Alian, R. Amslinger, M. Andreozzi, A. Armejach, N. Asmussen, B. Beckmann, S. Bharadwaj, et al. The gem5 Simulator: Version 20.0+. *Computing Research Repository (CoRR)*, arXiv:2007.03152, Sep 2020.

[39] S. O. H. Madgwick, A. J. L. Harrison, and R. Vaidyanathan. Estimation of IMU and MARG Orientation Using a Gradient Descent Algorithm. In *2011 IEEE International Conference on Rehabilitation Robotics*, pages 1–7, Jun 2011.

[40] R. Mahony, T. Hamel, and J.-M. Pflimlin. Nonlinear Complementary Filters on the Special Orthogonal Group. *IEEE Transactions on Automatic Control*, 53(5):1203–1218, Jun 2008.

[41] S. Mange, E. F. Helbling, N. Gravish, and R. J. Wood. An Actuated Gaze Stabilization Platform for a Flapping-Wing Microrobot. *Int'l Conf. on Robotics and Automation (ICRA)*, May 2017.

[42] V. Mayoral-Vilches, J. Jabbour, Y.-S. Hsiao, Z. Wan, M. Crespo-Alvarez, M. Stewart, J. M. Reina-Muñoz, P. Nagras, G. Vikhe, M. Bakhshalipour, M. Pinzger, S. Rass, S. Panigrahi, G. Corradi, N. Roy, P. B. Gibbons, S. M. Neuman, B. Plancher, and V. J. Reddi. RobotPerf: An Open-Source, Vendor-Agnostic, Benchmarking Suite for Evaluating Robotics Computing System Performance. *Computing Research Repository (CoRR)*, arXiv:2309.09212v2, Jan 2024.

[43] A. Naveen, J. Morris, C. Chan, D. Mhrous, E. F. Helbling, N.-S. P. Hyun, G. Hills, and R. J. Wood. Hardware-in-the-Loop for Characterization of Embedded State Estimation for Flying Microrobots, 2024.

[44] K. Nguyen, S. Schoedel, A. Alavilli, B. Plancher, and Z. Manchester. TinyMPC: Model-Predictive Control on Resource-Constrained Microcontrollers. *Int'l Conf. on Robotics and Automation (ICRA)*, May 2024.

[45] D. Nister. An Efficient Solution to the Five-Point Relative Pose Problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(6):756–770, Jun 2004.

[46] E. Olson. AXLE: Computationally-efficient Trajectory Smoothing Using Factor Graph Chains. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 7443–7448, Xi'an, China, May 2021. IEEE.

[47] R. S. Pierre and S. Bergbreiter. Gait Exploration of Sub-2 g Robots Using Magnetic Actuation. *IEEE Robotics and Automation Letters (RAL)*, 2(1):34–40, Jan 2017.

[48] H. K. H. Prasad, Y. M. C. Ravi Sankar Vaddi, E. Dedic, I. Novosselov, and S. B. Fuller. A Laser-Microfabricated Electrohydrodynamic Thruster for Centimeter-Scale Aerial Robots. *PLOS ONE*, 15:e0231362, Apr 2020.

[49] R. Raguram, O. Chum, M. Pollefeys, J. Matas, and J.-M. Frahm. USAC: A Universal Framework for Random Sample Consensus. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(8):2022–2038, Aug 2013.

[50] Z. Ren, S. Kim, X. Ji, W. Zhu, F. Niroui, J. Kong, and Y. Chen. A High-Lift Micro-Aerial-Robot powered by Low-Boltage and Long-Endurance Dielectric Elastomer Actuators. *Advanced Materials*, 34:2106757, Nov 2021.

[51] E. Rosten and T. Drummond. Machine Learning for High-Speed Corner Detection. In A. Leonardis, H. Bischof, and A. Pinz, editors, *Computer Vision – ECCV 2006*, volume 3951, pages 430–443. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006.

[52] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski. ORB: An Efficient Alternative to SIFT or SURF. In *2011 International Conference on Computer Vision*, pages 2564–2571, Barcelona, Spain, Nov 2011. IEEE.

[53] B. H. Shin, K.-M. Lee, and S.-Y. Lee. A Miniaturized Tadpole Robot Using an Electromagnetic Oscillatory Actuator. *Journal of Bionic Engineering*, Mar 2015.

[54] S. Singh, Z. Temel, and R. S. Pierre. Multi-Modal Jumping and Crawling in an Autonomous Springtail-Inspired Microrobot. *Int'l Conf. on Robotics and Automation (ICRA)*, May 2024.

[55] Y. S. Song and M. Sitti. STRIDE: A Highly Manueverable and Non-Tethered Water Strider Robot. *Int'l Conf. on Robotics and Automation (ICRA)*, May 2007.

[56] M. V. Srinivasan. An Image-Interpolation Technique for the Computation of Optic Flow and Egomotion. *Biological Cybernetics*, 71(5):401–415, Sep 1994.

[57] S. H. Suhr, Y. S. Song, S. J. Lee, and M. Sitti. Biologically Inspired Miniature Water Strider Robot. *Robotics: Science and Systems (RSS)*, 2005.

[58] Y. P. Talwekar, A. Adie, V. Iyer, and S. B. Fuller. Towards Sensor Autonomy in Sub-Gram Flying Insect Robots: A Lightweight and Power Efficient Avionics System. *Int'l Conf. on Robotics and Automation (ICRA)*, May 2022.

[59] Y. P. Talwekar, A. Adie, V. Iyer, and S. B. Fuller. Towards Sensor Autonomy in Sub-Gram Flying Insect Robots: A Lightweight and Power-Efficient Avionics System. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 9675–9681, Philadelphia, PA, USA, May 2022. IEEE.

[60] C. K. Trygstad, E. K. Blankenship, and N. O. Perez-Arancibia. A New 10-mg SMA-Based Fast Bimorph Actuator for Microrobotics. *Int'l Conf. on Intelligent Robots and Systems (IROS)*, Oct 2024.

[61] R. J. Wood. The First Takeoff of a Biologically Inspired At-Scale Robotics Insect. *IEEE Transactions on Robotics*, 24(2):341–347, Apr 2008.

[62] X. Yang, L. Chang, and N. O. Pérez-Arancibia. An 88-milligram Insect-Scale Autonomous Crawling Robot Driven by a Catalytic Artificial Muscle. *Science Robotics*, 5(45):eaba0015, Aug 2020.

[63] Z. Yu, J. Tran, C. Li, A. Weber, Y. P. Talwekar, and S. Fuller. TinySense: A Lighter Weight and More Power-Efficient Avionics System for Flying Insect-Scale Robots. *Computing Research Repository (CoRR)*, arXiv:2501.03416, Jan 2025.