

Optical Flow on a Flapping Wing Robot

Fernando Garcia Bermudez and Ronald Fearing

Department of Electrical Engineering and Computer Sciences
University of California, Berkeley, CA 94720
{fgb, ronf}@eecs.berkeley.edu

Abstract—Optical flow sensing techniques are promising for obstacle avoidance, distance regulation, and moving target tracking, particularly for small mobile robots with limited power and payload constraints. Most optical flow sensing experimental work has been done on mobile platforms which are relatively steady in rotation, unlike the pitching motion expected on flapping wing flyers. In order to assess the feasibility of using optical flow to control an indoor flapping flyer, an 7 gram commercially available ornithopter airframe was equipped with on-board camera and CPU module with mass of 2.5 grams and 2.6 gram battery. An experiment was conducted capturing optical flow information during flapping and gliding flight on the same platform. As expected, flapping introduced substantial systematic bias to the direction estimates to the point of flipping the true direction periodically. Nonetheless, since the optical flow results oscillated at the same frequency as the flapping wings, it is envisioned that one could disambiguate the jittering optic flow measurements by correlating these with real-time feedback from the motor current.

I. INTRODUCTION

Optical flow vision algorithms for use in robotic sensing have been implemented both in simulation [3] and in robotic platforms that have generally presented steady motion for the camera, such as in the case of wheeled robots [4], [12], [13], [19], fixed wing micro air vehicles [2], [19], [22], airships [8], [23], and tethered [16] and untethered [19] helicopters. One of the most unsteady platforms is a robot that mimics the fly motion [15], but its movement is constrained to stay within an artificially textured indoor arena. The group whose work is closest to the one presented in this paper, [21], proposes using optic flow for estimating the altitude of a flapping vehicle, but to date is mostly simulated, with the real video sequences used to test off-board algorithms having smooth motion.

The use of steady platforms for optical flow experimentation simplifies comparisons of several algorithms on sequences on which they all perform relatively well [1], [9], [10], [11], [12]. In addition, well structured environments also simplify extracting ground truth. There are some platforms that have been shown to work well in the outdoors [2], [19], but most of the indoor environments used to test real robots use artificially textured walls and objects to improve contrast and thus the performance of optic flow algorithms.

In contrast to robotic optic flow, insects such as flies and bees, who use optical flow for motion detection and

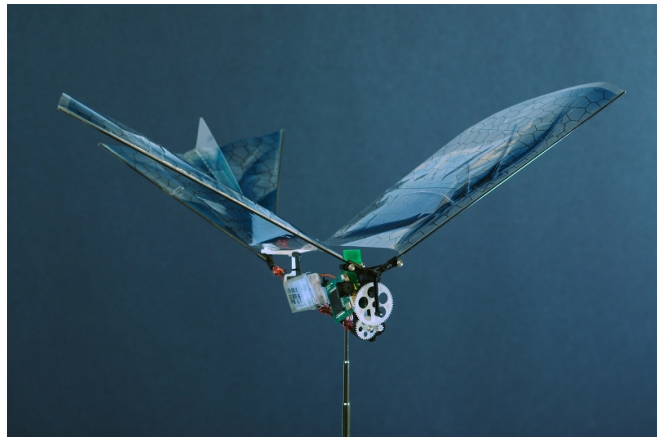


Fig. 1. The flying platform, a modified version of Interactive Toy’s VAMP RC ornithopter, includes custom electronics used for data acquisition and part of the image processing.

navigation [14], [18], [20], perform remarkably well in both outdoor and indoor environments. Their small size and power-to-weight ratio enable them to perform quick maneuvers [20] deemed impossible for larger platforms such as planes. They are also robust to outside disturbances and the occasional error that sends them crashing onto transparent surfaces like windows. However, flapping flight in insects increases unsteadiness in the visual input, which these insects generally account for by moving their heads to counteract it [7]. While a high speed camera or mirror mount could be used to compensate for body motion in flapping robot flight, we examine in this paper the significance of flapping artifacts in optical flow sensing.

II. ROBOTIC ORNITHOPTER PLATFORM

Fig. 1 shows the flying robot, a modified version of Interactive Toy’s VAMP RC ornithopter including custom electronics. The image processing board used to acquire and pre-process the data is pictured in more detail at the top part of Fig. 2. The board weighs 1.1grams, measures 15x35mm and is mainly comprised of a Microchip dsPIC33FJ128MC706 16bit microprocessor running at 40MHz, an OmniVision OV7660FSL VGA camera module, and an ATMEL AT45DB161D 2 megabyte (MB) DataFlash memory. The board was fabricated using a 25 μ m thick FR4 core printed circuit board (PCB).

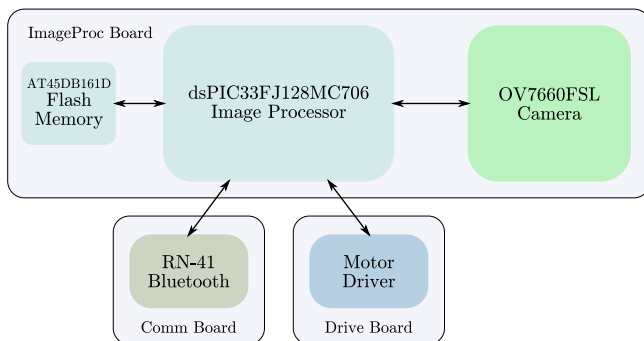
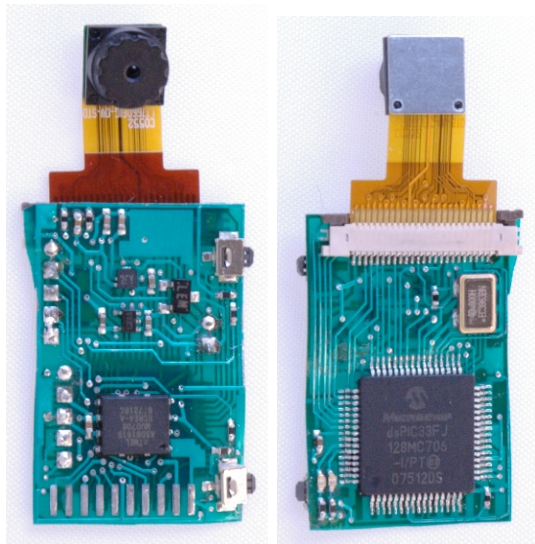


Fig. 2. Custom electronics: (Top) The front and back of the image processing board. The OmniVision OV7660FSL camera module is visible at the top, a Microchip dsPIC33FJ128MC706 microprocessor on the right picture, and an AT45DB161D flash memory on the left one. It weighs 1.1g and measures 15x35mm. (Bottom) Block diagram of the image processing board pictured above alongside the Bluetooth communication module and the motor driver board.

For wireless communication with a PC, a 1.3g Roving Networks RN-41 Bluetooth 2.0 module of roughly the same dimensions as the image processing board was connected through the dsPIC’s serial communication interface. The lower portion of Fig. 2 shows the block diagram representing these two boards as well as the 70mg motor driving board.

Data was acquired from the camera in black-and-white at 160x120 (QQVGA) resolution and a rate of 25 frames per second (fps) and then saved to the dsPIC’s 16KB random access memory (RAM). (The 2MB AT45DB161D memory was not utilized in this paper.) Thus, the RAM size limited the data acquisition to 60 frames of heavily subsampled images at a final resolution of 18x13, which comprised 2.4 sec of visual motion data. These data-sets were offloaded to a computer at 230.4 Kbps over the Bluetooth RS-232 link at the end of the acquisition. Note that the custom electronics as well as the robot’s flapping motor are running out of a 90mAh FULLRIVER lithium-polymer battery that weighs 2.6g.

The ornithopter uses a DC motor for each of flapping and steering. In the modified version used in this work, the Vamp’s RC electronics as well as the foam body are

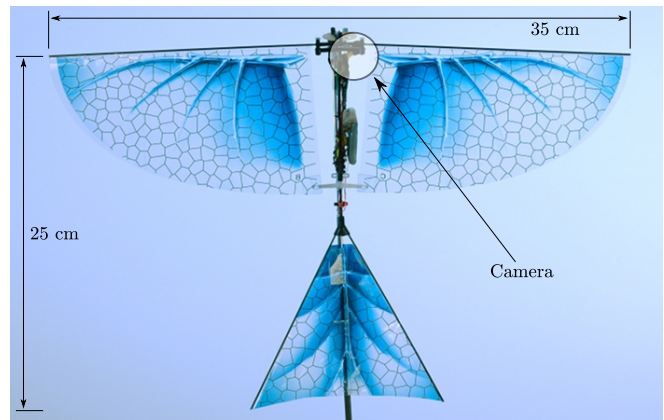


Fig. 3. Dimensions of the robotic platform. Note the camera module positioned to the side of the wing transmission mechanism. The optical axis is aligned with the direction of flight, which in the case of the figure would be the vertical axis.

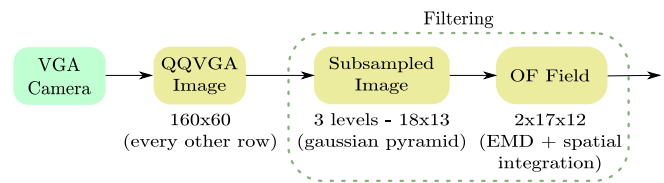


Fig. 4. Block diagram representing the image processing performed.

removed and the custom motor driving board used just actuates the flapping motor. The robot measures around 35x25cm, as can be seen in Fig. 3, and weighs 12.6g when unmodified. Normally, as it flies forward at full throttle, it interleaves climbing periods with stall recovery periods at around 1Hz. This, coupled to the fact that it flaps at around 12-17Hz dependent on battery charge, are the main sources of unsteadiness of this platform (see the top part of Fig. 6). The modified robot, though, weighs 13.6g and this results in a dampening out of the slow climb/stall oscillations because the ornithopter, unable to climb, essentially performs a smooth landing on its body. Thus, the only significant source of unsteadiness in these experiments comes from the flapping. Note that the 13.6g weight includes the 7g airframe, 2.5g of boards, a 2.6g battery, and 1.5g of wiring and mounting hardware.

III. OPTICAL FLOW COMPUTATION AND FILTERING

To reduce storage and transmission requirements, image data is reduced by sub-sampling and averaging. For applications such as wall following or terrain avoidance, a low resolution such as 18x13 is adequate. (For example, Barrows et al. [2] used a 1x18 array for ground height regulation.) Averaging also improves the signal-to-noise ratio for the image data.

The first filtering block shown in Fig. 4 outlines the image pre-processing happening at the camera board that yields the 18x13 frames. Basically, from the 160x120 image that the camera is sending to the dsPIC, the processor captures only

every other line, yielding a 160x60 frame. At this point, the processor convolves the image with a 3x3 pixel discrete gaussian filter,

$$f_{x,y} = \begin{pmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{pmatrix},$$

which can be shown to be equivalent to applying

$$f_x = (1 \ 2 \ 1)$$

to the rows and

$$f_y = \begin{pmatrix} 1 \\ 2 \\ 1 \end{pmatrix}$$

to the columns. Thus, in order to perform the subsampling, f_x is applied three times to each row as they arrive, discarding every other pixel at each step, while f_y is applied only twice to each column as soon as the 60 rows are received and discarding pixels in the same manner, which results in the final image size of 18x13. If it weren't for the fact that the processor is only capturing every other row of the input image, this subsampling would be equivalent to applying a 3-level gaussian pyramid to the image received as suggested by Fig. 4. Note that in this experiment the camera was mounted vertically, thus yielding a 13x18 image. Since the camera field of view is approximately 37° and 50° in the x and y axes respectively each reduced pixel subtends an angle of 4° and 4.5°.

Once the 60 subsampled frames are acquired, they are wirelessly sent to a PC to be further analyzed using Python¹. Even though this part of the processing is done off-board, the algorithms are still chosen according to the computational complexity that can be implemented on-board this type of hardware, since this is the end goal.

The optical flow algorithm chosen for this work is the standard elementary motion detector (EMD) correlation algorithm [6], [14], which is not only easy to implement on a fixed-point architecture such as the dsPIC's, but is also considered neurobiologically plausible in insects [5] and has been used in biological models of the fruit fly [15], [18], [20]. Fig. 5 shows the block diagram of an EMD, and is adapted from [14]. Explicitly, (1) shows the formulas that this block diagram represents for a local pixel patch transitioning from frame k to $k + 1$:

$$\begin{aligned} u_{i,j}(k) &= I_{i,j}(k+1) \cdot I_{i+1,j}(k) \\ &\quad - I_{i+1,j}(k+1) \cdot I_{i,j}(k), \\ v_{i,j}(k) &= I_{i,j}(k+1) \cdot I_{i,j+1}(k) \\ &\quad - I_{i,j+1}(k+1) \cdot I_{i,j}(k). \end{aligned} \quad (1)$$

u and v represent the horizontal and vertical optical flow component matrices while I is the pixel intensity matrix. Considering a maximum image shift of 1 pixel, and 25 fps, the maximum sensed velocity would be 72 and 70 degrees per second in x and y image plane axes.

¹Scientific Tools for Python: <http://www.scipy.org/>

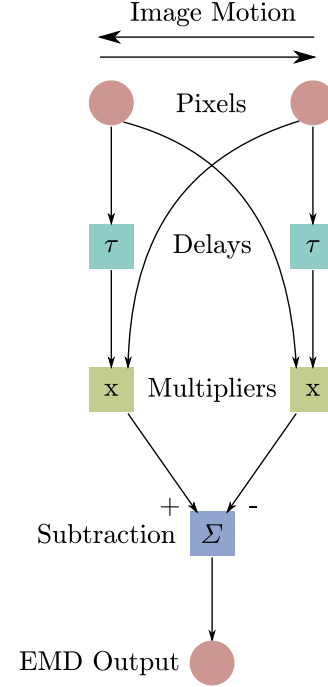


Fig. 5. Block diagram of an EMD (adapted from [14]). Note that, in our case, the delays are represented by consecutive frames in the video sequence captured on-board the platform.

The EMD algorithm was applied to each pair of pixels in the image both in the horizontal as well as the vertical direction. This yielded a 17x12 motion field for each frame pair, coming to a total of 59 fields. A further processing step, (2), integrated the fields spatially, summing the motion vectors over each motion field and normalizing by the corresponding Frobenius norm:

$$\begin{aligned} U(k) &= \frac{\sum_i \sum_j u_{i,j}(k)}{\sqrt{\sum_i \sum_j |u_{i,j}(k)|^2}}, \\ V(k) &= \frac{\sum_i \sum_j v_{i,j}(k)}{\sqrt{\sum_i \sum_j |v_{i,j}(k)|^2}}. \end{aligned} \quad (2)$$

The integrated optical flow, U and V , has some information about the overall flow field and thus about the general motion. It is known, though, that the optical flow field is a nonlinear representation of the true 3D motion field and thus doing a linear combination of its vectors will rarely yield accurate results. Nonetheless, there are neurobiological observations of the fly's nervous system, that support this computation [17].

IV. EXPERIMENTAL RESULTS AND DISCUSSION

To assess the effect on the optical flow calculations of the inherent pitch and roll oscillations related to flapping, reduced image data sets were captured while flapping or gliding in an indoor environment. The video sequences were collected using the same hardware in both cases and the flapping motor was either powered on (flapping) or off (gliding). During the flapping experiments, the robot

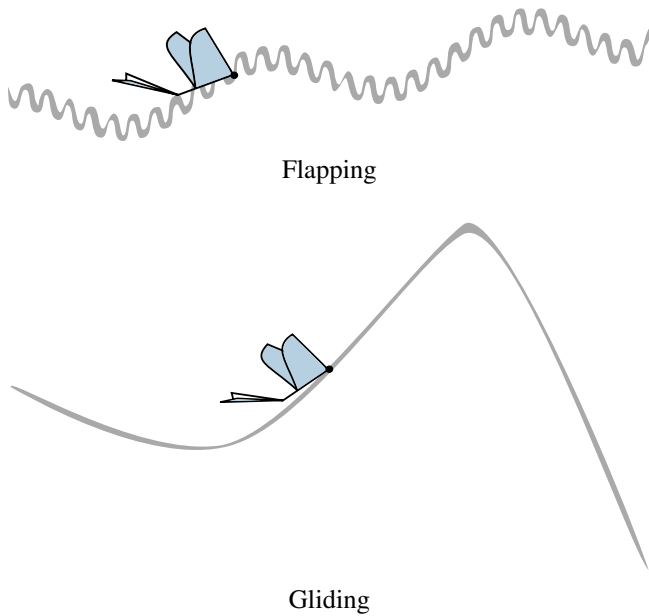


Fig. 6. Behavioral diagrams. (Top) Robot’s behavior while flapping, with a slow climb/stall frequency and a faster flapping frequency. (Bottom) Robot’s behavior while gliding, with a fast climb until stalling and a sharp nose-dive into the ground. Note the black dot where the robot intersects the trajectory as it traverses it. This is where the camera is positioned during flight with its axis pointing in the flight direction.

generally flew in a left circular trajectory with roughly a 5m radius (due to a slight weight imbalance) until it landed smoothly on its body. During the gliding flights, upon being launched manually forward, the robot usually climbed up quickly until stalling and then nose-dived into the ground (see the lower part of Fig. 6).

Fig. 7 shows three consecutive frames of a representative data-set for each experiment. These frames have the optical flow field overlaid on top of them as well as the integration result at the center of each frame. As one can visualize in the figure, the inferred direction that the optical flow integration outputs varies smoothly in the gliding experiment whereas it switches abruptly, frame to frame, in the flapping experiment. In the case of the gliding frame sequence, the inferred direction is that of motion of the robot with its nose diving into the ground. For the flapping frame sequence, the inferred direction is only correct in the outer two frames, since the robot is circling around that direction. The middle frame indicates the opposite motion most probably due to the flapping induced pitch oscillations, which introduce substantial fluctuations to an otherwise smooth circular trajectory of the robot. Although the gliding and flapping trajectories were quite different, the lighting conditions were almost the same since all experiments were done at the same time and in the same indoor environment.

We claim that the erroneous optical flow integration result for the flapping experiment is indeed due to the oscillations induced by flapping. To prove this point, the time-varying normalized vector signals $(U, V)^T$ were first filtered through a Hanning window of length 59 and then processed under a discrete Fourier transform, resulting in the plots of Fig. 8.

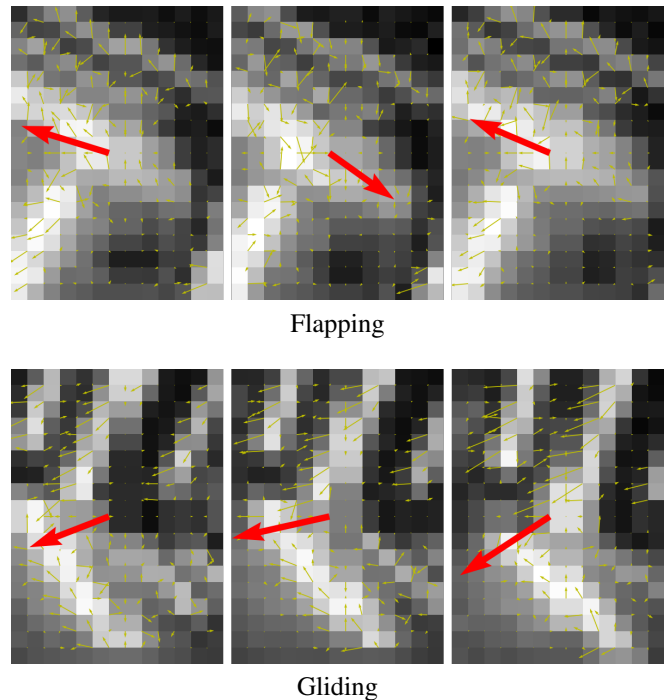


Fig. 7. Subsequent frames for both the flapping and gliding experiments, which include the optical flow field overlaid as well as the integration result at the center. Note that that this central arrow changes direction much more smoothly in the gliding experiment, while giving abrupt changes in direction for the flapping experiment. This behavior is consistent throughout the captured video sequences for each type of experiment.

As is evident from looking at the flapping results in Fig. 8, the optic flow vectors are oscillating at around 11-12Hz. There seems to be a small oscillation at around 2Hz, which could be explained as being related to the damped climb/stall cycle. It could also be related to the relatively short capture period of 2.4 sec, since a few coincidental events during this period can seem like a slow oscillation. This is in fact what can be seen in the gliding results, since in this case it is known that the capture took place just as the robot was reaching the maximum altitude, stalling, and recovering from the stall. Thus, the bump around 1Hz most probably comes from that single event during the 2.4 sec of capture.

To estimate the noise present in the camera, image capture, and optical flow estimation process, a control experiment that consisted of capturing a still image sequence under the same lighting conditions was performed, and the results of it are included at the bottom of Fig. 8. According to this figure, the level of error present in the system is around 10% with no camera motion, and thus argues that almost everything other than the larger peaks in the resulting frequency spectrums might be noise. The source of the peak at 6 Hz is not known, but this component is small compared to wing flapping or slow turning peaks.

In order to verify that during the flapping experiment the optical flow algorithm result was indeed oscillating at the frequency that the robot was flying at, the flapping trajectory was captured on high speed video. Fig. 9 shows a representative sequence of frames depicting a full flapping cycle of the

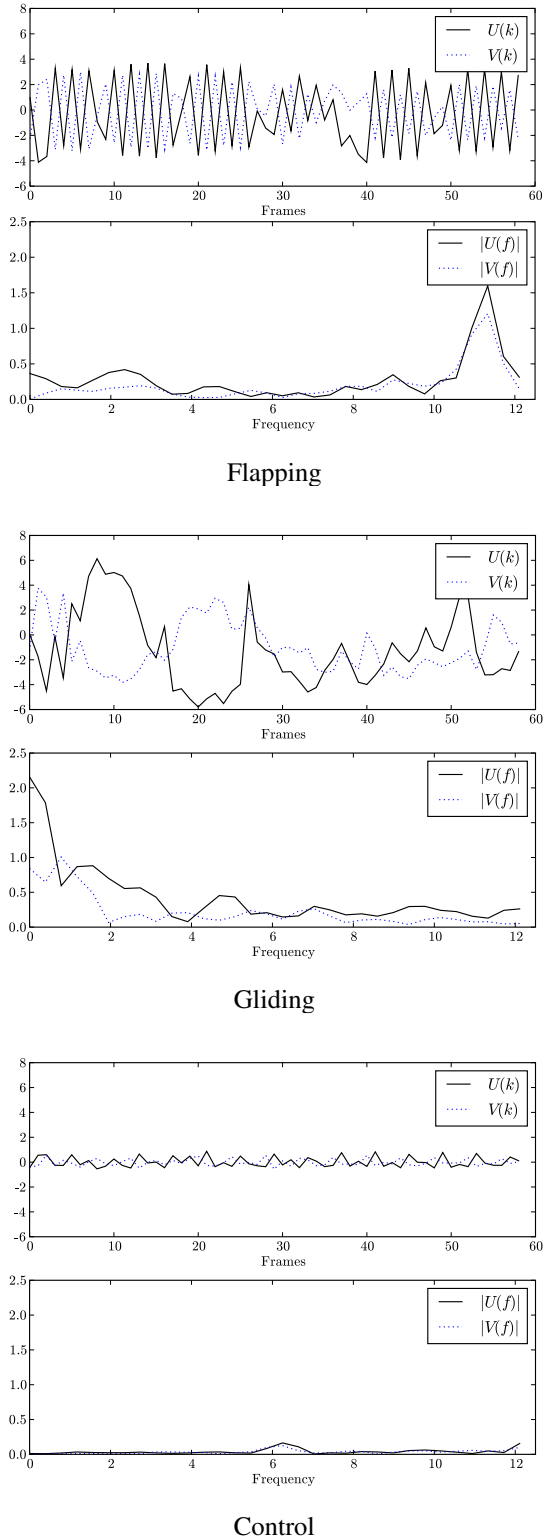


Fig. 8. Experimental results. In each: (Top) Components of the optical flow integration vector during the span of the captured data. (Bottom) Single-sided amplitude spectrum of the above signal. Note: (Flapping) the large peak at around 11-12Hz and the smaller peaks around 0-2Hz; (Gliding) the peak around 0-2Hz; (Control) that the error is around 10% of the previous signals.



Fig. 9. A sequence of frames representing a full flapping cycle of the robot.

robot during the same flapping experiment analyzed above. The total number of frames was 24, spanning 80ms if one takes into account that the video was captured at 300fps. This would indicate that at that point the robot was flapping at a low frequency of 12.5Hz, most probably due to low battery charge during the experiment. If one performs this same analysis at different positions throughout the trajectory, the same frequency is found. From the high speed video, the pitch range induced by flapping is estimated to be $\pm 5^\circ$.

In order to separate the pitch oscillation from the optical flow direction estimates when flapping, we propose to concurrently capture the motor current alongside the video sequence so as to later correlate optical flow integration errors to specific current profiles due to cyclic wing loading conditions. For example, the images could be captured in phase with the wing motion at top-dead-center and bottom-dead-center of the wing trajectory. This would enable almost exact nulling of the pitch rate disturbance, for example by calculating optic flow from pairwise frames $I_{i,j}(k)$ and $I_{i,j}(k+2)$.

V. CONCLUSIONS AND FUTURE WORKS

An order 10 gram robot ornithopter was constructed using a commercial platform combined with a lightweight cell phone camera interface and wireless interface. Subsampled, low resolution video data was captured during flapping or gliding flight and processed off board. This experiment, using a simple biomimetic optical flow algorithm which extracted net motion direction by averaging the flow field across the whole sensor, demonstrated the significance of pitch oscillations due to wing flapping on the optical flow direction estimates.

The small ornithopter used here demonstrates the coupling between body motion and optic flow sensing which can be expected without image stabilization mechanisms. The

strong optical flow signal corresponding to the wing flapping frequency appears readily separable by a notch filter or synchronized sampling. Hence, the active visual stabilization used by insects such as flies does not appear critical. We plan to add motor current measurement so as to enable synchronized sampling. In future work, optical flow information will be used for robot steering in behaviors such as wall following and obstacle avoidance.

VI. ACKNOWLEDGMENTS

The authors would like to thank Erik Steltz for his advice during the design, construction and debugging of the image processing board, Paul Birkmeyer for his help with the filming of the experiments, Prof. Jitendra Malik for his advice on discrete image filtering, Fred Cheng from OmniVision for his prompt technical support for the camera module as well as the advice and support of members of the Biomimetic Millisystems Lab at UC Berkeley. This work supported by NSF Grant IIS-0705249.

REFERENCES

- [1] J. Barron, D. Fleet, and S. Beauchemin, "Performance of optical flow techniques," *International Journal of Computer Vision*, vol. 12, no. 1, pp. 43–77, 1994.
- [2] G. L. Barrows, J. S. Chahl, and M. V. Srinivasan, "Biomimetic visual sensing and flight control," *The Aeronautical Journal*, vol. 107, no. 1069, pp. 159–168, March 2003.
- [3] S. Cameron, S. Grossberg, and F. H. Guenther, "A self-organizing neural network architecture for navigation using optic flow," *Neural Computation*, vol. 10, no. 2, pp. 313–352, February 1998.
- [4] D. Coombs and K. Roberts, "Bee-bot: using peripheral optical flow to avoid obstacles," in *Intelligent Robots and Computer Vision XI: Algorithms, Techniques, and Active Vision*, vol. 1825. SPIE, November 1992, pp. 714–721.
- [5] N. Franceschini, A. Riehle, and A. Le Nestour, "Directionally selective motion detection by insect neurons," in *Facets of vision*, D. G. Stavenga and R. Hardie, Eds. Berlin and New York: Springer, 1989, pp. 360–390.
- [6] V. B. Hassenstein and W. Reichardt, "Systemtheoretische analyse der zeit-, reihenfolgen- und vorseichenauswertung bei der bewegungsperzeption des rüsselkäfers *Chlorophanus*," *Zeitschrift für Naturforschung B*, vol. 11, no. 9, pp. 513–524, September 1956.
- [7] S. J. Huston and H. G. Krapp, "Visuomotor transformation in the fly gaze stabilization system," *PLoS Biology*, vol. 6, no. 7, pp. 1468–1478, July 2008.
- [8] F. Iida, "Biologically inspired visual odometer for navigation of a flying robot," *Robotics and Autonomous Systems*, vol. 44, no. 3-4, pp. 201–208, September 2003.
- [9] H. Liu, T.-H. Hong, M. Herman, T. Camus, and R. Chellappa, "Accuracy vs. efficiency trade-offs in optical flow algorithms," *Computer Vision and Image Understanding*, vol. 72, no. 3, pp. 271–286, December 1998.
- [10] M. Mammarella, G. Campa, M. L. Fravolini, Y. Gu, B. Seanor, and M. R. Napolitano, "A comparison of optical flow algorithms for real time aircraft guidance and navigation," in *Guidance, Navigation and Control Conference and Exhibit*. Honolulu, Hawaii: AIAA, August 2008.
- [11] B. McCane, K. Novins, D. Crannitch, and B. Galvin, "On benchmarking optical flow," *Computer Vision and Image Understanding*, vol. 84, no. 1, pp. 126–143, October 2001.
- [12] C. McCarthy and N. Barnes, "Performance of optical flow techniques for indoor navigation with a mobile robot," in *International Conference on Robotics & Automation*. New Orleans, LA: IEEE, April 2004.
- [13] F. Mura and N. Franceschini, "Obstacle avoidance in a terrestrial mobile robot provided with a scanning retina," in *Intelligent Vehicles Symposium*. Tokyo, Japan: IEEE, September 1996, pp. 47–52.
- [14] W. Reichardt, "Evaluation of optical motion information by movement detectors," *Journal of Comparative Physiology A: Neuroethology, Sensory, Neural, and Behavioral Physiology*, vol. 161, no. 4, pp. 533–547, July 1987.
- [15] M. B. Reiser and M. H. Dickinson, "A test bed for insect-inspired robotic control," *Philosophical Transactions: Mathematical, Physical and Engineering Sciences*, vol. 361, no. 1811, pp. 2267–2285, October 2003.
- [16] F. Ruffier and N. Franceschini, "Optic flow regulation: the key to aircraft automatic guidance," *Robotics and Autonomous Systems*, vol. 50, no. 4, pp. 177–194, March 2005.
- [17] S. Single and A. Borst, "Dendritic integration and its role in computing image velocity," *Science*, vol. 281, no. 5384, pp. 1848–1850, September 1998.
- [18] M. V. Srinivasan, M. Poteser, and K. Kral, "Motion detection in insect orientation and navigation," *Vision Research*, vol. 39, no. 16, pp. 2749–2766, August 1999.
- [19] M. V. Srinivasan, S. Zhang, J. Chahl, G. Stange, and M. Garrat, "An overview of insect-inspired guidance for application in ground and airborne platforms," *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, vol. 218, no. 6, pp. 375–388, 2004.
- [20] L. F. Tammero and M. H. Dickinson, "The influence of visual landscape on the free flight behavior of the fruit fly *Drosophila melanogaster*," *The Journal of Experimental Biology*, vol. 205, pp. 327–343, 2002.
- [21] C. D. Wagter, B. Bijnens, and J. Mulder, "Vision-only control of a flapping mav on mars," in *AIAA Guidance, Navigation and Control Conference and Exhibit*. Hilton Head, SC: AIAA, August 2007.
- [22] J.-C. Zufferey and D. Floreano, "Fly-inspired visual steering of an ultralight indoor aircraft," *IEEE Transactions on Robotics*, vol. 22, no. 1, pp. 137–146, February 2006.
- [23] J.-C. Zufferey, A. Guanella, A. Beyeler, and D. Floreano, "Flying over the reality gap: From simulated to real indoor airships," *Autonomous Robots*, vol. 21, no. 3, pp. 243–254, Nov. 2006.