Vision-Based Navigation using Multi-Rate Feedback from Optic Flow and Scene Reconstruction

Amanda R. Roderick*, Joseph J. Kehoe† and Rick Lind‡
University of Florida

Vision-based control is being aggressively pursued for autonomous systems. Such control is particularly valuable for path planning to achieve mission objectives like target tracking and obstacle avoidance. This paper presents a multi-rate strategy that utilizes a fast-rate optic flow approach and a slow-rate scene reconstruction approach. The resulting controller is able to generate flight trajectories and perform obstacle avoidance within a reasonable computational cost. A simulation demonstrates the performance of an aircraft that uses the multi-rate controller to avoid an obstacle which is only observed after a turn. Essentially, the fast-rate optic flow indicates the presence of the obstacle during the time that slow-rate scene reconstruction is being performed. The resulting flight path is able to follow mission objectives and avoid a collision.

I. Introduction

Autonomy is an enabling technology for many missions envisioned for future generations of unpiloted air vehicles (UAV). Some missions will use micro and small UAV platforms that will be difficult for a human operator to observe and command. Some missions will require rapid decision making based on more targets than a human can process. Finally, missions may utilize swarms of UAV systems for which the workload would be too immense for a human to control.

Vision-based feedback can be a critical component to autonomy. Inertial sensors, such as gyros and accelerometers, provide information about the state of the vehicle; however, they do not provide any information about the environment. Vision can be used to extract information, such as location of obstacles, about the flight space. As such, vision-based feedback provides a mechanism for autonomous navigation to maneuver through unknown or uncertain environments.

Many techniques have been explored that use vision for navigation. Researchers in robotics have been particularly active in this area along with more recent applications in aerospace and manufacturing. Most techniques share some commonality; namely, a sequence of image processing and vision processing are performed to extract information which is then analyzed to make an optimal control decision. The basic unit of information from an image is a feature point which indicates some pixel of particular interest due to, for example, color or intensity gradient near that pixel. Among the techniques that utilize feature points, the approaches related to optic flow and scene reconstruction are of direct interest to this paper.

Optic flow refers to the velocity of the feature points across images taken at different times.[1,2] Motion towards an obstacle will result in large flow that increases as the distance to that obstacle decreases. In this way, optic flow is used for control by indicating regions that may contain close obstacles. The approach has been used for various applications ranging from guide robots for the visually impaired[3] to micro air

*Graduate Student, Department of Mechanical and Aerospace Engineering, *amandar7@ufl.edu*, Student Member AIAA
†Graduate Student, Department of Mechanical and Aerospace Engineering, *jjk@ufl.edu*, Student Member AIAA
‡Assistant Professor, Department of Mechanical and Aerospace Engineering, *ricklind@ufl.edu*, Senior Member AIAA

vehicles.[4–7] Some succesful missions that demonstrated optic flow include obstacle avoidance for ground vehicles,[3, 8, 9] road navigation,[10] terrain following for helicopters,[4] and the autonomous landing of aerial vehicles.[8] A form of optic flow using probability distributions of correspondences between feature points has also been used for obstacle avoidance.[11]

The computation of optic flow has received considerable attention in the context of enabling autonomy. A fast algorithm was derived using intensity gradients as features for pattern matching in combination with a brightness constraint.[3] The computation time is further decreased by compressing the peripheral parts of optical flow to reduce the amount of input data.[9] Sensors with extremely fast rates of computing optic flow have been generated using inspiration from the biology of flying insects.[5]

Alternatively, the techniques using scene reconstruction have only limited experience with control. The approach for scene reconstruction attempts to estimate the relative locations, in an inertial coordinate system, of features points in a set of images. The underlying concept of scene reconstruction is actually straightforward to understand by considering projective geometry of aircraft positions and feature points; however, the mathematics required to accurately compute optimal solutions to the problem are quite complex.[12] The technique of structure from motion is perhaps the most matured approach to scene reconstruction.[13] A recent approach uses adaptive learning to estimate a detailed terrain from a set of feature points.[14] In each case, a strategy for path planning enables navigation by choosing a trajectory to avoid the estimated locations of obstacles. Scene reconstruction has been used for a variety of ground vehicles[15, 16] and aerial simulations.[14, 17, 18]

Each type of vision-based control, optic flow and scene reconstruction, has strengths and weaknesses. Optic flow presents only inferences about obstacle locations but can be computed extremely fast; conversely, scene reconstruction presents highly-detailed maps of obstacle location but requires an intense computational effort. Obviously autonomous aircraft will require both fast rate as the vehicle flies through cluttered environments and low computational cost due to limited processing speeds of on-board hardware.

This paper demonstrates a multi-rate strategy for navigation that combines optic flow and scene reconstruction. A slow loop operates on the scene reconstruction and, while the computations are being performed, a fast loop uses optic flow to avoid obstacles. In this way, the aircraft uses optic flow to ensure collisions will not occur during the scene reconstruction.

A series of simulations are presented to demonstrate the methodology. An aircraft maneuvers through a cluttered environment using only optic flow, only scene reconstruction, and the multi-rate strategy. The approaches using only one type of vision-based control are not able to satisfy the mission objectives whereas the multi-rate controller is able to navigate through the obstacles and still reach the target waypoint.

Also, the contribution of this paper lies mostly with optic flow and its use in multi-rate control. The discussion focuses on optic flow and multi-rate. A basic outline of scene reconstruction is presented, along with references that provide extensive details, because the associated computations are quite complicated and well beyond the scope of this paper.[12, 19]

II. Feature Points

This paper will use feature points as the foundation for any vision-based feedback. These points generally correlate to items in the environment of special significance. Some examples of items that often constitute feature points are corners, edges and light sources. Such feature points can provide information about the overall object in the sense that a set of corners can outline a building. Feature points do not necessarily provide enough information to completely describe an environment but, in practice, they usually provide sufficient information for obstacle avoidance.

A camera effectively maps the 3-dimensional environment onto a 2-dimensional image plane. This image plane is defined as the plane normal to the camera's central axis located a focal length, f, away from the origin of the camera basis. The image plane has coordinates bounded by the field of view such that the vertical component satisfies $\mu \in [\underline{\mu}, \overline{\mu}]$ and the horizontal component satisfies $\nu \in [\underline{\nu}, \overline{\nu}]$.

The mapping provided by the camera is described in Fig. 1. The vector, η, between the camera and

the environmental item is decomposed into components along and perpendicular to the camera axis or line of sight. The projection onto the image plane is then offset from the center of the image by μ and ν as a function of these components of η and the focal length of f.

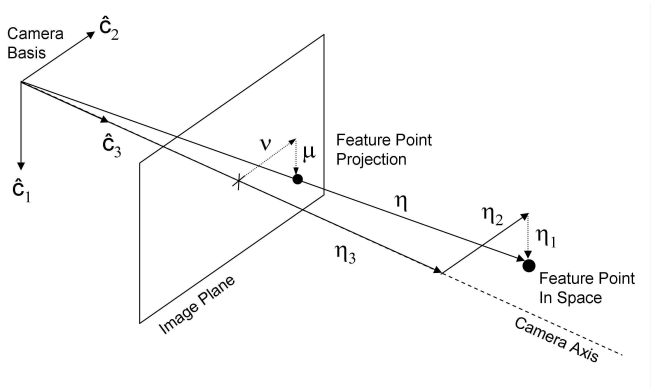

Figure 1. Mapping from Environment to Image Plane

The coordinates of the feature point in the image plane are given by Eq. 1 and Eq. 2. These equations utilize a standard pin-hole camera model. Also, the camera origin is assumed to coincide with the camera lens.

$$\mu = f\frac{\eta_1}{\eta_3} \tag{1}$$

$$\nu = f\frac{\eta_2}{\eta_3} \tag{2}$$

The extraction of feature points is critical to the success of the vision-based control described in this paper. Such extraction begins with identifying the features in the image using, among others, techniques based on edge detection, color distribution, intensity variation or basic differentiation of image properties.[20] The extraction continues as additional images are measured by tracking a feature point throughout these images using, among others, techniques for segmented correlation and Kalman filtering.[15,16,21] Extracting feature points is not trivial and degrades with increasing noise and increasing motion of the aircraft although the techniques are routinely used for robotic systems.

III. Vision-Based Control Strategies

A. Scene Reconstruction

1. Concept

Scene reconstruction attempts to actually estimate the inertial coordinates of obstacles within the environment observed by the camera. Essentially, this reconstruction seeks to create a virtual map of the environment. Such a map would obviously be highly beneficial for any autonomous system because a flight path could be computed with confidence. The techniques of scene reconstruction can roughly be built from elements of epipolar geometry, stereopsis, structure from motion, and terrain mapping.

Epipolar geometry is a fundamental computation used for computer vision.[22–24] The basic concept notes that a feature point which is observed on multiple images must satisfy a geometric relationship. In other words, the location of an obstacle can be determined by triangulation using the lines of sight from two camera positions. Of course, this concept requires the positions and attitudes of the cameras to be known.

The reconstruction of a complete scene, still assuming the positions and attitudes of cameras are known, can be achieved by stereopsis.[25, 26] The technique uses correspondences between multiple feature points across multiple frames to reconstruct an entire scene. Stereopsis is basically stereo vision that fuses features observed from images to estimate the depth of those features. The mathematics are obviously built upon the epipolar constraint but the ability to robustly account for a large number of features adds significant complexity.[27]

Relaxing the assumptions of known camera characteristics dramatically increases the difficulty of scene reconstruction. Structure from motion is introduced to simultaneously estimate both obstacle properties and camera properties.[13] Nonlinear relationships between the camera and obstacles are derived whose solutions correspond to the scene reconstruction. The relationships can be approximated as affine for small motions[28] and solved using a nonlinear least-squares factorization.[29] The full problem based on projective geometry can be addressed using, for example, conics and iterative solutions.[30]

Finally, recent advances consider scene reconstruction explicitly for vision-based control. These advances use an adaptive and multi-level method based on wavelets to provide various resolutions of block geometry.[31] The method actually uses learning theory in association with the adaptive wavelets such that information from multiple flights is used to methodically generate a virtual map that relates all feature points. Such a method can operate in near real-time speed and, consequently, provide information for obstacle avoidance during flight.

2. Strategy

Autopilot design is relatively straightforward using scene reconstruction. The computer vision provides a virtual map of the environment so the autopilot merely needs to find an optimal path through that map. The real difficultly arises with the need for on-line processing and the fast updates which are particularly important for maneuvering through cluttered environments.

Among the variety of approaches for path planning, a rapid algorithm has been directly developed for vision-based control with scene reconstruction.[32] The algorithm for this approach solves sequences of optimization problems based on local characteristics within the environment. The local solutions are then connected throughout the flight to generate a path through the environment. The approach does not yet guarantee stability; however, simulated demonstrations indicate good performance at a reasonable computational cost.

B. Optic Flow

1. Concept

Optic flow represents the velocity of feature points as projected onto the image plane.[8] Essentially, the relative motion between an aircraft and an obstacle results in movement of the associated feature points if that motion is not purely along the line of sight of the camera. The optic flow can thus provide inferences about the location of obstacles during flight.

The velocities associated with optic flow result from time derivatives of the feature-point positions in Eq. 1 and Eq. 2. The resulting expressions, given in Eq. 3 and Eq. 4, describe the motion in the two-dimensional image plane.

$$\dot{\mu} = f\left(\frac{\eta_3\dot{\eta}_1 - \eta_1\dot{\eta}_3}{\eta_3^2}\right) \tag{3}$$

$$\dot{\nu} = f\left(\frac{\eta_3\dot{\eta}_2 - \eta_2\dot{\eta}_3}{\eta_3^2}\right) \tag{4}$$

The concept of optic flow can be demonstrated for the artificial scenario in Figure 2. An aircraft is flown between two obstacles and images are recorded during the flight. In this case, the flight path is a northerly

trajectory at constant altitude. The obstacles lie equal distances to the east and west on either side of this trajectory; however, the obstacle to the east is closer to the initial location of the aircraft. Also, a set of feature points on each obstacle are denoted by the star symbols.

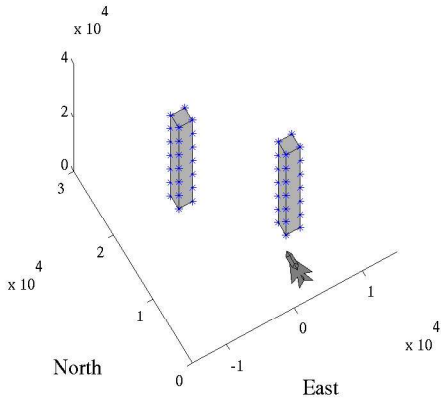

Figure 2. Scenario of Aircraft and Obstacles

Vision-based feedback is generated by simulating straight and level flight along the flight path. A pair of images, shown in Fig. 3, represent the feature points taken 0.04 *sec* apart. Several characteristics are notable for the feature points in each image. In particular, the feature points on the right side of each image are positioned farther from the vertical centerline, are spaced farther apart along the vertical direction, and are reduced in number when compared to the feature points on the left side. These characteristics uniformly result from the relative location of the camera to the obstacles. The obstacle to the east of the vehicle, which corresponds to the right side of each image, is closer to the camera so only a reduced portion of that obstacle lies within the field of view. The effect of distance, as described in Eq. 1 and Eq. 2, results in these visual characteristics.

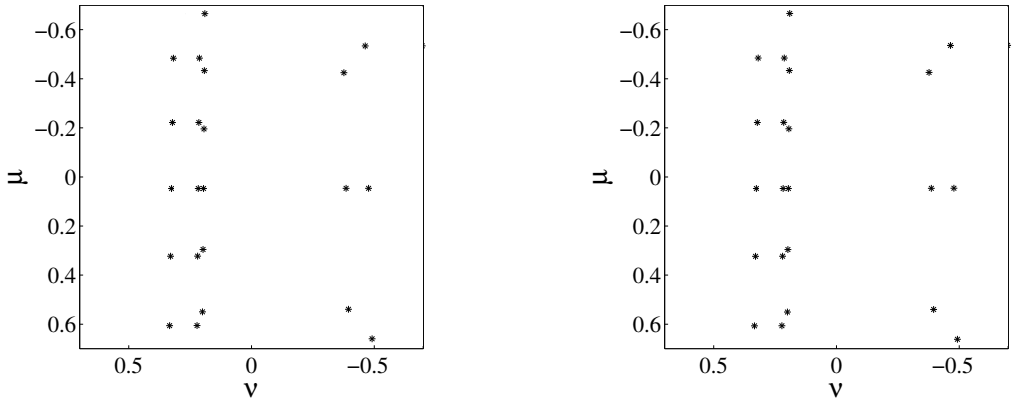

Figure 3. Feature Point Positions in Image Plane at Timesteps 1 (Left) and 2 (Right)

The optic flow of these feature points is given in Fig. 4. The vectors are actually scaled by a factor of 25 to increase their visibility and ease of presentation. These vectors clearly demonstrate the mathematical properties expected from Eq. 3 and Eq. 4 for such a scenario. The feature points tagged to the obstacle to the east, or to the right in the image plane, have larger optic flow due to the reduced η_3 distance. Similarly, the variation in η_1 causes points above or below the horizontal centerline to have a vertical velocity in the image plane.

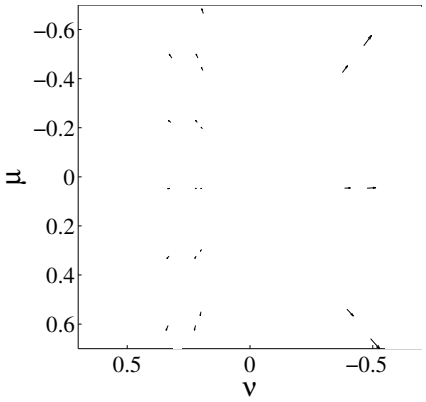

Figure 4. Optic Flow from Figure 3

This example is somewhat simplistic but clearly demonstrates a relationship between optic flow and obstacle detection. The magnitude and direction of optic flow can be used to infer information about relative position between obstacles and a camera. This argument is particularly strong when the obstacles are fixed in position but the general concept also applies to scenarios of moving obstacles.

2. Translational Optic Flow

The general expressions for optic flow, as given in Eq. 3 and Eq. 4, describe the velocity for any time-varying position of a feature point. A specific form of these expressions can be derived for the situation of an aircraft flying relative to fixed features. In this case, the expressions in Eq. 5 and Eq. 6 relate the optic flow in terms of aircraft states and camera parameters.[33]

$$\dot{\mu} = f\left[\left(q + p\nu + q\mu^2 + r\mu\nu\right) + \left(\frac{u\mu - w}{\eta_3}\right)\right] \tag{5}$$

$$\dot{\nu} = f\left[\left(r - p\mu + r\nu^2 + q\mu\nu\right) + \left(\frac{-u\nu + v}{\eta_3}\right)\right] \tag{6}$$

An evaluation of optic flow in terms of aircraft states is particularly valuable for obstacle avoidance. This optic flow, as evidenced by Eq. 5 and Eq. 6, can essentially be decomposed into translational and rotational parts. The primary risk of collision between an aircraft and a fixed obstacle is associated with translational motion; therefore, the rotational component of optic flow is not critical for this particular task.

The translational component of optic flow, which is of particular interest for obstacle avoidance, can be computed using measurements of the aircraft velocities. The desired values, $\tilde{\mu}$ and $\tilde{\nu}$, result from subtraction of the appropriate combinations of aircraft states and feature point coordinates from the measured optic flow as given in Eq. 7 and Eq. 8.

$$\dot{\tilde{\mu}} = \dot{\mu} - f\left(q + p\nu + q\mu^2 + r\mu\nu\right) \tag{7}$$

$$\dot{\tilde{\nu}} = \dot{\nu} - f\left(r - p\mu + r\nu^2 + q\mu\nu\right) \tag{8}$$

A comparison between optic flow and translational optic flow is demonstrated using the scenario in Fig. 2. In this new case, the aircraft flies due north but continuously rolls during the flight. The feature points in Fig. 5 correspond to images taken 0.04 *sec* apart while the aircraft has a roll rate of -2.608 *rad/sec*, a pitch rate of 0.0126 *rad/sec*, and a yaw rate of -0.1176 *rad/sec*.

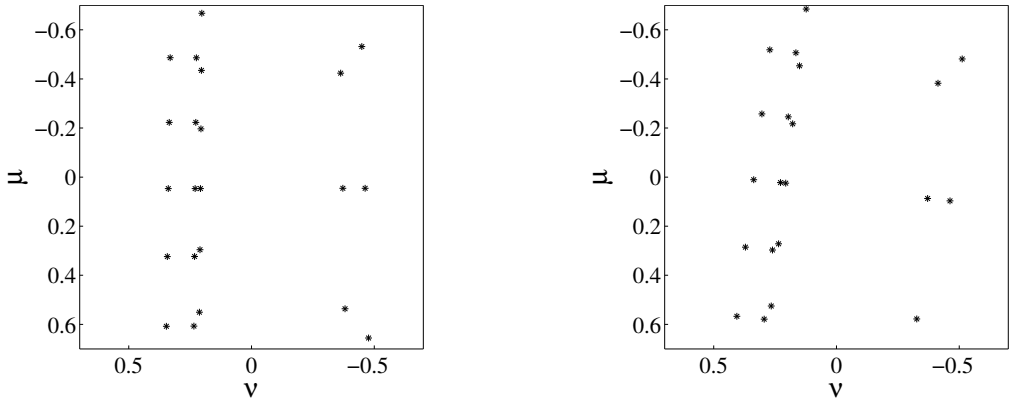

Figure 5. Feature Point Locations at timesteps 1 (Left) and 2 (Right)

The optic flow in Fig. 6 describes the motion of the feature points in Fig. 5. The optic flow is clearly influenced by the rotational velocities of the aircraft such that the feature points are primarily rotating in the image plane. Conversely, the translational component of optic flow shows considerably smaller magnitudes and different directions than the total optic flow. The translational component in Fig. 6 actually agrees quite well with the optic flow in Fig. 4 that resulted from purely translational motion of the aircraft along this same flight path. Note the translational component has been scaled by a factor of 25 for ease of presentations to ensure the vectors are visible.

3. Strategy

A control strategy is formulated to navigate the aircraft away from regions of high-magnitude optic flow or, alternatively, regions of nearby obstacles. Essentially, navigation commands are generated to minimize a cost function. This cost function is derived to account for characteristics, such as magnitude and location, of feature points and their associated optic flow. The regions in the image plane where this cost function is low indicate the desired directions towards which the aircraft should fly.

The magnitude of optic flow is obviously a metric of importance because of its inferred relationship to obstacle distance. As such, the cost function utilizes this velocity value to penalize obstacles which may present imminent threat. The optic flow is computed by tracking feature points, of which N are assumed, through a sequence of images. The n^{th} point has coordinates of μ_n and ν_n and translational optic flow of $\dot{\tilde{\mu}}$ and $\dot{\tilde{\nu}}$ with velocity given in Eq. 9.

$$V_n = \sqrt{\dot{\tilde{\mu}}_n^2 + \dot{\tilde{\nu}}_n^2} \tag{9}$$

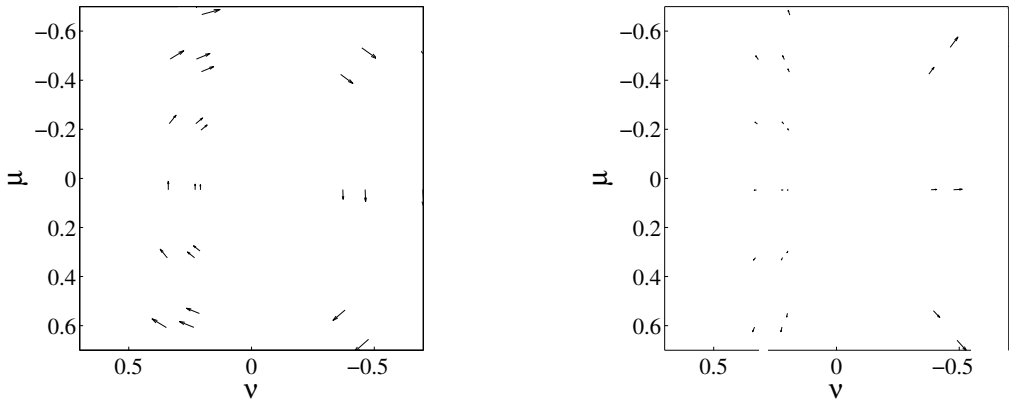

Figure 6. Optic Flow (Left) and Scaled-by-25 Optic Flow with Rotational Components Removed (Right)

Also, the cost function penalizes flight paths that pass close to potential obstacles. The two-dimensional image plane does not provide direct information for this penalty; however, the feature points can be related to such information. A point in the image plane correlates to a line of sight, as shown in Figure 1, so a flight path is safe if no obstacles lie along the line of sight associated with the flight path. As such, the cost function will consider distance, $d_n(\mu, \nu)$, from every point on the image plane to the n^{th} feature point as in Eq. 10.

$$d_n(\mu, \nu) = \sqrt{(\mu - \mu_n)^2 + (\nu - \nu_n)^2} \tag{10}$$

A cost function can be computed by relating the parameters for velocity and distance as in Eq. 11. This cost function essentially normalizes the quartic of the translational optic flow by the distance to the associated feature point. The quartic provides a disproportionately high cost for feature points with high velocities that lie close to the flight path.

$$J(\mu, \nu) = \sum_{n=1}^{N} \frac{V_n^4}{d_n(\mu, \nu)} \tag{11}$$

A flight path is chosen as the line of sight along which the cost function in Eq. 11 is minimal. This line of sight correlates to the image coordinates, μ_{opt} and ν_{opt}, with lowest cost as given in Eq. 12.

$$[\mu_{opt}, \nu_{opt}] = \arg \min_{\substack{\mu \in [\underline{\mu}, \overline{\mu}] \\ \nu \in [\underline{\nu}, \overline{\nu}]}} J(\mu, \nu) \tag{12}$$

Navigation commands are generated to steer the aircraft towards the optimal image coordinates or, alternatively, the optimal line of sight. The angles for heading and pitch are computed, assuming angle of attack is given by α and angle of sideslip is negligible, as in Eq. 13 and Eq. 14. These optimal commands, Δ_ψ and Δ_θ, can be directly extracted from Fig. 1 by including the angle of attack.

$$\Delta_\psi = -\tan^{-1}\left(\frac{\nu_{opt}}{f}\right) \tag{13}$$

$$\Delta_\theta = \alpha - \tan^{-1}\left(\frac{\mu_{opt}}{f}\right) \tag{14}$$

The control strategy is demonstrated using the scenario presented in Fig. 2. Simulated flights generated the optic flow in Fig. 4 or the translational optic flow in Fig. 6 based on feature points. The cost function, $J(\mu, \nu)$, associated with these optic flows is shown in Fig. 7. This cost function indicates the regions to the west or straight ahead are the safest. As such, the navigation commands of $\Delta_\psi = -0.61\ rad$ and $\Delta_\theta = 0$ are computed using Eq. 13 and Eq. 14. The commands indicate the strategy is able to steer the aircraft away from obstacles to minimize the threat of collision.

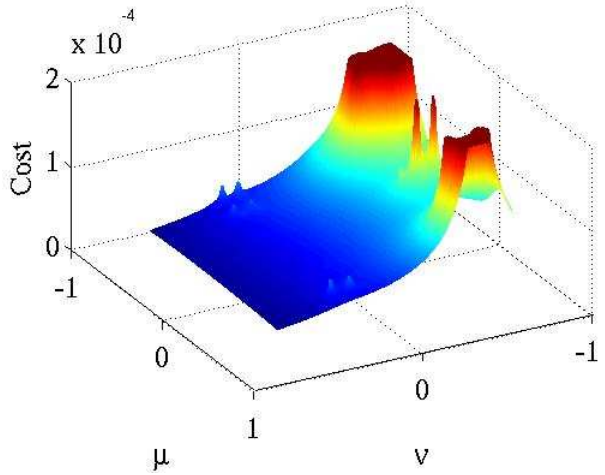

Figure 7. Cost Function for Fig. 3 and Fig. 6

IV. Multi-Rate Control

Certainly both vision-based control strategies, scene reconstruction and optic flow, present enabling technologies for autonomous aircraft but each approach also has limitations. Scene reconstruction can provide accurate maps of obstacles such that the resulting flight paths are confidently safe; however, the computational cost for scene reconstruction greatly limits its utility for on-board applications. Optic flow can actually be computed using hardware with extremely fast rates; however, the information provides only a rough inference about obstacles so the resulting flight paths are generated with some level of caution. As such, a clear trade-off is observed between the level of information for decision making and computation time required to provide that information.

A multi-rate controller is proposed that takes advantages of the strengths of each type of vision-based feedback while limiting the effects of their respect weaknesses. The fundamental scheme will utilize scene reconstruction in a slow loop while the optic flow runs in a fast loop. Each loop computes navigation commands, Δ_ψ and Δ_θ, so a switching element determines which set of commands are actually passed to the manuevering controller as in Fig. 8. Note that the time between updates from each controller is simply included as τ_1 and τ_2 to reflect the different rates.

The switching element is critical to effectively using both types of vision-based control. This switch is implemented based on a trigger that reflects the total magnitude of the optic flow field in the image as given in Eq. 15. In this case, the trigger uses the quartic of the optic flow for each feature point to disproportionately

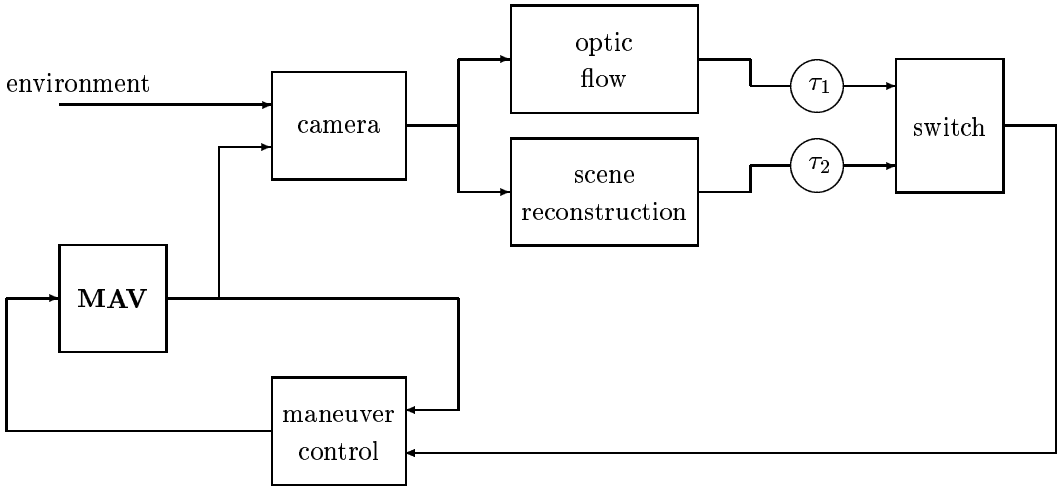

Figure 8. Closed-Loop System with Multi-Rate Control

emphasize the largest vectors which are assumed to relate to the closest obstacles.

$$F = \sum_{n=1}^{N} V_n^4 \tag{15}$$

The law governing the switch compares F to some threshold, σ, to determine which type of vision-based control should command the aircraft. Essentially, a large value of optic flow means either the vehicle is close to an obstacle and/or the vehicle is viewing a densely-cluttered environment. Either situation presents a need for fast-rate control so the optic flow should dominate. The resulting switch is given in Eq. 16.

$$\begin{bmatrix} \Delta_\phi \\ \Delta_\theta \end{bmatrix} = \begin{cases} \text{output of optic flow} & : if \;\; F > \sigma \\ \text{output of scene reconstruction} & : else \end{cases} \tag{16}$$

The multi-rate controller should enable to aircraft to operate better than a single-rate controller using either optic flow or scene reconstruction. Such control will be particularly beneficial when entering a cluttered environment in which some obstacles are hidden from view at various times during the flight. Consider an image taken at $t = 0$ during which the aircraft is turning. The controller using scene reconstruction will not respond to features in that image until $t = \tau_2$. The vehicle will remain turning during this time even though new features, including potentially unseen obstacles, will come into the field of view as the vehicle changes position and heading. The optic flow, which will evaluate the image considerably faster and be able to respond at $t = \tau_1$, will able to quickly respond to these newly-viewed obstacles. In this way, the multi-rate controller allows the high-resolution path planning associated with scene reconstruction but maintains obstacle avoidance during the computation time by utilizing optic flow.

Finally, the multi-rate controller may be beneficial but certain characteristics can not be guaranteed. The issue of stability is particularly important but can not be proven for the closed-loop system in Fig. 8. The commands from the optic flow and scene reconstruction may interfere and counteract each other and cause the aircraft to fly an undesired flight path. Such situations are certainly possible so the control strategy must be carefully evaluated for anticipated scenarios. In many simulated cases, as will be shown, the rate differences between the controllers is sufficient and the obstacle detection is pronounced such that the aircraft does actually fly a reasonable path.

V. Example

A. Mission

A set of simulations are presented to demonstrate the vision-based controllers. These simulations utilize a common scenario to compare the ability of each controller to steer an aircraft past obstacles. In this case, the scenario is a set of obstacles, of which one is large and seven are small, populating an area through which an aircraft must fly. The controllers are evaluated on the flight path each generates by considering time to reach a target waypoint and distance to obstacles during the flight.

The traversing of an urban environment by a micro air vehicle is a mission of particular interest to the University Florida. The scenario of these simulations is intended to mimic that mission; however, models of the flight dynamics of a micro air vehicle are not matured. Instead, the simulations utilize a high-fidelity simulation of an F-16.[34] The F-16 is not an ideal model for this simulation but the nonlinear dynamics provide an estimate of represenative flight dynamics. The simulations are able to show the controller performance on a realistic model so, like any controller design, the gains will need to be tuned for a different model but the basic approach can at least be validated.

Also, the model includes nonlinear open-loop dynamics and a maneuvering autopilot. This autopilot includes a stability augmentation system to smooth the flight path and a control augmentation system to track attitude commands. An additional outer-loop controller allows the vehicle to follow waypoints using the strategy from scene reconstruction.

The mission chosen for these simulations highlights the advantages of multi-rate control. A set of 7 obstacles is situated behind a long wall such that they are initially hidden from view. The mission objective seeks a path to the northwest. After the vehicle flies north around the wall, the hidden obstacles become visible. The objective is thus to maneuver between all 7 obstacles while navigating towards the indicated destination located to the west. The aircraft is also commanded to maintain a constant altitude.

The simulation represents responses from slow-rate control using only scene reconstruction, high-rate control using only optic flow, and multi-rate control using both scene reconstruction and optic flow. The optic flow algorithms are actually implemented and running; however, the scene reconstruction algorithms, which are extremely complex, have been replaced with a representative path planning approach. Such a simplification still presents the nature of the complex controller and thus is valid as proof of concept.

B. Control based on Optic Flow

A flight path is simulated in response to control using only optic flow. This controller uses the cost function in Eq. 11 to analyze the optic flow of the image plane at each timestep. It then selects a line of sight which is assumed to be the safest based on that cost function. The appropriate commands for heading and pitch are subsequently generated and passed to the maneuvering controller.

The resulting flight path is shown in Figure 9. The vehicle initially sees the optic flow to the left and consequently turns right. It therefore avoids the obstacles but does not fly towards the northwest; instead, it steers towards the northeast. This controller therefore does not accomplish the mission objectives.

The path in Figure 9 is not entirely unexpected given the simplistic nature of the implemented controller. Several approaches could be used that are much more advanced than simply steering towards least flow; however, this simplistic controller is used because it can operate at extremely high rate. As such, the results are not indicative of a limitation in optic flow as much as they are indicative of a limitation in high-rate optic flow.

C. Control based on Scene Reconstruction

The representative response to a controller using only scene reconstruction is also simulated. The simulation did not actually implement the entire scene reconstruction; instead, results from on-going research at the University of Florida and the University of South Carolina are used to mimic the anticipated performance.

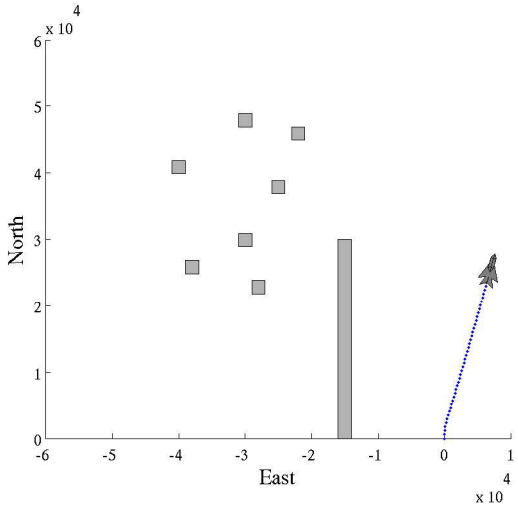

Figure 9. Flight Path using Optic Flow

The flight path is actually indicative of a controller that had perfect feature point extraction, perfect feature point tracking, perfect state estimation, perfect terrain mapping, and perfect path planning. Obviously such an assumption is unrealistic but it does provide a benchmark to demonstrate a potential measure of performance.

The fight path from this assumption of perfect analysis is presented in Figure 10. This result is only understood by noting that the updates are computed at the green points noted on the path. The vehicle records an image at the initial time $t = 0$, which shows only the wall. Assuming that the feature point analysis takes 2 minutes, or runs at 0.008 Hz, a resulting trajectory for this image is not created until point $t = 2$ *min*. More obstacles have come into view during the interval $t = 0$ *min* to $t = 2$ *min*; however, a trajectory resulting from the new image will not be computed until $t = 4$ *min*. By this time, the aircraft has already collided with an obstacle.

The images taken at $t = 0$ *min* and $t = 2$ *min* are shown in Fig. 11 to demonstrate the information available for scene reconstruction. Clearly the images indicate some obstacles and, assuming perfect reconstruction, can be used for path planning. The slow rate of the computation detracts significantly from the performance so the obstacles detected at $t = 2$ *min* do not cause deviation in the flight path until after a collision has occured at $t = 2.5$ *min*

The flight path in Figure 10 is meant to indicate a possible problem with scene reconstruction for aircraft. The approach has been used with considerable success for some systems, such as ground vehicles and even helicopters, that can stop and/or hover. An aircraft is continually moving forward so the computation delay can be devastating.

D. Multi-Rate Control

The multi-rate control scheme is also introduced to the simulation. This simulation uses the low-rate scene reconstruction to compute a path but watches for impending obstacles using the high-rate optic flow while following that path. Should a threat be detected, optic flow navigates the aircraft to safety. In this case, the optic flow operates at 100.0 *Hz* while the scene reconstruction operates at 0.008 *Hz*.

The initial commands from the two schemes, as shown in Fig. 9 and Fig. 10, try to steer the vehicles in different directions. So, the threshold, σ, in Eq. 16 is used such that the optic flow controller does not

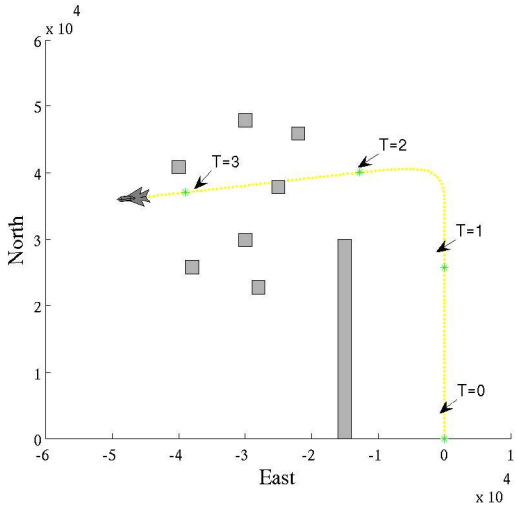

Figure 10. Flight Path using Scene Reconstruction

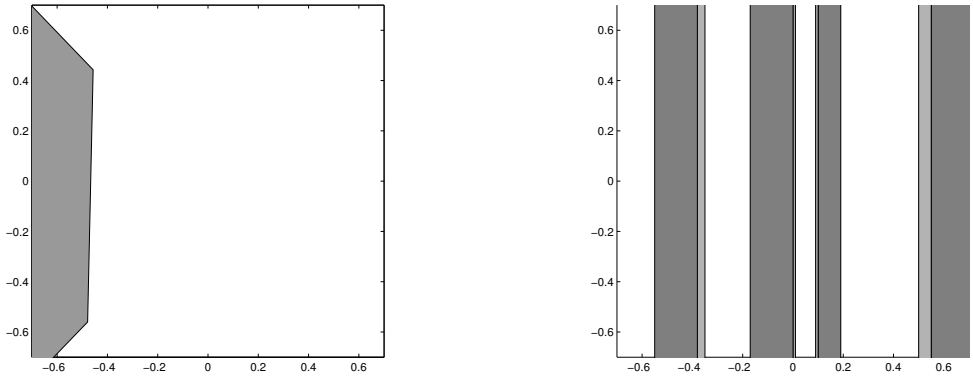

Figure 11. Camera View at T=0 (Left) and T=2 (Right)

change the vehicle path until the magnitude of flow passes a critical value. In this way, the vehicle follows the scene reconstruction but avoids obstacles.

Figure 12 presents the results of the integrated multi-rate controller. The path initially follows the commands from the scene reconstruction because the optic flow does not indicate any imminent threats. The command switches to optic flow at $t = 2.16$ *min* when the optic flow notes the aircraft is approaching a small obstacle. The vehicle continues using optic flow to avoid obstacles until $t = 2.85$ *min* at which time command was transferred back to scene reconstruction. The controller was again switched to optic flow at $t = 3.58$ *min* to avoid a final obstacle and continued to follow commands based on optic flow until $t = 3.81$ *min*. The controller based on scene reconstruction commands the vehicle during the remainder of the simulation until the final waypoint is reached.

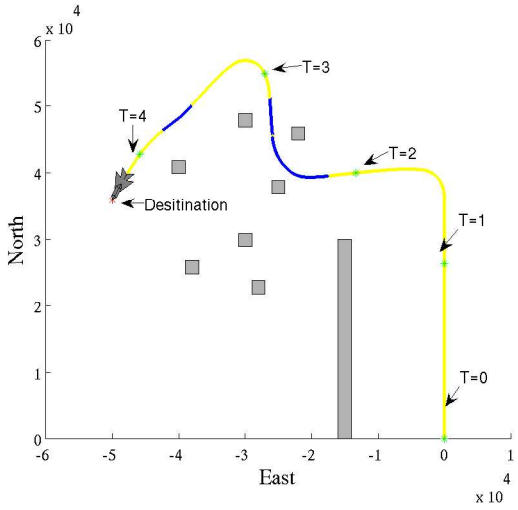

Figure 12. Flight Path using Multi-Rate Control

VI. Acknowledgments

This work was supported jointly by the Air Force Research Laboratory and the Air Force Office of Scientific Research under F49620-03-1-0381 with Johnny Evers, Neal Glassman, Sharon Heise, and Robert Sierakowski as project monitors.

VII. Conclusion

This paper demonstrates that a multi-rate scheme can be valuable for vision-based control. Both optic flow and feature point analysis are powerful approaches; however, each has relative suitability for on-board autonomous vehicles. The feature point analysis can clearly provide path plans but the computation time is excessive. The optic flow can be implemented on hardware that is fast but provides limited information. So, the high-rate optic flow merged with the low-rate feature analysis provides a beneficial structure. A simple simulation shows how an autonomous vehicle can utilize both types of vision-based feedback to avoid obstacles while still pursuing a mission objective.

References

[1]B. Lucas and T. Kanade, "An Iterative Image Registration Technique with an Application to Stereo Vision," *Proceedings of the DARPA Image Understanding Workshop*, 1981, pp. 121-130.

[2]B.K. Horn and B.G. Schunk, "Determining Optical Flow," *Artificial Intelligence*, 1981, Volume 17, pp. 185-203.

[3]K.-T. Song and J.-H. Huang, "Fast Optical Flow Estimation and Its Application to Real-time Obstacle Avoidance," *IEEE International Conference on Robotics and Automation*, 2001, Volume 3, pp. 2891-2896.

[4]F. Ruffier, S. Viollet, S. Amic and N. Franceschini, "Bio-Inspired Optical Flow Circuits for the Visual Guidance of Micro-Air Vehicles," *International Symposium on Circuits and Systems*, May 2003, Volume 3, pp. 846-849.

[5]G.L. Barrows, "Future Visual Microsensors for Mini/Micro-UAV Applications," *7th IEEE International Workshop on Cellular Neural Networks and their Applications*, July 2002, pp. 498-506.

[6]G.L. Barrows, J.S. Chahl and M.V. Srinivasan, "Biomimetic Visual Sensing and Flight Control," *2002 Bristol UAV Conference*, April 2002.

[7]G.L. Barrows and C. Neely, "Mixed-Mode VLSI Optic Flow Sensors for In-Flight Control of a Micro Air Vehicle," *SPIE*, July 2000.

[8]G.-S. Young, T.-H. Hong, M. Herman, J.C.S. Yang, and J.C.S., "Obstacle Detection for a Vehicle Using Optical Flow," *Intelligent Vehicles Symposium*, June 1992, pp. 185-190.

[9]G. Baratoff, C. Toepfer, M. Wende and H. Neumann, "Real-Time Navigation and Obstacle Avoidance from Optical Flow on a Space-Variant Map," *IEEE International Symposium on Intelligent Control*, September 1998, pp. 289-294.

[10]A.P.A. Castro, J.D. Silva and P.O. Simoni, "Image Based Autonomous Navigation with Fuzzy Logic Control," *International Joint Conference on Neural Networks*, July 2001, Volume 3, pp. 2200-2205.

[11]P.C. Merrell, D.J. Lee and R.W. Beard, "Obstacle Avoidance for Unmanned Air Vehicles Using Optical Flow Probability Distributions," *SPIE Optics East, Robotics Technologies and Architectures*, Mobile Robot XVII, October 2004, Volume 5609-04.

[12]Y. Ma, S. Soatto, J. Kosecka and S.S. Sastry, *An Invitation to 3-D Vision*, Springer-Verlag Publishers, New York, NY, 2004.

[13]S. Ullman, *The Interpretation of Visual Motion*, MIT Press, Cambridge, MA, 1979.

[14]A. Kurdila, M. Nechyba, R. Prazenica, W. Dahmen, P. Binev, R. DeVore and R. Sharpley, "Vision-Based Control of Micro Air Vehicle : Progress and Problems in Estimation," *IEEE Conference on Decision and Control*, December 2004, pp. 1635-1642.

[15]H. Wang and M. Brady, "A Structure-from-Motion Algorithm for Robot Vehicle Guidance," *Intelligent Vehicles Symposium*, July 1992, pp. 30-35.

[16]M.J. Stephens, R.J. Blissett, D. Charnley, E.P. Sparks and J.M. Pike, "Outdoor Vehicle Navigation Using Passive 3D Vision," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, June 1989, pp. 556-562.

[17]T. Kanade, O. Amidi and Q. Ke, "Real-Time and 3D Vision for Autonomous Small and Micro Air Vehicles," *IEEE Conference on Decision and Control*, December 2004, pp. 1655-1662.

[18]B. Sinopoli, M. Micheli, G. Donato and T.J. Koo, "Vision Based Navigation for an Unmanned Aerial Vehicle," *IEEE International Conference on Robotics and Automation*, 2001, Volume 2, pp. 1757-1764.

[19]D.A. Forsyth and J. Ponce, *Computer Vision : A Modern Approach*, Prentice-Hall Publishers, Upper Saddle River, NJ, 2003.

[20]L.M. Lorigo, R.A. Brooks and W.E.L. Grimsou, "Visually-Guided Obstacle Avoidance in Unstructured Environments," *IEEE International Conference on Intelligent Robots and Systems*, September 1997, Volume 1, pp. 373-379.

[21]Y.-S. Yao and R. Chellappa, "Dynamic Feature Point Tracking in an Image Sequence," *12th IAPR International Conference on Pattern Recognition*, October 1994, Volume 1, pp. 654-657.

[22]H. Longuet-Higgins, "A Computer Algorithm for Reconstructing a Scene from Two Projections," *Nature*, Vol. 293, 1981, pp. 133-135.

[23]T. Huang and O. Faugeras, "Some Properties of the E-Matrix in Two-View Motion Estimation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 11, No. 12, 1989, pp. 1310-1312.

[24]Z. Zhang, R. Deriche, O. Faugeras and Q.-T. Luong, "A Robust Technique for Matching Two Uncalibrated Images Through the Recovery of the Unknown Epipolar Geometry," *Artificial Intelligence Journal*, Vol. 78, 1995, pp. 87-119.

[25]W. Grimson, *"From Images to Surfaces: A Computational Study of the Human Early Visual System*, MIT Press, Cambridge, MA, 1981.

[26]O. Faugeras, *Three-Dimensional Computer Vision*, MIT Press, Cambridge, MA, 1993.

[27]M. Okutami and T. Kanade, "A Multiple-Baseline Stereo System," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 15, No. 4, 1993, pp. 353-363.

[28]J. Koenderink and A. Van Doorn, "Affine Structure from Motion," *Journal of the Optical Society of America : Part A*, Vol. 8, 1990, pp. 377-385.

[29]C. Tomasi and T. Kanade, "Shape and Motion from Image Strems under Orthography : A Factorization Method," *International Journal of Computer Vision*, Vol. 9, No. 2, 1992, pp. 137-154.

[30]R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, Cambridge, UK, 2000.

[31]R. DeVore and W. Dahmen, "Notes on Learning Theory," in press.

[32]M. Zabarankin, A. Kurdila, O. Prokopyev, A. Goel, V. Boginski, R. Causey, S. Uryasev and P. Pardalos, "Vision-Based Trajectory Planning for Autonomous Micro Air Vehicles in Urban Environments," *IEEE Robotics and Automation Magazine*, in review.

[33]R.S. Causey and R. Lind, "Aircraft-Camera Equations of Motion,"" *AIAA Journal of Aircraft*, In Progress.

[34]B.L. Stevens and F.L Lewis, *Aircraft Control and Simulation*, Wiley, 2003.