Abstract—Operating in rough, unstructured terrain is an essential requirement for any truly field-deployable ground robot. Search-and-rescue, border patrol and agricultural work all require operation in environments with little established infrastructure for easy navigation. This presents challenges for sensor-based navigation such as vision, where erratic motion and feature-poor environments test feature tracking and hinder the performance of repeat matching of point features. For vision-based route-following methods such as Visual Teach and Repeat (VT&R), maintaining similar visual perspective of salient point features is critical for reliable odometry and accurate localisation over long periods. In this paper, we investigate a potential solution to these challenges by integrating a gimbaled camera with VT&R on a Grizzly Robotic Utility Vehicle (RUV) for testing at high speeds and in visually challenging environments. We investigate the benefits and drawbacks of using an actively gimbaled camera to attenuate image motion and control viewpoint. We compare the use of a gimbaled camera to our traditional fixed stereo configuration and demonstrate cases of improved performance in Visual Odometry (VO), localisation, and path following in several sets of outdoor experiments.

I. INTRODUCTION

In order for field-robotic systems in applications to be effective tools in scenarios such as search-and-rescue, border patrol and agriculture, the sensor systems they use must be reliable in their intended environments. Rough, unstructured terrain is a frequent encounter in these deployments, and can challenge the ability of state-estimation algorithms that rely on sensing with limited update rates and limited perspective such as Light Detection And Ranging (LiDAR) and vision. Additionally, vision-based systems generally rely on complex, textured terrain invariant to environmental change in order to accurately track features, meaning that more uniform textures from specific viewpoints also cause issues with reliability. Generally, research demonstrations of visual navigation on ground robots are heavily biased towards smooth trajectories and carefully planned viewpoints to avoid poorly textured surfaces in order to achieve robust results.

VT&R is a path-following algorithm capable of autonomously driving a robot by following a previously driven route [1]. By extracting features taken from a monocular or stereo camera in the live view and matching them back to those extracted in a perspectively similar ‘teach’ view, relative path tracking error can be computed and sent to a path tracking controller to drive the robot along the path.

Computational complexity is constant with respect to map size, which enables long-distance operation, but the algorithm is highly dependent on matching Speeded-Up Robust Features (SURF) or other similar point features, and hence is susceptible to both perspective and appearance change. One of the key advantages of VT&R, however, is that it exploits the strengths of computer vision by keeping the viewpoint as similar as possible between repeat traverses of a path, i.e. as a crude version of active perception.

In extended field deployments, however, we have encountered situations that challenge our reliance on fixed cameras to achieve the goal of true long-term autonomy. These include areas of poor features directly in front of the vehicle (such as smooth asphalts, sandy or snow-covered areas), where looking in an alternative direction would improve matching performance. More regularly in our outdoor deployments, driving on steep inclines causes the sky to fill a large portion of the field of view, meaning feature tracking is impossible. Finally, driving at high speeds results in general VO failures in rough terrain due to loss of feature
tracks and poor localisation performance, which are highly coupled effects. For most activities, the robot is driven at well below walking pace and is still subject to a significant amount of careful perspective planning for reliable long-term operation. Addressing these deficiencies will result in a more reliable VT&R system and assist in bridging the gap between research testing and real-world deployments.

This paper presents a gimballed VT&R system, where the on-board camera is stabilised against pitch, roll, and yaw motions and actuated when necessary during the teach pass to improve performance. We describe our gimballed system, including both hardware and software changes, and we investigate the performance of this gimballed setup in comparison to our traditional fixed-camera VT&R when performing high-speed manoeuvres and in perceptually difficult situations.

The rest of this paper is outlined as follows: Section II discusses related work in vision-based navigation, focusing on the active perception problem, Section III describes the VT&R methodology and the improvements implemented to use a gimballed system with the framework, Section IV presents the experimental configuration and online testing framework for the gimballed VT&R algorithm and shows the results of this testing. Section V discusses the impact of these results and the paper is concluded in Section VI.

II. PREVIOUS WORK

A large and well-known number of visual feature detectors and descriptors exist [2], [3], [4], [5] for robustly matching keypoints in imagery. However, they are all subject to performance degradation due to lighting and perspective change. Many features have rotationally invariant options, but their reliance on computing a consistent orientation means matching often suffers when using these versions. By restricting some dimensions of motion when matching, such as driving on smooth ground or using active perception, performance is often significantly improved. VT&R already performs a crude form of active perception when repeating, maintaining as similar a viewpoint for matching as possible. This ensures perspective differences are minimised when repeating a route.

For vision-only odometry systems, breaks in feature tracks and smooth odometry due to rotational motion cause significant error build up in pose estimates over long trajectories. Visual-Inertial Navigation Systems (VINS) [6], [7], [8] can address the problems caused by rough or extreme motions very well by using complementary measurements to address the deficiencies of each sensor. By updating state through inertial measurements during breaks in visual tracking, accuracy over very long and extremely irregular trajectories are possible. Similarly, semi-dense methods such as SVO [9] allow accurate pose tracking under fast motions through very high frame-rates and the ability to track very weak features on uniform surfaces.

However, both VINS and semi-dense odometry methods do not solve the perspective problem and the inherent brittleness of matching point features under perspective change. Even non-feature-based methods are susceptible to perspective changes (e.g., driving in a different lane) [10]. Topological methods that are more robust to appearance change exist [11], but at the expense of metric localisation, which is critical to the online performance of VT&R. Using hardware can also be an effective way to improve robustness. We have investigated this before by using multiple stereo cameras [12] to look in opposing directions, countering difficulties caused by sun-glare and uniform surfaces. Alternatively, omnidirectional or catadioptric cameras improve invariance to perspective, but with an associated loss of metricality and consequent accuracy.

Surprisingly, there is little (to our knowledge) published work that includes active gimbaling in the performance of visual navigation on ground robots. Of course, visual servoing is a large and related field that addresses the issue of control to achieve a desired viewpoint of robot manipulators [13], [14]. Several approaches apply these concepts to mobile robots [15]. Our standard fixed-camera system can be considered a form of visual servoing, where, given the current pose and feedback from a visual sensor, a control system actuates the plant to pass through a set of desired viewpoints.

III. METHODOLOGY

In this paper, we use our well-established VT&R 2.0 software system as presented in [16] and extend this to take advantage of the gimballed camera. As in [16], the autonomous driving algorithm consists of separate teach and repeat phases. During the teach phase, the vehicle is manually driven by a human operator along a desired route, while the VT&R algorithm performs passive visual odometry; inserting the visual observations from this privileged experience into a relative map of pose and scene structure. During the repeat phase, without reliance on Global Positioning System (GPS) or other sensors, the vehicle should autonomously re-follow the route by visually localising to the map of the privileged path. The vehicle repeats a path by sending high-frequency localisation updates to a path-tracking controller [17]. In the following sections, we describe our VT&R system extensions to use a gimballed camera setup.

A. Gimballed Visual Odometry

During both teach and repeat phases, image pairs are captured by a calibrated stereo camera at a frame rate of ~15-20Hz, while the gimbal state (read as roll-, pitch-, and yaw-axis angular positions) is captured at approximately 30Hz. The gimbal state gives the pose of the camera in the vehicle frame by compounding the captured gimbal angles through a series of transforms with known translations. We denote the vehicle-to-sensor (camera) transform at time $t$ as $T_{sv}(t)$.

For each stereo image pair captured at time $t$, upright SURF features are extracted, descriptors generated and landmarks triangulated. Then, each feature in this latest frame-pair is matched to the last keyframe via SURF descriptor matching guided by a trajectory estimate. The raw matches
are then matched to landmarks using Maximum Likelihood Estimation SAmple Consensus (MLESAC) to find the relative (temporal) transform between the current frame and latest keyframe’s poses in the vehicle frame, using $T_{sv}$ at their respective time points. Finally, the temporal transform is optimised using our Simultaneous Trajectory Estimation And Mapping (STEAM) bundle adjustment engine. A trajectory estimate is also extracted at this point to guide matching extrapolate pose for the next frame, which is important for robust performance of our gimbaled system.

If a certain criterion is met, such as the number of inliers drops below a threshold or a component of 6-Degree of Freedom (DoF) motion exceeds a threshold, the frame is set as a keyframe and the features, new landmarks, and $T_{sv}$ are stored in a vertex in a pose graph for future retrieval. The relative transform is stored as an edge to the previous vertex. The vertex is marked as privileged if generated during the teach phase. Windowed bundle adjustment is then performed on the last 5-10 vertexes. For a more thorough explanation of this component, we direct the reader to our previous work [16].

B. Localisation

During the repeat phase, after a new vertex is created and windowed bundle adjustment completes, a separate localisation process is run to estimate the spatial transform and tracking errors between the current vehicle’s pose and the closest privileged vertex in the graph.

Here we introduce the ‘localisation chain’, a conceptual representation of the pose of the current keyframe, the spatially closest privileged vertex, the compounded transforms, and the path between them through the graph. These are visualised in Fig 2 as a snapshot of a repeat run. The latest vertex (a.k.a., keyframe) in the repeat is $V_b$, the live frame is $V_c$, and the current closest privileged vertex to $V_b$ is $V_e$. The chain is the transform $T_{eb}$, compounded through the shortest path from $V_e$ to $V_b$. If $V_b$ has just been added to the graph but not yet localised, the chain would follow $e, d, a, b$. The chain is updated after every live frame to keep track of the current transform, $T_{ec}$, and the current spatially closest privileged vertex is updated to $V_f$ by searching forward along the privileged edges’ graph and finding when the translational component of $T_{fe}$ is smaller than $T_{ec}$.

Localisation follows a similar basic process to the keyframing component of visual odometry, but with some distinct differences. We describe this process as if $V_b$ has just been added to the graph as a new vertex. First, a subgraph or window of vertexes (2-5 frames in both the forward and reverse directions) is extracted on the privileged (teach) path centered at the closest privileged vertex $V_e$. Within this window, all the landmarks from each vertex are migrated through their respective transforms to the privileged vertex $V_e$. This places them in a single local Euclidean frame centered at $V_e$. The respective descriptors of each migrated landmark area are then matched in order of vertex hops from $V_e$ to those in $V_b$. Positive matches and migrated landmarks are then used to find the new spatial transform $T_{eb}$ via MLESAC. This transform is then inserted in the graph, and the localisation chain from $V_e$ to $V_b$ is updated with the new transform following the shorter path. While this description assumes a single live pass and single privileged run, we leverage multiple experiences and use landmarks from selected intermediate runs to improve localisation, as described in [16].

C. Path-Tracking Control

We use the path-tracking controller first presented in [17] for following the teach pass during autonomous repeats. For the current section of the path to which the robot is estimated to be closest, the path-tracking controller sets a desired forward speed based on the curvature of the privileged path at that pose. Typically, higher curvature (smaller radius of turn) require slower speeds. For regular path following using our Clearpath Grizzly, we set a conservative speed profile, which can be described as a ‘strolling’ pace. Generally, faster speeds mean the likelihood of path-following errors (and consequently localisation failures) increase, as the fixed latency of the VO and localisation algorithms mean a delay in sending up-to-date cross-track errors to the path-tracking controller. For higher-speed operation, we set a speed profile with faster desired velocities for each curvature value. To achieve smooth and consistent performance, the desired speeds are subject to an acceleration constraint of 0.25m/s$^2$. This prevents large accelerations and decelerations when transitioning from tight turns to long, straight sections and vice versa, which tend to be a significant contributor to poor path-tracking accuracy.

To remove the influence of the Iterative Learning Control (ILC) algorithm of the path tracker on localization results, the learning component is turned off for the experiments presented in this paper. By using the ILC algorithm, later experiments would be biased to improved path tracking by learning terrain abberations. However, the standard path-tracking controller at the core of our other VT&$R$ work is still used.
D. Gimbal Control

There are two potential approaches to gimbal control from the perspective of VT&R; passive or active. In the passive strategy, the gimbal’s internal controller will stabilise pitch and roll in the camera frame, while using a smoothing controller on yaw to maintain an approximately forward facing viewpoint. This requires no additional control inputs from the VT&R system and can be considered an open-loop controller.

In the active strategy, the gimbal can be commanded to reduce angular error between the current (live) view and nearest privileged view, if knowledge of the transform between the current and the privileged poses is known. For this to occur, we leverage the methodology described in the previous subsections and pictorialise the strategy in Figure 2. First, we utilise STEAM to extrapolate the estimated pose at the current time, given latency in the visual processing algorithm, and include an additional fixed timestep to account for transmission delays. Typically, the latency between frame capture and output transform estimate, $T_{cb}$, is on the order of 60-100ms, but there is also a fixed delay of approximately 200ms between the sending of command and active motion of the gimbal due to message transmission latencies. We defined this extrapolated (predicted) pose as $V_p$.

Given $T_{pc}$, the localisation chain is queried for the nearest privileged pose in relation to $V_p$, denoted as $V_g$. The relative pose between $V_g$ and $V_p$ is then compounded through the transforms:

$$T_{pc}, T_{cb}, T_{bc}, T_{ef}, T_{fg}$$

This is the equivalent of the prior as stated in [16]. Of course, $V_f$ and $V_g$ could be equivalent, and transforms are compounded as needed by compounding transforms on the shortest path between $V_g$ and $V_p$.

The desired transform denoting the positional and rotational error between the closest privileged view and the live viewpoint is defined as:

$$T_{pg(s)} = T_{sv(p)} T_{pg} T_{sv(g)^{-1}}$$

where subscript $(s)$ denotes the sensor frame.

Given the rotational component of this transform, the desired gimbal angle at each of the control axes is sent to the gimbal controller. In this case, the roll is left as open-loop control, but pitch and yaw are closed-loop.

Both the active and passive strategies are trialled in Section IV to justify why active gimbal control is required in order for the gimballed system to outperform a static camera system.

IV. EXPERIMENTS

We use a Clearpath Grizzly RUV (Figure 1) as the base platform for two separate experiments. In the first experiment, we examine the performance of the gimballed system specifically in high-speed driving. In the second experiment, we examine the performance of the gimballed system in a long-term localisation scenario in tough outdoor conditions.

A. High-Speed Driving

In this experiment, the Grizzly RUV is fitted with our standard Point Grey XB3 camera system, placed facing forwards on a central mast (see Figure 1), as this will form the baseline to which we will compare our gimballed system. Specific to this paper, we also rigidly mount a DJI Matrice 600 multirotor body and DJI Ronin-MX gimbal to the mast of the Grizzly. The Matrice 600 provides the interface to the gimbal via a serial connection and sends the gimbal state (joint-angles) at up to 100Hz to a resolution of 0.1°. A Point Grey Bumblebee2 (BB2) stereo camera is placed on the gimbal, facing forwards. The gimballed system is placed specifically to closely match the mounting position of the XB3 to maintain as similar a perspective as possible, and the gimbal pitch is angled approximately 20° below horizontal to match that of the XB3. Each camera is connected to a separate Lenovo W541 laptop (8-core Intel Core-i7) for data processing purposes.

For this experiment, we use the short baseline of the XB3 camera (central and right cameras) to ensure fair comparison with the BB2 camera, which has the same 12cm baseline. The gimbal is commanded to actively stabilise pitch and roll, and we test both passive and active yaw control strategies. For the active strategy, the gimbal is not controlled by the operator during the teach phase but uses the closed-loop controller during repeats. Both the BB2 and XB3 are configured to generate images at 16Hz for comparative purposes, and both have an approximately 65° horizontal Field Of View (FOV).

The comparative performance of the fixed and gimballed systems are evaluated through a set of high-speed driving tests (Table II). For this comparison, a route covering approximately 100m is initially taught to both the fixed (XB3) and gimballed (BB2) camera systems covering the same trajectory (shown in Figure 3). Once the teach pass is complete and the endpoints of the path merged manually, the path is repeated multiple times over several hours by autonomously repeating the route interchangeably between actively gimballed, passively gimballed, and fixed camera systems, each controlling the vehicle during their respective tests, which we denote as ‘fixed’ for the traditional fixed XB3 rig, ‘passive’ for the passive gimballing strategy, and ‘active’, which uses our active gimbal control algorithm. The system is driven at the highest possible target speed given
TABLE I: Target speed profiles for differing radius of path curvature. The maximum speed for this profile can be described as a ‘running’ pace.

<table>
<thead>
<tr>
<th>Curvature (m⁻¹)</th>
<th>Target Speed (m/s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.01</td>
<td>2.25</td>
</tr>
<tr>
<td>0.2</td>
<td>2.75</td>
</tr>
<tr>
<td>1.5</td>
<td>3.5</td>
</tr>
<tr>
<td>5.0</td>
<td>4.25</td>
</tr>
<tr>
<td>10.0</td>
<td>5.0</td>
</tr>
</tbody>
</table>

TABLE II: Summary of the trials used for the experimental results

<table>
<thead>
<tr>
<th>Trial Number</th>
<th>Time started</th>
<th>Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Teach</td>
<td>13:00</td>
<td>passive+fixed</td>
</tr>
<tr>
<td>1, 2, 3</td>
<td>13:34, 13:36, 13:38</td>
<td>fixed</td>
</tr>
<tr>
<td>4, 5, 6, 7, 8</td>
<td>13:46, 14:05, 14:08</td>
<td>active</td>
</tr>
<tr>
<td>9, 10, 11</td>
<td>14:00, 14:05, 14:08</td>
<td>fixed</td>
</tr>
<tr>
<td>12, 13, 14, 15, 16</td>
<td>14:21,26,29,30,32</td>
<td>passive</td>
</tr>
</tbody>
</table>

acceptable safety limits, up to 5m/s (corresponding to path curvature as described in Section III-C). Each experiment or trial is repeated several times for consistency of results.

B. High-Speed Driving Results

To compare results and quantify the improvement of performance of the gimballed camera system, we evaluate metrics in several different ways: average feature track length during VO, total localisation matches, average localisation uncertainty, and camera actuation error.

Figure 4 shows the cumulative distribution of feature track lengths (the number of consecutive frames over which a landmark is tracked) during VO for the three different strategies over all repeat runs. This figure shows that a gimballed system improves the average track length over a static system by attenuating large image motions that cause tracks to break. The passive scheme can be seen to have a slight advantage over the active gimbaling scheme, potentially due to the smoother operation of the gimbal’s yaw control.

Turning to localisation, Figure 5 shows the mean number of MLESAC inliers for each keyframe for the three different strategies at each of the velocity profiles, and their 1-sigma standard deviation. While the mean number of inliers remains roughly equal on average, the standard deviation is significantly larger under the passively gimballed configuration, whereas the actively gimballed configuration achieves a slightly better variance than the fixed setup at the fastest speed profile.

Fig. 5: Mean localisation inliers at each localised keyframe for the three strategies. The markers are offset for clarity. While the mean inliers remains roughly equal on average, the standard deviation is significantly larger under the passively gimballed configuration, whereas the actively gimballed configuration achieves a slightly better variance than the fixed setup at the fastest speed profile.

The more important metric is the variance of the inliers for each strategy. Clearly, the passive strategy is inferior to the standard fixed-camera configuration, while active gimbaling shows a small improvement over the same.

Figure 6 shows the relative localisation yaw error for the different strategies over the same path. While the fixed and passive strategies show large angular errors due to the open-loop control, the active gimbaling strategy successfully attenuates large angular deviations during sections of poor path following.

Finally, the CDF of the localisation uncertainty for the same experiments is plotted in Figure 7, by velocity profile. In this setup, the actively gimballed system marginally improves localisation uncertainty throughout the dataset over both a fixed and passively gimballed strategy.

This can be attributed to two major components, the improved tracking performance of features in VO, meaning that landmarks are better triangulated with less average uncertainty compared to the fixed camera strategy, and more consistent perspective for localisation matching over the passive strategy. While the passive strategy will occasionally achieve reduced perspective error, performance in this metric is generally less consistent. In both the active and fixed strategies there were no localisation failures, but the passive strategy exhibited a 99.7% success rate.

C. Long-Term Experiment

In this experiment, the ability of the gimballed system to decrease the chance of localisation failure during a ‘grand-tour’ of deliberately challenging conditions is tested. A dataset is gathered on the robot over a period of two days within the meadows surrounding the University of Toronto Institute for Aerospace Studies (UTIAS) campus during mid-winter. At this time of year, the sun remains low in the sky, meaning sun glare and consequent image washout is a frequent occurrence. Also, snow cover is often significant and variable from day to day, meaning that certain areas have little salient texture and features change rapidly as snow falls and melts.

This setup, the actively gimballed system marginally improves localisation uncertainty throughout the dataset over both a fixed and passively gimballed strategy.
Fig. 6: Plot of relative localisation yaw error for the different strategies over three selected runs. While the fixed and passive strategies show large angular errors due to the open-loop control, the active gimballing strategy successfully attenuates large angular deviations during sections of poor path following.

Fig. 7: The CDF of translational localisation uncertainty during the high-speed experiment. The active gimballing strategy outperforms both the fixed and passive strategies.

For this experiment, a StereoLabs Zed stereo camera mounted on the gimbaled system is used, and the robot is taught a series of routes during the afternoon of the first day. The robot is first driven with the actively gimbaled camera across a snow-covered field where little permanent vegetation or structure is visible directly in front of the camera, followed by driving directly towards the sun, and then across a highly textured area to a large ditch through which a seasonal creek runs. In each of these sections, the camera is actuated (i.e., manually steered) to avoid the source of the degeneracy during the teach phase. This is in contrast to the high-speed experiments, where the gimbal was not actuated by the operator. In the snow-covered section, the camera is pointed towards vegetation, avoiding the poorly textured terrain covered by snow. When driven towards the sun, the camera is aimed below the horizon to avoid sun glare. In the ditch, the camera is pointed down to reduce the percentage of the image covered in sky, but is also allowed to automatically control pitch to maintain a stable orientation. These sections are visible in Figure 8.

The same set of routes is immediately re-taught while fixing the camera statically to face forward. In none of the aforementioned conditions is the static camera’s orientation changed. The robot is then commanded to re-traverse the route multiple times to build a set of experiences from which to match features using both the static and active strategies.

The next morning, the robot is commanded to re-follow the path using the experiences from the previous day. We record specific statistics for localisation throughout the experiment, such as the number of localisation inliers and localisation failures at each vertex of the traversed route. In total, each section of the route is traversed at least three times over the two day experiment. The passive strategy is not used in this experiment.

D. Long-Term Experiment Results

Results of the experiment are presented in Figures 8-11. Figure 8 shows the overall path followed by the robot for the two strategies. Each path is annotated with a circle at each vertex whose size denotes the average localisation inliers including all repeats. Larger-width paths show a greater average number of inliers at that vertex. Additionally, red circles are placed at each vertex where a localisation
failure occurred, whose size denotes the average number of localisation failures, following a log scale. During repeats, occasional failures do not necessarily require manual intervention due to reliance on VO, but a series of failures will ultimately cause the algorithm to exceed certainty bounds and stop the vehicle. Interesting segments of the path as described in Section IV-C are examined in Figures 9-11.

In Figure 9, the traversal of the ditch is highlighted. In all attempts, the fixed camera system fails to localise on the upward side of the path due to a significant view of sky. Each traversal required a manual intervention. Using the gimballed system, large sky views are avoided and no failures occur. Additionally, the average number of localisation inliers at each vertex remains high and stable.

In Figure 10, the snow-covered section is highlighted. Similarly to the ditch example, the average number of localisation inliers at each vertex remains high and stable using the gimballed system. In contrast, the fixed system again shows reduced inliers and far more frequent failures.

Finally, Figure 11 highlights the section affected by sun glare. Both strategies suffer in this section due to image saturation from directly viewing the sun. However, the actively gimballed system, by pointing away from the horizon, shows better performance and is able to successfully repeat the path without manual intervention. In contrast, the static camera system unfortunately fails in all cases and requires manual driving over a short 1m section.

Overall, the active strategy results in fewer localisation failures and generally higher numbers of feature inliers compared to the passive strategy. In the uphill section of Figure 8, even small amounts of sky-view cause the passive strategy to suffer due to fewer features in the upper portion of the image usually covered by trees.

V. DISCUSSION

From these experiments, the use of a gimbal to address operational limits has some intriguing outcomes. Surprisingly, the ability of the gimbal to improve performance during high-speed manoeuvres is marginal. While such a system is able to attenuate low-rate disturbances, the control envelope is insufficient to address large shocks that cause motion blur, which was a motivating use case of this system.

However, significant improvements are obvious when taking advantage of the gimbal’s ability to account for perspective. In the high-speed experiments, the gimbal is able to attenuate yaw error caused by poor path tracking (particularly on corners), while in the long-term experiment, active targeting of feature-rich and feature-stable areas drastically improves reliability over a static system. It is less effective in areas of rich texture. Additionally, little improvement was seen during fast turns.

Importantly, during the teach phase the gimbal must be controlled smoothly by the operator while avoiding sudden perspective changes. This is highlighted in the junction of Figure 8, where discontinuities in perspective cause frequent localisation failures for the active strategy. This requires the operators to be careful in planning where to point the camera during the teach phase. Our results suggest that further research into an autonomous attention model to actively point the camera may be fruitful.

VI. CONCLUSIONS

In this paper, we have shown the integration of a gimballed camera to VT&R to test the performance of localisation and path following in various conditions, including high-speed
driving and difficult localisation experiments. It has been shown that an actively gimbaled setup assists marginally in both VO performance and localisation uncertainty during high-speed driving, but shows significant improvement when faced with specific difficult cases of perspective that degrade the performance of a fixed camera.

Future work will focus on reducing the computational load for deployment on low-power embedded hardware and subsequent demonstration of the gimbaled VT&R on-board the DJI M600 multirotor vehicle.

ACKNOWLEDGMENT

This work was funded by Smart Computing for Innovation consortium (SOSCIP), Defense Research and Development Canada (DRDC), Drone Delivery Canada (DDC) and the Centre for Aerial Robotics Research and Education (CARRE), University of Toronto.

REFERENCES