A USDOT NATIONAL
UNIVERSITY TRANSPORTATION CENTER

Carnegie Mellon University

Aerial Contact Sensing for Improved Inspection of Transportation Infrastructure

Sebastian Scherer
orcid.org/0000-0002-8373-4688

Contract # 69A3551747111

1 Problem

An increasing number of bridges across the USA are considered deficient and require frequent inspection and repair. Sufficient inspection of transportation infrastructure requires an inspector to manually climb onto or underneath the structure to take hands-on measurements. This is not only expensive, but can also be dangerous.

The current state-of-the-art in using remotely controlled or small unmanned aerial systems (sUAS) for inspection focuses only on taking visual measurements from a distance of many meters. This is not a sufficient substitute for the inspection made via contact sensing by a human inspector. Furthermore, these systems often require global positioning (GPS) or a prior map of the environment to operate autonomously. Many inspection jobs (e.g., underneath a bridge) may not satisfy these requirements. The goal of this project is to develop an autonomous sUAS capable of taking contact sensing measurements in GPS-denied and previously unknown environments.

2 Approach

In this project we developed a first simple contact inspection system that has the potential to be low-cost and high-performance. This is achieved by using a novel configuration of tilted propellers with vision-based state estimation and control to allow flight without GPS.

Additionally, we focus on intuitive human interaction with the system to select regions of interest (ROI) for contact inspection. Simple, useful feedback is given if the system detects something is wrong. Visual servoing control is used for the approach and retreat from the contact surface to minimize required prior knowledge of the surface and the environment. State estimation of the robot is performed by combining measurements from various onboard sensors, while avoiding the use of expensive and heavy equipment such as LiDAR.

Standard quadrotor or hexarotor unmanned aerial vehicle (UAV) configurations are only capable of 4 degrees of freedom, and cannot move in lateral directions without changing roll and pitch angles. For contact inspection with an aerial platform, the forward-mounted contact sensor must be level before, during, and after contact to ensure a reliable measurement. We aim to forego an expensive gimbal-mounted sensor or actuated arm by developing a custom hexarotor with tilted rotors. This UAV has 6 degrees of freedom and is able to move and apply force in the lateral directions directly while maintaining a level roll and pitch. This allows a simple, rigid arm with a compliant end-effector to be sufficient for contact. While our progress towards developing more advanced control for this robot is not discussed in this report, the effort is ongoing in the interest of contact inspection with sUAS.

2.1 User Interaction

The sUAS should be autonomous such that construction workers or infrastructure inspectors may use the system without extensive prior training. This means that the user need not require a high level of experience in piloting UAVs. Additionally, inspectors should be able to interact with and use the system with tools with which they are already familiar.

A graphical user interface (GUI) was designed that contains all necessary communication between the user and the system. This GUI is launched from a WiFi-capable laptop computer (data is transmitted to and from the sUAS via WiFi), and allows the user to interact with the sUAS through mouse clicks. The user can command different states, such as Wait-hov (autonomous position hold in global frame), VS-hov (autonomous position hold in front of target), VS-fwd (autonomously approach target until within some set distance), and VS-bwd (autonomously retreat from target).

The GUI also provides a live video feed from the onboard camera. The user may select a ROI of custom size on this video feed and the corresponding bounding box area to be tracked will show up in an adjacent image. Feedback is given back to the user, such as "Please re-select ROI" and "Activating state: *state*". The user may re-select the ROI at any time and may choose to do so after being prompted by this feedback.

2.2 Perception

To guarantee reliable visual-based control of the robot it is necessary that the location of the ROI within the image is always known. For this purpose a tracking system was introduced that includes a depth-adaptive object tracker. The Intel D435 RGB-D camera is used. This sensor provides a RGB image and depth image, the latter formed either by interpreting distortion of a projected infrared pattern or via stereo image disparity calculations. Using both of these methods makes the sensor useful in a variety of indoor and outdoor environments.

Various state-of-the-art trackers were compared to assess usability for autonomous contact inspection. A framerate of at least 20Hz is necessary to match the rates of the control systems on board. High accuracy, demonstrated by few missed tracks and low false-positive rates, is also necessary to ensure the safety of the entire system during autonomous flight. As seen in Table 1 the `MOSSE` and `KCF` algorithms yield a high FPS but the tracking accuracy is not sufficient. On the other hand, the `CSRT` tracker achieved a reasonable performance in terms of frames per second (FPS) and good tracking results in terms of accuracy, so this was chosen for use in the perception package. Trackers were tested on a computer with an Intel i7 CPU with no discrete GPU.

Tracker	FPS	Accuracy
GOTURN	10	Medium
KCF	100-150	Medium
MOSSE	1000+	Low
BOOSTING	35	Low
MIL	11	High
TLD	6	High
CSRT	25	High

Table 1: Comparison of tracking algorithms on aerial data

Since the CSRT tracker does not reliably adjust the size of the tracked bounding box with distance to the tracked object (as distance decreases, size of the bounding box should increase), it is necessary for the outputted bounding box to be scaled manually with depth. The distance to the object is stored when the tracker is initialized with a ROI, and in subsequent tracked frames the bounding box is scaled relative to the measured depth from the sensor. This results in a bounding box that always spans the same area on the structure that is being tracked, regardless of the distance. Additionally, the framerate of the CSRT tracker decreases as the size of the bounding box increases. To alleviate this problem, we manually downscale the image for computing the updated track if the bounding box size surpasses some fraction of the image size. This way we ensure that the tracker will always run at least at the requried 20Hz.

2.3 Visual Servoing

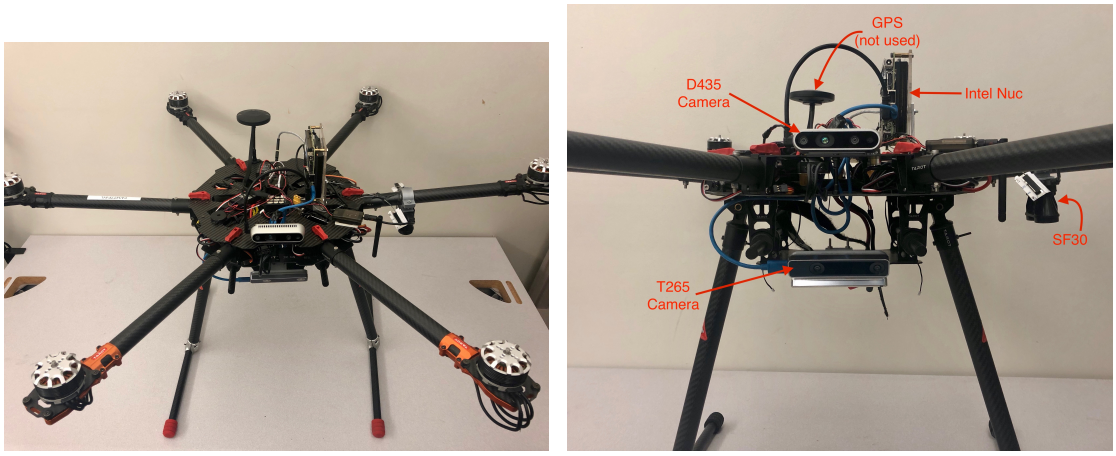
The sUAS should be able to autonomously approach a certain ROI for inspection as designated by the user. The goal for the UAV's control in this application is to ensure that the aerial trajectory coincides with the ROI upon contact. Any particular ROI should have only minimal prior informational requirements, not needing any prior geo-tagging or localization, be of any particular depth or size, or have been seen before by the system. This eliminates most control methods based on global positioning or learning.

Visual servoing uses only visual information to control the robot, which minimizes the constraints on what targets are viable for inspection. However, one of the major assumptions made for this project is that the surface within the selected ROI is roughly planar. The arm developed for this task is compliant in 4 degrees of freedom, and can adapt to some non-planar surfaces. However, the depth of the target taken from the D435 sensor is currently calculated as the median of the entire ROI, excluding points outside a certain depth range (for example, between 0.1 and 10.0 meters), which indirectly makes a planar surface assumption within the ROI.

There are many studied methods for visual servoing, generally classified into Image Based Visual Servoing (IBVS) and Pose Based Visual Servoing (PBVS). One IBVS method that directly translates pixel-distance errors into velocity control on the robot was tested in simulation and found to be unstable at varying distances from the target. PBVS was finally used because of the additionally available depth information from the D435 sensor. The target depth is combined with the ROI center's pixel-distance from the image center to find the x- and z-displacement of the target relative to the UAV in the body frame. These displacements were transformed into the global reference frame to form an estimated target position to be used in tandem with the developed state estimation package. This position was then used with the PX4 flight stack's position controller to control the UAV.

2.4 State Estimation

In order to robustly control the vehicle when inspecting infrastructure, localization in the world needs to be robust and locally accurate. This means in the long-term, some positional drift is acceptable but the system must be able to estimate its state accurately locally and in the short-term. Visual servoing techniques correct for long-term drift by estimating the target's position relative to the body and transforming the target point into the map (global) frame. However, an accurate state estimate is needed in order to precisely control the vehicle in order to reach the target position.



(a) Top view (b) Sensor payload

Figure 1: Aerial vehicle

Bridges, buildings, and other large pieces of infrastructure often block GPS signals and interfere with other sensors, rendering most off-the-shelf localization solutions inoperable. To solve this problem, an Intel T265 RealSense tracking camera is utilized, along with an SF30 Laser Rangefinder and the onboard IMU (see Figure 1). The T265 RealSense tracking camera utilizes localization methods such as SLAM to generate a 6 degree of freedom pose.

Readings from the onboard sensors are fused in a 15 degree of freedom Extended Kalman Filter (EKF) node provided by Robot Localization [1]. Utilizing the flexibility of the filter, the tracking camera was utilized for position and velocity in the horizontal plane and for absolute orientation. The SF30 provided a very accurate measure of the vehicle's altitude. Finally, the IMU provided linear and angular acceleration and orientation. This particular configuration provided the best performance in test flights.

2.5 Hardware Design

Infrastructure inspection is a demanding task that requires a rugged, robust, customizable, and powerful platform. The platform must be capable of operating in an outdoor environment under and over various forms of infrastructure (bridges, roadways, towers, etc.). It must be robust enough to make numerous and repeated contacts with the infrastructure under inspection. The platform must also be able to support various sensors, some of which may be relatively heavy.

For this project, we developed a fully-actuated hexarotor that uses a Tarot X6 frame, with a 0.96m motor-to-motor diameter and a maximum payload of 7.5kg (see Figure 1). The frame can be size-adjusted for specific tasks. Propulsion is achieved with six KDE Direct KDE4215XF-465 brushless motors and KDEXF-UAS55HVC electronic speed controllers (ESC). The flight controller is a Pixhawk 2 Cube running a modified version of the PX4 flight stack. On-board computation is achieved using an Intel NUC computer. An Intel T265 RealSense Camera is used to provide visual odometry information and a D435 RealSense Camera is used for RGB-D imaging, and an SF30 Laser Rangefinder is used to provide accurate height estimation.

The frame is relatively large, allowing for a large payload and numerous customization options. Additionally, the motors and propellers chosen allow for a long flight time, while also providing almost 29kg of peak thrust. Additionally, by utilizing the sensors listed above, robust state estimation in a GPS-denied environment was achieved at a relatively low-cost, when compared to state

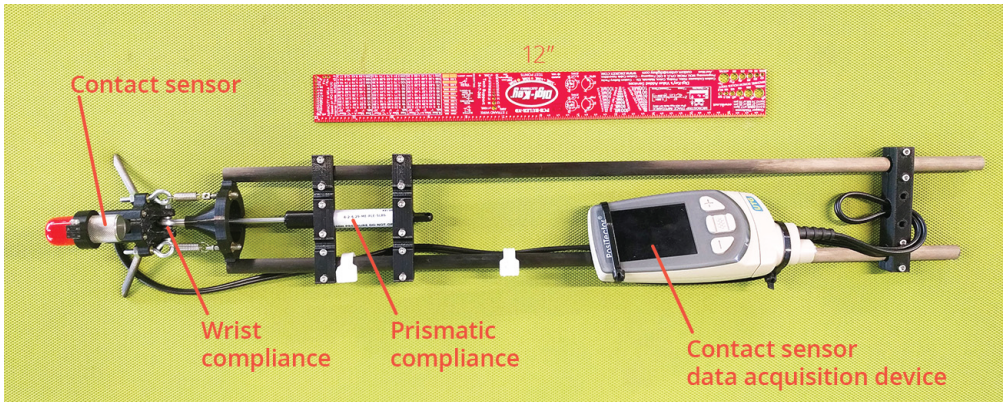


Figure 2: Contact sensor arm with compliant wrist

of the art localization techniques, and with little addition weight or power requirements.

An arm to be forward-mounted on the sUAS was used with a sonic depth gauge sensor attached to a compliant wrist. This was designed and built by Near Earth Autonomy. The end-effector has 4 degrees of freedom and three points of contact to achieve a stable contact with various smooth, textured, and non-planar surfaces. This arm was not mounted on the current UAV iteration, but was on a previous iteration.

3 Methodology

The system is built around ROS (Robot Operating System) and various ROS packages for the different sensors used. PX4 was the main flight stack, and the MAVROS package was used to interface with the rest of the system. Simulation was performed in Gazebo.

As described in a previous section, a number of state-of-the-art tracking algorithms were evaluated on real-world data taken using the D435 sensor. A perception package was built around the chosen tracker, which handled failure cases and made adjustments for the specific use-case of contact inspection. This perception package also included the GUI to be used by the user for selecting an ROI for inspection. The package was used as a tool in the development of the visual servoing control algorithm.

Visual servoing control algorithms were implemented and evaluated purely in simulation (see Figure 3) for determining the appropriate algorithm for this application. After the final algorithm was tested and tuned in simulation, it was run on the UAV without propellers. Hand-held control tests were used for validation prior to flight tests.

For state estimation, as stated previously, it is important for localization to be locally correct and accurate. Numerous approaches to the problem were attempted, all of which using the T265 tracking camera. Originally, the tracking camera's velocity estimate was fused with the rangefinder and IMU. However, through field testing it was found that fusing velocities alone caused significant drift (greater than roughly 20cm positional drift within less than 5 minutes of flight).

Additionally, it was discovered that no absolute orientation could be measured on-site, due to high magnetic interference with the onboard magnetometer from nearby large metal structures. As a result, it was decided to fuse the absolute position and orientation estimates provided by the T265. This approach eliminated most of the drift while maintaining a locally accurate estimate.

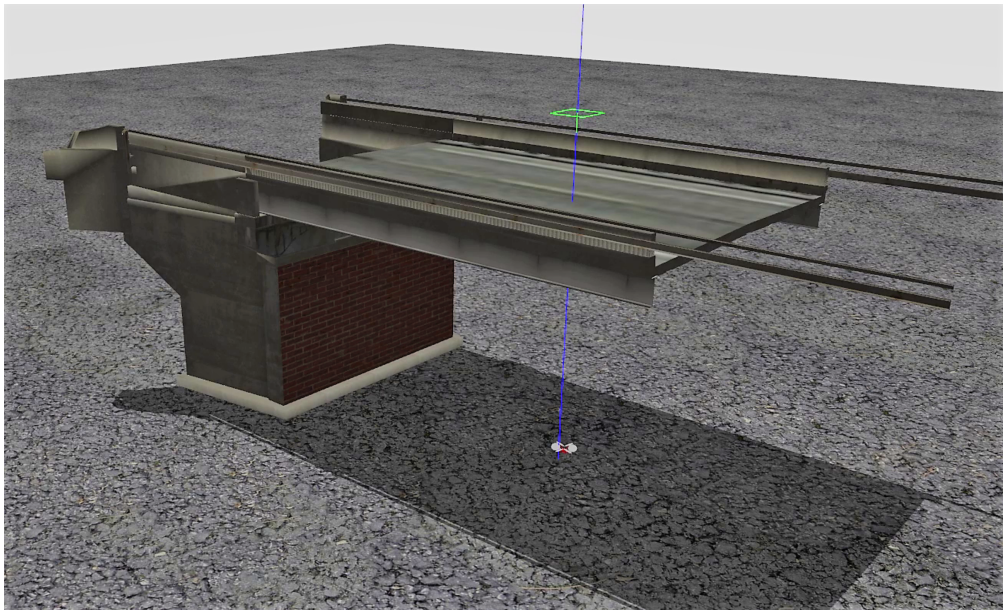


Figure 3: Gazebo simulation bridge environment for testing visual servoing algorithms

3.1 On-site testing at Charles Anderson Bridge, 7/15/19

Once each subsystem (perception, visual servoing, state estimation) was tested through simulation and indoor hand-held tests, final on-site subsystem validation tests were performed at the Charles Anderson Bridge in Pittsburgh, PA (see Figure 4). This took place on 7/15/19, under the supervision of Justin Bouscher. At all points in testing, safety was ensured by using safety glasses, high-visibility vests, hardhats, and warning passersby to the possible hazards when testing.

The control development for the tilted-rotor hexarotor platform is an ongoing effort, so this platform was not used for testing. Instead, a standard x-configuration hexarotor was used. First, every element of the sUAS was assessed: WiFi network connectivity, sensor measurements, flight readiness, user interaction, etc... Integration of all autonomous subsystems was a continuing challenge that ultimately prevented fully autonomous test flights during the planned time slot for testing. Instead, manual flights were performed at close proximity to the beam structures of interest to gather data and later test the perception and visual servoing systems with the collected real-world data. This data helped develop improved system robustness in the following days of work.

As this was the first time this sUAS was tested on-site, a significant portion of the allotted testing time was spent in ensuring safety measures and developing an efficient testing routine for future tests.

3.2 Outdoor testing at CMU, 7/21/19

Poor weather conditions and a large city-sponsored event in Schenley Park prevented access to the Charles Anderson Bridge on 7/21/19. Instead, tests were performed on the Carnegie Mellon University football field with appropriate safety precautions (see Figure 5). The makeshift surface for inspection composed of two cardboard boxes affixed together to be a height of about 2 meters.

At this point, the effort to develop stable control for the tilted-rotor hexarotor platform was still ongoing, so this platform was not used for testing. Instead, a standard x-configuration hexarotor was used. First tests involved manually controlled flights with an “inspector” using the GUI to

(a) Team members

(b) Manual test flight to collect data

Figure 4: Testing on-site at Charles Anderson Bridge, 7/15/19

(a) Team members

(b) sUAS hovering relative to selected ROI

Figure 5: Testing autonomous flight at CMU, 7/21/19

click points of interest to test the system's target tracking in-flight. Then, the first autonomous flights were attempted in the with the user having full control of the system via the GUI. In state Wait-hov, the UAV maintained constant position with only onboard, GPS-denied state estimation. This position hold was maintained for over 60 seconds with centimeter-level accuracy. The user then clicked a point on the inspection surface (placed about 7 meters away) and transitioned to state VS-hov. In this state, the UAV adjusted position to be directly in front of the selected target ROI (see Figure 6). It maintained this position for 30 seconds, at which point the user transitioned to Wait-hov and the safety pilot manually took over control and landed the robot. A third test was conducted that replicated the previous Wait-hov to VS-hov sequence, and then attempted VS-fwd to approach the inspection surface. However, in both modes Wait-hov and VS-hov the UAV slowly drifted and overshot setpoints in both the x-axis and y-axis of movement (both lateral axes). When transitioned to VS-fwd, the UAV steadily moved backward rather than forward. Further investigation showed that the position setpoints being sent to the system were to be indicating a desired backward movement, so likely an arithmetic flaw in these setpoint calculations is to blame for the error.

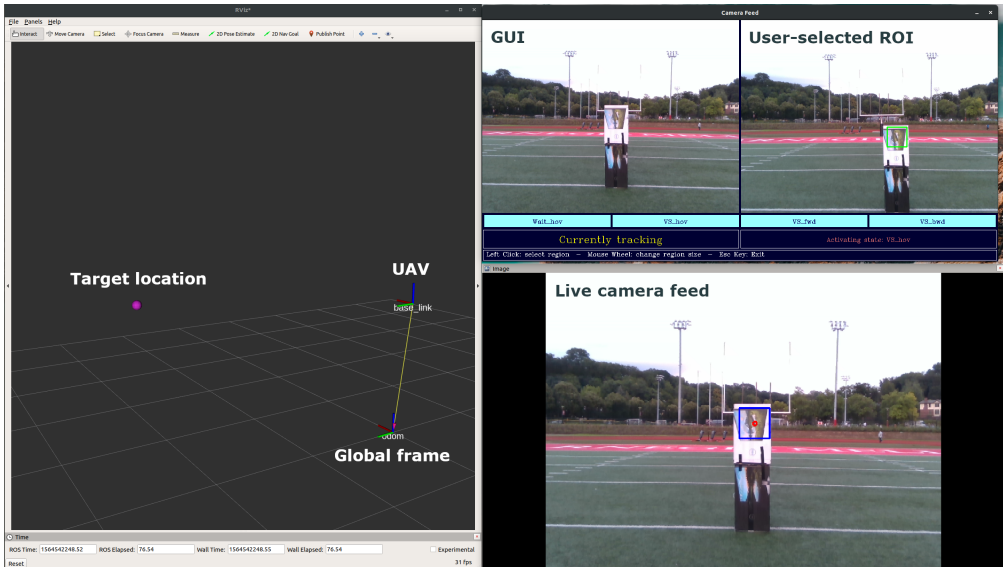


Figure 6: Robot and target frames, GUI, and live camera feed with tracked target during state VS-hov

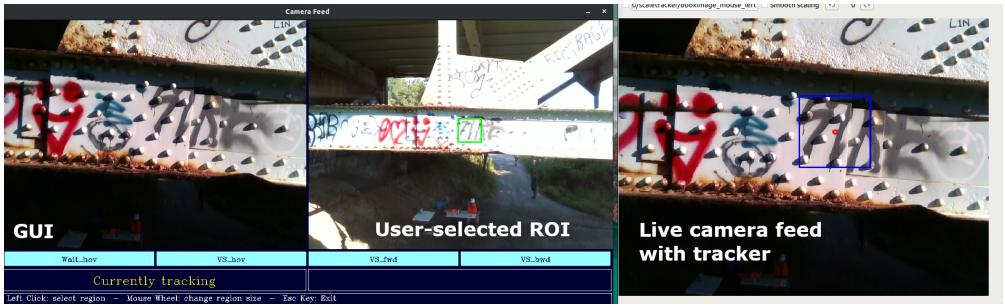


Figure 7: Perception package tracker robust to varying exposure and depth

4 Findings

4.1 Perception

Using the CSRT tracker for maintaining the target's location within the camera field of view worked very well in almost all tests. During flight tests, the track on a selected ROI never failed, with many flights reaching over a minute in length. Framerate was consistently over the 20Hz required minimum, the bounding box scaled with varying depth, and depth values were filtered appropriately to avoid unusable measurements. The only failure cases were when the inspection surface had very few features, which was only the case during hand-held perception subsystem tests. The assumption of sufficient features is typically a safe assumption for infrastructure inspection, since it is very likely for cracks, graffiti, or other structural components (e.g., rivets) to be within the chosen ROI (see Figure 7). However, when insufficient features are present in the scene (such as when inspecting a uniformly colored and non-textured wall), tests have shown that the tracker may snap the ROI bounding box to the closest, most feature-rich area in the image.

4.2 Visual Servoing

The calculation of the target position in the global odometry frame was shown to be accurate in-flight (see frame positions in Figure 6). The UAV maintained steady position hold relative to the selected target until the system state was changed. Later autonomous flight tests showed some instability in that the robot drifted and overshot the calculated setpoints, as discussed in Section 3.2. This likely indicates that the PX4 position controller needs to be tuned. Additionally, the unexpected behavior of retreating from the target when the sUAS should have been approaching the target was found to be due to an arithmetic error in the relative-to-target position setpoint calculations.

The primary finding for the visual servoing effort in this project was that commanding position setpoints based on RGB-D data of a given target is sufficient for control of a sUAS.

4.3 State Estimation

Overall, robust localization and state estimation was achieved utilizing the onboard sensor suite, despite the lack of conventional means of localization, such as GPS. This was demonstrated and tested on-site at the Charles Anderson Bridge with manual flights and on campus with autonomous flights. Estimates were consistent and correct and accurate, otherwise such autonomous flight would not be possible. Finally, the developed system, unlike conventional means of localization, can work in significantly more environments where GPS and other sensors can not operate.

5 Conclusions

This project developed a fully-actuated hexacopter based system and developed prototypes and integrated all the software and hardware component technologies for an aerial contact inspection system.

The components and system were validated in simulation and field experiments. While fully autonomous contact inspection results were not achieved, data to help with future development and a validation of component technologies was possible. The results indicate that state estimation, tracking, and control in a bridge environment is a valid concept and an aerial contact inspection system could be a feasible option to assess the state of transportation infrastructure.

Several open questions remain that could not be addressed in the current project: Can contact force be controlled precisely enough for contact inspection? How can the mechanical system be ruggedized for inadvertent contact (perhaps utilizing prop-guards or a protective cage)? Can the control be made more robust to wind disturbances? Is the chosen interaction paradigm the best paradigm? Is the user workload sufficiently low to be manageable by one person?

6 Recommendations

If an integrated system is developed, this type of aerial contact inspection could reduce the cost and impact (e.g., lane closures) of inspection significantly. Additionally, the increased speed of inspection and digital data capture could lead to a better future method of managing transportation infrastructure.

For this vision to be possible, further research in the following areas is recommended:

1. Improve the control approach for precise maneuvering in contact: the current tilted-hexacopter control is over simplfied and is not suitable for precise visual servoing as well as force control.

2. The mechanical system needs to be robust to contact from all sides. A robust system would maintain stability and continue flying through deliberate and inadvertent contact.
3. High winds are frequently present under bridges and the control as well as autonomy system need to be able to react to different wind conditions.
4. While the current control paradigm of manual defined contact inspection is very flexible, a longer term goal would be to completely automate the decisions of when and where to take contact measurements.

7 Shared Data

A point cloud of the underside of the East side of Charles Anderson Bridge was generated using another bridge inspection sUAS developed by the AirLab. The pointcloud can be viewed using software such as CloudCompare. The zipped folder can be found at this hyperlink, hosted by Google Drive (please request access permission if prompted to do so):
Charles Anderson Bridge Underside EastSide, July 15, 2019

References Cited

[1] T. Moore and D. Stouch, "A generalized extended kalman filter implementation for the robot operating system," in *Proceedings of the 13th International Conference on Intelligent Autonomous Systems (IAS-13)*, Springer, July 2014.