2022 NDIA MICHIGAN CHAPTER
GROUND VEHICLE SYSTEMS ENGINEERING
AND TECHNOLOGY SYMPOSIUM
AUTONOMY, ARTIFICIAL INTELLIGENCE & ROBOTICS TECHNICAL SESSION
AUGUST 16-18, 2022 - NOVI, MICHIGAN

SATELLITE IMAGE TEMPLATE MATCHING WITH COVARIANCE ESTIMATION FOR UNMANNED GROUND VEHICLE LOCALIZATION: ACTIVE TERRAIN LOCALIZATION IMAGING SYSTEM (ATLIS)

Kenneth Niles[1], Steven Bunkley[1], W. Jacob Wagner[2], Isaac Blankenau[2], Anton Netchaev,PhD[1], Ahmet Soylemezoglu,PhD[2]

[1]USACE ERDC ITL Vicksburg, MS
[2]USACE ERDC CERL, Champaign, IL

ABSTRACT

Robotic platforms require accurate geo-spatial localization for high-level mission planning, real-time site reconnaissance, and multi-machine collaboration. Global navigation satellite system (GNSS) receivers are most commonly used to provide UGVs with accurate geolocation. However, GNSS is not reliable in contested environments because it is vulnerable to jamming, spoofing and black-outs. To address these issues, the United States Army Corps of Engineers (USACE) - Engineer Research and Development Center (ERDC) has developed the Active Terrain Localization Imagery System (ATLIS) which uses on-board perception and a priori satellite imagery to eliminate reliance on GNSS for global positioning of a ground vehicle. Using LiDAR and camera imagery, ATLIS creates a vehicle-centric, orthorectified image that is compared to an a priori satellite image using template matching. It then produces a global position estimate for the vehicle. We develop a method to estimate the uncertainty of this position estimate, enabling fusion with other relative positioning sensors. We demonstrate the effectiveness of ATLIS in aiding localization of an unmanned ground vehicle (UGV) in a complex outdoor environment achieving an average planar euclidean distance error of 1.21m over a 5.1km run when compared to a GPS ground truth.

Citation: K. Niles, S. Bunkley, W. J. Wagner, I. Blankenau, A. Netchaev, A. Soylemezoglu, "Satellite Image Template Matching with Covariance Estimation for Unmanned Ground Vehicle Localization: Active Terrain Localization Imaging System (ATLIS)," In *Proceedings of the Ground Vehicle Systems Engineering and Technology Symposium* (GVSETS), NDIA, Novi, MI, Aug. 16-18, 2022.

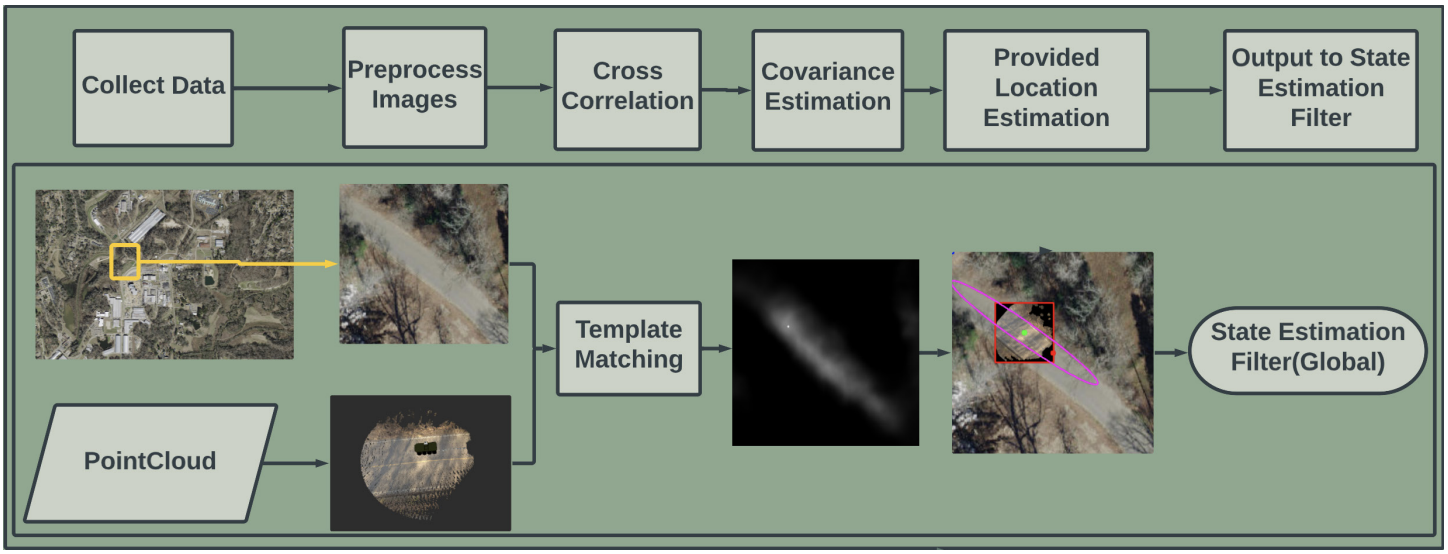

Figure 1: High level overview of the ATLIS workflow

1 INTRODUCTION

Advances in GNSS and geospatial localization technologies have resulted in their widespread use. However, the GNSS infrastructure is fragile and vulnerable to multiple failure modes [1]. The signal from the satellite must first pass through the earth's atmosphere, which attenuates the already low power signal, making it susceptible to offensive attacks like jamming and spoofing. Environmental factors that interrupt line of sight between a receiver and satellites can also impact accuracy and reliability.

In robotics, GNSS receivers are used to make the link between the UGV's internal relative motion frame and the global reference frame used by an operator. The UGV can maintain its local frame using techniques like dead reckoning and Simultaneous Localization And Mapping (SLAM). Without a GNSS signal however, a known collection of landmarks, or ranging beacon measurements, accumulated drift will prevent the UGV from successfully executing long-range missions that require autonomous navigation, reconnaissance, or multi-system collaboration [2].

In this paper, we propose a method that uses typical robotic sensors to accurately estimate its location in the global frame without the use of a GNSS receiver. Through real-world experimentation in complex outdoor environments, ATLIS has demonstrated an impressive average error of 1.21m over 5.1km when compared to a GNSS-based ground truth. This paper's key contributions are:

- A method to create dense and accurate top-down orthorectified images of a UGV's surroundings using lidar, cameras, and an inertial measurement unit (IMU).
- A method that estimates a UGV's global position and the uncertainty of this estimate by comparing the generated template image to a geo-referenced satellite image using template-matching
- Demonstration of this system integrated into a localization system achieving a 1.21m mean planar euclidean distance error over 5.1km.

2 BACKGROUND

Given the numerous current uses of GNSS, accurate geolocation can be a matter of personal safety, mission success, and national security. Several existing techniques rely on improving the signal quality and reliability of the GNSS system. In some cases, this requires extra antennas for beamforming at the receiver [3]. In others, new satellites are needed to transmit at higher power or over different frequency bands with added encryption.

If GNSS positioning is not reliable, techniques such as dead reckoning are used to maintain the state of the system between periods of degraded GNSS performance [3]. Kalman filters are used to help reduce errors in the systems state estimation [4, 5]. Often, local fixed frames are maintained internally to correct for drift using a family of techniques called Simultaneous Localization And Mapping (SLAM). SLAM can produce extremely accurate relative maps, but, without periodic GNSS corrections, the drift between the UGV's internal and global frames will still grow [6].

The research community has developed various approaches to overcome the GNSS-denied localization problem. Many of these methods assume *a priori* information in the form of a satellite image, point-cloud, digital elevation model (DEM), or star map. Star trackers use specialized sensors to estimate the system's location with respect to a star map using techniques similar to those used by ancient seafaring civilizations [3]. If the terrain contains many geometric features LiDAR matching to DEMs [7] or point clouds [8] can be used.

Cheung et al. describe an image-based method that uses template matching to compare the projection of an omni-camera sphere onto a flat plane with a satellite image to estimate a vehicle's location on the satellite map [9]. This method is promising as it only requires a satellite image, but it suffers from the distortion of the projected image due to deviation of the terrain from a plane and lack of on-board sensors for dead reckoning. In general, ground-to-aerial image matching is difficult because the perspective differences in how the images are obtained make feature-based matching challenging. To combat these distortions and better handle seasonal variation, other methods have been developed [10, 11, 12].

We propose an alternative approach that uses cross-correlation based image template matching, similar to [9], and bypasses these viewpoint issues by generating an orthorectified image of the ground through a combination of LiDAR point-clouds and camera imagery. The accuracy of the position estimates produced via template matching remains dependent on the environment, the quality of the *a priori* satellite image, and the quality of the vehicle-produced image. To take advantage of this method without introducing excess noise into our state estimation filter, it is essential to estimate the uncertainty of a given position estimate. This will enable context-dependent fusion of position estimates for improving ground vehicle localization accuracy.

Nickels et al. uses sum-of-squared-difference (SSD) template matching for a tracking application and lays out an approach to uncertainty estimation [13]. Their work builds on [14] who defines a response distribution Z based on the SSD metric

$$Z(i,j) = \exp(-aR_{\text{SSD}}(i,j)) \tag{1}$$

where a is a normalization factor. In [14], the value for a in Z is chosen so that the maximum value is 0.95. In [13], the value for a is instead chosen so that $\sum_{i,j} Z(i,j) = 1$. This has the effect of suppressing off-peak response compared to [14], i.e., values that are close to the peak response will not be scaled significantly, but values lower than the peak will quickly drop off. Both [13] and [14] interpret Z as a probability distribution of the true match location, where $Z(i,j) = P(L(i,j))$ is the probability of the true match location being $L(i,j)$ where $(x,y) = L(i,j)$ is a function that converts the result image

coordinates into a position in meters relative to the top left corner of the result image. To enable fusion of this estimate with other state estimates, Nickels et al. approximates $P(L(i,j)) \sim \mathcal{N}(\mu, \mathbf{\Sigma})$ using a 2D Gaussian distribution

$$(i^*, j^*) = \arg\max_{i,j} Z(i,j), \quad (\mu_x, \mu_y) = L(i^*, j^*)$$
$$\Sigma = f(\mu, Z) = \begin{bmatrix} \frac{\sum_{i,j} Z(i,j)(x-\mu_x)^2}{\sum_{i,j} Z(i,j)} & \frac{\sum_{i,j} Z(i,j)(x-\mu_x)(y-\mu_y)}{\sum_{i,j} Z(i,j)} \\ \frac{\sum_{i,j} Z(i,j)(x-\mu_x)(y-\mu_y)}{\sum_{i,j} Z(i,j)} & \frac{\sum_{i,j} Z(i,j)(y-\mu_y)^2}{\sum_{i,j} Z(i,j)} \end{bmatrix} \tag{2}$$

where $(x, y) = L(i, j)$ and $f(\mu, Z)$ is a standard weighted covariance function.

3 METHODOLOGY

ATLIS is a tool that produces a global position estimate by matching UGV's sensor observations with satellite imagery; see Figure 1 for an overview. It requires five inputs: a GeoTiff of the area, the current estimated UGV latitude and longitude, the current UGV heading, and a colorized point cloud capturing the current surroundings of the UGV. This software is designed to be used with a state estimation method such as an extended Kalman filter (EKF) that fuses the position estimate with other on-board sensors such as IMUs and wheel odometry.

3.1 Generate Image From Sensors

Fundamentally, ATLIS finds the geolocation of the UGV by matching a vehicle-centric orthorectified image to a georeferenced satellite image, i.e. a GeoTIFF [15] . Perspective differences between the UGV on the ground and the satellite in the sky pose the first major challenge to this approach. We address this problem using the ERDC-developed Sensor Fusion ROS node to project the camera's images onto the LiDAR point clouds producing a colorized point cloud in real time. This ensures depth accuracy, but the cloud is sparse due to the low resolution of the LiDARs. To create a denser cloud, the node also buffers the LiDAR point clouds as the UGV moves. This is accomplished by maintaining the UGV's location state through the transform tree in ROS and transforming each new cloud into a fixed frame temporarily before republishing in the sensor frame.

The template image is created by passing the point cloud to the ATLIS node with the heading estimate from the state estimation system. ATLIS uses the heading to rotate the point cloud to match the orientation of the satellite image. It then rasterizes the point cloud into a top-down 2D orthorectified image by mapping the x and y components of each point to an image bin based on the resolution of the satellite image. Note that it is important to keep track of the UGV's position with respect to this image to enable accurate position estimation.

3.2 Geolocation Estimation

ATLIS takes a template matching approach to align the locally obtained rasterized image $T \in \mathbb{R}^{W_T \times H_T}$ to the *a priori* satellite image $S \in \mathbb{R}^{W_S \times H_S}$. The satellite image for the current mission, in the form of a GeoTIFF, can be stored locally on the UGV before deployment. This step helps ensure the system can continue to operate if all network connections are severed.

To reduce the computational burden of matching against the entire satellite image, a sub-image $\bar{S} \in \mathbb{R}^{W_{\bar{S}} \times H_{\bar{S}}}$, centered on the current global position estimate is extracted from the satellite image. The initial global position estimate may be obtained from a human operator using a tablet interface with a satellite map overlay, or from a single GNSS estimate. An array of matching scores, $R \in \mathbb{R}^{(W_R \times H_R)}$ is produced in a moving window fashion, sliding T along $\bar{S}$ where $W_R = (W_{\bar{S}} - W_T)$ and $H_R = H_{\bar{S}} - H_T$, computing a similarity score between the two image patches at each window position.

OpenCV provides a template matching function matchTemplate() which exposes several methods for image comparison [16]. In our data, gaps in the LiDAR create black sections in the template image as shown in Figure 2. The chosen method must then be robust to errors in these dark pixels. For this reason Zero-Normalized Cross-Correlation Coefficient (ZNCC) was chosen. This method returns a fit score R_{ZNCC} ranging between positive and negative one. The positive side represents correlation with pixel brightness, while the negative side corresponds to pixel shadow [17, 18, 19, 20, 21]. We can then simply truncate the negative scores to negate the effect of the gaps in the template image [4]. Thus, the result image $R_{\text{ZNCC}}(u, v)$ is redefined as follows.

$$R(u, v) = \max\{R_{\text{ZNCC}}(u, v), 0\} \tag{3}$$

A position estimate can be found by searching for the maximum matching score in the result image $(i_R^*, j_R^*) = \arg\max_{i,j} R(i, j)$. It should be noted that the location found in the matched image must be offset by the UGV's position in the template image. ATLIS keeps track of the vehicle's known position within the source image $(i_{\bar{S}}^v, j_{\bar{S}}^v)$ and the known latitude and longitude $(x_{\bar{S}}, y_{\bar{S}})$ of the corner of the local source image $\bar{S}$ and combines this information with the match location (i_R^*, j_R^*) to produce a global position estimate $(\tilde{x}, \tilde{y})$ of the vehicle.

It is important to consider the age of the satellite imagery prior to using ATLIS. Often, the satellite data obtained for a given area is several months if not years old, and much of the environment may have changed in that time. It is preferable to use the most recently available imagery to enable optimal matching performance. Even with recent imagery, there is no guarantee that small details in the template image will be present on the satellite image.

A bilateral filter is applied to both the template and the satellite images to smooth some of these details and to reduce the image noise before cross-correlation. The bilateral filter was chosen over other standard image filters because it maintains definition over macro features [22]. Another optional step is to desaturate both images. This step can be helpful if there is a seasonal difference between when the satellite image was collected and the current season. These features help ensure the ATLIS estimates maintain robustness when given non-ideal priors.

Figure 2: The template image T overlaid on the satellite source patch image $\bar{S}$ at the match location. The red dot represents the UGV's location prior while the green dot shows the ATLIS proposed update. The pink ellipse displays the 95% confidence interval for the estimated covariance matrix.

3.3 Positional Uncertainty Estimation

The accuracy of the outlined template matching approach is dynamic, varying significantly with the location within the environment. Certain areas of the environment provide distinctive features that enable accurate matching and high scores for a small region of R, such as a road intersection. Other areas of

the environment are less informative and produce relatively uniform matching scores throughout the search region, e.g., a parking lot or grassy field. Often, the environment is informative along one direction but less informative in another, e.g., a long road, enabling accurate estimation of the vehicle position perpendicular to the road but an inaccurate estimate along the road, see Figure 2.

In addition to these environmental influences, if the environment has undergone significant changes since the satellite image was captured then there will be a mismatch between T and $\bar{S}$ resulting in poor matching. For example, this type of mismatch can occur due to new construction or seasonal differences. Errors in the creation of T can also result in matching problems. If multiple cameras are used to produce the colorized cloud, white balance differences can result in artificial color differences in the image.

To ensure that fusing ATLIS estimates into a localization system enables improved geospatial localization, we require a way of characterizing the uncertainty of the estimate. Directly using the ZNCC matching scores for the response distribution $Z = R$ and then computing the positional covariance using the weighted covariance calculation in Equation 2 results in a number of problems. First, by treating the raw ZNCC scores R as a probability distribution for the true position, too much weight is given to low value scores, which produces higher uncertainty position estimates than is desired. Second, the weighted covariance calculation is invariant to a linear transformation of Z. This means that higher matching scores will not lead to lower positional covariance estimates. Third, as the size of the source patch image $\bar{S}$ increases the estimated covariance also grows, even if no new peaks are observed in the expanded regions of R. Ideally, we do not want the size of R to significantly affect the covariance estimate if no new peaks are observed. We address these problems by:

1. Scaling R using a non-linear response distribution
2. Scaling the weighted covariance by a non-linear function of the highest match score
3. Normalizing the weighted covariance by the un-weighed covariance

The SSD metric is inversely related to the ZNCC metric, so the same response distribution function, Equation 1 will not work for our purposes; we want R to have a higher value for areas that indicate a more likely match and lower values for less likely matches. We modify Equation 1 to account for this difference

$$Z(i,j) = \frac{exp(aR(i,j)) - 1}{b} \tag{4}$$
$$\text{where } b = \frac{exp(aR(i_R^*, j_R^*)) - 1}{R(i_R^*, j_R^*))}$$

and where a is a constant chosen to attenuate the non-matched scores. In contrast to previous approaches [14, 13], we do not allow a to be chosen dynamically in an effort to enable more control over the covariance estimation. After producing Z, the covariance of the match, Σ, is computed using Equation 2. To address items 2 and 3, we then develop a function $g(\cdot)$ to scale Σ.

If the size of either the source patch image $\bar{S}$ or the template image T changes, the size of R will change as well. ATLIS computes the un-weighted covariance $\Sigma_u = f(\mu_c, W)$ where W is the same size as Z with all values equal to 1 and μ_c is the center of W. To enable ease of tuning and the ability for image sizes to change dynamically, we normalize the covariance Σ by the largest eigenvalue of Σ_u, $\lambda = \max(\text{eig}(\Sigma_{center}))$. We choose to compute the covariance matrix at the center of the result image and not at the max match location because as the max match location moves to the edges, the eigenvalues increase to account for the fact that we are observing less data near the max location. This is a useful feature because if our match is near the edge, we

want our system to be more uncertain than if it is at the center, i.e. our previous position estimate.

In addition, we would like higher scores, i.e., better matches, to produce smaller covariances. We therefore include a term in $g(\cdot)$ to accomplish scaling based on the score at the max match location. Combining the size and score scaling results in

$$\tilde{\Sigma} = g(\Sigma_u, R, \lambda | c, d) = \frac{c}{\lambda} \cdot \Sigma \cdot R(i_R^*, j_R^*)^{-d} \quad (5)$$

where $\tilde{\Sigma}$ is the scaled covariance, c is a parameter that linearly scales the covariance, which is very useful when tuning our EKF, and d is a parameter that controls how much to let the value of the maximum match affects the covariance. Note, if we get a perfect match, $R(i_R^*, j_R^*) = 1$, then scaling by the max match score has no effect.

3.4 State Estimation

We use the ROS Robot Localization Package implementation of an Extended Kalman Filter (EKF) for state estimation, which enables fusing of a variety of sensors[5]. Dead reckoning is achieved by fusing wheel odometry with orientation from an Inertial Measurement Unit (IMU) that provides a global north estimation using a magnetometer. The ATLIS-aided estimate is produced by fusing the same wheel odometry and IMU orientation with of position estimates from ATLIS. In certain cases, ATLIS provides a faulty match and the error in position estimation is not effectively accounted for by the covariance. We hypothesize that this may be because ATLIS assumes a Gaussian distribution where the template matching process can sometimes produce matching scores R that indicate a more complex distribution. For example, if the search area contains multiple similar features, e.g., parallel roads, a match may be made to the incorrect road. A Gaussian distribution may not capture this uncertainty accurately where a multi-modal distribution may be more appropriate.

To mitigate the effect of overconfident poor matches, the EKF discards ATLIS position estimates if they are too far away from the current filtered position estimate, as measured by the Mahalanobis distance [23]. The rejection threshold feature of Robot Localization is leveraged to handle this function [5]. As the uncertainty of the position estimate grows without observation of a global position, the filter begins to accept ATLIS position estimates that are further away from the current position estimate. This is the desired behavior as the longer the UGV travels without correction from ATLIS, the more drift will accumulate due to dead-reckoning. This scenario may occur when traveling down a long straight trail where very few features perpendicular to the road are present to match to. As the UGV encounters an intersection, ATLIS is able to produce an accurate match, and since the covariance of our position estimate has grown, the filter accepts the correction even though it is a significant distance away from our current position estimate. It is important to ensure the position estimate covariances produced by the EKF accurately reflect the true uncertainty so that the filter can correctly differentiate the accuracy of ATLIS corrections. However, the rejection threshold does not account for faulty matches that fall within the rejection threshold. Qualitatively though, matches close to the current position estimate tend to have appropriate covariances mitigating the effect of an incorrect match.

4 RESULTS

We evaluate the performance of ATLIS in aiding localization of a ground vehicle traversing a $5.1\mathrm{km}$ combat-trail like path with some off-road excursions. This experiment was performed in August of 2021 and the satellite image used by ATLIS was taken in 2016 with a $0.1\mathrm{m/px}$ resolution [24]. In addition, approximately $800\mathrm{m}$ of the path is obscured by thick vegetation on the satellite map, making it difficult for ATLIS to produce informative position estimates. We use a high-accuracy fiber optic gyro (FOG) dual antenna GNSS/inertial navigation

system (INS) to establish a ground truth and compare the dead-reckoning accuracy of our state estimation system with the ATLIS aided accuracy. The UGV used in this experiment is an ARGO J8, which shares the same chassis as the U.S. Army's Small Multipurpose Equipment Transport (S-MET) platform. In addition to the FOG GNSS/INS, the J8 is outfitted with three LiDARs, two 16 beam and one 64 beam, two forward facing cameras, an IMU, and wheel encoders.

Table 1: Comparison of localization accuracy

Method	Avg. Error (m)	Max. Error (m)
Dead-Reckoning	6.4	11.7
ATLIS-Aided	1.21	3.53

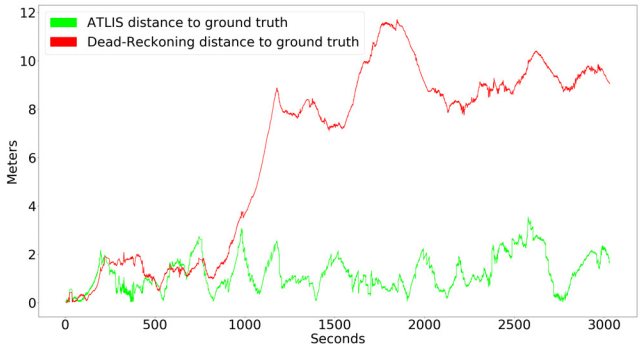

Figure 3: Line graph displaying planar euclidean distance error over time for dead-reckoning (red) and ATLIS-aided (green).

We track the planar euclidean distance absolute positional error over the course of the run for both dead-reckoning and ATLIS-aided localization, see Figure 3, and show large improvements in both the average and maximum planar euclidean distance error as shown in Table 1. The dead reckoning error is small at the beginning of the run, but over time this error compounds, causing the UGV's position to drift by nearly $12\mathrm{m}$ by the end of the run. The ATLIS-aided system does a much better job tracking the ground truth throughout the entirety of the run, deviating by more than $2.5\mathrm{m}$ for only a short period. This can be seen clearly at second 1000 on Figure 3 the two lines follow a similar shape and ATLIS is able to correct the drift that caused dead-reckoning to climb to $9\mathrm{m}$. The accuracy of dead reckoning vs ATLIS can also be seen in the satellite overlay in Figure 5. ATLIS is able to keep the vehicle position estimate on the road and near the ground truth, while dead-reckoning has the vehicle position off the road and in the ditch for a large portion of the time. These types of errors are problematic for waypoint based navigation as the planner will have difficulty reaching goals that appear to be within an obstacle due to this positional error.

Though ATLIS improves the average error of the run, there is a section between 700 seconds 800 seconds where ATLIS underperforms dead reckoning as seen in Figure 3; this is due to two major factors. The first of these factors is related to the location on the map where this performance degradation occurs. The location in question is the approximately $800\mathrm{m}$ stretch of road that is obfuscated by thick vegetation on the satellite map. Though most of the road is under tree cover, there are small sections of the road that are visible on the map. This situation is especially challenging for ATLIS because when a match is found, the unweighted covariance Σ_u will be small, expressing high confidence, because the system only matches well for that small section of uncovered road. If the max match score is not very high the $R(i_R^*, j_R^*)^{-d}$ component of Equation 5 will help to inflate this covariance. Despite this scaling, ATLIS is still occasionally overconfident. In this case, the EKF pose rejection can help to mitigate the effect of poor matches.

The other factor that contributes to the underperformance of ATLIS stems from a white balancing issue we observed during this run. As shown in Figure 4, the camera configuration on the UGV is using the two front cameras to project the color on our LiDAR point cloud. Because the two cameras were not properly white balanced with

each other, projection on the point cloud created a discoloration along the center of the cloud. This discoloration occasionally causes ATLIS to match to the side of the road instead of the center in scenarios where the road is less visible on the map. Since making this observation, the camera configuration has been changed so that the projection is only coming from a single camera directly in the center. This change helps eliminate this issue and will be discussed further in future work.

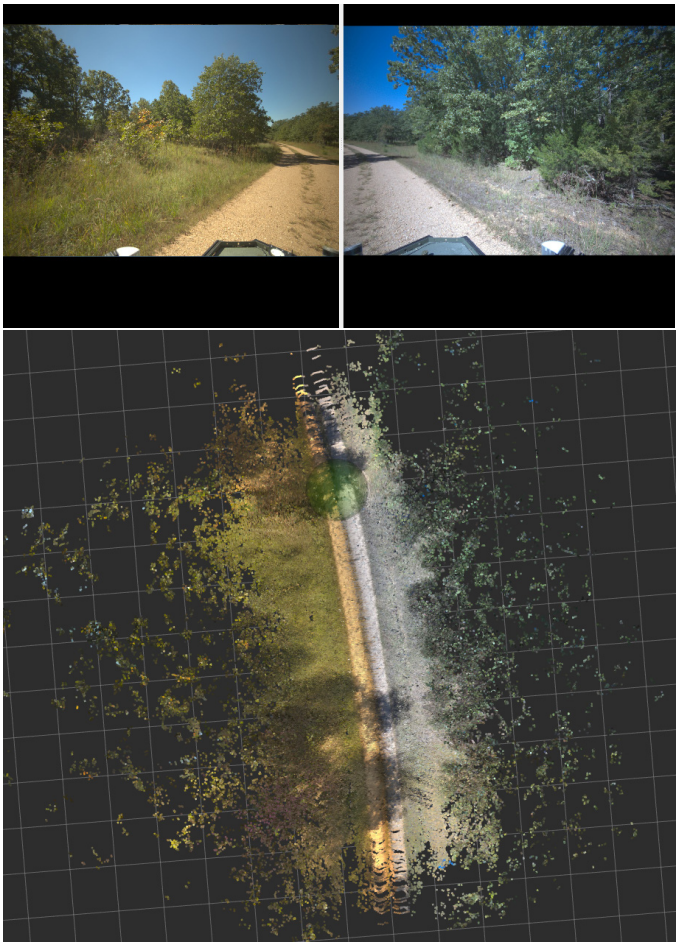

Figure 4: (Top) First person view of the camera data provided by the UGV to highlight white balancing issue. (Bottom) Top down view of the colorized point cloud provided by the UGV highlighting effect of white balancing on the cloud colorization.The green circle is the covariance around the UGV's position

5 CONCLUSION

ATLIS is a global localization system that enables a UGV to localize itself within an *a priori* satellite image. Onboard cameras are fused with LiDAR to generate a colorized point cloud of the ground. This cloud is rasterized into an orthorectified local image that enables accurate estimation of the vehicle position using template-matching. The covariance estimation method developed is shown to work well, correctly providing small covariances when there are salient features in the environment, large covariances when in more uniform and less informative areas, and elongated covariances in areas such as long straight roads, where confidence is higher along the width of the road and lower along the length. By combining these techniques, ATLIS is shown to perform similar to a GNSS based system in an experiment where our UGV traveled over 5.1km while maintaining an average error of 1.21m compared to ground truth.

From this experiment, we were able to identify a number of areas for improving the system. The camera-LiDAR fusion approach to handing perspective differences between the UGV and the satellite imagery is shown to be successful in most cases, but is limited for areas of the environment where the UGV is obscured from above by tree cover. We are interested in exploring methods to help increase the uncertainty in this scenario. Future plans also include efforts to minimize error due to hardware based limitations such as white balancing. ATLIS is shown to be effective for on-trail localization, but will likely have difficulty in open spaces due to the lack of distinctive visual features. However, if there are elevation changes, using LiDAR-DEM matching approaches may help to compensate. In the near-term, we plan to more rigorously evaluate ATLIS, comparing various matching approaches and image filtering methods such as canny edge detection across a variety of environments including desert and urban regions.

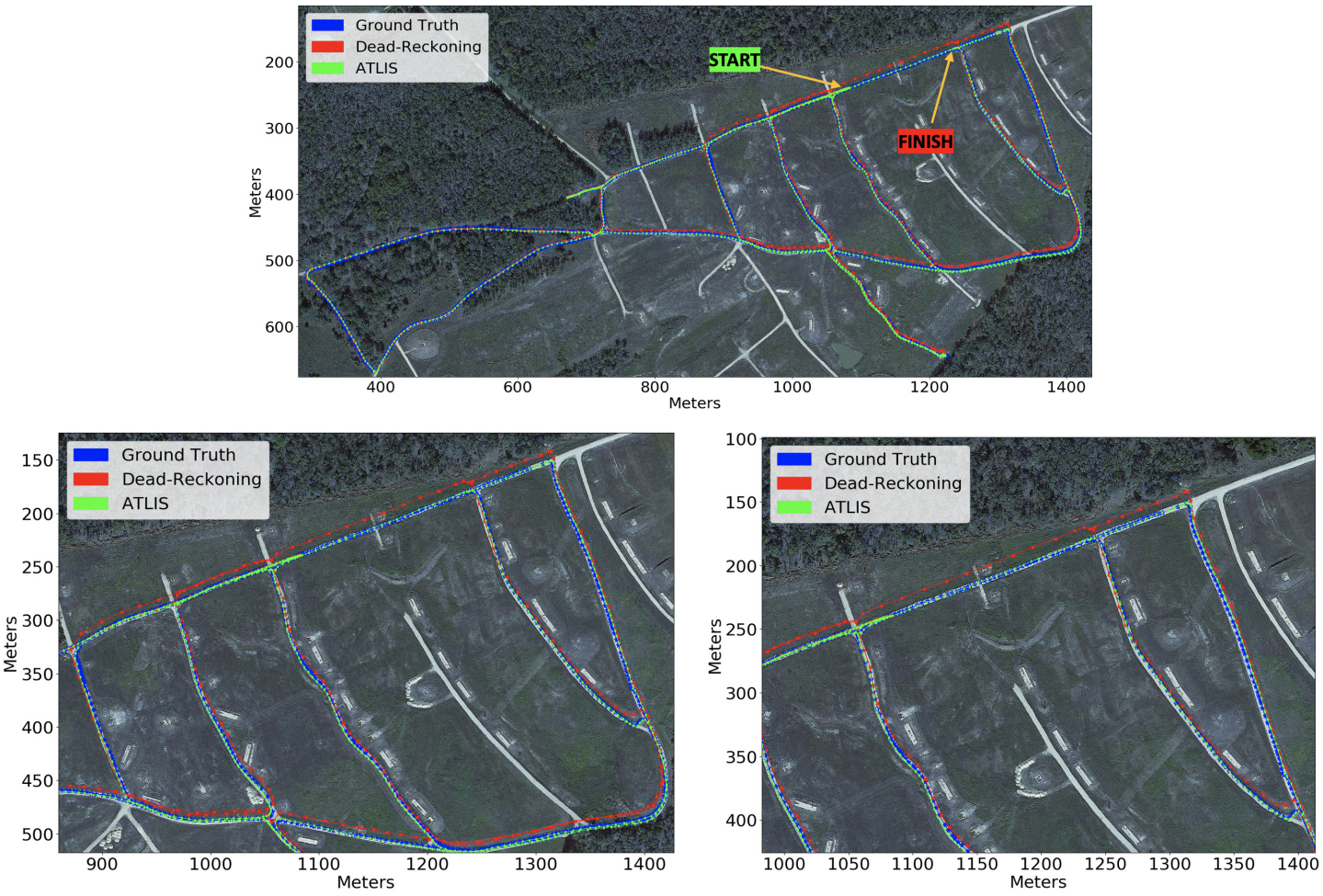

Figure 5: Satellite view of experiment area with an overlay of the path traveled by the UGV with ground truth in blue, dead reckoning in red, ATLIS-aided localization in green.

6 ACKNOWLEDGMENT

The authors would like to thank Chuck Ellison for help developing the Sensor Fusion node and Emily Leathers, Kendall Niles, and Kevin Murphy for their editorial assistance.

7 REFERENCES

[1] T. Nighswander, B. Ledvina, J. Diamond, R. Brumley, and D. Brumley, "GPS software attacks," *Proceedings of the ACM Conf. on Computer and Communications Security*, pp. 450–461, 2012.

[2] R. Siegwart, I. R. Nourbakhsh, and D. Scaramuzza, *Introduction to autonomous mobile robots*. MIT press, 2011.

[3] "GNSS denied solutions and upcoming technologies," *Advanced Navigation*, Feb. 2022. [Online]. Available: https://www.advancednavigation.com/knowledge-base/gnss-denied-solutions-and-upcoming-technologies/

[4] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*, 2005, vol. 1.

[5] T. Moore and D. Stouch, "A generalized extended Kalman filter implementation for the Robot Operating System," in *Proc. of the 13th Int. Conf. on Intelligent Autonomous Systems*. Springer, July 2014.

[6] G. He, X. Yuan, Y. Zhuang, and H. Hu, "An integrated GNSS/LiDAR-SLAM pose estimation framework for large-scale map building in partially GNSS-denied environments," *IEEE Transactions on Instrumentation and Measurement*, vol. 70, pp. 1–9, 2021.

[7] G. Hemann, S. Singh, and M. Kess, "Long-range GPS-denied aerial inertial navigation," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2016, pp. 1659–1666.

[8] P. J. Carle, P. T. Furgale, and T. D. Barfoot, "Long-range rover localization by matching LIDAR scans to orbital elevation maps," *Journal of Field Robotics*, vol. 27, no. 3, 2010.

[9] C. Cheung and S. Baek, "Cross correlating ground-level panoramas with satellite imagery for GPS-denied localization of autonomous ground vehicles," in *NDIA Ground Vehicle Systems Engineering and Technology Symposium*, 2018.

[10] S. Verde, T. Resek, S. Milani, and A. Rocha, "Ground-to-aerial viewpoint localization via landmark graphs matching," *IEEE Signal Processing Letters*, pp. 1490–1494.

[11] H. Goforth and S. Lucey, "GPS-denied UAV localization using pre-existing satellite imagery," in *IEEE Int. Conf. on Robotics and Automation*, 2019, pp. 2974–2980.

[12] S. Hu and G. H. Lee, "Image-based geo-localization using satellite imagery," pp. 1205–1219, 2020.

[13] K. Nickels and S. Hutchinson, "Estimating uncertainty in SSD-based feature tracking," *Image and Vision Computing*, vol. 20, no. 1, pp. 47–58, 2002.

[14] A. Singh and P. Allen, "Image-flow computation: an estimation-theoretic framework and a unified perspective," *CVGIP: Image Understanding*, vol. 56, no. 2, pp. 152–177, 1992.

[15] OGC 19-008r4, "GeoTiff Standard 1.1, Open Geospatial Consortium," Sept. 2019. [Online]. Available: https://docs.ogc.org/is/19-008r4/19-008r4.html

[16] G. Bradski, "The OpenCV library," *Dr. Dobb's Journal: Software Tools for the Professional Programmer*, vol. 25, no. 11, pp. 120–123, 2000.

[17] R. C. Gonzalez and Woods, *Digital image processing*. Pearson, 2008.

[18] B. Horn, B. Klaus, and P. Horn, *Robot vision*. MIT press, 1986.

[19] R. Szeliski, *Computer vision: algorithms and applications*. Springer Science & Business Media, 2010.

[20] D. A. Forsyth and J. Ponce, *Computer Vision - A Modern Approach, Second Edition.* Pitman, 2012.

[21] A. Kaehler and G. Bradski, *Learning OpenCV 3: Computer Vision in C++ with the OpenCV Library*, 1st ed. O'Reilly Media, Inc., 2016.

[22] C. Tomasi and R. Manduchi, "Bilateral filtering for gray and color images," in *Sixth Int. Conf. on Computer Vision*, 1998, pp. 839–846.

[23] S. T. Pfister, "Algorithms for mobile robot localization and mapping, incorporating detailed noise modeling and multi-scale feature extraction," Ph.D. dissertation, 2006.

[24] Google Earth 7.1, “Satellite imagery,” 2016. [Online]. Available: https://earth.google.com