Landmark Mapping from Unbiased Observations

Jason S. Ku, Stephen Ho, and Sanjay Sarma
Department of Mechanical Engineering
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139
Email: jasonku@mit.edu, ssh1@mit.edu, sesarma@mit.edu

***Abstract*—The starting point of any Smart City approach is knowing what is in the city and the location of city assets. We propose a general, automated approach to inventorying and monitoring outdoor city infrastructure using common sensors: namely a GPS, IMU, and camera. The presented mapping algorithm operates in the mobile sensing paradigm, using observations from a moving vehicle to construct a map of landmark location estimates whose uncertainty decreases linearly with the number of observations, robust to both translational and angular error to first order. The algorithm is adaptable to many applications given an appropriate image classifier. We apply our algorithm to automatically locate and inventory city streetlights and demonstrate its performance using both numerical simulation and field experiments.**

I. Introduction

Simultaneous localization and mapping, or SLAM refers to machines mapping their environment and positioning themselves accurately within that environment [1] [2] [3]. SLAM has been studied extensively in robotics and motion planning, typically implimented using range finders [4] and cameras combined with modeling of vehicle kinematics to inform an estimate. This paper ignores observer localization in favor of a limited but powerful model of landmark mapping with two main features: an environment containing a finite set of fixed landmarks; and an observer equipped with noisy estimates of its own position, and estimates of the direction in which landmarks are observed.

The motivation for the above formulation is twofold. First, we limit the environment to a sparse set of landmarks to simplify the typical computational complexity of trying to map an entire environment. Second, the modern ubiquity of smartphones provides a convenient platform to collect data consistent with this model: small computers equipped with a camera, IMU, and GPS. The IMU and GPS can give (possibly highly inaccurate) estimates of position and orientation, and with a sufficient image classifier for some class of landmarks, the camera may provide an estimate of the direction of landmarks in front of the camera. Using unbiased estimators like an IMU with magnetometer (compass) and GPS greatly decreases our reliance on sensitive odometry to estimate position that is prone to drifting over time.

Much of the existing work using a single camera without depth information for landmark mapping (called MonoSLAM) relies on continuous motion of a camera for feature mapping [5], or addresses the combinatorial problem of mapping and location given perfect observation measurements [6] [7]. Our estimation approach is robust to noise even for observations given in random order. Existing work in triangulation attempts to reconstruct the location of objects given only a few measurements [8] [9], where as our approach exploits multiple measurements to decrease the covariance matrix of our estimation as the number of observations increases.

While addressing this general formulation, we keep in mind the specific application of automatically locating and inventorying landmarks in cities. The starting point of any Smart City approach is knowing what is in a city, as well as the location of city assets. The observation system described in this paper could easily be mounted on existing municipal fleet vehicles and gather data as the vehicle moves about the city. The proposed estimation algorithm could be used to locate and inventory many types of landmarks, given an appropriate image classifier.

In this paper, we demonstrate the proposed framework by mapping the location of streetlights. This information is useful to cities to inventory and maintain their infrastructure. Streetlights comprise a significant part of most municipal budgets, and many municipalities still rely on self reporting to maintain them. If a city knows where its infrastructure is, the process of monitoring for maintenance and repair can be automated saving time and money. Section II develops the theoretical framework for the estimation algorithm. Section III discusses the numerical simulation used to verify the theory. Section IV describes experimental setup used to conduct field experiments, with Section V analyzing the collected data. Final thoughts are concluded in Section VI.

II. Theory

We would like to identify the location of certain landmarks in space. For now, we restrict our analysis to locating a single landmark. For practical applications, we will mostly be interested in landmarks in the plane or in three dimensions, but the theory developed extends naturally to higher dimensions.

A landmark exists at an unknown position p in $\mathbb{R}^d$. Unless otherwise stated, we assume vectors are column vectors so that $p \cdot p = p^T p$. Observations of the landmark are taken at different positions $x_i \in \mathbb{R}^d$ for $i \in \{1, \ldots, n\}$, observing the landmark in direction $u_i = (p-x)/\|p-x\|$ where $\|\cdot\|$ denotes the Euclidean norm. Observations are of the form $(\tilde{x}_i, \tilde{u}_i)$ where $\tilde{x}_i$ and $\tilde{u}$ are measurement estimates, functions only of observation position x_i and direction u_i respectively. We will use X to denote collectively the set of position observations, and similarly U for the set of direction observations.

A. Perfect Information

First assume perfect data, *i.e.* that $\tilde{x}_i = x_i$ and $\tilde{u}_i = u_i$. Then certainly any two observations (x_1, u_1) and (x_2, u_2) that

intersect ($u_1 \neq u_2$) will be sufficient to reconstruct p perfectly using methods of elementary geometry. Note that the line through p and x_i (which is parallel to u_i) contains a unique point q_i closest to the origin, with

$$q_i = p - u_i(u_i \cdot p) = x_i - u_i(u_i \cdot x_i) \tag{1}$$

for any perfect observation (x_i, u_i). Motivated by the symmetry of Equation 1, we define the *projector* of a unit vector $u \in \mathbb{R}^d$ as $\mathcal{P}(u) = \mathbb{I}_d - uu^T$, where $\mathbb{I}_d$ denotes the d dimensional identity matrix. Note that each $\mathcal{P}(u)$ is always Hermitian and singular, with positive elements along the diagonal and the property that $\mathcal{P}(u)\mathcal{P}(u) = \mathcal{P}(u)$. In this notation, Equation 1 can be written as $q_i = \mathcal{P}(u_i)p = \mathcal{P}(u_i)x_i$. Summing over n measurements gives:

$$\tilde{p} = \left(\sum_{i=1}^{n} \mathcal{P}(u_i)\right)^{-1} \sum_{i=1}^{n} \mathcal{P}(u_i)x_i = \mathbf{P}(U) \sum_{i=1}^{n} \mathcal{P}(u_i)x_i, \tag{2}$$

where $\mathbf{P}(U)$ denotes the inverse of the sum of observation projectors. Of course, $\mathbf{P}(U)$ may not exist if the sum of observation projectors is singular, but it will exist under the following condition.

Claim 1: The sum of observation projectors is singular if and only if all observation directions are parallel.

Proof: Certainly if all $u_i \in U$ are parallel, every column of every projector $\mathcal{P}(u_i)$ will be linearly dependent, so their sum will also be singular. It remains to show the reverse implication. Assume for contradiction there exist two nonparallel observations u_1 and u_2, and some finite $p' \in \mathbb{R}^d$ such that $\sum_{i=1}^{n} \mathcal{P}(u_i)p' = 0$. First observe that by construction, $\mathcal{P}(u_i)p'$ is the closest point on the line from p' to $\mathcal{P}(u_i)p'$, so the angle at $\mathcal{P}(u_i)p'$ between the origin and p' is $90°$. Then by Thales' theorem [10], each $\mathcal{P}(u_i)p'$ is also on the d-sphere with diameter from the origin to p', and thus exists in one halfspace $\mathcal{H}$ through the origin, with normal at the origin pointed at and containing p'. Since u_1 and u_2 are not parallel, $(\mathcal{P}(u_1)+\mathcal{P}(u_2))p'$ is also strictly in $\mathcal{H}$ so summing additional $\mathcal{P}(u_i)p' \in \mathcal{H}$ can never yield zero, a contradiction. ■

This may seem redundant since any pair of measurements is sufficient to accurately reconstruct p given perfect measurements. However, we can exploit the redundancy of Equation 2 to minimize the effect of noisy data.

B. Noisy Positions

Now consider observations that contain positional error, so that while direction observations are perfect with $\tilde{u}_i = u_i$, position observations $\tilde{x}_i$ are random variables. Assume that each $\tilde{x}_i$ is unbiased so that the expected value of the observed position equals the true position, with $\mathbb{E}(\tilde{x}_i) = x_i$. Let $\delta_i = \tilde{x}_i - x_i$ be the zero-centered random error of the observed position. Taking such a set of measurements and applying Equation 2 gives an estimate $\tilde{p}_x$ for p:

$$\tilde{p}_x = p + \mathbf{P}(U) \sum_{i=1}^{n} \mathcal{P}(u_i)\delta_i. \tag{3}$$

Since $\mathbb{E}(\delta_j) = 0$, by the linearity of expectation $\mathbb{E}(\tilde{p}_x) = p$, and this approach recovers p in expectation. If each δ_i is independent from each other, the covariance matrix of $\tilde{p}_x$ can be shown to be:

$$\Sigma(\tilde{p}_x) = \mathbf{P}(U) \left(\sum_{i=1}^{n} \mathcal{P}(u_i)\Sigma(\delta_i)\mathcal{P}(u_i)\right) \mathbf{P}(U). \tag{4}$$

If many observation directions are chosen randomly, the diagonal elements of $\mathbf{P}^{-1}(U)$ will increase with expected value n/d, while the off-diagonal elements will have zero value in expectation. Thus, for randomly chosen observation directions, $\mathbf{P}(U)$ approaches the identity matrix times d/n in the limit of large samples. Equation 9 then establishes that the effect of any single measurement's covariance on the estimate covariance decreases linearly with the number of observations. Particularly, if each position observation is an identically distributed with covariance $\sigma^2\mathbb{I}_d$, in the limit of a large number of random observations, $\Sigma(\tilde{p}_x) = O(\sigma^2/n)\mathbb{I}_d$.

C. Noisy Directions

Now consider observations that contain directional error only, so that while position observations are perfect with $\tilde{x}_i = x_i$, direction observations $\tilde{u}_i$ are random variables. Assume that each $\tilde{u}_i$ is unbiased so that the expected value of the observed direction equals the true direction, with $\mathbb{E}(\tilde{u}_i) = u_i$. We further assume the small angle approximation so that the variation of direction is relatively small with $\tilde{u}_i \approx u_i + \gamma_i\hat{u}_i$ where $\gamma_i = \cos^{-1}(\tilde{u}_i \cdot u_i)$ is the zero-centered angular random error of the observed direction and the unit vector $\hat{u}_i$ is in the null space of u_i in the direction of $\tilde{u}_i - \mathcal{P}(u_i)\tilde{u}_i$. This approximation holds to within one percent for $\gamma_i < 10°$. For such a set of measurements, a projector can be approximated to first order as:

$$\mathcal{P}\left(u_i + \gamma_i\hat{u}_i\right) \approx \mathcal{P}(u_i) - \mathcal{A}(u_i, \hat{u}_i)\gamma_i, \tag{5}$$

where we have used $\mathcal{A}(u_i, \hat{u}_i)$ to represent the cross-term matrix coefficient $u_i\hat{u}_i^T + \hat{u}_iu_i^T$. Applying Equation 2, the individual variations in $\mathbf{P}(U)$ are negligible in expectation, which to first order gives the following estimate $\tilde{p}_u$ for p:

$$\tilde{p}_u = p - \mathbf{P}(U) \sum_{i=1}^{n} \mathcal{A}(u_i, \hat{u}_i)x_i\gamma_i. \tag{6}$$

Since $\mathbb{E}(\gamma_i) = 0$, by the linearity of expectation $\mathbb{E}(\tilde{p}_u) = p$, again recovering p in expectation. Also if each γ_i and $\hat{u}_i$ is independent from each other and $\hat{u}_i$ is uniformly distributed ($\tilde{u}_i$ is symmetrically distributed about u_i), then $\mathcal{A}(u_i, \hat{u}_i)$ goes to the identity matrix in expectation and the covariance matrix of $\tilde{p}_u$ will be:

$$\Sigma(\tilde{p}_u) = \mathbf{P}(U) \left(\sum_{i=1}^{n} x_i^T\Sigma(\gamma_i)x_i\right) \mathbf{P}(U). \tag{7}$$

Again we find that if many observation directions are chosen randomly, the estimated covariance decreases linearly with the number of observations.

D. Noisy Position and Direction

Now consider observations that contain error in both position and direction using the same models for error as in Sections II-B and II-C. Then to first order keeping only terms

with zero or one of δ_i or γ_i, an estimate $\tilde{p}$ for the position of p can be obtained:

$$\tilde{p} = p + \mathbf{P}(U) \sum_{i=1}^{n} \mathcal{P}(u_i)\delta_i - \mathcal{A}(u_i, \hat{u}_i)x_i\gamma_i. \tag{8}$$

Again, this estimate recovers p in expectation. Assuming that δ_i and γ_i are independent, the covariance matrix evaluates to:

$$\Sigma(\tilde{p}) = \mathbf{P}(U) \left(\sum_{i=1}^{n} \mathcal{P}(u_i)\Sigma(\delta_i)\mathcal{P}(u_i) + x_i^T \Sigma(\gamma_i) x_i \right) \mathbf{P}(U), \tag{9}$$

with the same linear decrease in variance with the number of measurements, as expected.

E. Multiple Landmarks

The previous discussions focused on localizing a single landmark. In this section, we discuss localizing k landmarks in $\mathbb{R}^d$. Multiple landmarks can be substantially more difficult to localize given that it will be unknown a priori which observation corresponds to which landmark. If we knew which observations were associated with each landmark, then the estimation problem would simply reduce to the single landmark case. So our goal in this section will be to classify each observation according the landmark associated with it. If the landmarks and the observer exist in three dimensions and are not restricted to a common plane, we can exploit the dimensional gap between the 3D space and the 1D line observations to cluster measurements.

Observe that two random lines in a common plane will intersect with high probability, while two random lines in three dimensions will intersect with very low probability. Thus, in three dimensions using an online estimation algorithm, each time we make an observation, we can check its distance to either previous observation lines or current landmark estimates and associate it with the closest observation or landmark. The first method yields an expected $O(n^2)$ estimation algorithm while the second yields an $O(nk)$ algorithm. In two dimensions, both upper bounds will be slower as observation lines will cross every other observation line with high expectation, and all such intersections will be landmark candidates. In our implementations, we exploit the assumption that the observer's sight is bounded *i.e.* observed landmarks are relatively close to the observer leading to better performance, only requiring a search of local intersections. This heuristic seems to work well when landmarks are sufficiently separated in space.

III. Numerical Simulation

A computer simulation was developed to demonstrate the localization algorithm described in Section II and to observe the effect of observation error. The simulation was written using Processing[1] and is available as a Mac or Windows64 application by contacting the first author. A one minute video demonstration of the program can be found on Youtube[2]. See Figure 1 for screenshots of the program simulation. All errors were programmed to be symmetric and normally distributed. A range of observation variances were evaluated

[1]http://www.processing.org

[2]http://youtu.be/rUwpGqJMuuI

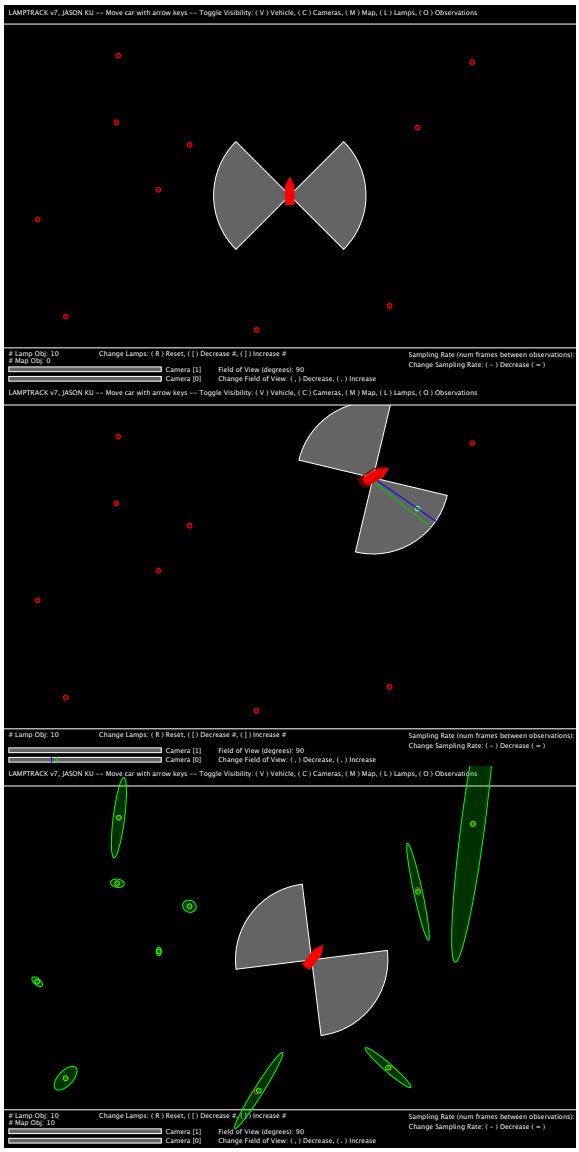

Fig. 1. Screenshots of the simulation program in action. A vehicle moves in a planar environment and takes observations of its surroundings. Landmarks are estimated using the proposed localization algorithm. [Top] Initial setup with many landmarks, observer is red pentagon with cameras observing gray areas on either side with limited field of view. [Center] The observer sees a landmark. A visualization of a noisy position and direction observation is overlaid on the actual position. [Bottom] After moving in the environment taking observations, a map of estimated landmark positions is created. The variance of each estimate is visualized as a green ellipse.

which confirmed the estimator to be unbiased, with estimated variance decreasing linearly with the number of measurements.

The simulation environment itself has many features. Arrow keys navigate a vehicle around an planar environment containing landmarks. When the vehicle observes a landmark, the noisy observation is used to update an internal map. The map displays each landmark estimate as an ellipse representing the covariance of the estimate. The visibility of different components of the simulation may be independently toggled on or off. Further, parameters of the simulation such as observation sampling rate, camera field of view, and the number and location of landmarks are all adjustable.

Fig. 2. Test route in Torrejón, Spain plotted on Google Earth. The route is shown in yellow.

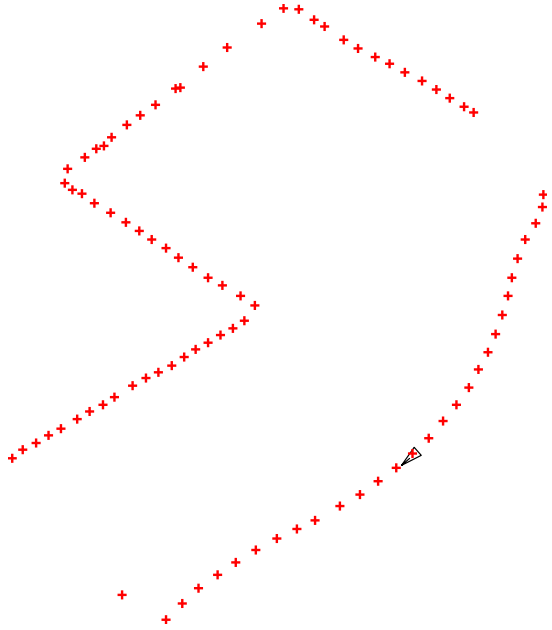

Fig. 3. Ground truth locations of streetlights using averaged GPS data. One position of the observer vehicle is plotted as a black triangle as shown. The vehicle traversed the route clockwise and used a camera to observe streetlights to the right of the vehicle. Road sections without marks did not have streetlights to the right of the vehicle.

IV. Data Collection

We implemented the proposed algorithm to automatically estimate the location of streetlights in an urban environment. Our algorithm requires observations consisting of two measurements: an estimate of the observer's position and an estimate of the direction of a landmark relative to the observer. To localize streetlights, we used a GPS to obtain the former, and an IMU and camera to obtain the latter. The IMU gives an estimate of the orientation of the observer, while the camera is used to detect a landmark (a streetlight) within its field of view. Note that when using a mobile sensing framework, GPS velocity estimates can be used as a replacement for observer orientation estimates in the absence of IMU data. Once measurements were collected, the localization algorithm described in Section II was used to provide estimates of streetlight positions.

A. Sensors

A u-blox MAX-M8 GPS receiver[3] was used to estimate vehicle position. The MAX-M8 module has a quoted positioning accuracy, Circular Error Probability c =2.0m, referring to the radius of a circle centered on the true value that contains 50% of actual GPS measurements given adequate satellite line of sight. We assume the GPS is an unbiased estimator of position. The GPS was sampled at its peak sampling rate of 15 Hz.

A UM7-LT IMU orientation sensor[4] was used to estimate vehicle heading. The IMU uses data from an on board magnetometer (compass), accelerometer, and gyroscope as input to an internal Kalman filter to compute a heading angle measurement accurate to within 0.5 degrees assuming a level IMU. We assume the IMU is an unbiased estimator of heading angle. The GPS was sampled at its peak sampling rate of 30 Hz.

A HackHD video camera[5] was used to estimate the direction of landmarks relative to the vehicle. The camera was mounted in a fixed known position and inclination relative to the vehicle. The camera was capable of capturing at 30 frames per second, each frame with 1920×1080 pixel resolution. We used a lens with 4.0mm focal length, F2.0 aperture, so that in the horizontal and vertical directions, each frame had an 80° and 64° field of view respectively.

All three sensors were mounted on top of a car with known relative geometry and driven around at night, with the camera pointed orthogonal to the direction of travel at known inclination angle. The test route is shown in Figure 2; the route was traversed clockwise, and only streetlights on the right side of the street (the inside of the loop) were observed. In the following subsection, we discuss retrieving direction observations $\tilde{u}_i$ from IMU and camera information.

In order to evaluate the accuracy of the proposed localization algorithm, independent measurements of each observed streetlight's position were required. To take these ground truth measurements, we manually recorded multiple GPS measurements under each streetlight using the same GPS discussed above, and averaged the readings. A plot of the calculated ground truth streetlight locations is shown in Figure 3 (compare to Figure 2). Figure 4 shows a view of these measurements overlaid on a Google Earth image.

[3] http://www.csgshop.com/product.php?id_product=171

[4] http://www.chrobotics.com/shop/um7-lt-orientation-sensor

[5] http://www.hackhd.com/tech.php

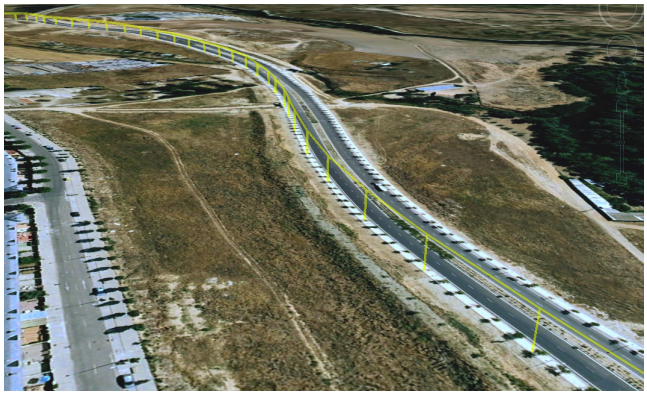

Fig. 4. View overlaid onto Google Earth showing the location of individual streetlights on the southeast road as measured by sampling GPS manually at each lamp.

Fig. 5. A camera frame showing a detected lamp centroid at the red star. This image processor is just one of many that can be used to identify city assets.

B. Direction Observations from IMU and Camera

The theory developed in Section II relied on direction observations from the observer to a landmark. Here we construct such observations from an IMU that gives an estimate of observer orientation and a camera which can detect landmarks. Consider a camera at x_i moving in space and a landmark located at position p. We assume a pinhole camera model with focal length f such that landmarks in real space scale to image space proportionally with the ratio of the focal length to the normal distance of the landmark to the camera plane [11]. Let the unit orthonormal basis $[\hat{x}, \hat{y}, \hat{z}]$ represent the orientation of the camera, with $\hat{x}$ pointing to the left of the image, $\hat{y}$ pointing to the top of the image, and $\hat{z}$ normal to the camera plane. Then the projection of the landmark on the camera image is:

$$p_i' = f\frac{(p - x_i)\cdot\hat{x}}{(p - x_i)\cdot\hat{z}}\hat{x} + f\frac{(p - x_i)\cdot\hat{y}}{(p - x_i)\cdot\hat{z}}\hat{y}. \tag{10}$$

If the IMU and camera are fixed to each other rigidly, then the IMU measurements can provide estimates of $[\hat{x}, \hat{y}, \hat{z}]$. Then, a direction observation can be computed directly as:

$$u_i = \frac{f\hat{z} + p_i'}{\|f\hat{z} + p_i'\|}. \tag{11}$$

Thus given the location of a landmark in a camera frame, a direction observation can be computed. This framework is quite general and can be used with any number of existing image processors to accurately locate many different types of objects. Detecting streetlights in camera images taken at night is a fairly simple image processing task, which we discuss in the following section.

In general, tracking 3D orientations can be very difficult to work with and compute [5]. In our experimental setup, our sensors were attached to a car so the camera position was constrained to lie in a plane parallel to the road, with known fixed roll and pitch but unconstrained yaw (heading). This setup greatly simplified implementation, only requiring a single angular measurement from the IMU.

C. Lamp Detection

Detecting the location of a streetlight bulb in an image taken at night can be accomplished using well known existing techniques. If the view from the camera to the bulb is unobstructed, the light from the bulb will be seen as a bright spot in the camera frame. The exposure of our camera was adjusted to increase the contrast of the bulb to the surrounding darkness. Standard thresholding, component labeling, and segmentation techniques can then be used to find the centroids of bright objects in an image [12] [13]. The standard MATLAB image processing toolbox was used in our implementation. Lamp detection using this method was particularly robust even for lamps obstructed by trees or signs. Additionally, to lower the possibility of false positives, our camera was tilted upward so that bright objects far from the street would not be observed, again exploiting the flexibility of taking 1D measurements in 3D. A representative camera frame with detected centroid is shown in Figure 5.

V. Experimental Results

The test route was traversed by a vehicle equipped with a GPS, IMU, and camera. The three sensors were synchronized and logged using an Arduino Mega 2560 R3[6]. The data was then post-processed into position and direction observations, then finally position estimates as discussed in the previous sections.

Figure 6 shows a sample of estimated locations of streetlights obtained using the proposed algorithm compared with ground truth data. The direction of travel of the vehicle is from right to left. The top image shows the vehicle path in black, ground truth positions in red, and observations lines in blue. Only a subset of observations are shown for clarity. The bottom image shows the same route with the estimated streetlight locations with the variance of each estimate depicted as an ellipse. As expected, the variance of each estimate is smaller parallel and larger orthogonal to the direction of travel because observations are only taken to one side of each streetlight from the street. We also notice that estimated position tends to lag behind the ground truth position by about a meter. This offset may be explained by the speed of the vehicle relative to the update response of the GPS. This lag is a function of vehicle speed and the specifications of the GPS used, and may be calibrated experimentally for a particular implimentation.

[6] http://www.arduino.cc/en/Main/ArduinoBoardMega2560

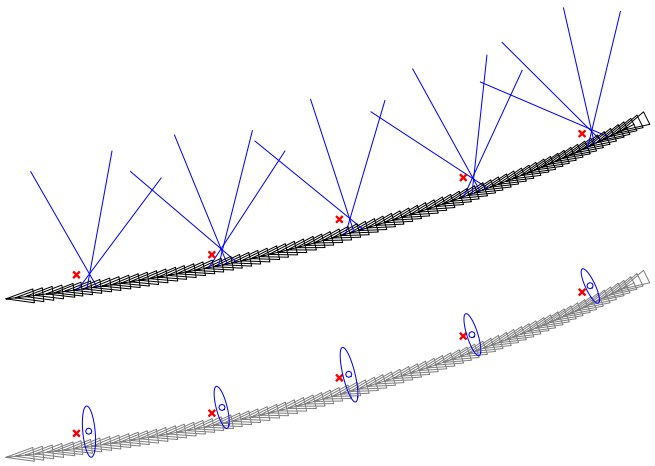

Fig. 6. [Top] A sample of observations taken during a trial experiment. Vehicle observation positions are shown by black triangles, and observation directions are shown by blue lines. Ground truth positions are shown in red. [Bottom] Same route showing estimated positions based on the proposed algorithm, with each covariance matrix represented by a blue ellipse.

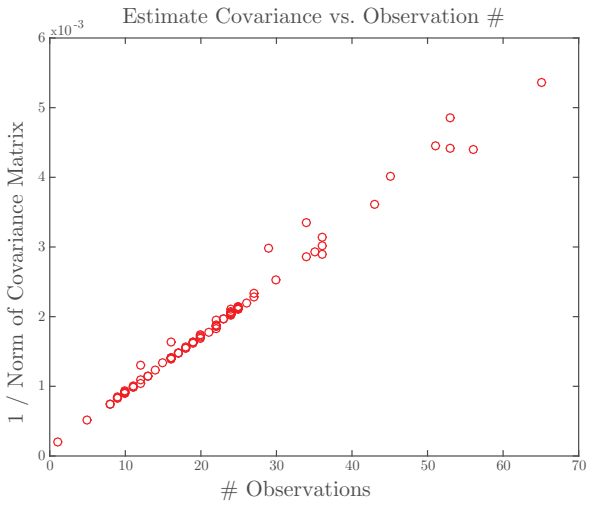

Fig. 7. A plot demonstrating the linear relationship between the number of observations of a landmark and the norm of the covariance matrix of the position estimate of the landmark, obtained by the proposed algorithm. The correlation coefficient is 0.9942.

Figure 7 shows a plot of the inverse norm of the covariance matrix $\|\Sigma(\tilde{p})\|^{-1}$ from each estimated landmark position as it varies with the number of observations of each streetlight. As predicted in Section II, this relationship is clearly linear with a correlation coefficient of 0.9942.

VI. Conclusion

The landmark mapping algorithm presented in this paper is very general and can be used to inventory and monitor a variety of outdoor city assets in conjunction with appropriate image classifiers, both existing and under active research. Once the locations of city assets are known precisely, the same cameras used for mapping can be used to automatically build realtime image libraries of assets already inventoried, informing decision making and allowing for automatic maintenance prioritization in the future. Because of the magnitude of urban scale, mobile sensing will be at the center of any truly Smart city, with municipal vehicles acting as "white blood cells", carrying sensors to monitor the health and well-being of the city. This mapping algorithm allows sensors to see the city in 3D using nothing more than the ubiquitous technology already in your pocket. Additionally, because the algorithm can accept observations in any order, a database of images taken by multiple observers tagged with position and orientation, each observing the same object might be used to build 3D reconstructions, enabling localization from crowd sourced imagery. A smart phone application implementing this algorithm is currently being developed.

Acknowledgment

This work was supported in part by Ferrovial, a multinational company supporting municipal infrastructure headquartered in Madrid, Spain. The authors would like to thank Dr. Sumeet Kumar, Prof. Erik Demaine, Craig Gordon, and Erick Fuentes for helpful discussions.

References

[1] H. Durrant-Whyte and T. Bailey, "Simultaneous Localisation and Mapping (SLAM): Part I The Essential Algorithms," *Robotics & Automation Magazine, IEEE*, vol. 13, no. 2, pp. 99–110, June 2006.

[2] M. R. Blas and S. Riisgaard, "SLAM for Dummies," *MIT Open Courseware*, 2005.

[3] M. Dissanayake, P. Newman, H. Durrant-Whyte, S. Clark, and M. Csorba, "An experimental and theoretical investigation into simultaneous localisation and map building," in *Experimental Robotics VI*, ser. Lecture Notes in Control and Information Sciences. Springer London, 2000, vol. 250, pp. 265–274. [Online]. Available: http://dx.doi.org/10.1007/BFb0119405

[4] N. Karlsson, E. Di Bernardo, J. Ostrowski, L. Goncalves, P. Pirjanian, and M. Munich, "The vSLAM Algorithm for Robust Localization and Mapping," in *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, April 2005, pp. 24–29.

[5] N. M. Andrew J. Davison, Ian Reid and O. Stasse, "MonoSLAM: Real-Time Single Camera SLAM," *IEEE Trans. PAMI*, 2007.

[6] D. Avis and H. Imai, "Locating a robot with angle measurements," *Journal of Symbolic Computation*, vol. 10, no. 2, pp. 311–326, 1990.

[7] E. D. Demaine, A. López-Ortiz, and J. I. Munro, "Robot localization without depth perception," in *Proceedings of the 8th Scandinavian Workshop on Algorithm Theory (SWAT 2002)*, ser. Lecture Notes in Computer Science, vol. 2368, Turku, Finland, July 3–5 2002, pp. 249–259.

[8] R. I. Hartley and P. Sturm, "Triangulation," *Computer vision and image understanding*, vol. 68, no. 2, pp. 146–157, 1997.

[9] S. Recker, M. Hess-Flores, and K. I. Joy, "Statistical angular error-based triangulation for efficient and accurate multi-view scene reconstruction," in *Applications of Computer Vision (WACV), 2013 IEEE Workshop on*. IEEE, 2013, pp. 68–75.

[10] T. Heath, *A history of Greek mathematics. Volume II. , From Aristarchus to Diophantus*. New York: Dover, 1981. [Online]. Available: http://opac.inria.fr/record=b1089407

[11] E. Hecht, *Optics*. Addison-Wesley, 2002. [Online]. Available: http://books.google.com/books?id=7aG6QgAACAAJ

[12] G. Stockman and L. G. Shapiro, *Computer Vision*, 1st ed. Upper Saddle River, NJ, USA: Prentice Hall PTR, 2001.

[13] M. B. Dillencourt, H. Samet, and M. Tamminen, "A general approach to connected-component labeling for arbitrary image representations," *J. ACM*, vol. 39, no. 2, pp. 253–280, Apr. 1992. [Online]. Available: http://doi.acm.org/10.1145/128749.128750