2008 IEEE International Conference on
Robotics and Automation
Pasadena, CA, USA, May 19-23, 2008

Localization in Urban Environments by Matching Ground Level Video Images with an Aerial Image

Keith Yu Kit Leung
PhD Candidate
University of Toronto
Institute of Aerospace Studies
Toronto, Ontario, Canada, M3H 5T6
keith.leung@robotics.utias.utoronto.ca

Christopher M. Clark
Assistant Professor
Department of Computer Science
California Polytechnic State University
San Luis Obispo, CA 93407
cmclark@calpoly.edu

Jan P. Huissoon
Professor
Department of Mechanical
and Mechatronics Engineering
University of Waterloo
Waterloo, Ontario, Canada, N2L 3G1
jph@uwaterloo.ca

Abstract— **This paper presents the design of a monocular vision based particle filter localization system for urban settings that uses aerial orthoimagery as the reference map. One of the design objectives is to provide a low cost method for outdoor localization using a single camera. This relaxes the need for global positioning system (GPS) which may experience degraded reliability in urban settings. The second objective is to study the achievable localization performance with the aforementioned resources. Image processing techniques are employed to create a feature map from an aerial image, and also to extract features from camera images to provide observations that are used by a particle filter for localization.**

I. INTRODUCTION

The ability to localize is essential for an autonomous mobile robot to successfully navigate in its workspace. This paper proposes the design of an urban outdoor localization system that uses a high resolution aerial image to create a feature map of an operation workspace, and uses a single monocular camera as an exteroceptive sensor. An assumption made in the design is that most observable features of buildings in the workspace are either orthogonal or parallel with the ground plane, which is the case in many urban settings. To use the aerial image, processing is required to transform it into a representation that is usable by a robot. Being able to achieve this will increase the degree of autonomy for a robot system. In urban environments, beacon based sensing and localization such as with the use of global positioning system (GPS) may become impractical or has degraded performance due to buildings in the operation area that interfere with beacon signals [1]. The end objective of the design is to achieve autonomous localization in an outdoor urban environment defined by an aerial image without knowledge of the initial position (x, y) and heading (ϑ).

Other researchers have also used monocular vision in localization because of the simplicity of hardware involvement. In recent publications, researchers have tried using a database of images to serve as the map in localization. Zhang et al. [2] captured images at various points in an operating workspace and tagged them with GPS position readings to create an image database. Matching of features between the database images and an on board camera images for localization was carried out by performing scale invariant feature transform (SIFT). In the work by Johns et al. [3], the appearances of city skylines from various locations were used collectively as the map. Similar work have also been done by looking at the details of building facades. In the most related and recent work presented in [1], a robot is first guided through a course as it records a video of the surrounding. This information is used offline where distinct image features are selected to generate a three dimensional map. The robot is then shown to localize itself using on board camera images while navigating a trajectory close to the path which the robot first took in generating the map. The methods highlighted share the similarity of requiring a map to be first created by capturing images at known locations. This map is subsequently compared to on board camera images when localization is performed. The approach taken by the proposed localization method is different in that it is not necessary to obtain on board images of the environment before performing localization. Instead, this information will come from an aerial image. Aerial orthoimages are geographic information system (GIS) resources that are becoming more readily available, and are usually obtainable through government sources or private agencies.

Image processing techniques are applied to an aerial image to highlight building boundaries (walls). The details of this process is presented in section II. Building boundaries are considered good features to detect because they can be seen from both an aerial image and from an on board camera equipped by a robot on the ground. Thus there is a similar type of object that can be compared for localization. The identification of relevant wall features from camera images is presented in section III, which involves the use of vanishing point analysis in order to infer 3d information from 2d images. The orientation of building boundaries are compared to determine the importance factor of particles in the particle filter. This particle filter is used because of its ability to perform state estimation with unknown initial pose. In the testing of the particle filter, a camera is moved manually while information required to generate

978-1-4244-1647-9/08/$25.00 ©2008 IEEE.

state transition (odometric data) is recorded. The system is tested offline using saved camera footage, and the results are presented in section IV.

II. Feature Map Generation

Aerial images are resources that are becoming readily available. The goal of extracting boundary features from aerial imagery is similar to the practice of building detection. Automatic detection of buildings from aerial images is of great interest in many geographic information system (GIS) related fields [4]. In general there are three ways to approach the building detection problem: stereo vision, line analysis, and using auxiliary information. However, there exists no single method that can perfectly detect all buildings in every aerial image. For the proposed localization system, the line analysis approach is adopted to extract features from an aerial image. Lines are considered appropriate since most man made structures are rectangular or contain mostly straight edges [5]. Most building detection methods start with low level image processing methods of edge and line detection. The problem is made difficult with the presence of shadows, surface markings, vegetation, and other distractions which may add unwanted boundary lines or fragmented boundaries of interest. These effects together are known as the figure-ground problem, and it has a much more significant impact compared with sensory noise [6]. As a result the resulting feature map is not an absolutely accurate representation of the environment.

The aerial maps that will be used in the localization system are orthoimages, which are images derived from normal perspective images in a way such that displacements caused by sensor (camera) placement and relief of terrain are removed. These high resolution images are in the format of grayscale bitmaps, where $10cm$ in real life resolves to approximately 1 pixel length. The aerial image of the $220m \times 180m$ workspace where the proposed localization system will be tested is shown in figure 1.

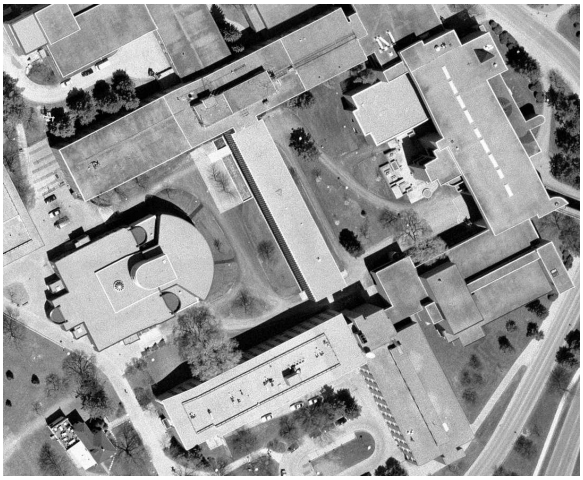

Fig. 1. The aerial image of the proposed localization system test site, located at the University of Waterloo

The first step in obtaining a feature map is the processing of raw image data using Canny's edge detector [7] to create an edge map which highlights the pixels that are likely to be part of building boundaries. The next step is to remove the effects of shadows. Shadows can be easily identified in an aerial image as they appear much darker compared to all other objects. To correctly remove the effects of shadow, it is necessary to distinguish whether a shadow edge is shared with a building boundary (which should not be removed), or incident with the ground (which should be removed). One approach to this is to take into account the source of illumination (sunlight). Sobel operators are used to estimate the intensity gradient over the edge of a shadow. If the intensity gradient increases in the direction of illumination (away from the light source), the corresponding edge is considered a shadow edge and is eliminated from the edge map.

The edge map is further filtered by masking edges that may have been generated from distractions in the aerial image, such as vegetation in the environment. In general image intensities in distraction areas vary in a way that give the appearance of rough texture. A corner response measure obtained using the Harris corner detector [8] is used to discriminate these distractions. It was found that most distractions that appear in an aerial image are within a certain range of corner response values. Any pixel with a corner response value within the range will contribute to a mask that is applied over the edge map. The morphological closing operation is used on the mask prior to its usage to fill in small gaps.

From the filtered edge map, high level line segment (boundary) features are identified using a modified version of the Progressive Probabilistic Hough Transform (PPHT) [9], summarized in figure 2. The modified PPHT algorithm is designed to alleviate the figure-ground problem. The algorithm uses multiple edge maps as inputs derived from different edge detection settings and is able to vary its threshold parameters as it iterates so that longer line segments are always extracted first. Furthermore, the algorithm discourages the extraction of multiple overlapping line segments in areas of the edge map where edge density is high. For more detail, refer to [10].

Extracted line segments are further processed so that segments that are almost parallel and close to each other are merged to a single line. A simple building model is then used to distinguish whether a line segment is a real boundary. A building should be enclosed and therefore the line segments making up its boundary should theoretically have overlapping endpoints. However, since the edge map is not perfect, this requirement is relaxed by accepting endpoints that are near one another. Line segments that fail this test are removed from the map. The final result of the map generation process can be seen in figure 3. The feature map is not perfect, meaning that a robot using this map will have errors in its interpretation of the real world. Fortunately, Bayes filters are known to remain robust even with such discrepancies.

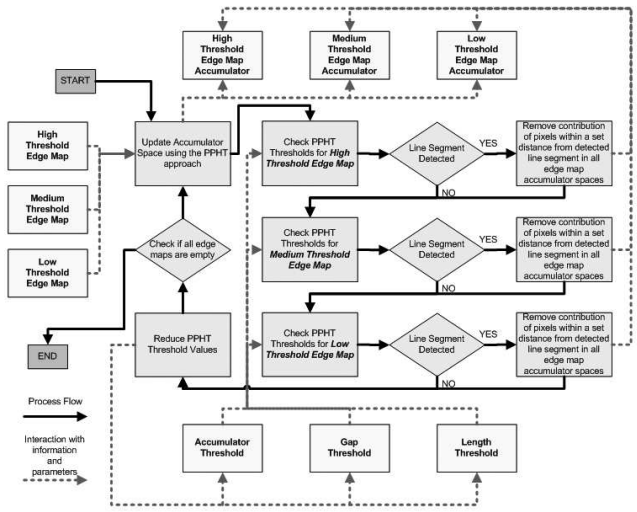

Fig. 2. The modified progressive probabilistic Hough transform algorithm

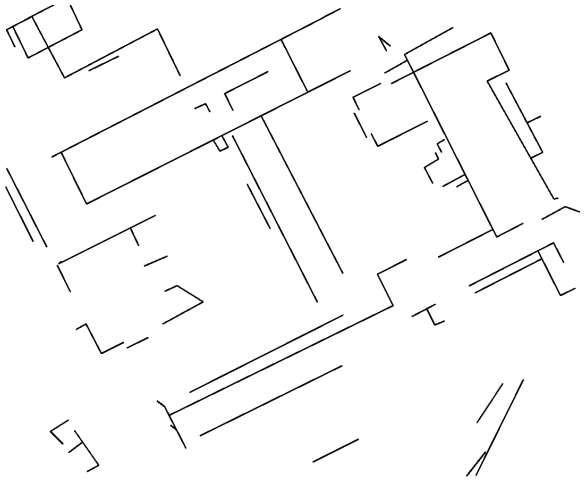

Fig. 3. The feature map derived through image processing of Fig.1

III. Camera Observations

The orientations of observed building walls needs to be interpreted from camera images. This information can be recovered from the effect of perspective using vanishing point analysis. A benefit of using vanishing point analysis and selecting building orientation as a measure for comparison is that it makes the system tolerant to significant camera rolling and tilting. Edge detection and the PPHT algorithm are used again to extract useful features from an image scene.

The concept of vanishing point has been known for centuries. When parallel lines in 3d space (or object space) are projected onto an image plane using a central projection model, the lines on the image plane will intersect at a point known as the vanishing point [11] [12]. The Gaussian sphere [13] shown in figure 4 was introduced as a method of quantifying the location of vanishing points. It is a unit sphere centered on the focal point of a vision system. Using the Gaussian sphere, a vanishing point can be defined by its projection on the sphere, where it has a unique coordinate (azimuth and elevation). When a vanishing point has been identified in an image, it is possible to infer the orientation of 3d objects from an image [14].

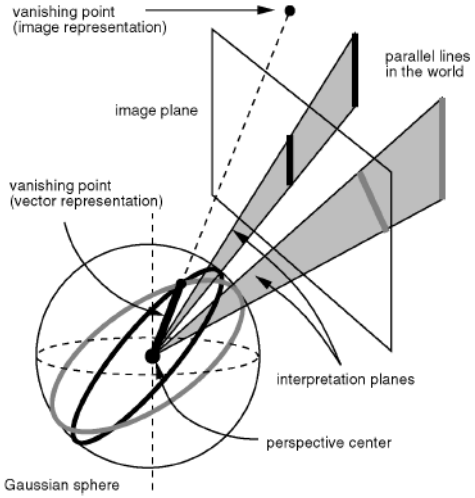

Fig. 4. The Gaussian Sphere [15]

To autonomously identify vanishing points in an image, an approach similar to that presented by Gallagher [16] is used. Intersections of line segments discovered by the PPHT are projected onto the Gaussian sphere. Vanishing points are likely to appear where there is a high density of intersection points, which are identified by subtractive clustering [17] [18]. To aid the clustering process, it is assumed that the on board camera will normally remain leveled and not experience severe tilting and rolling. With this assumption, it follows that vanishing points will likely appear near the pole (from vertical lines in an image) and the equator (from horizontal lines in object space) of the Gaussian sphere. Therefore, the clustering problem can be divided into two parts; one that searches above a certain angular elevation for a vanishing point near the pole, and one that searches between two elevations slightly above and below the horizon. This implementation has been tested and confirmed to generate better and more accurate vanishing point detection results.

Line segments previously identified by the PPHT are classified and associated with a vanishing point to determine their orientation in 3d space. This is done by projecting each line segment onto the Gaussian sphere and measuring the arc distance to each identified vanishing point. Membership of a class is won by the shortest distance measure. Figure 5 is an example of the result of vanishing point analysis and line segment classification. In this figure, detected line segments are colour coded according to an associated vanishing point. Red lines belong to the first vanishing point, green lines belong to the second, and blue lines belong to the third. A black line segment is one which its orientation is unknown

due to failure to associate it with any vanishing point. The location of the corresponding vanishing points on the Gaussian sphere has also been identified in the figure by their azimuth and elevation in units of degrees. The azimuth is of greater interest as it indicates the orientation of the associated line segments (and the wall they belong to) in 3d space.

Fig. 5. Result of the vanishing point analysis on an image captured by the camera. The location where the image is taken is within in the workspace defined by fig.1 at the University of Waterloo

IV. Particle Filter Implementation

The particle filter is an implementation of the Bayes filter using a finite number of particles in state space to describe the belief state probability density distribution. The algorithm propagates particles through time using the survival of the fittest concept. Initially, the particles are uniformly distributed throughout the workspace to represent the unknown starting state. Then, the algorithm iterates with two steps, the first of which propagates particles forward in time based on control inputs. In the second step, sensor measurements are used to determine an importance factor for each particle. This factor is used in particle resampling. A more detailed explanation of the particle filter can be found in [19]. A benefit of the particle filter is that it works for probability distributions of any shape and is able to achieve localization with an unknown initial state.

To experimentally validate the proposed localization system, on board camera images were rerecorded for conducting offline localization. Unfortunately, neither the vehicle odometry nor control inputs were available directly, which are typically required for the algorithm propagation step. As a substitute, the vehicle forward velocity (v) and yaw rate (ω) were extracted by differencing the position and yaw between images. Zero-mean Gaussian noise was then added to account for actuator uncertainties. For the results presented, the standard deviations for forward velocity and yaw rate are $10\frac{cm}{s}$ and $\frac{5^o}{s}$ respectively.

The state transition model shown in equation (1) is used in the propagation step of the algorithm. For each iteration, this model propagates the particles forward according to the velocities v and ω.

$$\begin{bmatrix} x_t \\ y_t \\ \vartheta_t \end{bmatrix} = \begin{bmatrix} v\Delta t \cos\left(\vartheta + (\omega\Delta t)/2\right) \\ v\Delta t \sin\left(\vartheta + (\omega\Delta t)/2\right) \\ \omega\Delta t \end{bmatrix} + \begin{bmatrix} x_{t-1} \\ y_{t-1} \\ \vartheta_{t-1} \end{bmatrix} \quad (1)$$

The importance factor is a weighting for particles that indicates the likelihood of the particle state being the true vehicle state. This factor is determined by observations made by exteroceptive sensors (in this case the camera), and is a function of the similarity between expected measurements and observed measurements. The expected measurement are determined for each particle. By referring to the feature map, each particle can determine the relative orientation of features observable in its field of view as a function of bearing $\psi_r = \psi_r(\alpha)$.

The observed measurements come from the vanishing point analysis. Depending on the scene, it may be possible to observe multiple features at various relative orientation $\psi_{s,i}(\alpha)$ (where i is the index for different features). To keep track of multiple features at a given bearing, the camera image is divided into view sections e (bounded by starting and ending bearing limits $\alpha_{e,start}$, $\alpha_{e,end}$), within which the non-empty set of observable features remain constant.

The importance factor for each particle W_m is evaluated by first considering the similarity between observed and expected measurements in individual view sections e. This similarity (η_e) is determined heuristically according to equation (2).

$$\eta_e = \max_i \left[1 - \frac{1}{1 + \exp\left(\frac{20 - |\psi_{s,i} - \psi_r|}{2}\right)}\right] \quad (2)$$

A weighting factor ρ_e is also determined for each view section depending on its size with respect to the combined size of all view sections, as expressed in equation (3).

$$\rho_e = \frac{\alpha_{e,end} - \alpha_{e,start}}{\sum_{e=1}^{e_{\max}} \left(\alpha_{e,end} - \alpha_{e,start}\right)} \quad (3)$$

The importance factor for a particle is then calculated by considering all view sections using equation (4), and the low variance sampling method [20] is used during particle resampling in each iteration of the particle filter. It should be noted that particles that move through a building boundary on the feature map are automatically regenerated with a new random state.

$$W_m = \sum_{e=1}^{e_{\max}} \rho_e \eta_e \quad (4)$$

Through numerous trials with different sizes of particle sets, it has been shown that offline localization can be achieved with as little as 300 particles. However, it was determined that 2000 particles are required to assure a high likelihood of convergence.

For the path shown in figure 6, snapshots of the particle set for a particular execution of the particle filter process are

shown in figures 7 through 9. To evaluate the performance of the particle filter, the positioning error is tracked over the test course. This error is determined by the difference between the true position and the location where the belief state probability density is determined to be a maximum by using the mean shift clustering algorithm [21]. The result for several runs of the particle filtering process is shown in figure 10. This figure indicates that convergence of particles is achieved for all test runs at around 150s, after which a small positioning error is maintained.

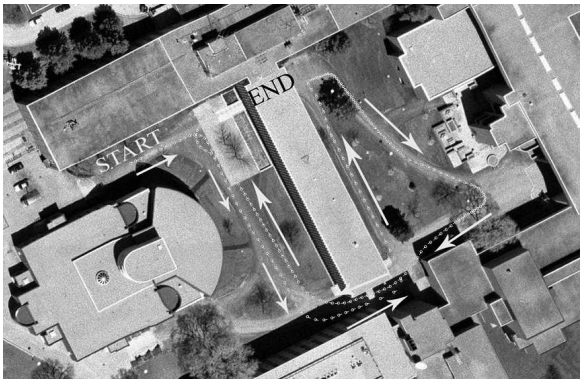

Fig. 6. Particle filter localization test course with the true path shown

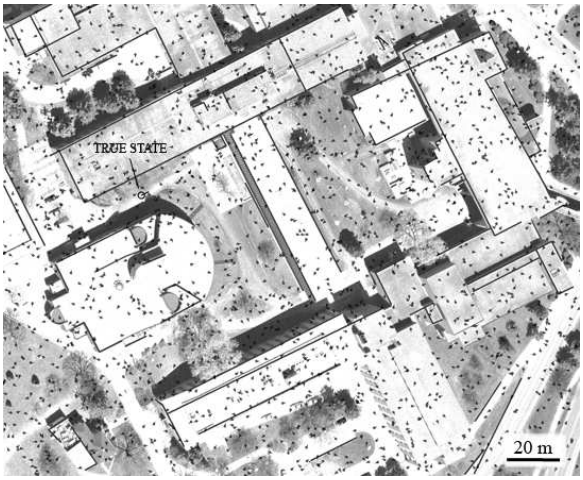

Fig. 7. Particle filter localization result - elapsed time: 0s

Closer examination reveals that during a couple of the test runs (1 and 2), the positioning error became very low but increased again. This occurred because there were multiple dominant particle clusters existing concurrently which exchanged the role of the most dominant cluster in a back and forth manner. To show that once convergence is maintained once it has been achieved, the particle filter process is executed by initiating particles at the true starting position. Positioning error for this case is shown in figure 11 and it can be seen to remain relatively stable (note the difference in scale compared to figure 10). The main source of error in this particle filter localization process is the imperfect map. By

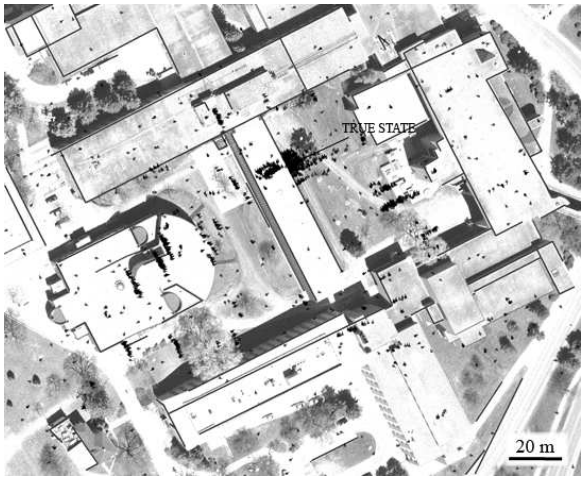

Fig. 8. Particle filter localization result - elapsed time: 60s

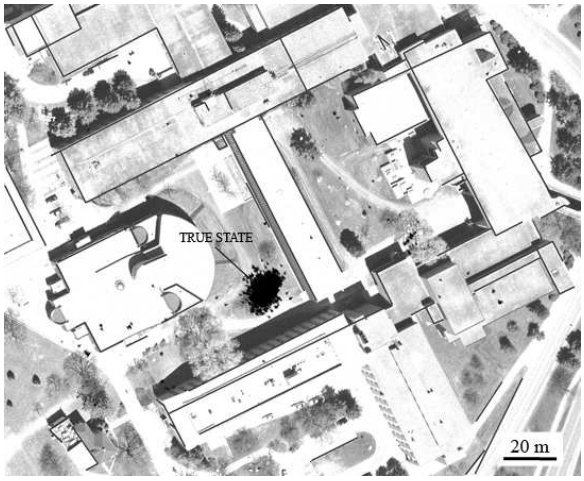

Fig. 9. Particle filter localization result - elapsed time: 150s

examining the trajectory taken in the test course, instances where the positioning error was relatively high in figure 11 corresponded to inaccuracies in the feature map. The average positioning error is determined to be $3.40m$, with 95% of the error measurements being below $4.80m$. This result is comparable to that of the standard positioning service for GPS, which can achieve a positioning accuracy of 4.83 in 95% of horizontal (2d) measurements according to [22]. However, the particle filter process takes considerable time before particle convergence is achieved. On the other hand, GPS availability and performance are not reliable in some urban settings due to multipath. Therefore, the particle filter process presented may serve as an alternative localization tool.

Timing analysis indicates that the current implementation of the particle filter on a computer with a $1.5GHz$ Pentium M processor is unable to achieve real time operation while processing camera images at $1Hz$. A large proportion of computational time is spent on processing images captured from the camera and performing the vanishing point analysis.

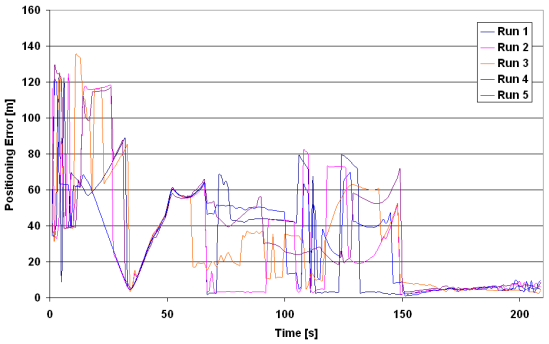

Fig. 10. Localization Errors Recorded in 5 instances of the particle filter process with unknown initial state

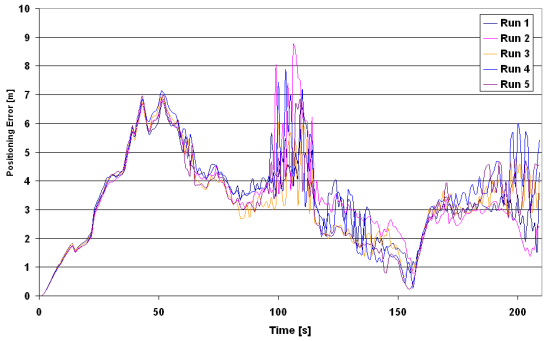

Fig. 11. Localization Errors Recorded in 5 instances of the particle filter process with known initial state

On average an image frame requires about $9s$ of processing time. However, it is predicted that the system real time implementation may be feasible given a faster computer and through program code optimization.

V. Conclusions

The design of a monocular vision based particle filter localization system was presented. The system is designed for urban settings consisting mainly of orthogonal structures. An aerial image is given from which information is extracted to autonomously generate a feature map for localization. Image processing techniques and the vanishing point analysis are used to estimate building wall orientations as observations for the particle filter. Experimentation was conducted in a large urban environment and the results indicate that localization is achievable by the system. Positioning error is determined to be comparable to that of a GPS receiver using standard positioning service, but a considerable duration of movement in the workspace is necessary. Still, this particle filter process may be a useful alternative if GPS performance is degraded due to multipath or when GPS service is not available. Although testing was conducted at only one urban setting, it is hypothesized that similar results can be achieved in other urban settings provided that it consists mainly of orthogonal structures. In the future, further verification in other urban environments will be performed using a real robot.

Acknowledgment

Funding of this research is provided by Auto21 Canada. The aerial images used in this research are properties of the Regional Municipality of Waterloo, Ontario, Canada

References

[1] E. Royer, M. Lhuillier, M. Dhome, and J.-M. Lavest, "Monocular vision for mobile robot localization and autonomous navigation," *Internation Journal of Computer Vision*, vol. 74, no. 3, pp. 237–260, 2007.

[2] W. Zhang and J. Kosecka, "Image based localization in urban environemnts," in *Proceedings of the International Symposium on 3D Data Processing, Visualization, and Transmission*, 2006.

[3] D. J. amd G. Dudek, "Urban position estimation from one dimensional visual cues," in *Proceedings of the Canadian Conference on Computer and Robot Vision*, 2006.

[4] J. Shufelt and J. D.M. McKeown, "Fusion of monocular cues to detect man-made structures in aerial imagery," *CVGIP: Image Understanding*, vol. 57, no. 3, pp. 307–330, 1993.

[5] A. Croitoru and Y. Doytsher, "Right-angle rooftop polygon extraction in regularised urban areas: Cutting the corners," *The Photogrammetric Record*, vol. 19, no. 118, pp. 311–341, 2004.

[6] C. Lin and R. Nevatia, "Building detection and description from a single intensity image," *Computer Vision and Understanding*, vol. 72, no. 2, pp. 101–121, 1998.

[7] J. Canny, "A computational approach to edge detection," *IEEE Trans. Pattern Analysis and Machine Intelligence*.

[8] C. Harris and M. Stephens, "A combined edge and corner detector," in *4th Alvey Vision Conference*, 1998.

[9] J. Matas, C. Galambos, and J. Kittler, "Robust detection of lines using the progressive probabilistic hough transform," *Computer Vision and Image Understanding*, vol. 78, pp. 119–137, 2000.

[10] K. Leung, "Monocular vision based particle filter localization in urban environments," Master's thesis, University of Waterloo, Waterloo, ON, Canada.

[11] F. van den Heuvel, "Vanishing point detection for architectural photogrammetry," *International Archives of Photogrammetry and Remote Sensing*, vol. 32, no. 5, pp. 652–659, 1998.

[12] V. Vantoni, L. Lombardi, M. Porta, and N. Sicard, "Vanishing point detection: Representation analysis and new approaches," in *Proceedings of the Internation Conference on Image Analysis and Processing*, 2001.

[13] S. Barnard, "Interpreting perspective images," *Artificial Intelligence*, vol. 21, no. 4, pp. 435–462, 1983.

[14] A. Tai, J. Kittler, M. Petrou, and T. Windeatt, "Vanishing point detection," *Image and Vision Computing*, vol. 11, no. 4, pp. 240–245, 1993.

[15] J. Shufelt, "Performance evaluation and analysis of vanishing point detection techniques," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 3, pp. 282–288, 1999.

[16] A. Gallagher, "A ground truth based vanishing point detection algorithm," *Pattern Recognition*, vol. 35, no. 7, pp. 1527–1543, 2002.

[17] R. Yager and D. Filev, "Generation of fuzzy rules by mountain clustering," *Journal of Intelligent and Fuzzy Systems*, vol. 2, no. 3, pp. 209–219, 1994.

[18] S. Chiu, "Fuzzy model identification based on cluster estimation," *Journal of Intelligent and Fuzzy Systems*, vol. 2, no. 3, pp. 267–278, 1994.

[19] M. Arulampalam, S. Maskell, N. Gordon, and T.Clapp, "A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking," *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, 2002.

[20] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotcs*. USA: The MIT Press, 2005.

[21] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 603–619, 2002.