Robust Outdoor Visual Localization Using a Three-Dimensional-Edge Map

Stephen Nuske*
School of Information Technology
and Electrical Engineering
University of Queensland
St Lucia, Queensland, 4072
Australia
Autonomous Systems Lab
CSIRO ICT Centre
P.O. Box 883
Kenmore, Queensland, 4069
Australia

Jonathan Roberts
Autonomous Systems Lab
CSIRO ICT Centre
P.O. Box 883
Kenmore, Queensland, 4069
Australia
e-mail: jonathan.roberts@csiro.au

Gordon Wyeth
School of Information Technology
and Electrical Engineering
University of Queensland
St Lucia, Queensland, 4072
Australia

Received 23 February 2009; accepted 27 July 2009

Visual localization systems that are practical for autonomous vehicles in outdoor industrial applications must perform reliably in a wide range of conditions. Changing outdoor conditions cause difficulty by drastically altering the information available in the camera images. To confront the problem, we have developed a visual localization system that uses a surveyed three-dimensional (3D)-edge map of permanent structures in the environment. The map has the invariant properties necessary to achieve long-term robust operation. Previous 3D-edge map localization systems usually maintain a single pose hypothesis,

*Currently Postdoctoral Fellow at the Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213 (e-mail: nuske@cmu.edu).

Journal of Field Robotics 26(9), 728–756 (2009) © 2009 Wiley Periodicals, Inc.
Published online in Wiley InterScience (www.interscience.wiley.com). • DOI: 10.1002/rob.20306

making it difficult to initialize without an accurate prior pose estimate and also making them susceptible to misalignment with unmapped edges detected in the camera image. A multihypothesis particle filter is employed here to perform the initialization procedure with significant uncertainty in the vehicle's initial pose. A novel observation function for the particle filter is developed and evaluated against two existing functions. The new function is shown to further improve the abilities of the particle filter to converge given a very coarse estimate of the vehicle's initial pose. An intelligent exposure control algorithm is also developed that improves the quality of the pertinent information in the image. Results gathered over an entire sunny day and also during rainy weather illustrate that the localization system can operate in a wide range of outdoor conditions. The conclusion is that an invariant map, a robust multihypothesis localization algorithm, and an intelligent exposure control algorithm all combine to enable reliable visual localization through challenging outdoor conditions.

1. INTRODUCTION

Heavy industry has recently begun to explore the use of automated mobile equipment in their operations, usually in an attempt to resolve safety and/or productivity issues. Over the past 5 years, our research group has been developing and testing navigation systems for large forklift-type vehicles that are used in the aluminum production industry to handle large loads, such as containers of molten aluminum or large anodes used in the smelting process. The nature of these applications dictates that the vehicle's navigation system must be dependable. Of the sensing options available for localization, our research has investigated three different sensor systems: two-dimensional (2D) scanning laser range finders, global positioning system (GPS) receivers, and fish-eye cameras.

We are not focused on finding a single solution that outperforms the others in all cases; we are instead looking at combining the complementary and redundant properties of many independent localization systems. A higher level localization system is currently under development that takes redundant inputs from multiple independent sublocalization systems and compares and arbitrates their pose estimates in order to formulate a single, highly reliable, confident pose estimate for the vehicle. The initial instance of this redundant localization system was published in an earlier paper, in which results from fully autonomous experiments were shown (Roberts, Tews, & Nuske, 2008). The continued development of the fish-eye camera system is the focus of this paper. An early implementation of the fish-eye camera system was published in Nuske, Roberts, and Wyeth (2008).

The three types of sensors each has different performance characteristics. The GPS is most reliable in open outdoor areas but is not operational indoors. When operating outdoors with buildings nearby, the GPS receiver reports a mix of signal dropouts, confident accurate measurements, and the most dangerous case for an autonomous navigation system, that is, measurements that are both confident and inaccurate. The laser scanner system (Roberts, Tews, Pradalier, & Usher, 2007; Tews, Pradalier, & Roberts, 2007) has proven to be a reliable system across the entire work site and has been used as the basis of long-duration, accurate autonomous navigation and precision load transfer maneuvers. An early implementation of the fish-eye camera system, published in Nuske et al. (2008), showed accuracy suitable for autonomous navigation but not necessarily suitable for precise load transfer maneuvers. [Another vision-based system has been developed for these precise maneuvers (Pradalier, Tews, & Roberts, 2008).] This paper continues the development of the fish-eye camera system, and the testing again looks at its performance in outdoor lighting conditions.

The reason for focusing on outdoor tests is the dynamic lighting conditions that make the outdoor areas the most challenging part of the application environment for a vision system. The sun often appears directly in the image, causing issues for exposure control, addressed in this paper in Section 4. Furthermore, the ever-changing position of the sun in the sky causes drastic changes in the shadows and shading in the scene and, as a result, has a large impact on the information captured by the camera. It is impossible in all situations to completely decouple the effects of lighting when extracting information from images, which are fundamentally an array of light measurements. This is a significant concern for the choice of visual map. A visual map can be created automatically with information extracted from images, but this map may be specific to a certain lighting condition and may not be useful for localization as the appearance of the environment changes after the sun's position in the sky moves.

Autonomously generated visual maps have often been built from image-point features, such as in the work of Lowe, Se, and Little (2002), which do possess some robustness to lighting conditions. However, these image-point features are only pseudo-invariant. Many authors have tested image-point features through lighting changes (Cummins & Newman, 2008; Lowe, 2004; Mikolajczyk & Schmid, 2005; Sim & Dudek, 2003; Valgren & Lilienthal, 2007), and there appears to be enough evidence—both in the literature and in our experience—suggesting that image-point features can be unstable as the lighting changes, and this should be a concern for a system that must operate outdoors through extreme lighting conditions.

We wish to localize using image features that are associated with permanent structures. The environment is somewhat dynamic in nature as there are other moving vehicles and other items being moved around the site, changing its appearance; but most of the environment is physically the same, day after day. We therefore propose a vision-based localization system that uses three-dimensional (3D) edges of buildings as the features of interest. Edges are far more robust to lighting changes in our target environment as they tend to be larger scale features, such as those created at the boundaries of buildings with the sky, large doorways, and other large-scale building features, such as overhangs.

We have developed an edge-based localization system inspired by the work in Klein and Murray (2006). This algorithm uses a particle filter that maintains multiple pose hypotheses in which each particle represents a possible pose of the vehicle. This is different from traditional edge-model localization frameworks, which maintain a single hypothesis. Multihypothesis particle filters have some notable advantages:

- the ability to commence operation with large uncertainty in the initial pose of the vehicle
- the ability to deal with local minima caused by spurious image edges by forming multimodal distributions

Our contribution lies in the area of the observation function, where we have developed a new function that further improves the initialization process and robustness to local minima in the particle filter. The new observation function conducts, for each particle, a search in the image outward along the normals to the projected 3D edges in the map, whereas the previous observation functions consider only image edges that directly align with the projected 3D-edge map. The implicit downside of these previous functions is that a small change in pose causes a large drop in probability, requiring a tight distribution of many particles. We will show situations illustrating the benefits of our new observation function in which the filter can converge reliably even when given sparse particle distributions.

An additional contribution of this paper is the development of an intelligent exposure control algorithm to deal with the issues of nonuniformity of lighting across the scene (in particular shadows) and the problem of direct sunlight in the camera's field of view (FOV).

The system was evaluated in an outdoor test environment using a vehicle fitted with two fish-eye cameras mounted facing either side of the vehicle. The experiments were designed to assess the system's performance in the outdoor lighting conditions. First, the system was tested for a continuous 30-min period on a bright sunny day when the vehicle covered a distance of 1.5 km. The second test was conducted over the period of an entire day when the weather was again bright and sunny. The vehicle was driven along a path at the beginning of each hour from just after sunrise at 7 a.m. to just before sunset at 5 p.m. The final test was conducted during rainy weather when there were raindrops sitting directly on the camera's lenses, partially obscuring their view.

The remainder of the paper is structured as follows. In Section 3, an edge-based localization technique is described that can be used to estimate the position and heading of the mobile vehicle given a sparse 3D-edge map of the doors, walls, and roofs in the buildings in the environment. Section 4 discusses the issue of camera exposure control and shows how knowledge of the scene can be used to intelligently adjust the camera exposure parameters to improve the quality of information in the image. Results from experiments on a vehicle in a real work site are presented for initialization of the filter in Section 6 and for extended outdoor operations of the moving vehicle in Section 7. Finally, conclusions are drawn in Section 8. Video attachments depicting the results of this paper can be found online at http://www.cat.csiro.au/ict/download/nuske/ and are listed in Section 9.

2. RELATED WORK

Most visual localization work has been performed by aligning the 3D world locations of features to their corresponding 2D locations in the image plane, the work of Davison and Molton (2007) and Lowe et al. (2002) being notable examples.

For many years computer vision was restricted to indoor environments, but recently more outdoor experiments are being performed in which the challenges of the lighting conditions must be met. The work of Cummins and Newman (2008) and Valgren and Lilienthal (2008) both present recent results that improve operations even when lighting changes cause disruptions in the image-point feature descriptions. However, these authors are solving a problem different from ours—that of appearance-based localization (recognizing a previously visited place) rather than geometric localization (deriving an explicit orientation and pose of a vehicle with enough accuracy to allow autonomous navigation).

The monocular visual odometry system of Nistér, Roberts, and Wyeth (2006) and the stereo visual odometry system of Maimone, Cheng, and Matthies (2007) are examples of geometric localization in outdoor environments. Both odometry systems present very impressive and useful results, although for long-term autonomous operations the frame-to-frame error (however small) will accumulate and must be removed using a map of the environment.

Outdoor visual localization examples that do perform geometric localization with a map can be seen in the dense 3D-point cloud stereo vision work of Marks, Howard, Bajracharya, Cottrell, and Matthies (2008) in unstructured natural environments, and Paz, Pinies, Tardos, and Neira (2008) show a stereo system operating in an urban environment. Stereo is well known to be suited to environments with a lot of small-scale texture and not suited to environments with large homogeneous texture-less walls, where the dominant visual features are at the edges of the doors, walls, windows, and rooflines of the buildings.

There is a class of localization algorithms that compare edges extracted from images with 3D-edge maps. One of the initial edge-based localization techniques was developed in Kosaka and Kak (1992) for navigating indoor hallways using a 3D-edge map of the doors and walls. This type of technique has, for the most part, been applied in indoor environments. A more recent real-time technique developed in Drummond and Cipolla (2002) has been applied outdoors in Reitmayr and Drummond (2006). A graphics processing unit (GPU)–accelerated version of this system was presented in Michel et al. (2007), and another outdoor example of this type of system was presented in Georgiev and Allen (2002).

The main limitation of the 3D-edge localization approaches listed above is that they need to be initialized with an accurate prior estimate of the pose and must remain with an accurate estimate at all times. This is because they maintain only a single pose hypothesis at a time, which makes them susceptible to the edges of scene clutter and shadows extracted in the camera image that are not in the 3D-edge map.

However, the multiple pose hypothesis particle filter has the ability to deal with spurious edges detected by the camera by forming multimodal distributions that correctly account for ambiguities in the observation function. A 3D-edge particle filter was presented in Klein and Murray (2006), which uses an observation function for each particle that measures the quality of alignment of the 3D-edge map with the camera image. This system is tested and evaluated against our proposed system.

The first instantiation of our system computes the observation function with image edge weighted equally (Nuske et al., 2008), rather than weighting each pixel equally as in Klein and Murray (2006). Computing the observation function on a local basis gives equal weighting to each edge, avoiding the observation function being dominated by large edges, so that smaller edges are not ignored in situations in which they are in fact providing useful localization information. The results of Nuske et al. (2008) showed that the filter using an observation function that weights edges equally was more accurate at estimating the orientation of the vehicle.

Even with an observation function that is computed on a per-edge basis, small changes of pose produce large changes of likelihood in the observation function. This is because the alignment between the 3D-edge map and camera image is evaluated only at the locations of the projected 3D-edge map. This paper proposes a new observation function that searches outward from the projected edges for the nearest edge in the camera image. The design of this new observation function is aimed at allowing the filter to converge even when given sparse particle distributions. Convergence is more likely because small changes of pose will not produce large changes in the likelihood measured by the observation function. Particles that are moderate distances or orientations away from the correct pose will still generate moderate alignment probabilities.

3. 3D-EDGE MAP LOCALIZATION

Our goal of a robust outdoor visual localization system led us to explore the use of a 3D map of the edges of the buildings in the environment. Edges are found on the buildings themselves (e.g., doorways, windows, ventilation openings) and at the visual boundary of the buildings with the background (e.g., the roofline against the sky and at the sides of buildings with the general background). The attractive property of the building edges in this environment is that they are static and we can measure their precise location in three dimensions.

3.1. A Sparse Map

The 3D-edge map of industrial buildings can be extremely sparse. The edge features stored in the 3D map include the building rooflines (typically, the gutter lines), the door frames (the edges of the large industrial roller doors), and on some of the buildings, some of the lines that define the edges of some other significant features in the building shape (such as overhangs). For a building with three roller doors and a single visible roofline, this translates into just 14 3D data points in the model (4 for each door and 2 for the roofline). This sparsity of data means that it is both feasible (in terms of cost and effort) and desirable (in terms of confidence in accuracy) to manually survey the model data points. Such a survey ensures that only permanent parts of the buildings are included—which is difficult to guarantee in an autonomous map-building system. An example of the map is shown in Figure 1(a). To give the reader an idea of scale, the roofline in the figure is 9 m high and the doors are 4 m wide × 6 m tall.

The cost of the survey was approximately US$4,000, which is an insignificant cost considering the large value of the raw minerals transported by the vehicle and also can be considered minor when compared with the price of a standard laser scanner, which can be more than US$4,000. The map took a single day to survey, which is a one-off delay that is not a concern for an application in which the vehicles operate day after day for several years of productive operation. The survey was of 19 industrial buildings ranging in height from 6 to 17 m and together having a footprint on the ground of approximately 290 × 100 m. The experiments in this paper are conducted in the industrial compound portion of the site, which is surrounded by seven buildings that have a footprint of around 70 × 45 m. Figure 2 presents an overhead view of the site and survey, showing both the dimensions of the overall footprint and the compound footprint.

The survey was taken using a total station, and the professional surveyor conducting the survey quoted the accuracy of the measurements to be within 50 mm. The accuracy depends on the surface properties of the point being measured and the viewing angle and the distance to the point.

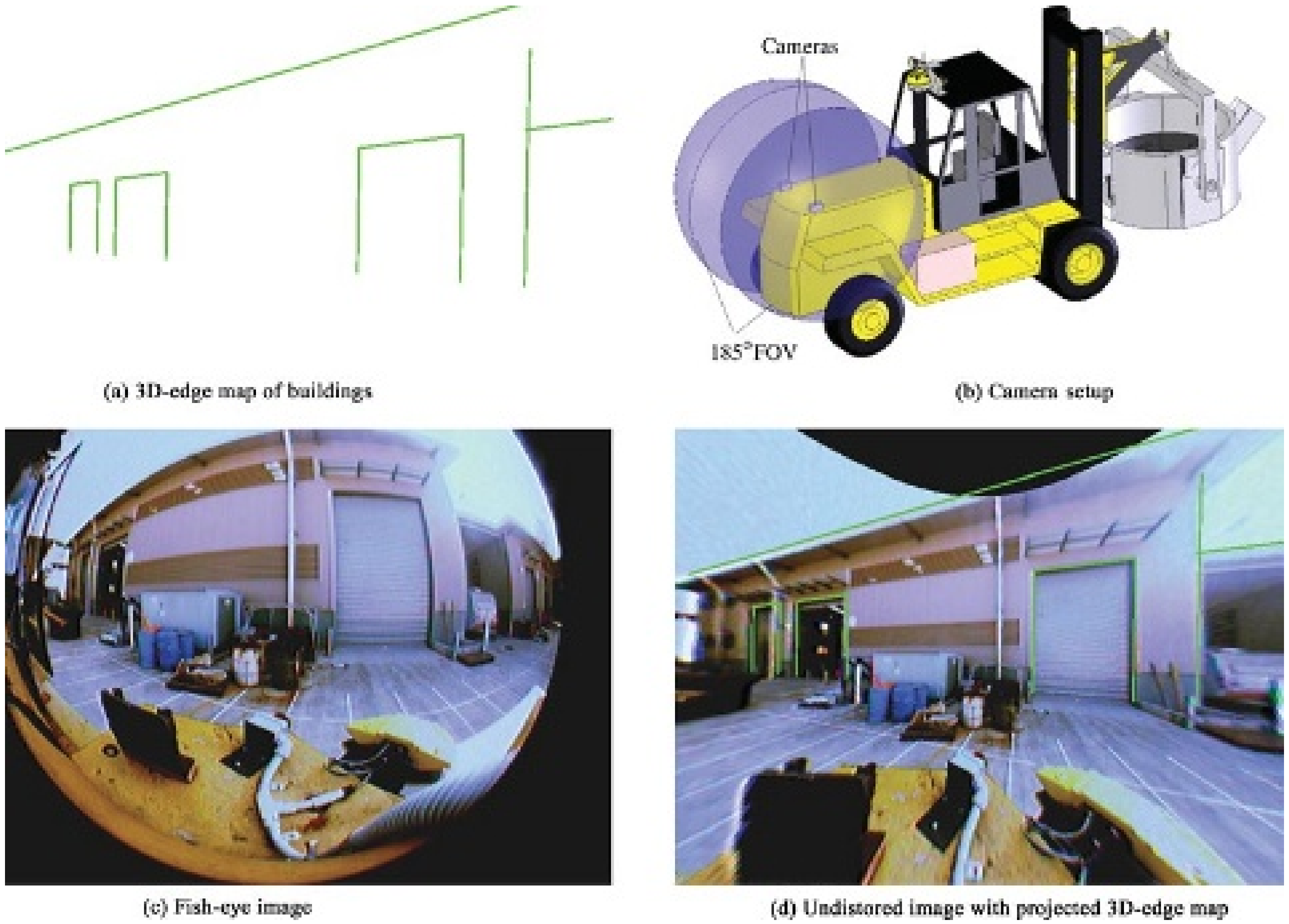
3.2. Wide-FOV Imaging

In our target application, the distance between the vehicle (and hence the cameras) and the buildings can vary from 1 to 50 m. This geometry indicates that we need an imaging system that can cover as much of the environment as possible at all times in order to guarantee that we can see edges in our model. We used two fish-eye cameras (with 185-deg FOV) mounted on the vehicle [Figure 1(b)]. A typical image from one of these cameras is shown in Figure 1(c).

A specialized lens model is adopted for calibration from Geyer and Danilidis (2001), which was shown to be applicable to fish-eye cameras by Ying and Hu (2004). The model assumes that all pixels in the fish-eye image map onto a sphere located in front of the image plane. The model consists of four parameters: m, the distance from the center of the sphere to the image plane; C_x and C_y, the center of projection on the image plane; and l, the distance from the center of the sphere to the intersecting focus point of the light rays and the center of projection line.

These calibrated parameters enable the fish-eye image to be transformed into an undistorted image. Examples of distorted/undistorted images can be seen in Figure 1. The corrected image is generated as follows: for each pixel coordinate $[U_u, U_v]$ in the corrected image [Figure 1(d)], with center $[C_x, C_y]$, the corresponding distorted coordinate $[D_u, D_v]$ in the fish-eye image [Figure 1(c)] is calculated by

$$D_u = R\cos\left[\operatorname{atan}\left(\frac{U_v}{U_u}\right)\right] + C_x, \qquad (1)$$



(a) 3D-edge map of buildings

(b) Camera setup

(c) Fish-eye image

(d) Undistored image with projected 3D-edge map

Figure 1. Examples of the surveyed 3D-edge map, fish-eye camera setup, and calibration. Two fish-eye cameras are placed at the front of the vehicle facing sideways. The hemispheres represent the FOV of the cameras.

$$D_v = R \sin\left[\operatorname{atan}\left(\frac{U_v}{U_u}\right)\right] + C_y, \tag{2}$$

where

$$R = \frac{\sin(\theta)(m+l)}{\cos(\theta)+l} \tag{3}$$

and

$$\theta = \operatorname{atan}\left(\frac{\sqrt{U_u^2 + U_v^2}}{f}\right), \tag{4}$$

where f is the effective focal length of the undistorted projective image, measured in pixels from the center of the sphere. f is also used in the projection model to project the 3D-edge map onto the image plane and can be selected according to the effective FOV that is required. The resulting undistorted image is converted into an edge image using Canny's (1986) algorithm with a 3 × 3 kernel.

3.3. Particle Filter Localization

Most previous 3D-edge based techniques calculate only a single pose estimate for each iteration, which requires a precise initial estimate of the pose and is susceptible to failure. Recently in Klein and Murray (2006), a particle filter method was presented that maintains many pose estimates per frame. The technique was applied to the tracking of a single object, such as a printer, from a range of just a few meters in a regular indoor environment. We too have chosen a particle filter implementation. The comparison between the map and the camera images is calculated for each pose hypothesis in a particle filter and provides a likelihood measure, discussed in Section 3.4.

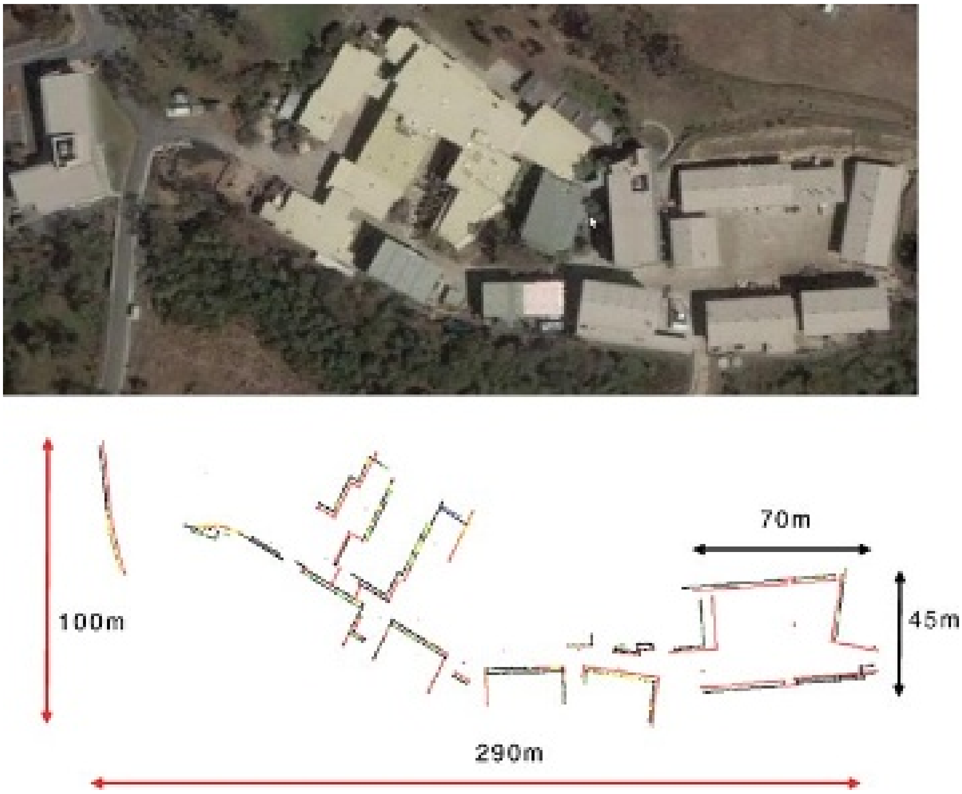


Figure 2. Top: Aerial image of the site. Bottom: Overhead view of the survey of the buildings.

In Thrun, Burgard, and Fox (2005) the use of a particle filter for pose estimation is described in detail. In brief, the filter is a set of N pose hypotheses (particles) $X_t = x_t^{(1)}, x_t^{(2)}, x_t^{(3)} \ldots, x_t^{(N)}$. The pose is a six-degree-of-freedom translation and rotation $(t_x, t_y, t_z, r_x, r_y, r_z)$. The dominant degrees of freedom are 2D horizontal translations and a rotation around the vertical axis (t_x, t_y, r_z). The additional degrees of freedom are used to deal with any deviations in the ground plane causing slight rolls and pitches in the vehicle and slight changes in the vertical displacement. The coordinate system is defined from the center point of the axle joining the two rear wheels, and positive rotations in the vehicle's heading are defined by anticlockwise rotations around the vertical axis. Vehicle odometry from wheel and steering encoders is used to estimate changes in horizontal translation and heading angle. The other degrees of freedom are not measured by additional sensors but are included as small perturbations in the filter; more details on the propagation model are in Section 3.6.

The set of poses is sampled from the previous set X_{t-1} using a propagation model m_t and a corresponding set of weights (probabilities), W. The weight of particle n is calculated at time k as follows:

$$W_k^{(n)} = p\big(y_k | x_k^{(n)}\big), \tag{5}$$

where y is the comparison between the edge image extracted from the camera image and the 3D-edge map projected to the image plane from the pose of each particle x. The comparison between the map and camera image provides a likelihood measurement based on observation functions described in the next section. The concept is that the particles nearest the correct pose will have the highest likelihood measure, because their projection of the 3D-edge map will have the best alignment with the camera image. These particles will have the highest probability of being resampled for the next iteration. To extract the current pose estimate of the vehicle from the filter, the mean pose of the most highly weighted particles is calculated. Here, the nonweighted mean of the 5% most highly weighted particles is used.

3.4. Observation Function

The likelihood measure for each particle is generated through a comparison with the edge image and the 3D-edge map. The map is projected onto the image plane so that a direct comparison can be made. A fast method is presented in Klein and Murray (2006) that performs this computation on a GPU counting the number of aligning pixels over the whole image. This section will first describe Klein and Murray's metric and then describe modifications that improve performance.

3.4.1. Klein and Murray

Klein and Murray's method was implemented by first placing the undistorted edge image into the GPU's texture memory. For each particle, the OpenGL projection matrix was set and the 3D-edge map was called to be rendered for each particle by a custom fragment shader program. The program permits the counting of the visible edge pixels of the 3D-edge map that align with edge pixels in the undistorted edge image. The custom fragment shader program permits pixels only to pass through the pipeline that align with edge pixels in the edge image. The pixels that pass this custom fragment shader program are counted using the OpenGL occlusion query extension (NVIDIA Corporation, 2007).

Klein and Murray present the likelihood measure of the particle, $W_t^{(n)}$, as a ratio between the count of aligning edge pixels (a) and the total number of visible edge pixels (v), calculated as follows:

$$W_t^{(n)} = p\left(y_t | x_t^{(n)}\right) \propto \exp\left(\kappa \frac{a}{v}\right), \tag{6}$$

where κ is a constant that weights the observation function.

Klein and Murray show that this metric can successfully track objects, but the simple ratio of pixel counts leads to the situation in which large edges, such as the rooflines of the buildings, dominate other smaller edges, such as the door edges. This is simply because the majority of pixels are in the roof edges. Smaller edges provide important localization information and should have more consideration in the observation function.

3.4.2. Per-Edge Function

The first implementation of our localization system (Nuske et al., 2008) presented a modification to Klein and Murray's function that incorporated per-edge measurements instead of a sum over the whole image. The new metric calculates the ratio of aligning-to-visible edge pixels for each edge, j. This is calculated using occlusion queries for each edge, giving the two measurements a_j and v_j. The first component of the new metric is the original Klein and Murray global ratio; the second component is the mean of the ratios of each individual edge and is calculated as follows:

$$W_t^{(n)} = p\left(y_t | x_t^{(n)}\right) \propto \exp\left(\kappa \frac{a}{v} + \lambda \frac{\sum_{j=0}^{m} \frac{a_j}{v_j}}{m}\right), \tag{7}$$

where m is the number of edges. The second component of this equation treats each edge equally, regardless of its size. This penalizes particles with edges that are smaller and misaligning, even if the overall count of aligning pixels is high. The filter will prefer particles with the combination of a reasonably high overall count of aligning pixels and smaller aligning edges. This allows the filter to maintain a better track of the smaller door edges in the environment. The new per-edge component has its own constant λ, and this has to be tuned in conjunction with κ, thus striking a balance between the global and per-edge components, although as shown later in Section 5, the function behaves well when these values are equal.

3.4.3. Nearest-Edge Function

An issue with the above two functions is that the peaks in the functions are narrow—a small change in pose causes a large change in likelihood value due to the binary comparison (aligned or not aligned) at the core of the functions. Therefore a tight pack of many particles is required to correctly maintain track of the narrow peaks. Such a distribution in the filter will be susceptible to local maxima. This susceptibility will manifest itself in two ways: converging at an incorrect estimate at initialization and also making the filter unable to recover after slightly losing track of the 3D-edge map.

Klein and Murray proposed two methods to overcome this issue. The first is to dilate the edge image, creating thicker image edges. However, creating thicker edges will serve only to flatten and plateau the observation function, causing a loss in accuracy. The other solution is to use a two-stage filter with the first stage being performed in low resolution, essentially creating thicker map edges. The second, high-resolution stage can provide added accuracy; however, the particles resampled for the second stage may not be resampled near the narrow peaks because of the flat function in the first stage. This will cause the filter to jitter and to lose track; both problems are reported in Klein and Murray (2006).

We propose a new metric: the distance to the nearest edge pixel, which is an improved measurement regime to the binary alignment. A similar method is proposed in Drummond and Cipolla (2002), though they used a set of nearest-edge measurements to extract a single pose hypothesis. In our algorithm we incorporate nearest-edge measurements into the multiple hypothesis particle filter. The nearest-edge metric not only considers pixels aligning with the projected edge but searches outward in the image. This provides a wider, sloping, function that will be easier for the particle filter to remain converged and recover from divergence and also importantly will be more robust when initializing the filter with a large/sparse particle distribution.

Drummond and Cipolla's method is to take sample points along each edge at regular pixel increments. At each sample point a search is conducted outward along the edge normal to find the nearest image edge. Here the same sampling and searching strategy is adopted. A sample is taken every 20 pixels on each 3D edge, s. A search is conducted in both the positive and negative directions of the 3D edge's normal in the image plane. The search distance in the real-world coordinate frame is a constant, D_w. The search distance in pixels D is determined according to the focal length f in pixels and the depth of the sample point E_z:

$$D = D_w \frac{f}{E_z}. \tag{8}$$

An example of the sample and search for the nearest edge can be seen in Figure 3. The distance to the nearest edge, d, is calculated in pixels and is normalized to a value between 0 (which represents

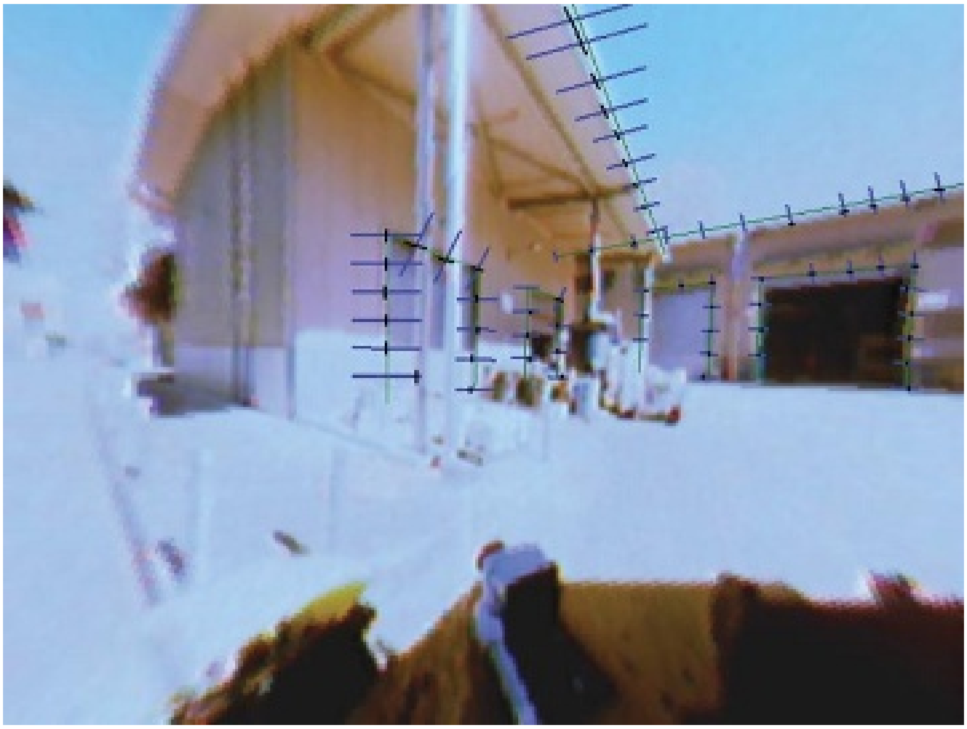


Figure 3. Nearest-edge search. The largest lines are the projected edges, the smaller perpendicular lines are the search along the edge normal, and the smallest lines indicate the nearest detected edge.

a zero search distance) and 1 (which represents the maximum search distance D). A Gaussian function converts d to add weight to closer edges:

$$g(d) = \exp\left(-\frac{d^2}{2\sigma^2}\right), \tag{9}$$

where the constant σ must be selected to weight the output appropriately. σ could be set to $\frac{1}{3}$ to give no importance [$g(1) \approx 0$] for when the nearest edge was found at the end of the search (that is, $d \approx 1$). However, an edge found near the end of the search should hold greater importance than not finding an edge at all. Therefore σ is set to $\frac{2}{3}$ to give the output value $g(1) \approx 0.3$, and when no edge is found, g evaluates to 0.

The output of these samples [$g(d)$] is formed into the final observation function by first aggregating the samples for each edge to give a likelihood l for the s samples on the edge:

$$l = \frac{\sum_{i=0}^{s} g(d_i)}{s}. \tag{10}$$

The likelihood of all of the m edges is aggregated as

$$W_t^{(n)} = p\big(y_t | x_t^{(n)}\big) \propto \exp\left(\kappa \frac{\sum_{k=0}^{m} l_k}{m}\right). \tag{11}$$

This observation function is designed to allow the filter to converge given sparse particle distributions. Particles that are moderate distances away from the correct pose will still provide moderate likelihood scores, whereas the other observation functions will assign very low weights to these particles.

The performance of the three different observation functions presented above are compared in initialization in Section 6. For ease of identification in the remainder of this paper, the three functions are referred to as follows:

- Eq. (6) is named *Klein and Murray*
- Eq. (7) is named *Per-edge*
- Eq. (11) is named *Nearest-edge*

3.5. Occlusions

Self-occlusions, when one building occludes another (known occlusions), can be dealt with by the depth buffer. A real-time technique is presented in Klein and Murray (2006) using a subsampled depth buffer. The technique is to render faces of the buildings to the depth buffer; then only edges that are in front of the faces will pass through. The depth buffer is limited in resolution, which leads to the problem of a surface blocking its own edges. To avoid this, the surfaces are recessed back a small distance from the edge. The offset distance between surface and edge needs to be larger than the resolution of the buffer at that depth.

3.6. Propagation Model

Motion measurements from the vehicle can be formed into a model that propagates the particle filter. The uncertainty in the motion model m_t is defined by a Gaussian distribution φ as follows:

$$m_t = \varphi(\sigma^2, \mu). \tag{12}$$

This propagation distribution, φ, is defined by the mean, μ, and variance, σ^2, as follows:

$$\mu = \delta, \tag{13}$$

$$\sigma^2 = \beta\delta + \alpha. \tag{14}$$

The vehicle's wheel encoders and steering encoders form δ as a 2D translation, t_x, t_y, and a rotation around the vertical axis, r_z. The model propagates the particle distribution with the odometry estimate, δ, and perturbs the distribution proportional to the odometry estimate according to the constant β. This constant increases the variance in the distribution as the vehicle's velocity and angular velocity increase and increases the variance in the direction of the velocity.

The ground in the environment is not perfectly flat, and therefore slight vertical translations, t_z, and roll and pitching, r_x, r_y, of the vehicle must be taken into account. No sensors are used to measure these additional degrees of freedom; these unknown degrees of freedom are included in the propagation model by small perturbations across all six degrees of freedom, defined by the constant α.

3.7. Initialization

The initialization process begins by distributing the particles roughly around the approximated vehicle location. The particle filter is then iterated to converge around a pose estimate.

The variance (uncertainty) in the initial distribution is set according to how accurately the vehicle's initial pose is known. If the pose is known only approximately, the variance is set high and more particles are needed to cover the larger search space. The larger number of particles slows the system during this initialization phase. An adaptive particle filter is required, which gradually reduces the number of particles to transition into a faster operating frame rate. Fox (2003) presented a particle filter that adapts the number of samples in the distribution according to the spread of the distribution over the state space. Fox discretized the state space into bins and used the number of occupied bins to set the desired number of samples. In this section a simpler method is developed that does not require discretizing the state space. Here the adaptive filter sets the number of particles according to the variance in the distribution, using the following equation:

$$n_{t+1} = \max\left(n_0 \frac{v_t}{v_0}, n_d\right), \tag{15}$$

where n_{t+1} is the number of particles for the next iteration, n_0 is the initial number of particles, n_d is the desired number of particles after full convergence, v_t is the current translational variance in the particle distribution, and v_0 is the initial variance. The effect of this equation is to reduce the particle count as the filter converges, until the desired number of particles is reached to achieve a processing rate suitable for operation. Admittedly Fox's method of adapting the particle filter will behave more reasonably in the case of multimodal distributions that are widely separated but individually are tight distributions. In that type of distribution the proposed method will wrongly use a large number of particles because a single-mode assumption will calculate a large variance, whereas Fox's method captures the true variance of the multiple modes in the distribution. However, the experiments presented later demonstrate that the proposed adaptive filter still gives desirable results.

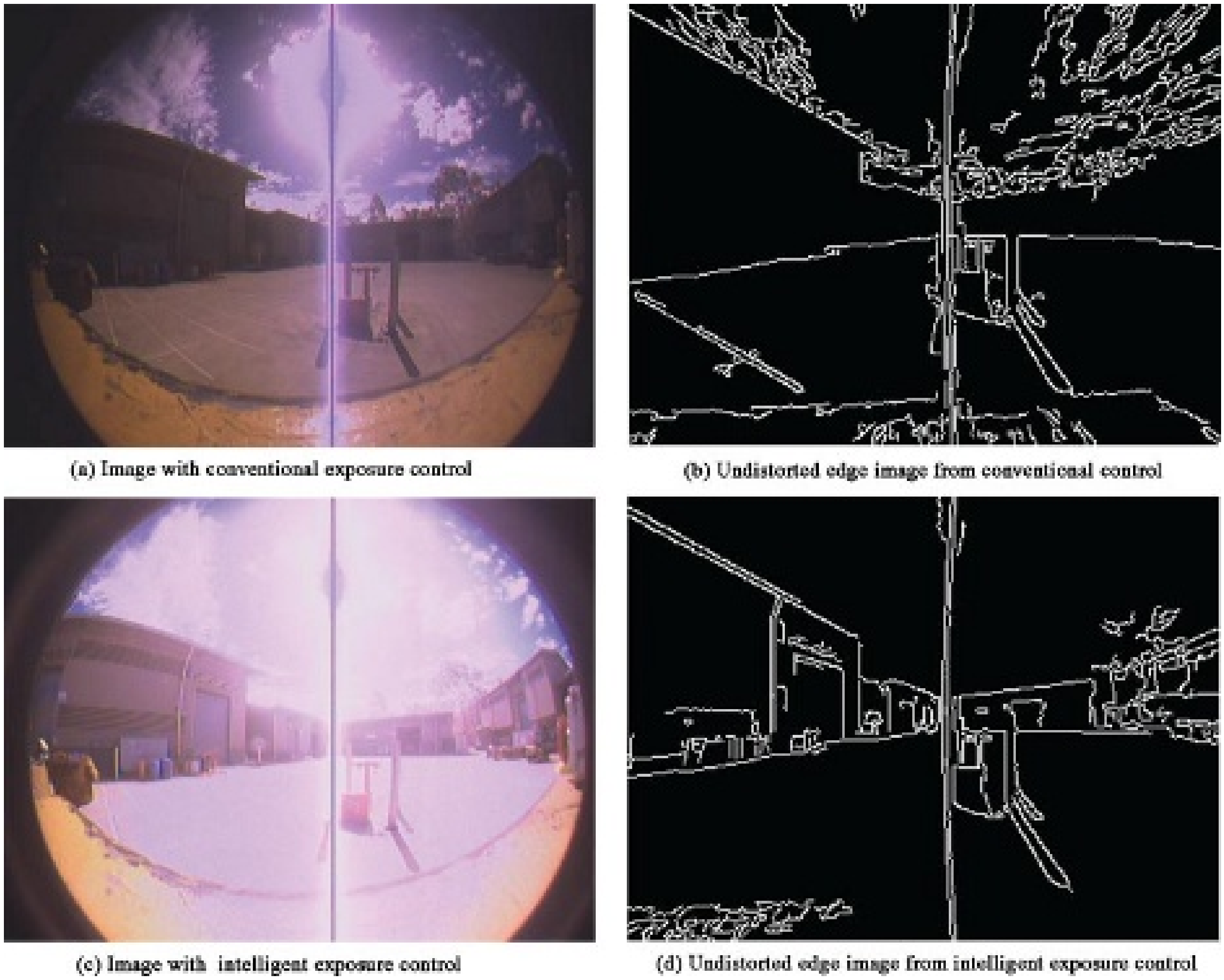
4. INTELLIGENT EXPOSURE CONTROL

The lighting conditions of our application environment are harsh, with the robot vehicle operating in bright sunlight. The nature of the built environment (tall buildings with gaps in between) results in multiple areas of shadow and of full sunlight. The use of fish-eye cameras also results in the sun itself appearing in the images most of the time. This is a challenge for standard cameras in which the built-in exposure control algorithms use a gray-world assumption. These algorithms aim to control the mean intensity value over the whole image to a predefined intensity value, regardless of the content of the scene. The conventional approach to exposure control causes overcorrection, resulting in an image that contains incorrectly exposed areas. An example of overcorrection is shown in Figure 4(a), which shows a lens flare running down the image. But more importantly, Figure 4(b) shows that there is too much correction for the sunny sky with a standard exposure control algorithm, leading to underexposure of the building fronts and no door edges being visible.

Exposure control is a task often undertaken without regard for the specific objects that are in the FOV and is instead based purely on statistical information, such as in Shimizu, Kondo, Kohashi, Tsurata, and Komuro (1992). One example in which exposure control is directed toward specific objects of interest is in the work of Yang, Wu, Crenshaw, Augustine, and Mareachen (2006), in which a face detection algorithm is investigated to find the areas of the image that are used to control exposure.

In our previous work (Nuske et al., 2006) we demonstrated how a standard camera can be used to create a high-dynamic-range image. The technique used multiple exposures (low, medium, and high) that were combined to form a single image that successfully captured image features in dark shadows and full sunlight. This method is particularly applicable to our application in that the resultant high-dynamic-range image is an edge image. However, because this technique requires multiple exposures (at least three) to create an image, the effective frame rate of the imaging system is reduced (by at least a third). Here in this work we show that a single exposure is sufficient when it is intelligently controlled to correctly expose the specific areas of interest in the image.

Owing to the nature of our proposed localization method, specific areas of interest in the scene (that is, the doors and rooflines of the buildings) must be correctly exposed, and because we are effectively tracking these (via the estimated location of the vehicle), we know where in the image the features should lie. We have therefore developed an exposure control algorithm that aims to maximize the strength of image edges corresponding to 3D-map edges, while ignoring all nonessential areas of the image. The algorithm first samples the intensity values of pixels near the tracked edges. [The intensity values are taken from the luminance channel of the YCrCb space after



(a) Image with conventional exposure control

(b) Undistorted edge image from conventional control

(c) Image with intelligent exposure control

(d) Undistorted edge image from intelligent exposure control

Figure 4. Example of the bright lighting conditions. The sun causes a flare in the fish-eye lens and a dark line down the image owing to errors in the sensor's response. Overcompensation for the sunlight can occur using naive exposure control found on most cameras. (a) An example of overcompensation where the buildings are underexposed. (b) The corresponding edge image where no edge features are detected on the doors of the buildings. (c) shows that with the use of the intelligent exposure control algorithm, the buildings are correctly exposed. As a result, the edges are detected on the doors and on the other areas on the buildings in (d).

conversion from the camera's native red–green–blue (RGB) format).] The edges are projected into the image according to the current pose estimate, and short scans of pixels are taken along the normal of the edges. All edges except the roofline edges are used for sampling. The roof edges are ignored to avoid sampling pixels from the bright sky that would otherwise heavily weight the sampling against increasing the exposure of the camera. We assume here that if the doors are correctly exposed, then so too will be the roofline. Figure 5(a) shows which pixels are sampled from the scans. The control algorithm is as follows:

$$\xi_t = \xi_{t-1} + [1.0 - (\epsilon e)], \tag{16}$$

where ξ_t is the exposure level at time t. The IEEE1394 IIDC (1394-Trade-Association, 2000)–compliant digital cameras used in experimentation have two exposure parameters available, an analog-to-digital gain and a digital shutter time. These two parameters are

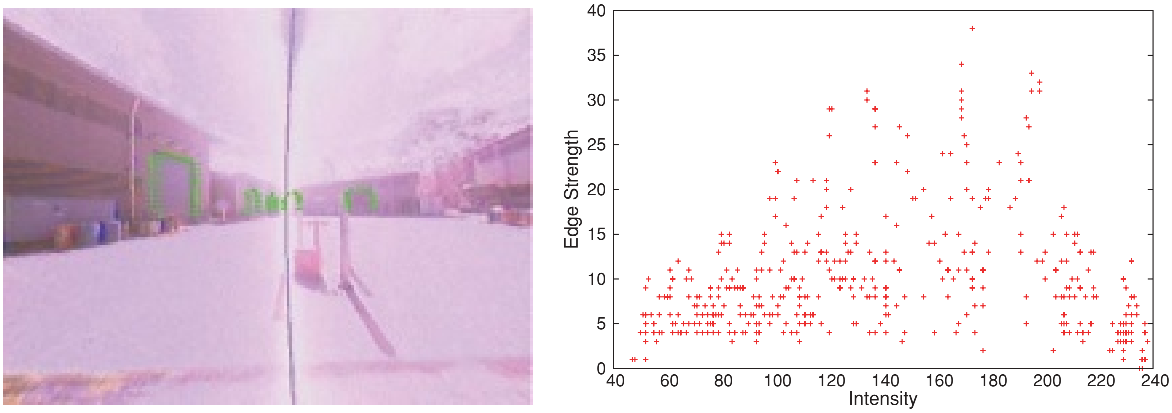


Figure 5. Left: Pixels are sampled along the normal to the tracked door edges. The mean intensity value of these pixels are used as input for the exposure control algorithm. Right: The edge strength sampled from the doors of buildings, plotted against mean 8-bit pixel intensity of the sample. Edge strength is defined as the average intensity difference in a 3 × 3 pixel neighborhood. This graph was recorded over a period of time as the exposure of a stationary camera was incrementally increased.

scaled between 0 and 1 and combined linearly into one parameter ξ. A constant to determine the rate of adjustment is ϵ, and e is the error, calculated as the ratio between the mean intensity value, I_m, from the pixel scans [Figure 5(a)] and the goal intensity value I_d:

$$e = \frac{I_m}{I_d}. \tag{17}$$

Based on the plot seen in Figure 5(b) the edge strength is at a maximum when the mean intensity of the sampled pixels is $I_d = 180$, defined on an 8-bit intensity scale. The damping constant ϵ has been set to 0.02, after empirical tests showed this value to provide a balance between quick response to lighting changes and stable control. This value has not been adjusted since being set and has been used successfully in a wide range of lighting ranging from dark clouds to bright sunlight. Figure 4(c) shows a typical result for this intelligent exposure control algorithm, where the buildings are properly exposed. Edges on the doors and other areas of the buildings are now clearly detected [Figure 4(d)].

5. EXPERIMENTAL SETUP

The vehicle is fitted with two IEEE1394 cameras with fish-eye lenses that are mounted facing sideways on the vehicle [Figure 1(b)]. The intrinsic and extrinsic camera parameters were calculated and verified by projecting the edge model into the image plane using the ground truth and ensuring that the edges were aligned correctly with the recorded video stream. Figure 1 shows the undistorted image using the calibrated fish-eye model. An in-house camera driver was implemented with a double buffer so that the current image being processed is delayed by at most one frame and the image transfer time.

The upper and lower hysteresis edge-detection thresholds for the Canny (1986) edge detector were adjusted manually to the minimal values that still permitted the edges on the buildings to be reliably detected. The threshold values resulting from these empirical tests were 30 and 100.

The number of particles is a crucial parameter and will ideally be large enough to sample a greater portion of the solution space, but this comes with a larger computation cost. We keep the filter, once converged, with 500 particles and at initialization the number of particles is dependent on how well the initial pose is known.

The remaining parameters for the particle filter, the likelihood constants and the motion model parameters, cannot be chosen analytically and require quantitative data for tuning. The approach used to tune these parameters was to record a short sequence of video, odometry, and ground-truth pose data (see the following section for information on ground truth data) from the vehicle traveling around the environment covering most areas and orientations.

Table I. Details of motion model parameters.

Parameter name	t_x	t_y	t_z	r_x	r_y	r_z
α for Klein and Murray	1.5	1.5	0.1	0.01	0.01	0.3
α for Per-edge	1.5	1.5	0.1	0.01	0.01	0.3
α for Nearest-edge	2.0	2.0	0.1	0.01	0.01	0.5
β for Klein and Murray	0.3	0.3	0.0	0.0	0.0	0.5
β for Per-edge	0.3	0.3	0.0	0.0	0.0	0.5
β for Nearest-edge	0.3	0.3	0.0	0.0	0.0	0.5

The particle filter was then run offline several times through the same sequence of recorded data to optimize these parameters. Different parameters were tested each cycle through the data, and the average pose estimate error was recorded. The three different observation functions behave differently, but the parameters need only minor adjustment when switching between the different functions. In fact the parameters are kept the same for the *Klein and Murray* function and the *Per-edge* function, which optimal values were found to be as follows: the motion model parameters, from Eqs. (12) and (14) (one for each pose dimension; $t_x, t_y, t_z, r_x, r_y, r_z$), are calibrated and shown in Table I.

The observation functions constants are shown in Table II, where λ is specifically for the second component of the *Per-edge* function, which behaves well when κ and λ are equal. Noticeably a lot of these parameters are not too different from the other functions' parameters, and in fact adjusting any parameter up or down by 30% does not significantly change the filter's behavior.

To evaluate the visual localization system presented in this paper, we compare its pose estimates to those coming from a laser-scanner localization system described in Tews et al. (2007). Our laser localization system was extensively compared with real time kinematic (RTK)–GPS and shown to give full coverage around the site, whereas GPS was found to experience dropouts and multipath errors in some locations. Given that the laser system is more reliable in terms of coverage and has suitable accuracy in the appropriate areas of the site, it was chosen as the source localization to evaluate the visual localization system.

Table II. Details of observation function parameters.

Parameter/observation function	Value
κ for Klein and Murray	5
κ for Per-edge	5
κ for Nearest-edge	3
λ for Per-edge	5
D_w for Nearest-edge	0.5 m

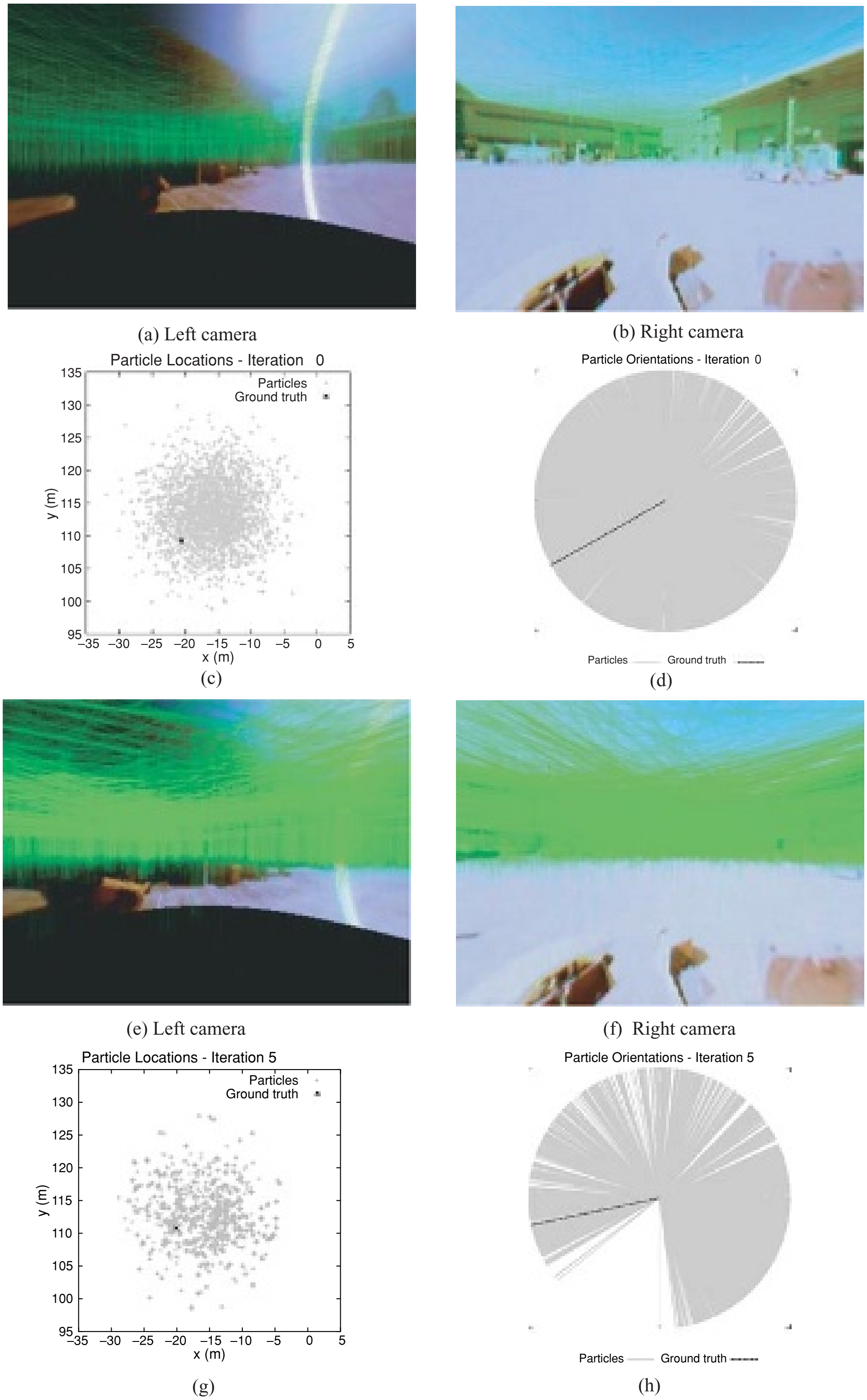
The accuracy of this laser-scanner system varies depending on the density of the surrounding laser-reflecting beacons. In some areas of the site high accuracy is not required, and here the laser beacons are sparsely placed, and hence, the accuracy of the laser-localization system is lower in these areas. In some areas the error is as low as 0.97 m in comparison with RTK–GPS, as reported in Tews et al. (2007). Even though the laser localization system is known to have 0.97-m error in some areas of the site, the area where the experiments are performed in this paper has a dense placement of laser beacons and the laser localization system is much more accurate here. In this area the laser localization system has proven to be a reliable basis for closed-loop control of precise load transfer maneuvers (Roberts et al., 2008). These maneuvers have been performed repeatedly for hours and require accuracy on the order of 100 mm to place the pickup hook inside a small eyelet. Therefore in this area of the site the laser localization system is known to be an appropriate source of ground truth localization to evaluate the proposed visual localization system.

6. INITIALIZATION EXPERIMENT

The experiments conducted on this system are split into two sections; here the initialization of the filter is presented, and the following section presents results of the filter operating for extended periods outdoors. All experiments are conducted in a 70 × 45 m industrial courtyard surrounded by seven buildings, ranging from 6 to 9 m in height (see Figure 2).

The first initialization experiment demonstrates the initialization process of the filter using the adaptive particle filter technique described in Section 3.7. In this example, 4,000 particles were scattered across a 40-m-diameter area and the full 360 deg in the orientation. In other words, the pose of the vehicle is known to lie within a 40-m diameter, and the orientation of the vehicle is completely unknown. The center of the initial distribution is randomly chosen to lie off-center of the correct pose.

Images of the process are shown in Figures 6–8. In this example the *Nearest-edge* observation function is used, and in the following section the three observation functions are compared by running the


(a) Left camera

(b) Right camera

(c)

(d)

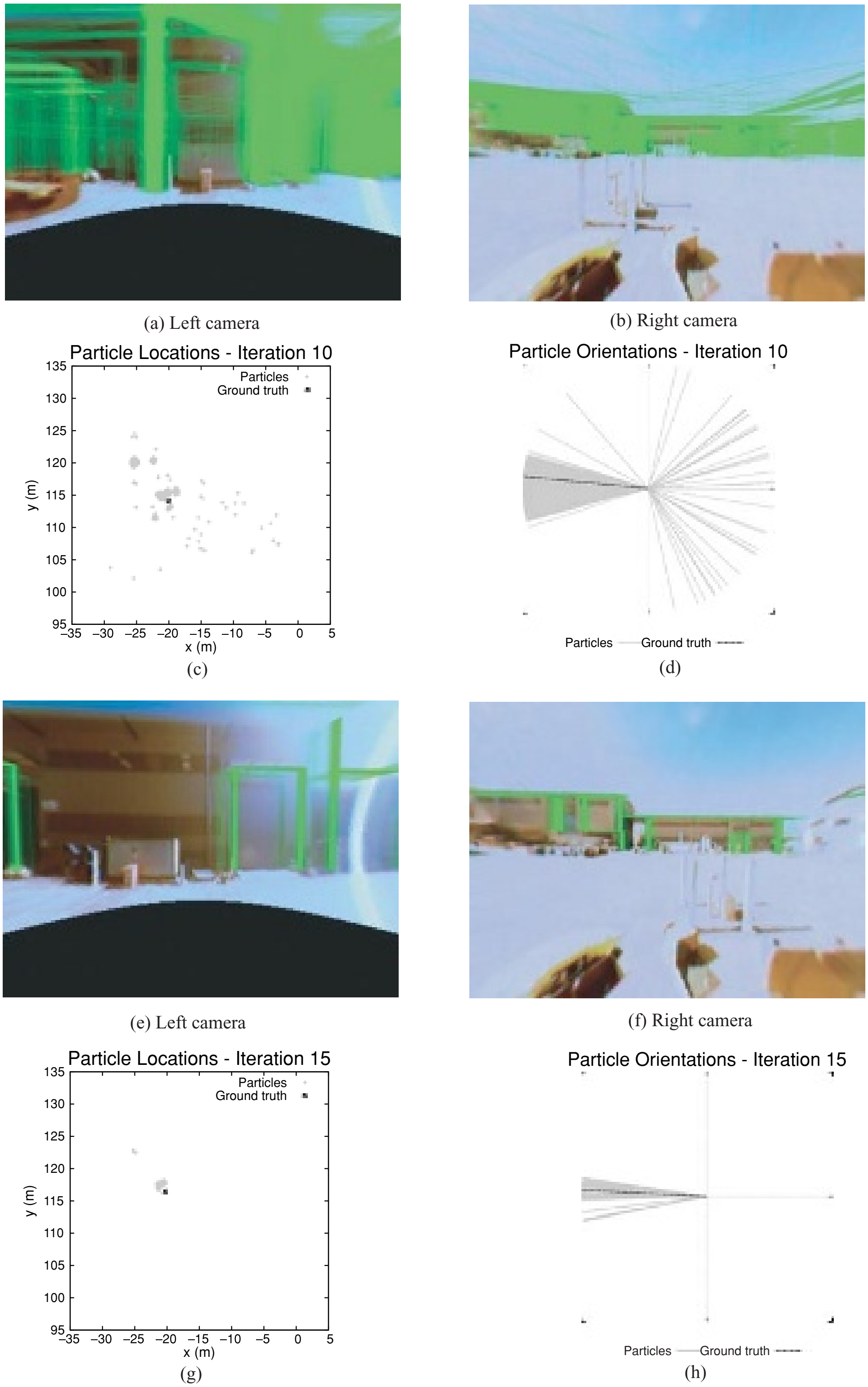
(e) Left camera

(f) Right camera

(g)

(h)

Figure 6. Initialization of the particle filter showing the image overlaid with 3D-edge map, at iterations 0 and 5.



(a) Left camera

(b) Right camera

(c)

(d)

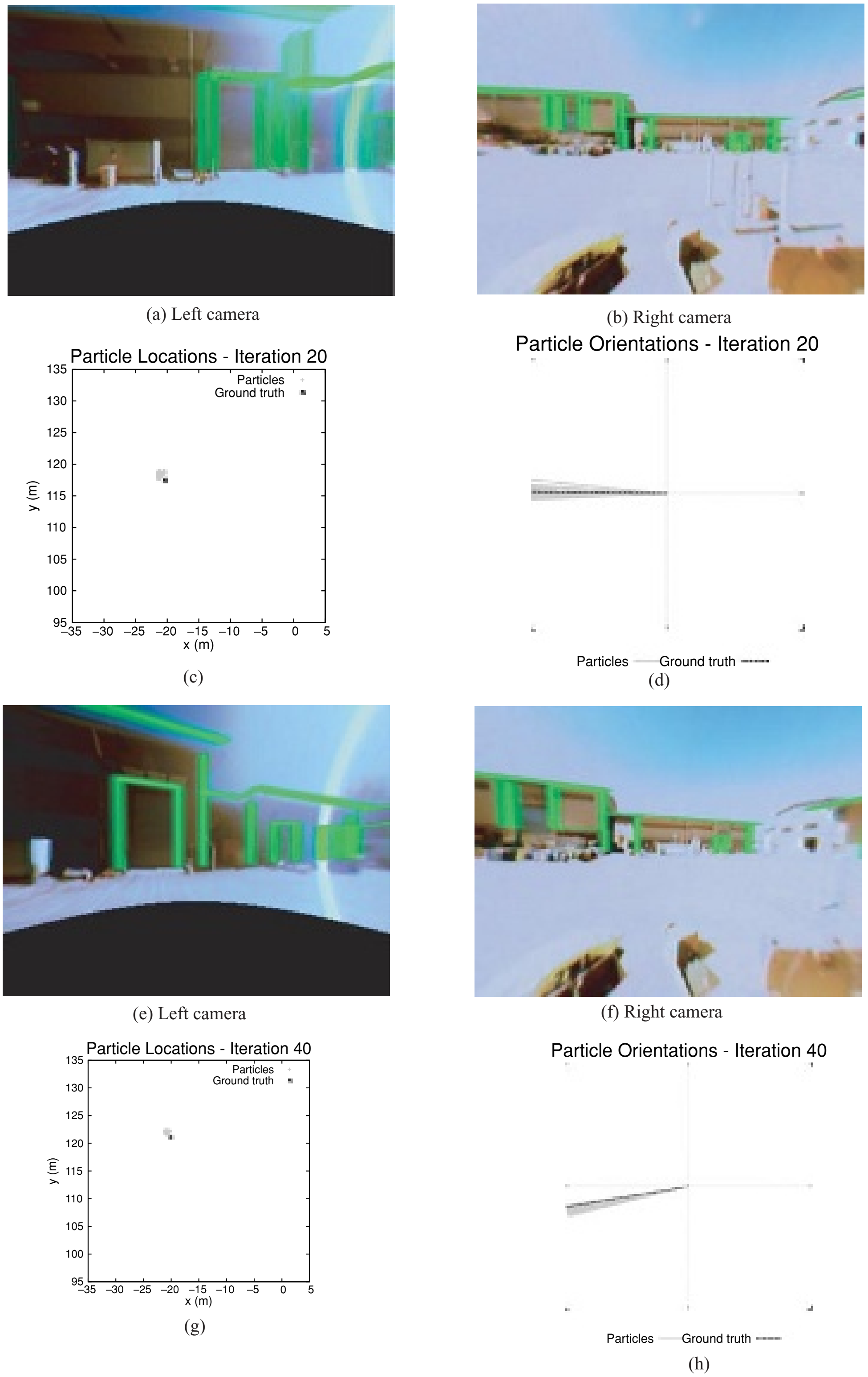
(e) Left camera

(f) Right camera

(g)

(h)

Figure 7. Initialization of the particle filter showing the image overlaid with 3D-edge map, at iterations 10 and 15.



(a) Left camera

(b) Right camera

(c)

(d)

(e) Left camera

(f) Right camera

(g)

(h)

Figure 8. Initialization of the particle filter showing the image overlaid with 3D-edge map, at iterations 20 and 40.

initialization process many times at many locations/orientations. Inspecting the figures, at iteration 10, the particle filter has converged into several distinct distributions, seen in the overlay in Figure 7(c), illustrating that there are several local maxima near the correct pose. After further iteration many of these distributions become downweighted and disappear. Yet later, at iteration 15, there are still two distinct particle distributions visible in Figure 7(g). The ability of the particle filter to maintain multiple estimates is a powerful feature lacking in previous single-hypothesis methods. By iteration 40, the particle filter had converged around one hypothesis, which was centered within 1 m of the ground truth position (the online video attachment named initialisation.mpg shows this initialization process).

The three observation functions described in Section 3.4 (the functions are labeled *Klein and Murray*, *Per-edge*, and *Nearest-edge*) are compared here to evaluate their abilities on convergence.

The proposed *Nearest-edge* function is designed to improve limitations of the other two functions. The other two functions consider alignment only at the projected 3D-edge map of each particle, meaning that a small change in pose will cause a large change in the likelihood measured by these functions. By comparison the *Nearest-edge* function searches outward from the 3D-edge map projected by each particle to find the nearest image edge. In theory this will enable the function to correctly converge at initialization with a large and sparse distribution of particles, whereas the other functions are more likely to converge at incorrect estimates.

To evaluate whether the proposed *Nearest-edge* function does in fact perform better at initialization, an experiment is set up to initialize the particle filter with the three functions 50 times at many locations/orientations over the course of an entire sunny day. The three functions are used to initialize the filter on the same data, and the results are compared against the ground-truth pose given by the laser-scanner system. The initial distribution is the same as the one seen in Figure 6, which is a distribution of 4,000 particles across a 40-m-diameter area and around the full 360 deg in the orientation.

Table III presents the statistics of the error recorded against the ground truth at the point when the particle filter has converged. The error is calculated against the ground-truth pose (laser-scanner system) from the mean pose of the 5% highest weighted particles. Convergence is defined as the iteration when the desired number of particles (in this case 500) is reached in the filter as controlled by the adaptive particle filtering technique described in Section 3.7. The position error is the Euclidean distance of the horizontal translation errors.

The table noticeably shows that the *Nearest-edge* function outperforms the other two in all the statistical indicators of position error. The median and interquartile range of the position error of the *Nearest-edge* function are 0.7 and 1 m, respectively, whereas for *Klein and Murray* they are 3.9 and 6.8 m, and for *Per-edge* they are 1.4 and 1.9 m. The *Nearest-edge* function also outperforms in orientation error, although the *Per-edge* function is more similar in orientation and the *Klein and Murray* function is by far the worst. The median and interquartile range of the orientation errors of the *Nearest-edge* are 0.6 and 1.6 deg, respectively, and for *Per-edge* 1.0 and 2.2 deg and *Klein and Murray* 2.8 and 24.4 deg. The maximum errors are all quite large and are from the cases in which the filter fails to converge and the map is completely misaligned with incorrect edges in the camera image. In these cases the filter can drift unpredictably in the wrong direction and orientation. A success threshold is defined to separate the situations in which the filter fails to converge from the successful convergences. The success threshold is defined at 1 m and 2 deg, which are reasonable bounds of the requirements to successfully commence autonomous control of the vehicle. This threshold makes the success rate of the three functions *Nearest-edge* 71%, *Klein and Murray* 15%, and *Per-edge* 38%.

The time taken for the filter to converge during the experiment is also presented in Table III. The median time and quartile times for convergence of the *Nearest-edge* function and the *Klein and Murray* function are longer than those of the *Per-edge* function, though the difference is not significant. If the filter can remain converged for long periods after initialization, then a delay of up to 20 s at start-up is not a major concern. The improvements gained in the position and orientation estimates and the success rate from the *Nearest-edge* function far outweigh its time penalty.

7. OPERATION EXPERIMENTS

7.1. Bright Sunlight

An experiment was conducted with an extended period of manual operation (30 min) of the vehicle at 2 p.m. on a sunny day. At this time, the lighting

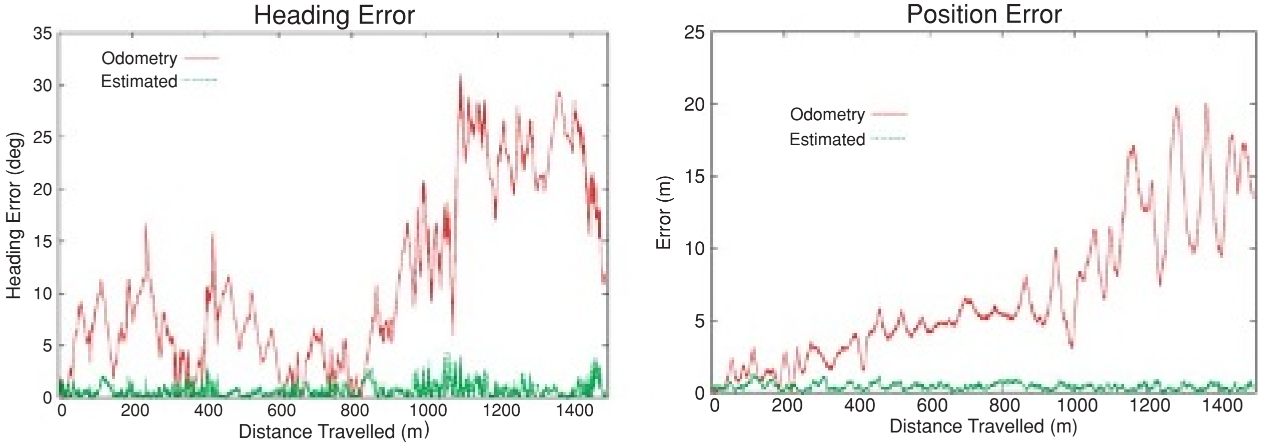
Table III. Comparison of observation functions after convergence at initialization.

		Klein and Murray	Per-edge	Nearest-edge
Position error (m)	Median	3.91	1.41	0.71
	25th Percentile	1.52	0.76	0.39
	75th Percentile	8.33	2.65	1.44
	Max	50.167	26.84	19.29
Rotation error (deg)	Median	2.8	1.0	0.6
	25th Percentile	0.8	0.3	0.3
	75th Percentile	25.2	2.5	1.9
	Max	179.9	90.7	102.9
Convergence time (s)	Median	8.66	6.1	7.9
	25th Percentile	6.46	5.2	6.5
	75th Percentile	10.1	9.1	9.7
	Max	10.6	20	18.8
Success rate (%)		15	38	71

conditions were challenging, because the sun was bright and at an angle in the sky. When the vehicle turned and drove into a shadow, the exposure control algorithm had to adjust quickly according to the direction in which the cameras were facing. The vehicle was driven along an arbitrary path for a total distance of 1.5 km during the experiment. The vehicle traveled through a wide range of positions and orientations, ensuring that the system was well tested.

Figure 9(a) presents the heading estimate error of the vision system that was maintained at an average error of 0.62 deg, as opposed to the accumulated odometry error, which drifted to a maximum error of 30 deg. These errors were recorded from the particle filter using the *Per-edge* observation function. The pose estimate is extracted from the filter as the mean pose of the 5% most highly weighted particles.

Figure 9(b) shows the position error of the visual localization. The vehicle's position was correctly estimated to an average error of 0.44 m over the 1.5-km run. The maximum error at one stage crept out to 1.4 m and 4.4 deg. The average errors indicate that the system is sufficiently accurate for autonomous navigation around the site. Example images from the experiment are shown in Figure 10. (The online video attachment named extended-operation.mpg presents



(a)

(b)

Figure 9. Results from the extended operation of the visual localization system. The position error is Euclidean distance of the horizontal translation errors.

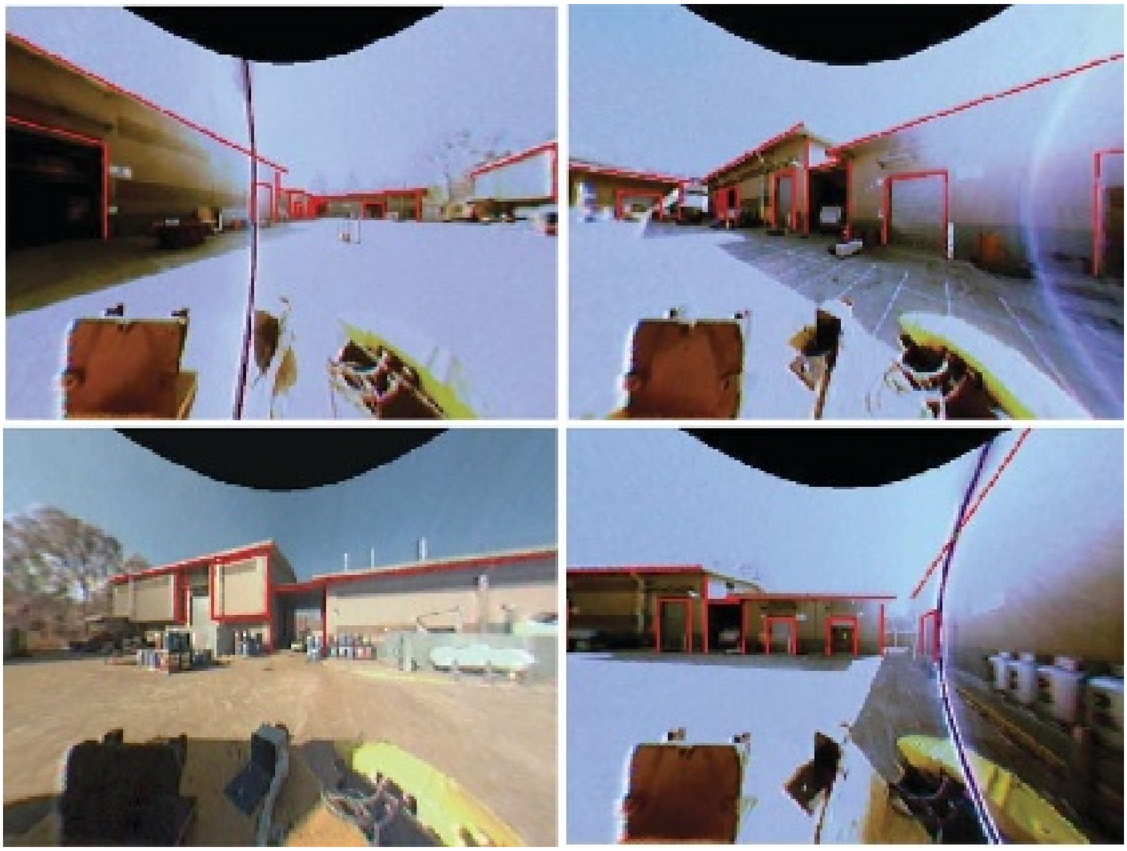


Figure 10. Undistorted images taken from the extended operation experiment, overlaid with the 3D-edge map projected from the estimated pose.

a sped-up sequence from the left camera of this experiment.)

7.2. All-Day Experiment

Here the system is evaluated over the full range of lighting conditions experienced on a bright sunny day. The test consists of a 110-m path that began and ended at the same position and orientation. Two three-point turns were completed during the path, simulating the dropping off and picking up of loads. The path driven by the vehicle is shown in Figure 11. Initially, the path was completed twice just after sunrise at 7 a.m. The path was then repeated twice at the beginning of every hour until just before sunset at 5 p.m. The path was driven manually and only approximately visited the same locations, as seen in the figure.

The two repetitions of the path were completed in approximately 3 min at an average velocity of more than 1 m/s and maximum velocity of 3 m/s. Table IV reports the distance traveled, the overall rotation of the vehicle, and the maximum velocity. Table V presents the error at the end of the path, where the position error is the Euclidean distance of the horizontal translation errors. These errors are those from the filter while it was using the *Nearest-edge* observation function. The visual localizer can be seen to remove the accumulating odometry error. The odometry error on average was approximately 20 m and 25 deg sampled at the end of the path. The visual localizer's position error for 7 of the 11 runs was within 0.5 m, and the maximum error was 1.86 m at 10 a.m. The rotation error for 8 of the 11 runs was within 1 deg, and the maximum error was 3.6 deg recorded again at 10 a.m.

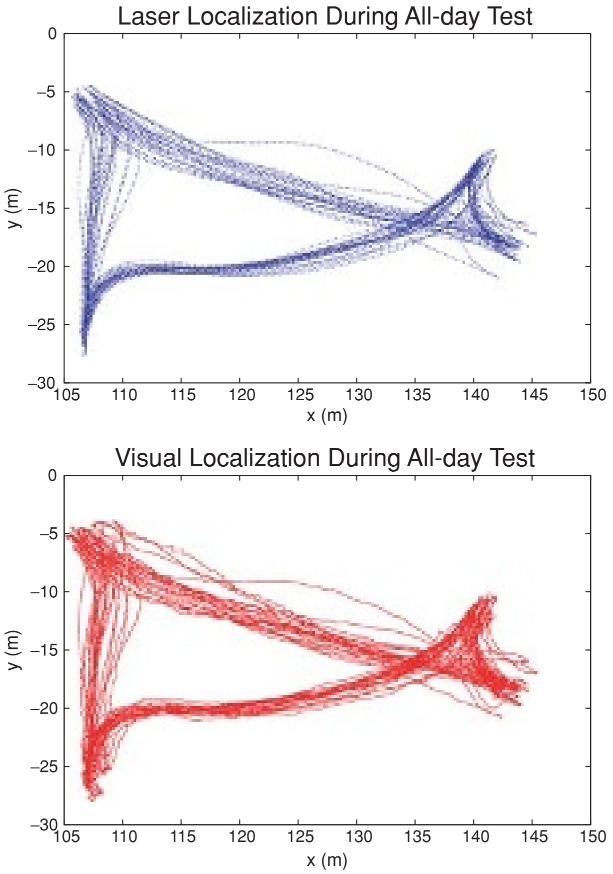


Figure 11. Path traveled every hour during the all-day experiment. The path from both the laser-localization ground-truth system and the visual-localization system (using the *Nearest-edge* observation function) are presented.

Table IV. Details of paths traveled.

Time	Distance traveled (m)	Total rotation (deg)	Max velocity (m/s)
7:00	220	669	2.8
8:00	211	771	2.7
9:00	214	719	2.4
10:00	211	697	3.1
11:00	214	735	3.3
12:00	223	720	3.2
13:00	229	709	2.5
14:00	225	664	2.7
15:00	228	721	3.4
16:00	211	790	2.3
17:00	218	708	2.4
Overall	2,176	7,903	3.4

Table V. Error at the end of path.

	Final position error (m)		Final rotation error (deg)	
Time	Visual localizer	Odometry	Visual localizer	Odometry
7:00	0.41	22.33	0.4	30.3
8:00	1.40	21.25	2.4	27.6
9:00	0.25	19.46	0.9	22.6
10:00	1.86	21.82	3.6	27.1
11:00	0.71	22.47	0.6	31.4
12:00	0.28	21.83	0.3	31.0
13:00	0.64	21.69	0.5	26.3
14:00	0.47	22.19	0.2	26.8
15:00	0.07	26.06	0.2	34.8
16:00	0.31	18.22	1.2	23.5
17:00	0.32	22.93	0.1	29.3
Mean	0.61	21.81	0.9	28.81

Figure 12 presents example images from the experiment. (The online video attachment labeled all-day.mpg is a 10× sped-up movie of the entire all-day experiment including both the left and right video streams.)

The example images show the successful tracking of the buildings across the entire day. The system can maintain track of the buildings even in the early morning and late afternoon when the low angle of the sun in the sky causes large lens flares to block out most of the image. The exposure control algorithm samples pixels within the lens flare, and as a result, door edges not in the flare are left underexposed. The system does not fail in this situation because the roof edges are still visible and, more importantly, the camera facing in the other direction is facing away from the sun.

At 8 a.m, the right camera entered a peculiar state in which the captured images are gray and washed out; this can be seen in the second row and second column of Figure 12. The camera was power-cycled when the vehicle completed the path, and subsequently the images appeared normal. This peculiar camera state did increase the error in the system as seen in Figure 13. However, the increase in error was not drastic and there were other times during the day where the error was similar.

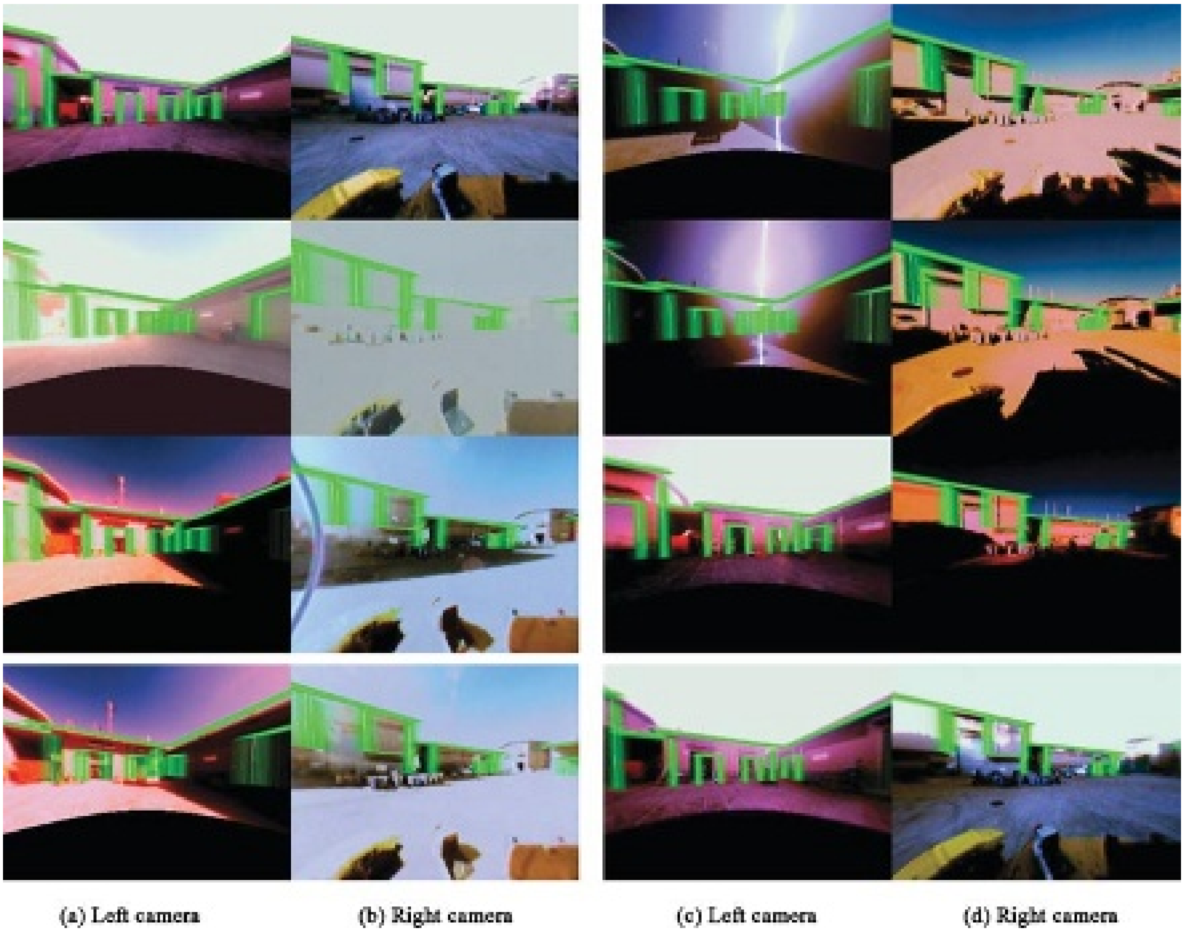


Figure 12. Tracking results from all-day experiment. The two left-hand columns are from the morning hours, and the two right-hand columns are from the afternoon hours. The 3D-edge map is projected from each particle and overlaid on the camera image.

7.3. Comparison of Observation Functions

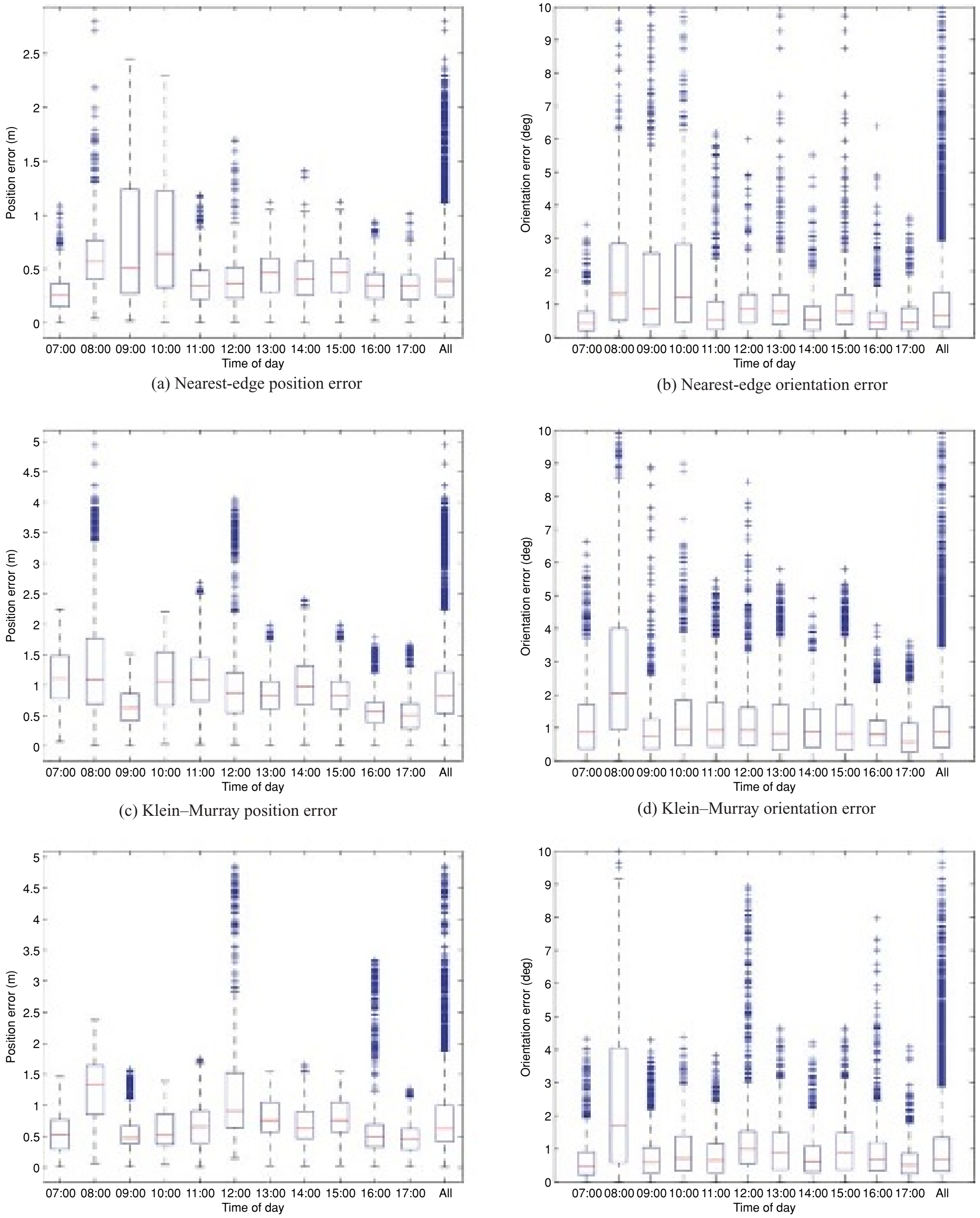
A comparison is made between the three observation functions on the all-day data: the *Per-edge* function [Eq. (7)], the *Nearest-edge* function [Eq. (11)], and the *Klein and Murray* function from Klein and Murray (2006) [Eq. (6)].

The rates at which the three functions can be processed with the 500 particles are listed in Table VI. All timings are recorded on a laptop with a Intel dual-core 2.33-GHz CPU and a NVIDIA Quadro FX 350-M GPU, which is carried onboard the vehicle in the cabin and powered through the vehicle's power supply. The video was captured at 15 Hz, and therefore only Klein and Murray's metric is efficient enough to process the video at the full 15 Hz. The other two functions are slower and process a smaller number of frames. The software implementation of the *Nearest-edge* function is by no means optimal in terms of computation. Aspects of the software implementation that can be optimized are as follows:

- square root calculations, moving from one sample point on the edge to another, could be avoided

Table VI. Comparison of likelihood functions on processing rate.

Observation function	Frame rate (Hz)
Nearest-edge	4.41
Klein and Murray	15.27
Per-edge	8.33



(a) Nearest-edge position error

(b) Nearest-edge orientation error

(c) Klein–Murray position error

(d) Klein–Murray orientation error

(e) Per-edge position error

(e) Per-edge orientation error

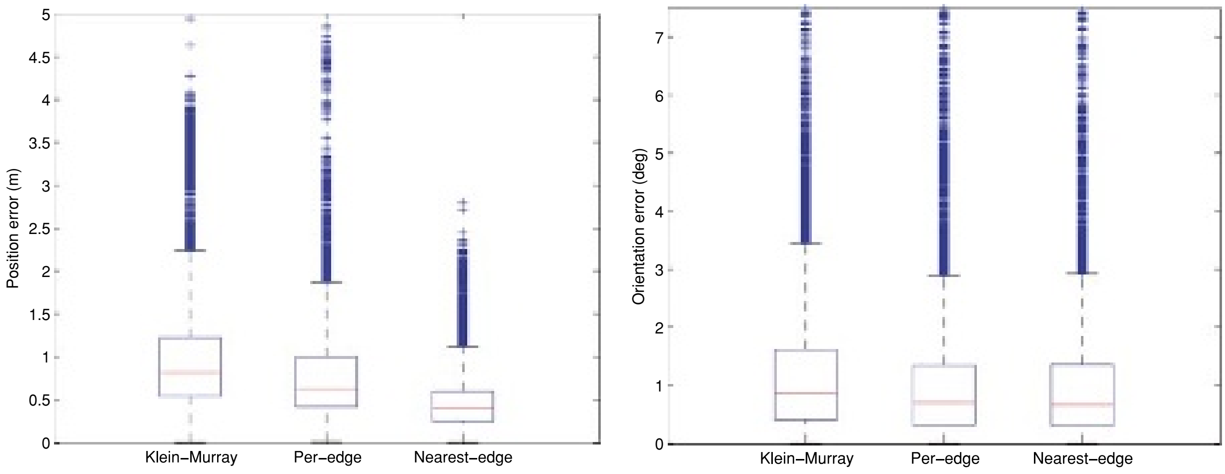
Figure 13. Box plots of the three different observation functions during the all-day test. The position error is the Euclidean distance in the horizontal translation errors. The boxes in the plots represent the interquartile range, the whiskers the minimum and maximum values, the lines inside the boxes the median, and the crosses outside the whiskers the outliers, which are values more than 1.5 times outside the interquartile range.

- matrix multiplications, which project the 3D-edge points onto the image plane, could be optimized
- calls to access the pixels in the edge image could be more efficient
- conversions between normalized image coordinates and image pixel coordinates could be avoided

The three different metrics were each evaluated against the laser-based localizer on the same logged sensor data from the all-day test. The statistics of the recorded errors over the whole day are shown in the box plots in Figure 13, and the statistics of the individual times are shown in Figure 14. The error is recorded at each iteration, and during the entire day there were totals of 29,378, 15,633, and 7,369 iterations for the *Klein–Murray*, *Per-edge*, and *Nearest-edge* versions of the system, respectively. The position error is the Euclidean distance of the horizontal translation errors.

In a previous paper (Nuske et al., 2008) the *Per-edge* function was shown to be better at orientation estimation than Klein and Murray's original function from Klein and Murray (2006). This is because the *Per-edge* function enables better tracking of the smaller door edges and, therefore, better orientation estimation. However, in the experiments presented here, the orientation estimation of the two functions was more comparable. In earlier reported experiments (Nuske et al., 2008), in which the *Per-edge* function performed better, both functions were operated in an offline manner, one frame per iteration, essentially being processed at the same frame rate. Here the experiments were run in an online manner, where if the function operated slower than the frame rate, then frames were dropped. The Klein and Murray function is more efficient, processing twice the number of frames than are processed by the *Per-edge* function, and it is thought that this is why the orientation estimation is more comparable, as seen in Figure 14. Here the median orientation error over the whole day for the *Klein–Murray* function is 0.8 deg and for the *Per-edge* is 0.65 deg.

The observation function proposed in this paper, the *Nearest-edge* function, was by far the better of the three at initializing from particle distributions with large uncertainty, as seen in Section 6. The improvements of the *Nearest-edge* function in pose estimation during operation are not quite as profound. After



(a) Position error

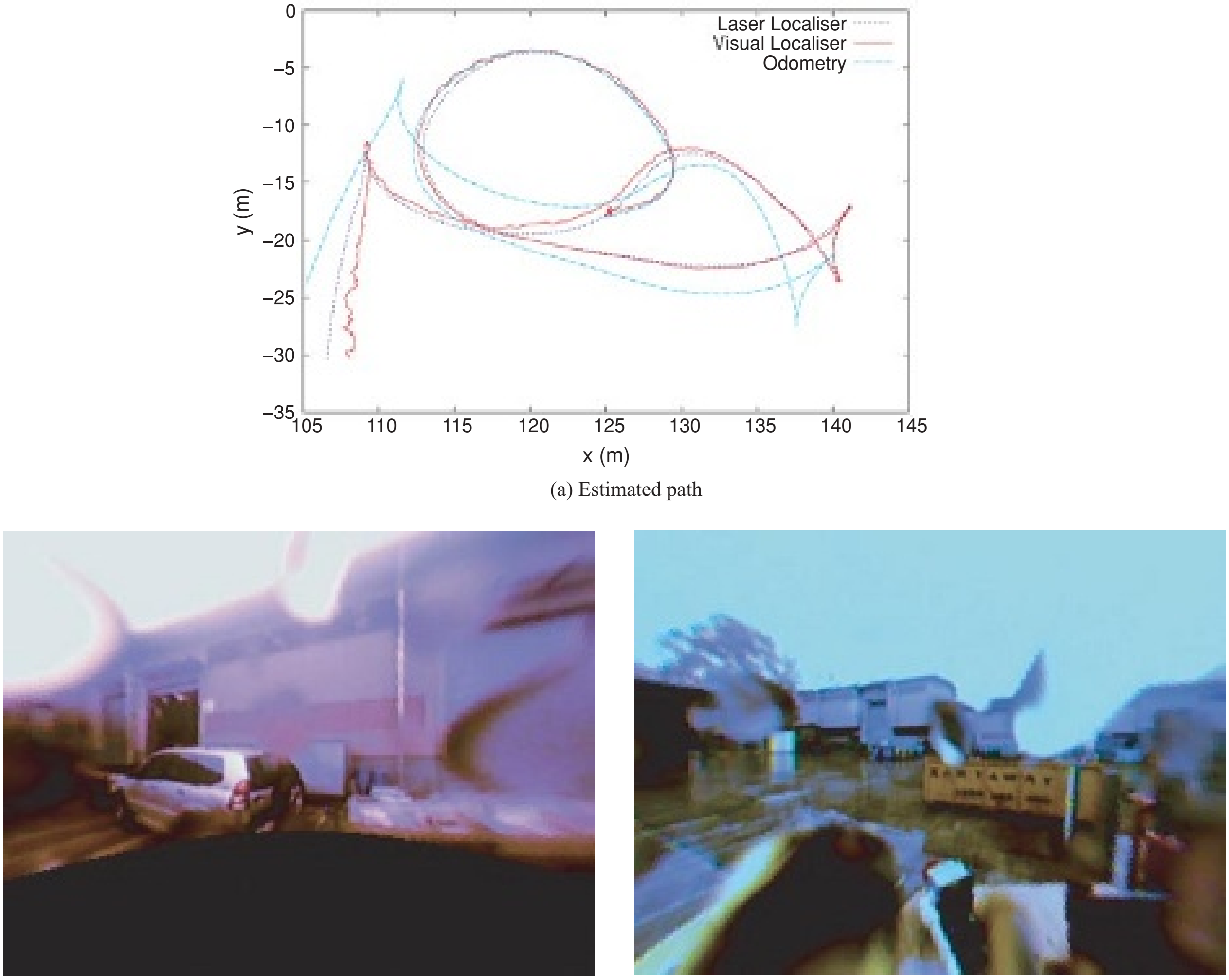
(b) Orientation error

Figure 14. Box plots of the pose estimation errors of the different times during the all-day test of the three different observation functions. The position error is the Euclidean distance of the translation errors. The boxes in the plots represent the interquartile range, the whiskers the minimum and maximum values, the lines inside the boxes the median, and the crosses outside the whiskers the outliers, which are values more than 1.5 times outside the interquartile range. The error is calculated against the ground-truth pose (laser-scanner system) from the mean pose of the 5% highest weighted particles at each iteration of the filter. Some of the outliers are cropped off this figure in order to zoom in on the interquartile ranges, though all the outliers are visible in Figure 13.

inspecting Figures 13 and 14, there are noticeable improvements in using the *Nearest-edge* function, especially in position error, where the median error is 0.45 m as contrasted to 0.7 and 0.8 m of the *Per-edge* and *Klein–Murray* functions, respectively. The orientation error of the three is similar when the median errors of the three are between 0.6 and 0.8 deg. Importantly, the median pose estimation errors of the three functions are all sufficient to be used as a basis of navigating the vehicle around the site.

The system is not accurate enough to repeatedly perform precise load pickups. However, another vision system (using a camera pointing toward the load-carrying point of the vehicle) has been developed specifically for this task (Pradalier et al., 2008), which can provide the level of accuracy required to pick up a load.

The outliers seen in Figures 13 and 14 drift to several meters and several degrees of error, which is a concern for a vehicle navigating near buildings. These errors occur for short periods on the order of 5–10 s when one of two things occurs with the filter's particle distribution: either the distribution drifts into and out of a local maxima located away



(a) Estimated path

(b) Left camera

(c) Right camera

Figure 15. Estimated path traveled during the rain experiment and example camera images. Raindrops on the lenses are noticeably disrupting the camera's view.

from the correct pose, or the filter forms multimodal distributions—such as the one seen in Figures 7(c) and 7(g)—and the filter's pose estimate (mean of the 5% highest weighted particles) switches between the distinct distributions. Most often these situations happen during sharp turns and the distribution returns to the correct pose estimate once the turn is complete, as reflected by the relatively low median errors and also reflected in the errors at the end of the path displayed in Table V. An obvious solution to this issue is a more accurate odometry source that can better propagate the filter during quick turns, which will be investigated in future work.

7.4. Rain Experiment

In addition to the difficulties of direct sunlight, vision systems operating outdoors also face the problems of rainy weather. Here is a brief experiment showing that it is possible to operate the proposed system even with raindrops sitting on the lenses of the cameras, disrupting the view. The vehicle is driven a path around the industrial courtyard while it is raining.

The estimated path can be seen in Figure 15(a), and the errors recorded against the ground truth can be seen in Table VII. The median position error is 0.5 m, and the maximum error is 1.7 m. The median orientation error is 0.7 deg, and the maximum error 4.7 deg recorded during a turn. These errors are similar in comparison to those of the experiments in clear conditions presented in earlier sections and are promising for the possibility of operating the visual-localization system in a wide range of outdoor conditions. Example camera images are seen in Figure 15, where the raindrops on the lenses can be seen to dramatically obscure the view. (An online video attachment of this test is named rain.mp4.)

Table VII. Errors recorded during the rain test.

		Visual localizer	Odometry
Position error (m)	Median	0.45	1.54
	25th Percentile	0.29	0.33
	75th Percentile	0.63	4.80
	Max	1.71	6.73
Rotation error (deg)	Median	0.71	10.8
	25th Percentile	0.32	2.2
	75th Percentile	1.23	16.2
	Max	4.79	22.3

8. CONCLUSION

This paper has investigated the use of vision-based localization for a ground vehicle operating in an outdoor industrial setting. The system developed here fits into our project's overarching plan of utilizing several independent localization systems based on different physical properties, enabling a level of redundancy, confidence, and robustness in localization required for truly long-term autonomous operation.

The visual localization system uses a manually surveyed 3D-edge map of the permanent buildings in the environment. The sparse 3D-edge map is not specific to any lighting condition and includes only permanent information. Two sideways-facing fisheye cameras are calibrated and used to project the 3D-edge map onto the image plane for comparison with the detected edges in the camera image. A particle filter is used to localize the vehicle and is proven reliable for initialization given a very coarse initial pose estimate and also for extended operation in changing outdoor lighting conditions.

The first experiment of the localization system evaluated the ability of the system to initialize given only a coarse estimate of the vehicle's location and no indication of the vehicle's initial orientation. The experiment provided the system 50 different opportunities to initialize the vehicle from a range of poses and across many different times during a sunny day. A novel observation function was compared to two existing functions on the 50 different examples and performed far better at initializing the filter. Not only was there less error in the filter's converged estimate but also the function was far more successful at converging.

Three experiments were conducted to evaluate the performance of the localization system in challenging outdoor conditions. One demonstrated the system on a vehicle while it was driven around a test site for an extended period during which the vehicle covered a total distance of 1.5 km. The pose estimates from the vision-based localizer were compared to a laser-based localizer, which acted as ground truth. The pose of the vehicle was estimated by the visual localizer to an average position error of 0.44 m and average rotation error of 0.62 deg over the 1.5-km path. The second experiment was carried out over the period of an entire day. At every hour during the day, the localization system was evaluated as the vehicle was driven along a 220-m path. The localization system was evaluated and shown to successfully localize the vehicle in all the lighting conditions over the whole day. The final experiment evaluated the system operating in rainy weather with raindrops sitting on the lenses obscuring the view.

In addition to the particle filter, an intelligent exposure control algorithm was presented that enabled operation in the highly nonuniform outdoor lighting conditions. The algorithm developed uses knowledge of the scene to adjust the camera exposure and hence improve the quality of the important information in the image.

In conclusion, the combination of a sparse but high-quality invariant map, a robust localization algorithm, and an intelligent exposure control algorithm all combined to produce dependable visual localization in difficult outdoor lighting conditions.

8.1. Intelligent Exposure Control

The intelligent exposure control developed significantly contributed to the positive results achieved by the system. There are, however, situations in which the intelligent exposure control algorithm does not control to a suitable exposure level. These are situations when a lens flare covers parts of the buildings. The algorithm then samples pixels from the bright flare and undercorrects the exposure, leaving parts of the buildings underexposed that are not in the flare. Figure 16 presents an example of this problem. This situation, which arose many times during the all-day test, did not cause irreversible failures in the localization system because there are still visible roof edges that can still be tracked, in addition to the second camera facing in the opposite direction to the sun.

The exposure control algorithm should be improved to better deal with this situation, by recognizing which areas of the image have the lens flares and not sampling the pixels from these areas. Because the time of day is known and the pose of the vehicle is known, the location of the sun in the sky (and hence in the image) could be calculated. This estimated sun position could then be used to calculate where lens flares will occur and hence mask them out. Another improvement would be to physically block lens flares using a physical light shade that could be automatically moved between the sun and the lens (similar to the shades used in cars by humans when driving in the early morning or late afternoon).

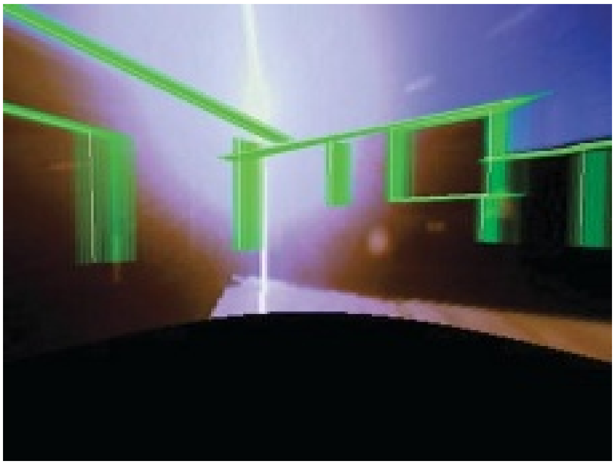


Figure 16. An example image of a limitation of the intelligent exposure control algorithm. Pixels are sampled for the control algorithm that are inside the lens flare, leaving parts of the buildings either side of the lens underexposed. These parts either side of the flare could be visible with a higher exposure.

8.2. Better Cameras

The model of camera used in this work was a consumer-grade product, and its sensor will obviously not perform as well as the sensor of a higher quality camera. Future work should therefore include the rerunning of the experiments using higher quality cameras and higher resolution images.

8.3. Night Operation

The application areas of interest, that of the movement of materials around an industrial setting, often demand 24-h operation and hence operation at night. It is desirable to have a localization system that can therefore work in near darkness as well as in full sunshine. Future work will therefore include the evaluation of the system with the addition of IR illumination and standard vehicle lighting systems.

9. RESULTS VIDEOS

Many of the experimental results of this paper are best presented in video format. Table VIII lists the videos that accompany this paper online at http://www.cat.csiro.au/ict/download/nuske/.

Table VIII. Table of video files.

File	Description
initialisation.mp4	Example showing the initialization process described in Section 6.
extended-operation.mpg	Sped-up video showing localization from the 30-min extended operation experiment presented in Section 7.1. Localized 3D-edge map is projected in red.
allday.mpg	Sped-up video showing the localization of the vehicle every hour from 7 a.m. to 5 p.m., described in Section 7.2. Images from both left- and right-facing cameras are shown. The 3D-edge map is projected from each particle in green, and the mean pose of the top 5% most highly weighted particles is projected in white.
rain.mp4	Video showing the visual localization system operating in rainy weather.

ACKNOWLEDGMENTS

This work was a part of Stephen Nuske's Ph.D. research that was funded equally by the CSIRO ICT Centre and the School of Information Technology and Electrical Engineering at the University of Queensland. The material resources required for this work were provided by CSIRO's Light Metals Flagship Autonomous Hot Metal Carrier Project. The authors gratefully acknowledge the following members of the CSIRO ICT Centre's Autonomous Systems Lab team for their assistance with the setup, calibration, and experimental work conducted during this project: Cédric Pradalier, Ashley Tews, Peter Hansen, Paul Flick, Polly Alexander, Felix Duvallet, and Felix Ruess.

REFERENCES

1394-Trade-Association (2000). IIDC 1394-based digital camera specification, 1.30 ed. 1394-Trade-Association, Santa Clara, CA.

Canny, J. (1986). A computational approach to edge detection. IEEE Transactions on Pattern Analysis and Machine Intelligence, 8(6), 679–698.

Cummins, M., & Newman, P. (2008). FAB-MAP: Probabilistic localization and mapping in the space of appearance. International Journal of Robotics Research, 27(6), 647–665.

Davison, A. J., & Molton, N. D. (2007). MonoSLAM: Real-time single camera SLAM. IEEE Transactions on Pattern Analysis and Machine Intelligence, 29(6), 1052–1067.

Drummond, T., & Cipolla, R. (2002). Real-time visual tracking of complex structures. IEEE Transactions on Pattern Analysis and Machine Intelligence, 24(7), 932–946.

Fox, D. (2003). Adapting the sample size in particle filters through KLD-sampling. International Journal of Robotics Research, 22, 985–1005.

Georgiev, A., & Allen, P. (2002, September). Vision for mobile robot localization in urban environments. In IEEE/RSJ International Conference on Intelligent Robots and Systems, 2002, Lausanne, Switzerland (vol. 1, pp. 472–477).

Geyer, C., & Danilidis, K. (2001). Catadioptric projective geometry. International Journal of Computer Vision, 45(3), 223–243.

Klein, G., & Murray, D. (2006, September). Full-3D edge tracking with a particle filter. In British Machine Vision Conference, Edinburgh, UK.

Kosaka, A., & Kak, A. (1992, July). Fast vision-guided mobile robot navigation using model-based reasoning and prediction of uncertainties. In Proceedings International Conference on Intelligent Robots and Systems, Raleigh, NC.

Lowe, D. (2004). Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision, 60(2), 91–110.

Lowe, D., Se, S., & Little, J. (2002). Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks. International Journal of Robotics Research, 21(8), 735–758.

Maimone, M., Cheng, Y., & Matthies, L. (2007). Two years of visual odometry on the Mars exploration rovers. Journal of Field Robotics, 24(3), 169–186.

Marks, T., Howard, A., Bajracharya, M., Cottrell, G., & Matthies, L. (2008, May). Gamma-SLAM: Using stereo vision and variance grid maps for SLAM in unstructured environments. In IEEE International Conference on Robotics and Automation, 2008. ICRA 2008, Pasadena, CA (pp. 3717–3724).

Michel, P., Chestnut, J., Kagami, S., Nishiwaki, K., Kuffner, J., & Kanade, T. (2007, October). GPU-accelerated real-time 3D tracking for humanoid locomotion and stair climbing. In IEEE/RSJ International Conference on Intelligent Robots and Systems, 2007. IROS 2007, San Diego, CA (pp. 463–469).

Mikolajczyk, K., & Schmid, C. (2005). A performance evaluation of local descriptors. IEEE Transactions on Pattern Analysis and Machine Intelligence, 27(10), 1615–1630.

Nistér, D., Naroditsky, O., & Bergen, J. (2006). Visual odometry for ground vehicle applications. Journal of Field Robotics, 23, 3–20.

Nuske, S., Roberts, J., & Wyeth, G. (2006, May). Extending the dynamic range of robotic vision. In Proceedings of the IEEE International Conference on Robotics and Automation, Orlando, FL (pp. 162–167).

Nuske, S., Roberts, J., & Wyeth, G. (2008, May). Outdoor visual localisation in industrial building environments. In Proceedings of the IEEE International Conference on Robotics and Automation, Pasadena, CA.

NVIDIA Corporation (2007). NVIDIA OpenGL Extension Specifications. Santa Clara, CA: NVIDIA Corporation.

Paz, L., Pinies, P., Tardos, J., & Neira, J. (2008). Large-scale 6-DOF SLAM with stereo-in-hand. IEEE Transactions on Robotics, 24(5), 946–957.

Pradalier, C., Tews, A., & Roberts, J. (2008). Vision-based operations of a large industrial vehicle: Autonomous hot metal carrier. Journal of Field Robotics, 25(4–5), 243–267.

Reitmayr, G., & Drummond, T. (2006, October). Going out: Robust model-based tracking for outdoor augmented reality. In International Symposium on Mixed and Augmented Reality, Santa Barbara, CA (pp. 109–118).

Roberts, J., Tews, A., & Nuske, S. (2008, May). Redundant sensing for localisation in outdoor industrial environments. In Proceedings of the 6th IARP/IEEE-RAS/EURON Workshop on Technical Challenges for Dependable Robots in Human Environments, Pasadena, CA.

Roberts, J., Tews, A., Pradalier, C., & Usher, K. (2007). Autonomous hot metal carrier—Navigation and manipulation with a 20 tonne industrial vehicle. In Proceedings of IEEE International Conference on Robotics and Automation, Rome, Italy (pp. 2770–2771, video paper).

Shimizu, S., Kondo, T., Kohashi, T., Tsurata, M., & Komuro, T. (1992). A new algorithm for exposure control based on fuzzy logic for video cameras. IEEE Transactions on Consumer Electronics, 38(3), 617–623.

Sim, R., & Dudek, G. (2003, August). Comparing image-based localization methods. In International Joint Conference on Artificial Intelligence, Acapulco, Mexico.

Tews, A., Pradalier, C., & Roberts, J. (2007). Autonomous hot metal carrier. In Proceedings of IEEE International Conference on Robotics and Automation, Rome, Italy (pp. 1176–1182).

Thrun, S., Burgard, W., & Fox, D. (2005). Probabalistic robotics. Cambridge, MA: MIT Press.

Valgren, C., & Lilienthal, A. (2007, September). SIFT, SURF and seasons: Long-term outdoor localization using local features. In European Conference on Mobile Robots, Freiburg, Germany.

Valgren, C., & Lilienthal, A. (2008, May). Incremental spectral clustering and seasons: Appearance-based localization in outdoor environments. In IEEE International Conference on Robotics and Automation, Pasadena, CA.

Yang, M., Wu, Y., Crenshaw, J., Augustine, B., & Mareachen, R. (2006). Face detection for automatic exposure control in handheld camera. In IEEE International Conference on Computer Vision Systems, 2006 ICVS '06 (pp. 17–17).

Ying, X., & Hu, Z. (2004). Can we consider central catadioptric cameras and fisheye cameras within a unified imaging model? In Computer Vision—ECCV 2004 (vol. 3021/2004, Lecture Notes in Computer Science, pp. 442–455). Berlin: Springer.