A Visual-Sensor Model for Mobile Robot Localisation

Matthias Fichtner Axel Großmann

Artificial Intelligence Institute
Department of Computer Science
Technische Universität Dresden

Technical Report WV-03-03/CL-2003-02

Abstract

We present a probabilistic sensor model for camera-pose estimation in hallways and cluttered office environments. The model is based on the comparison of features obtained from a given 3D geometrical model of the environment with features present in the camera image. The techniques involved are simpler than state-of-the-art photogrammetric approaches. This allows the model to be used in probabilistic robot localisation methods. Moreover, it is very well suited for sensor fusion. The sensor model has been used with Monte-Carlo localisation to track the position of a mobile robot in a hallway navigation task. Empirical results are presented for this application.

1 Introduction

The problem of accurate localisation is fundamental to mobile robotics. To solve complex tasks successfully, an autonomous mobile robot has to estimate its current pose correctly and reliably. The choice of the localisation method generally depends on the kind and number of sensors, the prior knowledge about the operating environment, and the computing resources available. Recently, vision-based navigation techniques have become increasingly popular [3]. Among the techniques for indoor robots, we can distinguish methods that were developed in the field of photogrammetry and computer vision, and methods that have their origin in AI robotics.

An important technical contribution to the development of vision-based navigation techniques was the work by [10] on the recognition of 3D-objects from unknown viewpoints in single images using scale-invariant features. Later,

this technique was extended to global localisation and simultaneous map building [11].

The FINALE system [8] performed position tracking by using a geometrical model of the environment and a statistical model of uncertainty in the robot's pose given the commanded motion. The robot's position is represented by a Gaussian distribution and updated by Kalman filtering. The search for corresponding features in camera image and world model is optimised by projecting the pose uncertainty into the camera image.

Monte Carlo localisation (MCL) based on the condensation algorithm has been applied successfully to tour-guide robots [1]. This vision-based Bayesian filtering technique uses a sampling-based density representation. In contrast to FINALE, it can represent multi-modal probability distributions. Given a visual map of the ceiling, it localises the robot globally using a scalar brightness measure. [4] presented a vision-based MCL approach that combines visual distance features and visual landmarks in a RoboCup application. As their approach depends on artificial landmarks, it is not applicable in office environments.

The aim of our work is to develop a probabilistic sensor model for camera-pose estimation. Given a 3D geometrical map of the environment, we want to find an approximate measure of the probability that the current camera image has been obtained at a certain place in the robot's operating environment. We use this sensor model with MCL to track the position of a mobile robot navigating in a hallway. Possibly, it can be used also for localisation in cluttered office environments and for shape-based object detection.

On the one hand, we combine photogrammetric techniques for map-based feature projection with the flexibility and robustness of MCL, such as the capability to deal with localisation ambiguities. On the other hand, the feature matching operation should be sufficiently fast to allow sensor fusion. In addition to the visual input, we want to use the distance readings obtained from sonars and laser to improve localisation accuracy.

The paper is organised as follows. In Section 2, we discuss previous work. In Section 3, we describe the components of the visual sensor model. In Section 4, we present experimental results for position tracking using MCL. We conclude in Section 5.

2 Related Work

In classical approaches to model-based pose determination, we can distinguish two interrelated problems. The correspondence problem is concerned with finding pairs of corresponding model and image features. Before this mapping takes place, the model features are generated from the world model

using a given camera pose. Features are said to match if they are located close to each other. Whereas the pose problem consists of finding the 3D camera coordinates with respect to the origin of the world model given the pairs of corresponding features [2]. Apparently, the one problem requires the other to be solved beforehand, which renders any solution to the coupled problem very difficult [6].

The classical solution to the problem above follows a hypothesise-and-test approach:

(1) Given a camera pose estimate, groups of best matching feature pairs provide initial guesses (hypotheses).

(2) For each hypothesis, an estimate of the relative camera pose is computed by minimising a given error function defined over the associated feature pairs.

(3) Now as there is a more accurate pose estimate available for each hypothesis, the remaining model features are projected onto the image using the associated camera pose. The quality of the match is evaluated using a suitable error function, yielding a ranking among all hypotheses.

(4) The highest-ranking hypothesis is selected.

Note that the correspondence problem is addressed by steps (1) and (3), and the pose problem by (2) and (4).

The performance of the algorithm will depend on the type of features used, e.g., edges, line segments, or colour, and the choice of the similarity measure between image and model, here referred to as error function. Line segments is the feature type of our choice as they can be detected comparatively reliably under changing illumination conditions. As world model, we use a wire-frame model of the operating environment, represented in VRML. The design of a suitable similarity measure is far more difficult.

In principle, the error function is based on the differences in orientation between corresponding line segments in image and model, their distance and difference in length, in order of decreasing importance, in consideration of all feature pairs present. This has been established in the following three common measures [10]. e_{3D} is defined by the sum of distances between model line endpoints and the corresponding plane given by camera origin and image line. This measure strongly depends on the distance to the camera due to back-projection. $e_{2D,1}$, referred to as infinite image lines, is the sum over the perpendicular distances of projected model line endpoints to corresponding, infinitely extended lines in the image plane. The dual measure, $e_{2D,2}$, referred to as infinite model lines, is the sum over all distances

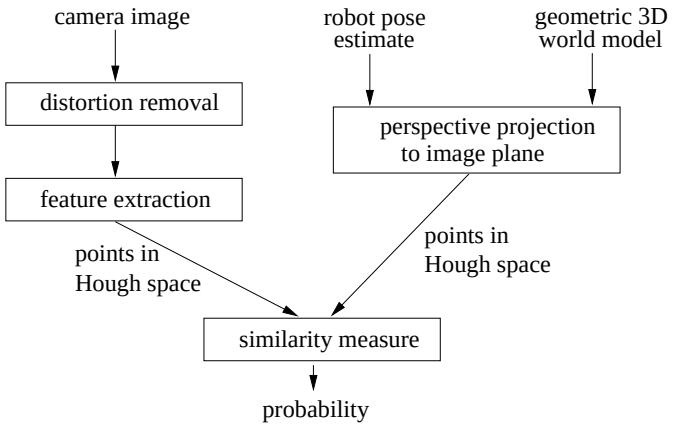

Fig. 1: Processing steps of the visual-sensor model.

of image line endpoints to corresponding, infinitely extended model lines in the image plane.

To restrict the search space in the matching step, [10] proposed to constrain the number of possible correspondences for a given pose estimate by combining line features into perceptual, quasi-invariant structures beforehand. Since these initial correspondences are evaluated by $e_{2D,1}$ and $e_{2D,2}$, high demands are imposed on the accuracy of the initial pose estimate and the image processing operations, including the removal of distortions and noise and the feature extraction. It is assumed to obtain all visible model lines at full length. [12, 9] demonstrated that a few outliers already can severely affect the initial correspondences in Lowe's original approach due to frequent truncation of lines caused by bad contrast, occlusion, or clutter.

3 Sensor Model

Our approach was motivated by the question whether a solution to the correspondence problem can be avoided in the estimation of the camera pose. Instead, we propose to perform a relatively simple, direct matching of image and model features. We want to investigate the level of accuracy and robustness one can achieve this way.

The processing steps involved in our approach are depicted in Figure 1. After removing the distortion from the camera image, we use the Canny operator to extract edges. This operator is relatively tolerant to changing illumination conditions. From the edges, line segments are identified. Each line is represented as a single point (ρ, θ) in the 2D Hough space given by $\rho = x \cos\theta + y \sin\theta$. The coordinates of the end points are neglected. In this representation, truncated or split lines will have similar coordinates in

the Hough space. Likewise, the lines in the 3D map are projected onto the image plane using an estimate of the camera pose and taking into account the visibility constraints, and are represented as coordinates in the Hough space as well. We have designed several error functions to be used as similarity measure in the matching step. They are described in the following.

Centred match count (CMC)

The first similarity measure is based on the distance of line segments in the Hough space. We consider only those image features as possible matches that lie within a rectangular cell in the Hough space centred around the model feature. The matches are counted and the resulting sum is normalised. The mapping from the expectation (model features) to the measurement (image features) accounts for the fact that the measure should be invariant with respect to objects not modelled in the 3D map or unexpected changes in the operating environment. Invariance of the number of visible features is obtained by normalisation. Specifically, the centred match count measure s_{CMC} is defined by:

$$\begin{aligned} s_{\mathrm{CMC}} &= (1/\left|H_e\right|) \#_{i=1}^{|H_e|} \min\left(1, \#_{j=1}^{|H_m|} p(h_{e_i}, h_{m_j})\right) \\ p(a,b) &\equiv |\rho_a - \rho_b| < t_\rho \wedge \|\theta_a - \theta_b\| < t_\theta \end{aligned}$$

where the predicate p defines a valid match using the distance parameters (t_ρ, t_θ) and the operator $\#$ counts the number of matches. Generally speaking, this similarity measure computes the proportion of expected model Hough points $h_{e_i} \in H_e$ that are confirmed by at least one measured image Hough point $h_{m_j} \in H_m$ falling within tolerance (t_ρ, t_θ). Note that neither endpoint coordinates nor lengths are considered here.

Grid length match (GLM)

The second similarity measure is based on a comparison of the total length values of groupes of lines. Split lines in the image are grouped together using a uniform discretisation of the Hough space. This method is similar to the Hough transform for straight lines. The same is performed for line segments obtained from the 3D model. Let $l_{m_{i,j}}$ be the sum of lengths of measured lines in the image falling into grid cell (i,j), likewise $l_{e_{i,j}}$ for expected lines according to the model, then the grid length match measure s_{GLM} is defined as:

$$s_{\mathrm{GLM}} = \frac{1}{\#_i^I \#_j^J (l_{e_{i,j}} > 0)} \sum_i^I \sum_{\substack{j \\ l_{e_{i,j}} > 0}}^J \min\left(1, \frac{l_{m_{i,j}}}{l_{e_{i,j}}}\right)$$

For all grid cells containing model features, this measure computes the ratio of the total line length of measured and expected lines. Again, the mapping is directional, i.e., the model is used as reference, to obtain invariance of noise, clutter, and dynamic objects.

Nearest neighbour and Hausdorff distance

In addition, we experimented with two generic methods for the comparison of two sets of geometric entities: the nearest neighbour and the Hausdorff distance. For details see [7]. Both rely on the definition of a distance function, which we based on the coordinates in the Hough space, i.e., the line parameter ρ and θ, and optionally the length, in a linear and exponential manner. See [5] for a complete description.

Common error functions

For comparisons, we also implemented the commonly used error functions e_{3D}, $e_{2D,1}$, and $e_{2D,2}$. As they are defined in the Cartesian space, we represent lines in the Hessian notation, $x \sin\phi - y \cos\phi = d$. Using the generic error function f, we defined the similarity measure as:

$$s = 1/\left(1 + \frac{1}{|M|} \sum_{m \in M} \left(\min_{e \in E} f(m, e) \right) \right) \tag{1}$$

where M is the set of measured lines and E is the set of expected lines.

In case of $e_{2D,1}$, f is defined by the perpendicular distances between both model line endpoints, e_1, e_2, and the infinitely extended image line m:

$$\begin{aligned} f_{2D,1}(m, e) &= f'_{2D,1}(m, e_1) + f'_{2D,1}(m, e_2) \\ f'_{2D,1}(m, p) &= \mid p_x \sin(\phi) - p_y \cos(\phi) - d \mid \end{aligned}$$

Likewise, the dual similarity measure, using $e_{2D,2}$, is based on the perpendicular distances between the image line endpoints and the infinitely extended model line.

Recalling that the error function e_{3D} is proportional to the distances of model line endpoints to the view plane through an image line and the camera origin, we can instantiate Equation 1 using $f_{3D}(m, e)$ defined as:

$$\begin{aligned} f_{3D}(m, e) &= \mid \vec{n}^{\circ}_m \vec{e}_1 \mid + \mid \vec{n}^{\circ}_m \vec{e}_2 \mid \\ \vec{n}_m &= \vec{m}_1 \times \vec{m}_2 \end{aligned}$$

where $\vec{n}_m$ denotes the normal vector of the view plane given by the image endpoints $\vec{m}_i = [m_x, m_y, w]^T$ in camera coordinates.

Obtaining probabilities

Ideally, we want the similarity measure to return monotonically decreasing values as the pose estimate used for projecting the model features departs from the actual camera pose. As we aim at a generally valid yet simple visual-sensor model, the idea is to abstract from specific poses and environmental conditions by averaging over a large number of different, independent situations. For commensurability, we want to express the model in terms of relative robot coordinates instead of absolute world coordinates. In other words, we assume

$$p(m \mid l_m, l_e, w) = p(m \mid \triangle l, w) \tag{2}$$

to hold, i.e., the probability for the measurement m, given the pose l_m this image has been taken at, the pose estimate l_e, and the world model w, is equal to the probability of this measurement given a three-dimensional pose deviation $\triangle l$ and the world model w.

The probability returned by the visual-sensor model is obtained by simple scaling:

$$p(m|l, w) = \left(\int s(m, \triangle l, w) \mathrm{d}(\triangle l) \right)^{-1} s(m, l, w)$$

4 Experimental Results

We have evaluated the proposed sensor model and similarity measures in a series of experiments. Starting with artificially created images using idealised conditions, we have then added distortions and noise to the tested images. Subsequently, we have used real images from the robot's camera obtained in a hallway. Finally, we have used the sensor model to track the position of the robot while it was travelling through the hallway. In all these cases, a three-dimensional visualisation of the model was obtained, which was then used to assess the solutions.

Simulations using artificially created images

As a first kind of evaluation, we generated synthetic image features by generating a view at the model from a certain camera pose. Generally speaking, we duplicated the right-hand branch of Figure 1 onto the left-hand side. By introducing a pose deviation $\triangle l$, we can directly demonstrate its influence on the similarity values. For visualisation purposes, the translational deviations $\triangle x$ and $\triangle y$ are combined into a single spatial deviation $\triangle t$. Initial experiments have shown only insignificant differences when they were considered independently.

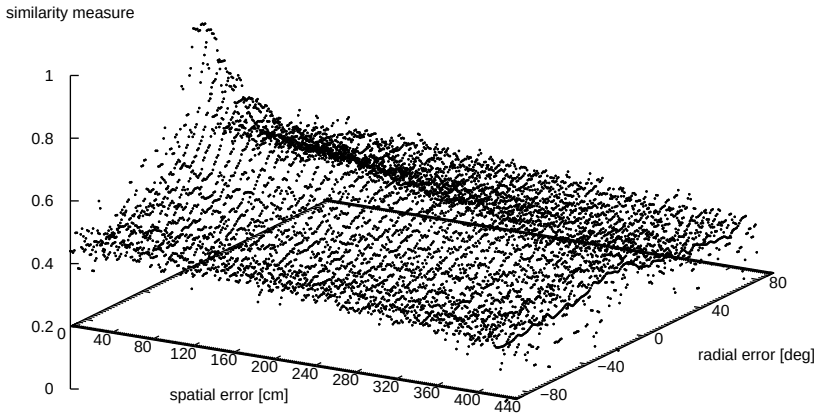

Fig. 2: Performance of CMC on artificially created images.

For each similarity measure given above, at least 15 million random camera poses were coupled with a random pose deviation within the range of $\triangle t <$ 440cm and $\triangle\theta < 90^\circ$ yielding a model pose.

The results obtained for the CMC measure are depicted in Figure 2. The surface of the 3D plot was obtained using GNUPLOT's smoothing operator dgrid3d. We notice a unique, distinctive peak at zero deviation with monotonically decreasing similarity values as the error increases. Please note that this simple measure considers neither endpoint coordinates nor lengths of lines. Nevertheless, we obtain already a decent result.

While the resulting curve for the GLM measure resembles that of CMC, the peak is considerably more distinctive. This conforms to our anticipation since taking the length of image and model lines into account is very significant here. In contrast to the CMC measure, incidental false matches are penalised in this method, due to the differing lengths.

The nearest neighbour measure turned out to be not of use. Although linear and exponential weighting schemes were tried, even taking the length of line segments into account, no distinctive peak was obtained, which caused its exclusion from further considerations.

The measure based on the Hausdorff distance performed not as good as the first two, CMC and GLM, though it behaved in the desired manner. But its moderate performance does not pay off the longest computation time consumed among all presented measures and is subsequently disregarded.

So far, we have shown how our own similarity measures perform. Next, we demonstrate how the commonly used error functions behave in this framework. The function $e_{2D,1}$ performed very well in our setting. The resulting curve closely resembles that of the GLM measure. Both methods exhibit a unique, distinctive peak at the correct location of zero pose deviation. Note that the length of line segments has a direct effect on the similarity value returned by measure GLM, while this attribute implicitly contributes to the measure $e_{2D,1}$, though both linearly. Surprisingly, the other two error functions $e_{2D,2}$ and e_{3D} performed poorly.

Toward more realistic conditions

In order to learn the effect of distorted and noisy image data on our sensor model, we conducted another set of experiments described here. To this end, we applied the following error model to all synthetically generated image features before they are matched against model features. Each original line is duplicated with a small probability ($p = 0.2$) and shifted in space. Any line longer than 30 pixel is split with probability $p=0.3$. A small distortion is applied to the parameters (ρ, θ, l) of each line according to a random, zero-mean Gaussian. Furthermore, features not present in the model and noise are simulated by adding random lines uniformly distributed in the image. Hereof, the orientation is drawn according to the current distribution of angles to yield fairly 'typical' features.

The results obtained in these simulations do not differ significantly from the first set of experiments. While the maximum similarity value at zero deviation decreased, the shape and characteristics of all similarity measures still under consideration remained the same.

Using real images from the hallway

Since the results obtained in the simulations above might be questionable with respect to real-world conditions, we conducted another set of experiments replacing the synthetic feature measurements by real camera images.

To compare the results for various parameter settings, we gathered images with a Pioneer 2 robot in the hallway off-line and recorded the line features. For two different locations in the hallway exemplifying typical views, the three-dimensional space of the robot poses (x, y, θ) was virtually discretized. After placing the robot manually at each vertex $(x, y, 0)$, it performed a full turn on the spot stepwise recording images. This ensures a maximum accuracy of pose coordinates associated with each image. That way, more than 3200 images have been collected from 64 different (x, y) locations. Similarly to the simulations above, pairs of poses (l_e, l_m) were systematically chosen

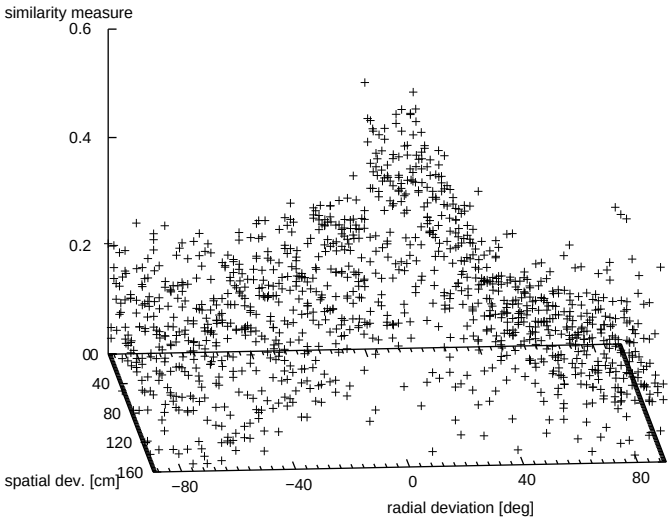

Fig. 3: Performance of GLM on real images from the hallway.

from with the range covered by the measurements. The values computed by the sensor model referring to the same discretized value of pose deviation $\triangle l$ were averaged according to the assumption in Equation 2.

The resulting visualisation of the similarity measure over spatial (x and y combined) and rotational deviations from the correct camera pose for the CMC measure exhibits a unique peak at approximately zero deviation. Of course, due to a much smaller number of data samples compared to the simulations using synthetic data, the shape of the curve is much more bumpy, but this is in accordance with our expectation.

The result of employing the GLM measure in this setting is shown in Figure 3. As it reveals a more distinctive peak compared to the curve for the CMC measure, it demonstrates the increased discrimination between more and less similar feature maps when taking the lengths of lines into account.

Monte Carlo Localisation using the visual-sensor model

Recalling that our aim is to devise a probabilistic sensor model for a camera mounted on a mobile robot, we continue with presenting the results for an application to mobile robot localisation.

The generic interface of the sensor model allows it to be used in the correction step of Bayesian localisation methods, for example, the standard

Fig. 4: Image and projected models during localisation.

version of the Monte Carlo localisation (MCL) algorithm. Since statistical independence among sensor readings renders one of the underlying assumptions of MCL, our hope is to gain improved accuracy and robustness using the camera instead of or in addition to commonly used distance sensors like sonars or laser.

In the experiment, the mobile robot equipped with a fixed-mounted CCD camera had to follow a pre-programmed route in the shape of a double loop in the corridor. On its way, it had to stop at eight pre-defined positions, turn to a nearby corner or open view, take an image, turn back and proceed. Each image capture initiated the so-called correction step of MCL and the weights of all samples were recomputed according to the visual-sensor model, yielding the highest density of samples at the potentially correct pose coordinates in the following resampling step. In the prediction step, the whole sample set is shifted in space according to the robot's motion model and the current odometry sensor readings.

Our preliminary results look very promising. During the position tracking experiments, i.e., the robot was given an estimate of its starting position, the best hypothesis for the robot's pose was approximately at the correct pose most of the time. In this experiment, we have used the CMC measure. In Figure 4, a typical camera view is shown while the the robots follows the requested path. The grey-level image depicts the visual input for feature extraction after distortion removal and pre-processing. Also the extracted line features are displayed. Furthermore, the world model is projected according to two poses, the odometry-tracked pose and the estimate computed by MCL which approximately corresponds to the correct pose, between which we observe translational and rotational error.

The picture also shows that rotational error has a strong influence on the degree of coincidental feature pairs. This effect corresponds to the results

presented above, where the figures exhibit a much higher gradient along the axis of rotational deviation than along that of translational deviation. The finding can be explained by the effect of motion on features in the Hough space. Hence, the strength of our camera sensor model lays at detecting rotational disagreement. This property makes it especially suitable for two-wheel driven robots like our Pioneer bearing a much higher rotational odometry error than translational error.

5 Conclusions and Future Work

We have presented a probabilistic visual-sensor model for camera-pose estimation. Its generic design makes it suitable for sensor fusion with distance measurements perceived from other sensors. We have shown extensive simulations under ideal and realistic conditions and identified appropriate similarity measures. The application of the sensor model in a localisation task for a mobile robot met our anticipations. Within the paper we highlighted much scope for improvements.

We are working on suitable techniques to quantitatively evaluate the performance of the devised sensor model in a localisation algorithm for mobile robots. This will enable us to experiment with cluttered environments and dynamic objects. Combining the camera sensor model with distance sensor information using sensor fusion renders the next step toward robust navigation. Because the number of useful features varies significantly as the robots traverses an indoor environment, the idea to steer the camera toward richer views (active vision) offers a promising research path to robust navigation.

References

[1] F. Dellaert, W. Burgard, D. Fox, and S. Thrun. Using the condensation algorithm for robust, vision-based mobile robot localisation. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, 1999.

[2] D. DeMenthon, P. David, and H. Samet. SoftPOSIT: An algorithm for registration of 3D models to noisy perspective images combining Softassign and POSIT. Technical report, University of Maryland, MD, 2001.

[3] G. N. DeSouza and A. C. Kak. Vision for mobile robot navigation: A survey. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 24(2):237–267, 2002.

[4] S. Enderle, M. Ritter, D. Fox, S. Sablatnög, G. Kraetzschmar, and G. Palm. Soccer-robot localisation using sporadic visual features. In *Intelligent Autonomous Systems 6*, pages 959–966. IOS, 2000.

[5] M. Fichtner. A camera sensor model for sensor fusion. Master's thesis, Dept. of Computer Science, TU Dresden, Germany, 2002.

[6] S. A. Hutchinson, G. D. Hager, and P. I. Corke. A tutorial on visual servo control. *IEEE Trans. on Robotics and Automation*, 12(5):651–670, 1996.

[7] D. P. Huttenlocher, G. A. Klanderman, and W. J. Rucklidge. Comparing images using the Hausdorff distance. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 15(9):850–863, 1993.

[8] A. Kosaka and A. C. Kak. Fast vision-guided mobile robot navigation using model-based reasoning and prediction of uncertainties. *Computer Vision, Graphics, and Image Processing – Image Understanding*, 56(3):271–329, 1992.

[9] R. Kumar and A. R. Hanson. Robust methods for estimating pose and a sensitivity analysis. *Computer Vision, Graphics, and Image Processing – Image Understanding*, 60(3):313–342, 1994.

[10] D. G. Lowe. Three-dimensional object recognition from single two-dimensional images. *Artificial Intelligence*, 31(3):355–395, 1987.

[11] S. Se, D. G. Lowe, and J. Little. Vision-based mobile robot localisation and mapping using scale-invariant features. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, pages 2051–2058, 2001.

[12] G. D. Sullivan, L. Du, and K. D. Baker. Quantitative analysis of the viewpoint consistency constraint in model-based vision. In *Proc. of the 4th Int. IEEE Conf. on Computer Vision*, pages 632–639, 1993.