A Robust Method for Computing Vehicle Ego-motion

Gideon P. Stein Ofer Mano
MobileEye Ltd.
gideon,ofer@mobileye.com

Amnon Shashua
Hebrew University, Jerusalem
shashua@cs.huji.ac.il

Abstract

We describe a robust method for computing the ego-motion of the vehicle relative to the road using input from a single camera mounted next to the rear view mirror. Since feature points are unreliable in cluttered scenes we use direct methods *where image values in the two images are combined in a global probability function. Combined with the use of probability distribution matrices, this enables the formulation of a robust method that can ignore large number of outliers as one would encounter in real traffic situations. The method has been tested in real world environments and has been shown to be robust to glare, rain and moving objects in the scene.*

1 Introduction

Accurate estimation of the ego-motion of the vehicle relative to the road is a key component for autonomous driving and computer vision based driving assistance. We describe a robust method for computing the ego-motion of the vehicle relative to the road using input from a single camera rigidly mounted next to the rear view mirror. The method has been tested in real world environments and has been shown to be robust to glare, rain and moving objects in the scene. Using vision instead of mechanical sensors for computing ego-motion allows for a simple integration of ego-motion data into other vision based algorithms, such as obstacle and lane detection, without the need for calibration between sensors. This reduces maintenance and cost.

The challenge of achieving a high-level of robustness in ego-motion estimation for real-life conditions, such as dense traffic, can be traced to the following two points:

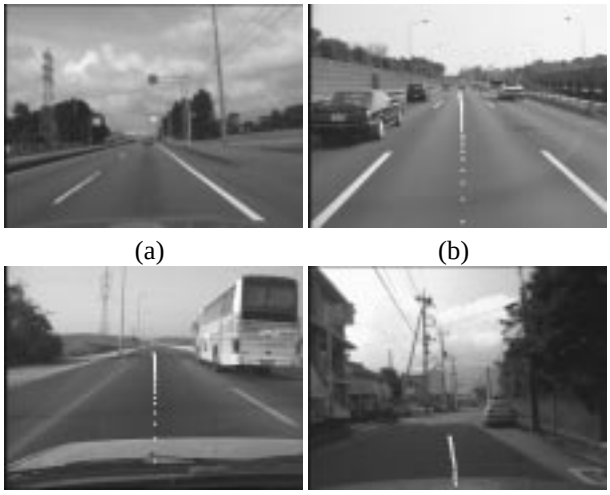
1. Typical roads have very few feature points, if at all. Most of the measurable image structure is linear — like lane marks. On the other hand, the background image structures may contain many feature points (like those on other vehicles, trees, buildings, etc.). Therefore, an optic-flow based calculation would be very problematic in practice.

2. A typical scene may contain a large amount of outlier information. For example moving traffic violates the rigid world assumption and thus contributes false measurements for ego-motion calculation; also rain drops, wiper moving in rain conditions, glare, and so forth, all contribute false measurements for ego-motion recovery.

To overcome these problems, first and foremost, we propose an approach based on a *direct method* [4, 1, 11, 10] where *each* pixel contributes a measurement. These measurements are then combined in a global probability function for the parameters of the ego-motion model. The "direct" approach has the advantage of avoiding the calculation of optic-flow and in turn avoids the use of feature tracking. As a result, the collinear image structures that are prevalent in typical roadways contribute measurements for the ego-motion model. Second, we reduce the number of estimated parameters to a minimum of three parameters. This has the advantage of disambiguating typical ambiguous situations by decoupling rotational and translational motion, and most importantly facilitates the use of *robust* estimation using sampling [7].

In the direct estimation model we make the assumption that the roadway is a planar structure and focus the measurements on the road itself. In other words, all image measurements that violate the rigid world assumption (like moving vehicles and moving shadows) *and* all image structure above the road are considered outliers. The planar assumption makes the ego-motion estimation a parametric estimation model (see [1]) with 8 parameters (described in the sequel). In our work we found that it is very important to reduce the number of estimated parameters to a minimum in order to facilitate a robust estimation. Fortunately, in the typical driving scenario, the road forms a planar structure and leads to a simple parametric model with only 3 dominant parameters: forward translation, pitch and yaw. With few parameters it is possible to devise a robust method and the computation can be performed at frame rate on standard hardware.

Fig. 1 shows a typical set of road images with varying degrees of difficulty. Fig. 1a shows a highway scene with a clearly dominant ground plane. The car in the distance can



(a) (b)

(a) (b)

Figure 1: Typical images from camera mounted inside a car. (a) a simple highway scene with a small number of outliers to the planar assumption. (b) a more challenging scene with denser traffic. (c) The large moving bus passing on the right does not divert the robust motion estimate. (The white dots at the center of the image mark the predicted path 50 frames ahead.) (d) a typical urban street. The road features are weak and unstructured. The estimated heading clearly marks the shift to the left to pass the parked car.

be regarded as an outlier and systems such as [5, 2, 12] might work well. In fig. 1b the traffic is denser. Many features belong to the moving cars and few features lie on the road.

Fig. 1c shows a bus passing on the right which takes up a significant part of the image. The passing bus will cause a non-robust method to estimate a motion to the right (towards the bus). There is still texture on the road that can give us the ego-motion but the road is no longer the dominant planar structure. Fig. 1d shows a typical urban street where the road features are weak and unstructured. The weak features are combined globally to give good motion estimates and the shift to left to pass the parked car is clearly visible.

2 Mathematical foundation

2.1 The Motion Model

We will assume a calibrated camera system where the Z axis of the world coordinate system is aligned with the camera optical axis and the X and Y axes are aligned with the image axes x and y. The focal length f is assumed to be known.

Let $t = (t_x, t_y, t_z)^\top$ and $w = (w_x, w_y, w_z)^\top$ be the unknown instantaneous camera translation and rotation. Then following [6] the flow vector (u,v) for a point $p = (x,y)^\top$ in the image is given by:

$$\begin{aligned} u &= \frac{1}{Z} S_1^\top t + (\hat{p} \times S_1)^\top w \\ v &= \frac{1}{Z} S_2^\top t + (\hat{p} \times S_2)^\top w \end{aligned} \tag{1}$$

where:

$$S_1 = \begin{pmatrix} f \\ 0 \\ -x \end{pmatrix} \quad S_2 = \begin{pmatrix} 0 \\ f \\ -y \end{pmatrix} \quad \hat{p} = \begin{pmatrix} \frac{x}{f} \\ \frac{y}{f} \\ 1 \end{pmatrix} \tag{2}$$

For points on the plane we have:

$$AX + BY + CZ = 1. \tag{3}$$

Dividing through by Z we get:

$$\frac{1}{Z} = ax + by + c \tag{4}$$

where: $a = \frac{A}{f}$, $b = \frac{B}{f}$ and $c = C$. We then substitute (4) in (1) to get:

$$\begin{aligned} u &= (ax + by + c) S_1^\top t + (\hat{p} \times S_1)^\top w \\ v &= (ax + by + c) S_2^\top t + (\hat{p} \times S_2)^\top w. \end{aligned} \tag{5}$$

Expanding these equations we have:

$$\begin{aligned} u &= -(ct_z + aft_x)x + (bft_x - w_z)y + (fw_y + cft_x) + \\ &\quad (\frac{w_y}{f} - at_z)x^2 - (\frac{w_x}{f} + bt_z)xy \end{aligned} \tag{6}$$

$$\begin{aligned} v &= (w_z + aft_y)x + (-ct_z + bft_y)y - (fw_x + cft_y) + \\ &\quad (\frac{w_y}{f} - at_z)xy - (\frac{w_x}{f} + bt_z)y^2 \end{aligned} \tag{7}$$

These equations are a special case (the calibrated camera case) of the 8-parameter model for a camera moving relative to a plane:

$$u = \alpha_1 x + \alpha_2 y + \alpha_3 + \alpha_7 x^2 + \alpha_8 xy \tag{8}$$

$$v = \alpha_4 x + \alpha_5 y + \alpha_6 + \alpha_7 xy + \alpha_8 y^2. \tag{9}$$

Given the optical flow (u,v) one can recover the parameters $\alpha_i, i = 1 \ldots 8$ from which one can recover the motion parameters [12]. The problem is that due to the large number of parameters it is hard to devise a robust method which rejects outliers. Furthermore, it is hard to differentiate between flow due to rotation around the X and Y axes and translation along the Y and X axes respectively.

It is therefore advantageous to reduce the number of motion parameters to a minimum. The motion of a car on the

road can be modeled as a translation along the Z axis and rotation around the X and Y axes. Limiting ourselves to this motion model, eq. (5) becomes:

$$u = -(ax+by+c)xt_z - \frac{xy}{f}w_x + (f+\frac{x^2}{f})w_y \qquad (10)$$
$$v = -(ax+by+c)yt_z - (f+\frac{y^2}{f})w_x + \frac{xy}{f}w_y.$$

In our particular setup we rectify the images so that the ground plane is parallel to the XZ plane (i.e. $a = 0$ and $c = 0$) and thus:

$$u = -bxyt_z - \frac{xy}{f}w_x + (f+\frac{x^2}{f})w_y \qquad (11)$$
$$v = -by^2t_z - (f+\frac{y^2}{f})w_x + \frac{xy}{f}w_y.$$

In order to rectify the images correctly one must calibrate the camera. The calibration process is described in section 4.

2.2 Combining geometric and photometric constraints

Solving eqs. (11) for t_z, w_x and w_y would require first computing the optical flow (u,v) (i.e. point correspondences). Finding corresponding points in two images is based on the *photometric constraint*[3]:

$$I(x,y,t) - I(x+u\delta t, y+v\delta t, t+\delta t) = 0. \qquad (12)$$

This equation states that the irradiance of an image point (x,y) at time t is equal to the irradiance of the corresponding point at time $t+\delta t$. In practice eq. (12) does not hold exactly due to noise. If we model the noise for every pixel as zero mean Gaussian noise we get:

$$P\left(I(x,y,t) - I(x+u\delta t, y+v\delta t, t+\delta t)\right) = N\left(\sigma^2, 0\right) \qquad (13)$$

and a *maximum likelihood* solution is sought.

Using eq. (12) alone to find correspondences has proven to be difficult and computationaly expensive. To avoid this step we use the direct method approach of [4] (see also [1, 11, 10]). Following this approach we compute the motion parameters directly from the images by combining the geometric constraints embodied in eq. (11) together with the photometric constraints (12).

Given two consecutive images $\psi(x,y)$ and $\psi'(x,y)$, our goal is to compute the probability of a motion $\hat{m} = (t_z, w_x, w_y)$ given the two images:

$$P\left(\hat{m}|\psi,\psi'\right). \qquad (14)$$

The motion that maximizes (14) is our estimate of the camera motion between those two frames.

We derive the probability distribution in a similar manner to [8]. Using Bayes rule:

$$P\left(\hat{m}|\psi,\psi'\right) = \frac{P\left(\psi'|\psi,\hat{m}\right)P\left(\hat{m}\right)}{P\left(\psi'\right)} \qquad (15)$$

$P\left(\hat{m}\right)$ is the *a priori* probability that the motion is $\hat{m}$. We will assume a uniform probability in a small region $\hat{M}$ around the previous estimate. The denominator $P\left(\psi'\right)$ does not depend on $\hat{m}$ and thus does not affect the search for a maximum.

We now develop an expression for $P\left(\psi'|\psi,\hat{m}\right)$, the probability of observing an image ψ' given the previous image ψ and a motion $\hat{m}$. Given the motion $\hat{m} = (t_z, w_x, w_y)$ the *sum squared difference* (SSD) between the two patches is:

$$S(\hat{m}) = \frac{1}{N}\sum_{x,y\in R}\left(\hat{\psi}'(x,y) - \psi(x,y)\right)^2 \qquad (16)$$

where $\hat{\psi}'$ is image ψ' warped according the motion $\hat{m}$ and R is the set of all the pixels in ψ that belong to the road. N_r is the number of pixels in the set R. Using this SSD criteria:

$$P\left(\psi'|\psi,\hat{m}\right) = ce^{\frac{-S(\hat{m})}{\sigma^2}} \qquad (17)$$

where c is a normalization factor and we have modeled the noise as zero mean Gaussian noise with variance σ^2.

Therefore the problem of finding the maximum likelihood motion $\hat{m}$ for a patch ψ is that of finding the maximum of the function:

$$P\left(\hat{m}|\psi,\psi'\right) = ce^{\frac{-S(\hat{m})}{\sigma^2}} \qquad (18)$$

for $\hat{m} \in \hat{M}$. Since the set R is not known we next consider the problem of robust estimation.

2.3 Robust Implementation

The basic idea is to tessellate the image into a set of patches W_i. We then sum the probability densities for each patch W_i weighted by our confidence λ_i that the patch comes from the road and β_i, a measure of the gradient information in the patch.

$$P\left(\hat{m}|\psi,\psi'\right) = c\frac{\sum_i P\left(\hat{m}|W_i,W_i'\right)\lambda_i}{\sum_i \lambda_i} \qquad (19)$$

The motion $\hat{m} \in \hat{M}$ that maximizes eq. (19) is our motion estimate given images ψ and ψ'. For each patch W_i the set R_i includes all the pixels in the patch.

To compute the weight λ_i we observe that for patches which do not belong to the road (such as the rear ends and sides of cars) the motion model (eq. 11) is not a good fit. A better fit can be obtained using some other motion of the patch. Furthermore, for planar objects moving on the road surface such as moving shadows the maximum of eq. (18)

will occur far away from the initial guess. The weight λ_i is then the ratio between the best fit using the motion model in a local region near the initial guess ($\hat{M}$) and the best fit using any motion model over a large search region.

Let:

$$P_1 = \max\left(\exp(\frac{-S_i(\hat{m})}{\sigma^2})\right) \quad (20)$$

for all $\hat{m} \in \hat{M}$ be the score for the best fit in a local search region. We have used $S_i()$ to denote the SSD over all pixels in the patch i. Let:

$$P_2 = \max\left(\exp(\frac{-S_i(\hat{m})}{\sigma^2})\right) \quad (21)$$

for all $\hat{m} \in L$ be the score for the best fit for all feasable image motions, not limiting ourselves to the particular motion model (11). Then:

$$\lambda_i = \frac{P_1}{P_2}. \quad (22)$$

In practice P_2 as defined is too expensive to compute. It is sufficient to consider only integer image translations over a range of ± 7 pixels in the x and y directions.

In order to reduce the effect of patches with little gradient information we define:

$$\beta_i = \left(\sum_{\hat{m} \in L} \exp(\frac{-S_i(\hat{m})}{\sigma^2})\right)^{-1}. \quad (23)$$

For a uniform patch the SSD will be low for all motions and therefore β_i will be low. For patches with texture the SSD will be high for most motions and therefore β_i will be high.

3 The Algorithm

We now describe the complete algorithm for computing ego-motion from two frames.:

1. Start with an initial guess which is based on the previous motion estimate and information from other sensors if available, such as the speedometer.
2. For each patch, warp image 2 towards image 1 using the initial guess.
3. In a 15×15 region around that point compute the SSD and the fit value. The best fit is P_2 (eq. 21). The sum of the fits values is $\frac{1}{\beta_i}$ (eq. 23).
4. In a small 3D space of motions ($t_z, w_x, w_y \in \hat{M}$) around the initial guess, search for the best fit value for that patch. This is P_2 (eq. 20). This search can be performed using gradient descent limited to a cube shaped region.
5. Compute λ_i from P_1 and P_2 (eq. 22).
6. Search for the motion $\hat{m}$ that maximizes eq. (19). This search can be performed using gradient descent limited to a cube shaped region.

This algorithm is extended to a motion sequence by using the new estimate to adjust the initial guess. The size of the region $\hat{M}$ can also be adjusted adaptively. As a starting guess, if we do not have a speedometer reading we use 40kmh as the initial speed and zero values for yaw and pitch. The algorithm then converges to the correct value after a few hundred frames (2-3 seconds of motion).

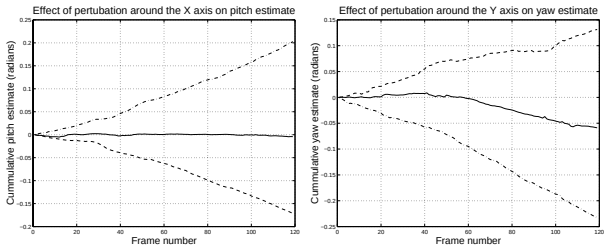
4 Calibration

In eq. (11) we assume a coordinate frame in which the ground plane is parallel to the XZ plane of our camera coordinate system and that the optical axis is parallel to the Z axis. This requires that the images be rectified (in software) prior to computing the ego-motion. We now describe a procedure for calibrating the system and determining the correct rectification required.

Let us first consider the effects of incorrect calibration. Let us assume the car is driving down a straight road (e.g. fig. 1a). If the camera optical axis is aligned with the direction of motion then the image flow field will be an expansion field with the *focus of expansion* (FOE) located at the image center $(0,0)$. If the camera was mounted with a small rotation around the Y axis then the FOE will be located at some other point along the x axis. The motion model defined in eq. (11) cannot account for this flow field but it will be well approximated by a forward translation and a rotational velocity w_y around the Y axis.

Thus, errors in the orientation of the camera calibration around the Y axis will create a bias in our rotation estimate. The system will estimate a curved path when the car is in fact driving down a straight road. In a similar fashion, errors in the camera orientation around the X axis will cause a bias in the pitch estimates. Based on these observation we can come up with a simple calibration procedure. We use an image sequence where the car is driving down a straight road. We estimate the car motion and search for rectification parameters that will give us ego-motion estimates that integrate into a straight path.

Fig. 2a shows the motion estimates (w_y) using various values of rotation around the Y axis for rectification. Fig. 2b shows the effect of the rectification value of rotation around the X axis on the pitch estimate (w_x). Using this error measure the correct calibration can easily be found using simple search techniques such as gradient descent.



(a) (b)

Figure 2: (a) Cumulative rotation (yaw) estimates from a sequence of 120 images where a car is driving down a straight highway (fig. 1a). Solid lines indicate correct calibration. The broken lines show the effect of a rectification error of ± 5 pixels along the X axis (i.e. a rotation error round the Y axis). (b) Similar graph showing the effects of error in rectification of ± 10 along the Y axis.

5 Experiments and results

The images were captured using a camcorder (Cannon Optura) rigidly mounted close to the passenger side of the rear view mirror. The horizontal field of view is 50^o. The images were digitized at 30fps and then processed offline. Image resolution was 320×240 pixels.

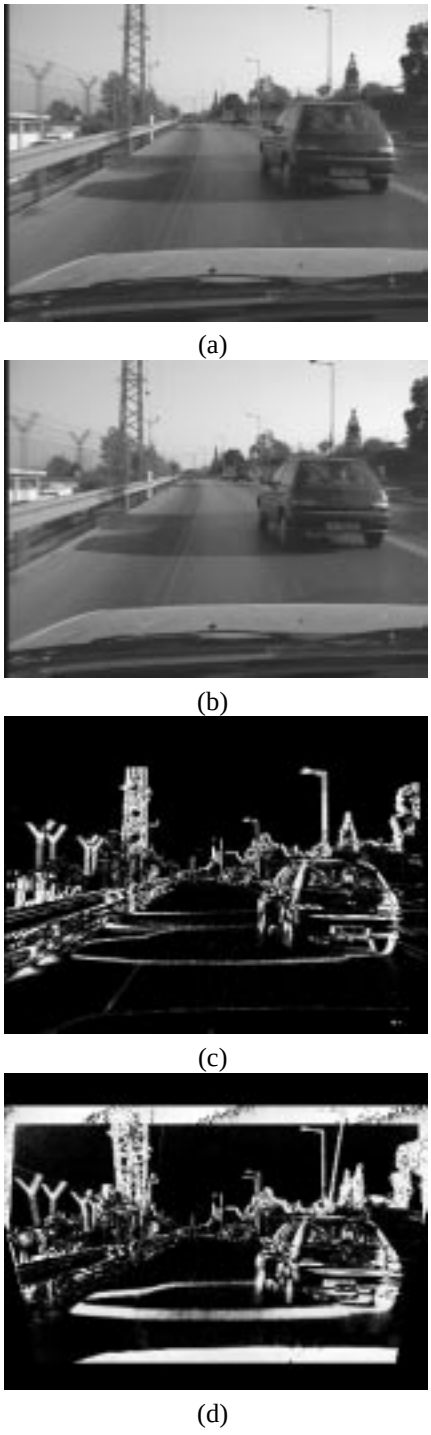
5.1 Robust estimation

Figs. 3a and 3b show two consecutive images from a motion sequence. This is a challenging scene since we have to differentiate between the moving shadow of the car and the stationary shadow on the road. Fig. 3c shows the result of simply subtracting the two images prior to alignment. Note how the shadow of the moving car is almost stationary in the image. Fig. 3d shows the result of subtracting the images after alignment. As we can see the shadow of the car (and the car itself) stand out while the static shadow on the road and the lane markings are well aligned and disappear from the difference image.

5.2 Accuracy test

In order to test the accuracy of the ego-motion estimation, the car was driven around a traffic circle. Samples of the image sequence are show in fig. 4. Note images fig. 4b and fig. 4e. The x coordinate of the distant rectangular structure which appears at the top left of the images is the same in both images. This means that the car has completed exactly 360^0 rotation and this provides us with an accurate ground truth measurement. Summing up the rotation estimates over that part of the sequence results in 366.5^o, a 1.9% error or an average error of 0.017^o per frame.

We have no accurate ground truth measure of the distance traveled but the inner diameter of the traffic circle was $20m$ and the outer diameter $32m$. So the actual distance traveled



(a)

(b)

(c)

(d)

Figure 3: Robust alignment in the presence of outliers. (a) (b) two consecutive images. (c) difference image prior to alignment. Note how the shadow of the moving car is almost stationary. (d) difference between the images after alignment. The shadow of the car stands out while the shadow just in front of it is well aligned.

(a) (b) frame 362

(c) (d)

(e) frame 775 (f)

Figure 4: Examples from a sequence of 800 images from a car driving round a traffic circle. Between frame 362 and frame 775 the car has completed a full circle.

by the car (πd) was between 63m and 100m. The distance estimated using the ego-motion algorithm was 67.5m.

5.3 Adverse lighting conditions

The system was tested also on rainy day conditions. In fig. 5 a moderate rain was falling, rain was splattering on the windshield and the windshield wipers were working. The system manages to ignore the distractors and correctly detect the car rotation.

5.4 Adapting the algorithm to night scenes

During night driving, the scene is illuminated the headlights of our vehicle. Therefore the main light source is moving with the car. The illumination is not uniform but changes very slowly over the image. This low frequency signal will bias the motion estimate towards zero motion. Following [9] we preprocess the image by a bandpass filter to remove the very low spatial frequencies (20 pixels or larger). Fig. 6 shows the system working under night conditions.

Figure 5: Motion is correctly estimated during moderately rainy condition. The windshield wipers and rain drops on the windshield are ignored by the robust algorithm.

Figure 6: The markings on the road illuminated by the car headlights are sufficient for motion estimation.

6 Discussion and future work

We have presented a new method for robust estimation of vehicle ego-motion. It is based on a few key ideas:

- Reduce the motion model to 3 essential parameters. This makes handling the probability density feasible. It also eliminates the ambiguity between rotations and translations.
- Instead of tracking features compute a probability density function for each image patch and model the uncertainty due to the aperture problem explicitly.
- Combine together the probability functions from all patches. Prior motion estimates give low weight to patches that are unlikely to come from the road.

This method proves to be robust in a large number demanding, real life situations including dense traffic and moderately bad weather. The latest version of the software can process images at 30fps on a dual Pentium III computer.

6.1 Adding speedometer information

So far we in our experiments have used only vision information. There is nearly always enough vertical texture in the image to give good rotation estimates but there are times when there is no horizontal texture. This might be the case when driving on stretches of new highway. In this case rotation can still be accurately estimated but it is difficult to estimate the magnitude of the forward motion (t_z). If this happens for short segments of up to 30 frames (1sec.) the system can cope but for a general solution we are investigating the use of speedometer information.

References

[1] J. R. Bergen, P. Anandan, K. J. Hanna, and R. Hingorani. Hierarchical model-based motion estimation. In *Proceedings of the European Conference on Computer Vision*, Santa Margherita Ligure, Italy, June 1992.

[2] M. J. Black and P. Anandan. Robust dynamic motion estimation over time. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 296–302, June 1991.

[3] B. K. P. Horn and B. G. Schunk. Determining optical flow. *Artificial Intelligence*, 17:185–203, 1981.

[4] B. K. P. Horn and E. J. Weldon, Jr. Direct methods for recovering motion. *International Journal of Computer Vision*, 2:51–76, 1988.

[5] M. Irani, B. Rousso, and S. Peleg. Recovery of ego-motion using image stabilization'. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 454–460, Seattle, Washington, June 1994.

[6] H. Longuet-Higgins and K. Prazdny. The interpretation of a moving retinal image. *Proceedings of the Royal Society of London B*, 208:385–397, 1980.

[7] P. Meer, D. Mintz, D. Kim, and A. Rosenfeld. Robust regression methods for computer vision: A review. *International Journal of Computer Vision*, 6(1):59–70, 1991.

[8] Y. Rosenberg and M. Werman. A general filter for measurements with any probability distribution. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 654–659. IEEE Computer Society Press, June 1997.

[9] C. Y. S. Negahdaripour and A. Shokrollahi. Recovering shape and motion from undersea images. *IEEE Journal of Oceanic Engineering*, 15(3):189, 1987.

[10] G. P. Stein. *Geometric and Photometric Constraints: Motion and Structure from Three Views*. PhD thesis, M.I.T Artificial Intelligence Laboratory, February 1998.

[11] G. P. Stein and A. Shashua. Model based brightness constraints: On direct estimation of structure and motion. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Puerto Rico, June 1997.

[12] T. Suzuki and T. Kanade. Measurement of vehicle motion and orientation using optical flow. In *IEEE Conference on Intelligent Transportation Systems*, Tokyo, Japan, October 1999.