Direct Passive Navigation: Analytical Solution for Planes

S. Negahdaripour and B.K.P. Horn

The Artificial Intelligence Laboratory
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

Abstract: In this paper, we derive a closed form solution for recovering the motion of an observer relative to a planar surface directly from image brightness derivatives. We do not compute the optical flow as an intermediate step, only the spatial and temporal intensity gradients at a minimum of 8 points. We solve a linear matrix equation for the elements of a 3x3 matrix. The eigenvalue decomposition of its symmetric part is then used to compute the motion parameters and the plane orientation.

1. Introduction

The problem of determining rigid body motion and surface structure from image data has been the topic of many research papers in the area of machine vision [1-22]. Many approaches based on tracking feature points [5,11,19,20] or contours [9], using optical flow [1,3,4,10,12,16,17,21,22], texture [2], or image intensity gradients [14,15] have been proposed in the literature.

In the feature point matching schemes, information about a finite number of well-separated points is used to recover the motion (general 8-point 2-frame algorithms of Longuet-Higgins [11], Tsai and Huang [20], Buxton et al. [5], and the algorithm of Tsai, Huang and Zhu [19] for planar surfaces). These methods require identifying and matching feature points in a sequence of images. The minimum number of points required depends on the number of image frames. With 2 frames, in most cases, a minimum of 5 points results in a unique solution from a set of nonlinear equations. However, using 8 points, as in algorithms cited above, one only solves linear equations. Here, it is assumed that the more difficult problem of establishing point correspondence has already been solved. In general, this involves determining corners along contours using iterative searches. For images of smooth objects, it is difficult to find good features or corners.

For the smooth surfaces, Longuet-Higgins and Prazdny [11] suggested a method that uses the optical flow and its first and second derivatives at a single point. Later, Waxman and Ullman [21] developed this into an algorithm for recovering the structure and motion parameters from a set of nonlinear equations. Subbarao and Waxman [17] recently found a closed form solution to the original formulation in [21] for planar surfaces. These methods, while mathematically elegant, are very sensitive to errors in the optical flow data since second order derivatives of noisy data are used.

At the expense of more computation, more robust algorithms have been suggested using the optical flow at every image point [1,3,4]. Longuet-Higgins [12] has presented a closed form solution for planar surfaces, very similar to ours, using the coefficients of the second order optical flow equations. However, it is assumed that both components of the flow field have already been computed for a minimum of 5 image points.

By representing a planar surface in the form of a closed contour, Kanatani [9] has shown that the surface and motion parameters can be computed by measuring "diameters" of the contour using line and surface integrals. Here, no point correspondence is required. Assuming that the planar surface has a uniform texture density, Aloimonos and Chou [2] have presented a procedure for computing the motion and surface orientation from texture.

In much of the research work in recovering surface structure and motion from the optical flow field, it is assumed that a reasonable estimate of the full optical flow field is available. In general, the computation of the local flow field exploits a constraint equation between the local intensity changes and the two components of the optical flow. However, this only gives the component of the flow in the direction of the intensity gradient. To compute the full flow field, one needs additional constraints such as the heuristic assumption that the flow field is locally smooth [7,8]. This, in many cases, leads to optical flow fields that are not consistent with the true motion field.

In an earlier paper, we presented an iterative scheme for recovering the motion of an observer relative to a planar surface directly from the image brightness derivatives, without the need to compute the local flow field [14,15]. Further, using a compact vector notation, we showed that, at most, two interpretations are possible for planar surfaces and derived the relationship between them. Here, we present a closed form solution to the same problem. We first solve a linear matrix equation for the elements of a 3x3 matrix using intensity derivatives at a minimum of 8 points. The special structure of this matrix allows us to compute the motion and structure parameters very easily.

2. Preliminaries

We first recall some details about perspective projection, the motion field, the brightness change constraint equation, rigid body motion and planar surfaces. This we do using vector notation in order to keep the resulting equations as compact as possible.

2.1. Perspective Projection

Let the center of projection be at the origin of a Cartesian coordinate system. Without loss of generality we assume that the effective focal length is unity. The image is formed on the plane $z = 1$, parallel to the xy-plane, that is, the optical axis lies along the z-axis. Let $\mathbf{R}$ be a point in the scene. Its projection in the image is $\mathbf{r}$, where

$$\mathbf{r} = \frac{1}{\mathbf{R}\cdot\hat{\mathbf{z}}}\mathbf{R}.$$

The z-component of $\mathbf{r}$ is clearly equal to one, i.e., $\mathbf{r}\cdot\hat{\mathbf{z}} = 1$.

2.2. Motion Field and Optical Flow

The *motion field* is the vector field induced in the image plane by the relative motion of the observer with respect to the environment. The *optical flow* is the apparent motion of brightness patterns. Under favourable circumstances the optical flow is identical to the motion field (Moving shadows or uniform objects in motion could create discrepancies between the motion field and the optical flow. Here, we assume that the motion flow field and the optical flow are the same). The velocity of the image $\mathbf{r}$ of a point $\mathbf{R}$ is given by

$$\frac{d\mathbf{r}}{dt} = \frac{d}{dt}\frac{1}{\mathbf{R}\cdot\hat{\mathbf{z}}}\mathbf{R}.$$

For convenience, we introduce the notation $\mathbf{r}_t$ and $\mathbf{R}_t$ for the time derivatives of $\mathbf{r}$ and $\mathbf{R}$, respectively. We then have

$$\mathbf{r}_t = \frac{1}{\mathbf{R}\cdot\hat{\mathbf{z}}}\mathbf{R}_t - \frac{1}{(\mathbf{R}\cdot\hat{\mathbf{z}})^2}(\mathbf{R}_t\cdot\hat{\mathbf{z}})\mathbf{R},$$

that can also be written in the compact form

$$\mathbf{r}_t = \frac{1}{(\mathbf{R}\cdot\hat{\mathbf{z}})^2}\left(\hat{\mathbf{z}}\times(\mathbf{R}_t\times\mathbf{R})\right),$$

since $\mathbf{a}\times(\mathbf{b}\times\mathbf{c}) = (\mathbf{c}\cdot\mathbf{a})\mathbf{b} - (\mathbf{a}\cdot\mathbf{b})\mathbf{c}$. The vector $\mathbf{r}_t$ lies in the image plane, and so $(\mathbf{r}_t\cdot\hat{\mathbf{z}}) = 0$. Further, $\mathbf{r}_t = \mathbf{0}$, if $\mathbf{R}_t \parallel \mathbf{R}$, as expected.

Finally, noting that $\mathbf{R} = (\mathbf{R}\cdot\hat{\mathbf{z}})\mathbf{r}$, we obtain

$$\mathbf{r}_t = \frac{1}{\mathbf{R}\cdot\hat{\mathbf{z}}}\left(\hat{\mathbf{z}}\times(\mathbf{R}_t\times\mathbf{r})\right).$$

2.3. Rigid Body Motion

In the case of the observer moving relative to a rigid environment with translational velocity $\mathbf{t}$ and rotational velocity ω, we find that the motion of a point in the environment relative to the observer is given by

$$\mathbf{R}_t = -\omega\times\mathbf{R} - \mathbf{t}.$$

Since $\mathbf{R} = (\mathbf{R}\cdot\hat{\mathbf{z}})\mathbf{r}$, we can write this as

$$\mathbf{R}_t = -(\mathbf{R}\cdot\hat{\mathbf{z}})\,\omega\times\mathbf{r} - \mathbf{t}.$$

Substituting for $\mathbf{R}_t$ in the formula derived above for $\mathbf{r}_t$, we obtain

$$\mathbf{r}_t = -\left(\hat{\mathbf{z}}\times\left(\mathbf{r}\times\left(\mathbf{r}\times\omega - \frac{1}{\mathbf{R}\cdot\hat{\mathbf{z}}}\mathbf{t}\right)\right)\right).$$

It is important to remember that there is an inherent ambiguity here, since the same motion field results when distance and the translational velocity are multiplied by an arbitrary constant. This can be seen easily from the above equation since the same image plane velocity is obtained if one multiplies both $\mathbf{R}$ and $\mathbf{t}$ by some constant.

2.4. Brightness Change Equation

The brightness of the image of a particular patch of a surface depends on many factors. It may for example vary with the orientation of the patch. In many cases, however, it remains at least approximately constant as the surface moves in the environment. If we assume that the image brightness of a patch remains constant, we have

$$\frac{dE}{dt} = 0,$$

or

$$\frac{\partial E}{\partial\mathbf{r}}\cdot\frac{d\mathbf{r}}{dt} + \frac{\partial E}{\partial t} = 0,$$

where $\partial E/\partial\mathbf{r} = (\partial E/\partial x, \partial E/\partial y, 0)^T$ is the image brightness gradient. It is convenient to use the notation $E_{\mathbf{r}}$ for this quantity and E_t for the time derivative of the brightness. Then we can write the brightness change equation in the simple form

$$E_{\mathbf{r}}\cdot\mathbf{r}_t + E_t = 0.$$

Substituting for $\mathbf{r}_t$ we obtain

$$E_t - E_{\mathbf{r}}\cdot\left(\hat{\mathbf{z}}\times\left(\mathbf{r}\times\left(\mathbf{r}\times\omega - \frac{1}{\mathbf{R}\cdot\hat{\mathbf{z}}}\mathbf{t}\right)\right)\right) = 0.$$

Now

$$E_{\mathbf{r}}\cdot(\hat{\mathbf{z}}\times(\mathbf{r}\times\mathbf{t})) = (E_{\mathbf{r}}\times\hat{\mathbf{z}})\cdot(\mathbf{r}\times\mathbf{t}) = \left((E_{\mathbf{r}}\times\hat{\mathbf{z}})\times\mathbf{r}\right)\cdot\mathbf{t},$$

and by similar reasoning

$$E_{\mathbf{r}}\cdot\left(\hat{\mathbf{z}}\times\left(\mathbf{r}\times(\mathbf{r}\times\omega)\right)\right) = \left(((\mathbf{E}_{\mathbf{r}}\times\hat{\mathbf{z}})\times\mathbf{r})\times\mathbf{r}\right)\cdot\omega,$$

so we have

$$E_t - \left(((E_{\mathbf{r}}\times\hat{\mathbf{z}})\times\mathbf{r})\times\mathbf{r}\right)\cdot\omega + \frac{1}{\mathbf{R}\cdot\hat{\mathbf{z}}}((\mathbf{E}_{\mathbf{r}}\times\hat{\mathbf{z}})\times\mathbf{r})\cdot\mathbf{t} = 0.$$

To make this constraint equation more compact, let us define $c = E_t$, $\mathbf{s} = (E_{\mathbf{r}}\times\hat{\mathbf{z}})\times\mathbf{r}$, and $\mathbf{v} = -\mathbf{s}\times\mathbf{r}$; then, finally,

$$c + \mathbf{v} \cdot \omega + \frac{1}{\mathbf{R} \cdot \hat{\mathbf{z}}} \mathbf{s} \cdot \mathbf{t} = 0.$$

This is the brightness change equation in the case of rigid body motion.

2.5. Planar Surface

A particularly impoverished scene is one consisting of a single planar surface. The equation for such a surface is

$$\mathbf{R} \cdot \mathbf{n} = 1,$$

where $\mathbf{n}/|\mathbf{n}|$ is a unit normal to the plane, and $1/|\mathbf{n}|$ is the perpendicular distance of the plane from the origin. Since $\mathbf{R} = (\mathbf{R} \cdot \hat{\mathbf{z}})\mathbf{r}$, we can write this as

$$\mathbf{r} \cdot \mathbf{n} = \frac{1}{\mathbf{R} \cdot \hat{\mathbf{z}}},$$

so the constraint equation becomes

$$c + \mathbf{v} \cdot \omega + (\mathbf{r} \cdot \mathbf{n})(\mathbf{s} \cdot \mathbf{t}) = 0.$$

This is the brightness change equation for a planar surface. Note again the inherent ambiguity in the constraint equation. It is satisfied equally well by two planes with the same orientation but at different distances provided that the translational velocities are in the same proportions.

2.6. Essential Parameters for Planar Surfaces

The brightness change equation can be written as

$$c + (\mathbf{r} \times \mathbf{s}) \cdot \omega + (\mathbf{r} \cdot \mathbf{n})(\mathbf{s} \cdot \mathbf{t}) = 0.$$

Using the identity $(\mathbf{r} \times \mathbf{s}) \cdot \omega = \mathbf{r} \cdot (\mathbf{s} \times \omega)$, we obtain

$$c - \mathbf{r} \cdot (\omega \times \mathbf{s}) + (\mathbf{r} \cdot \mathbf{n})(\mathbf{s} \cdot \mathbf{t}) = 0.$$

We now use the isomorphism between vectors and skew-symmetric matrices. Let us define

$$\mathbf{\Omega} = \begin{pmatrix} 0 & -\omega_3 & +\omega_2 \\ +\omega_3 & 0 & -\omega_1 \\ -\omega_2 & +\omega_1 & 0 \end{pmatrix},$$

then, $\mathbf{\Omega s} = (\omega \times \mathbf{s})$, and we conclude that

$$c - \mathbf{r}^T \mathbf{\Omega s} + (\mathbf{r}^{\mathrm{T}} \mathbf{n})(\mathbf{t}^{\mathrm{T}} \mathbf{s}) = 0,$$

or

$$c + \mathbf{r}^T(-\mathbf{\Omega} + \mathbf{n}\mathbf{t}^{\mathrm{T}})\mathbf{s} = 0.$$

If we define

$$\mathbf{P} = \begin{pmatrix} p_1 & p_2 & p_3 \\ p_4 & p_5 & p_6 \\ p_7 & p_8 & p_9 \end{pmatrix} = -\mathbf{\Omega} + \mathbf{n}\mathbf{t}^{\mathrm{T}},$$

we can finally write

$$c + \mathbf{r}^T \mathbf{P s} = 0.$$

We will refer to $\{p_i\}$ as the *essential parameters* (in agreement with Tsai and Huang [20]) since these parameters contain all the information about the planar surface and motion parameters. The above constraint equation is linear in the elements of $\mathbf{P}$. Several such equations, for different image points, can be used to solve for these parameters. We will show how the special structure of $\mathbf{P}$ can be exploited to recover the motion and plane parameters very easily.

Note that the essential parameters are not independent. This is because $\mathbf{P}$ is not an arbitary 3x3 matrix. It has a special structure as a result of the fact that it is the sum of a skew-symmetric matrix and a dyadic product. It takes three parameters to specify ω (and hence $\mathbf{\Omega}$), three to specify $\mathbf{n}$, and another three for $\mathbf{t}$. The matrix $\mathbf{P}$, however, is unchanged if we replace $\mathbf{n}$ by $k\mathbf{n}$ and $\mathbf{t}$ by $(1/k)\mathbf{t}$ for any nonzero k. Thus, there are actually only eight degrees of freedom, not nine.

Equivalently, we can say that there is one constraint on $\mathbf{P}$. Since $\mathbf{\Omega}^{\mathrm{T}} = -\mathbf{\Omega}$, it follows that

$$\mathbf{P}^* = \mathbf{P} + \mathbf{P}^T = \mathbf{n}\mathbf{t}^T + \mathbf{t}\mathbf{n}^T.$$

A dyadic product has rank one, or less. The sum of two dyadic products has at most rank two. So we conclude that

$$\det(\mathbf{P} + \mathbf{P}^T) = 0.$$

This constraint can be expressed in terms of the essential parameters as

$$p_1(p_5p_9 - p_6p_8) + p_4(p_3p_8 - p_2p_9) + p_7(p_2p_6 - p_3p_5) = 0.$$

We can use this equation, for example, to solve for p_9 given $p_1, p_2, ..., p_8$. It is difficult to use this equation directly when one attempts to find $\mathbf{P}$ from image brightness measurements.

There is a simple way around this problem, however. Note that $\mathbf{r}^T\mathbf{s} = 0$, because $\mathbf{s} = \left((E_r \times \hat{\mathbf{z}}) \times \mathbf{r}\right)$. So $\mathbf{r}^T\mathbf{I s} = 0$, and

$$c + \mathbf{r}^T(\mathbf{P} + l\mathbf{I})\mathbf{s} = 0,$$

for arbitrary l. If we let $\mathbf{P}' = \mathbf{P} + l\mathbf{I}$, we can write

$$c + \mathbf{r}^T\mathbf{P}'\mathbf{s} = 0,$$

and conclude that we cannot recover $\mathbf{P}$ from image brightness measurements alone. To find $\mathbf{P}$, we must impose the constraint $\det(\mathbf{P} + \mathbf{P}^T) = 0$. To avoid dealing directly with the resulting non-linear relation between the essential parameters, we first find any $\mathbf{P}'$ that satisfies the above brightness change constaint equation for all image points being considered, and then determine l such that $\mathbf{P} = \mathbf{P}' - l\mathbf{I}$ satisfies

$$\det(\mathbf{P} + \mathbf{P}^T) = 0.$$

Now,

$$\det(\mathbf{P} + \mathbf{P}^T) = \det(\mathbf{P}' + \mathbf{P}'^T - 2l\mathbf{I}) = 0,$$

so that $2l$ must be an eigenvalue of the real symmetric matrix

$$\mathbf{P}'^* = \mathbf{P}' + \mathbf{P}'^T.$$

It will become apparent, in the next section, that we ought to choose the middle one of the three real eigenvalues of $\mathbf{P}''$ for $2l$.

In summary, the overall plan is to find any matrix $\mathbf{P}'$ that satisfies the image brightness constraint equation,

$$c + \mathbf{r}^T \mathbf{P}' \mathbf{s} = 0,$$

at a suitable number of image points and consequently determine $\mathbf{P}''$. We can then solve for the middle eigenvalue of $\mathbf{P}''$ (which is $2l$) so as to construct the singular matrix $\mathbf{P} = \mathbf{P}' - l\mathbf{I}$, and from that we finally determine $\mathbf{n}$ and $\mathbf{t}$ as well as $\mathbf{\Omega}$ (and hence ω) using the relationship

$$\mathbf{P} = -\mathbf{\Omega} + \mathbf{n}\mathbf{t}^{\mathbf{T}}.$$

3. Recovering Essential Parameters

We are looking for a matrix $\mathbf{P}'$ that satisfies the brightness change equation,

$$c + \mathbf{r}^T \mathbf{P}' \mathbf{s} = 0,$$

at a chosen number of image points. Now,

$$\mathbf{r}^T \mathbf{P}' \mathbf{s} = \mathrm{Trace}\{(\mathbf{s}\mathbf{r}^T)\mathbf{P}'\},$$

or

$$\mathbf{r}^T \mathbf{P}' \mathbf{s} = \mathrm{Flat}(\mathbf{s}\mathbf{r}^T) \,.\, \mathrm{Flat}(\mathbf{P}'),$$

where $\mathrm{Flat}(\mathbf{M})$ is the vector obtained from the matrix $\mathbf{M}$ by adjoining its rows. So we can write the brightness change equation in the form

$$c + \mathbf{a}^T \mathbf{p}' = 0,$$

where

$$\mathbf{p}' = (p'_1, p'_2, \ldots, p'_9)^T,$$

$$\mathbf{a} = (r_1 s_1, r_1 s_2, r_1 s_3, r_2 s_1, r_2 s_2, r_2 s_3, r_3 s_1, r_3 s_2, r_3 s_3)^T.$$

We first consider finding $\mathbf{p}'$ from the image brightness derivatives at the minimum number of points necessary. Later, we consider instead a least-squares procedure that takes into account information in a whole image region.

From the derivatives of the brightness at the i^{th} image point considered, we can construct the vector $\mathbf{a}_i$ such that

$$\mathbf{a}_i^T \mathbf{p}' = -c_i.$$

As discussed above, there are really only eight independent degrees of freedom. So we can arbitrarily fix one of the components of the vector $\mathbf{p}'$. This means that we can solve for the other eight using constraint equations derived from eight image points.

Let $\mathbf{p}' = (p'_1, p'_2, \ldots, p'_8, 0)^T$ denote the solution obtained by setting the last element equal to zero. If we define

$$\tilde{\mathbf{p}}' = (p'_1, p'_2, \ldots, p'_8)^T,$$

$$\tilde{\mathbf{a}} = (r_1 s_1, r_1 s_2, r_1 s_3, r_2 s_1, r_2 s_2, r_2 s_3, r_3 s_1, r_3 s_2)^T,$$

then the above constraint equation reduces to

$$\tilde{\mathbf{a}}_i^T \tilde{\mathbf{p}}' = -c_i.$$

Using eight independent points, we can solve the following linear matrix equation:

$$\mathbf{A}\tilde{\mathbf{p}}' = -\mathbf{c},$$

where

$$\mathbf{A} = (\tilde{\mathbf{a}}_1, \tilde{\mathbf{a}}_2, \ldots, \tilde{\mathbf{a}}_8)^T, \quad \mathbf{c} = (c_1, c_2, \ldots, c_8)^T.$$

The solution of the above equation is

$$\tilde{\mathbf{p}}' = -\mathbf{A}^{-1}\mathbf{c}.$$

Image intensity values are corrupted with sensor noise and quantization. These inaccuracies are further accentuated by methods used for estimating the brightness gradient. Thus it is not advisable to base a method on measurements at just a few points. Instead we propose to minimize the error in the brightness constraint equation over the whole region I in the image plane. So we choose the vector $\tilde{\mathbf{p}}'$ that minimizes

$$\iint_I (\tilde{\mathbf{a}}^T \tilde{\mathbf{p}}' + c)^2 \, dx \, dy.$$

The solution, in this case, is given by

$$\tilde{\mathbf{p}}' = -\left(\iint_I \tilde{\mathbf{a}}\tilde{\mathbf{a}}^T \, dx \, dy\right)^{-1} \left(\iint_I c\tilde{\mathbf{a}} \, dx \, dy\right).$$

In either case, we construct $\mathbf{p}'$ by adjoining a zero to the vector $\tilde{\mathbf{p}}'$. The result immediately gives us the matrix $\mathbf{P}'$. We determine the eigenvalues of $\mathbf{P}''$ so that we can construct $\mathbf{P}^{\circ}$ by subtracting the identity matrix times twice the middle eigenvalue from $\mathbf{P}''$. We can also determine $\mathbf{P}$ by subtracting the identity matrix times the middle eigenvalue from $\mathbf{P}'$. At this point, we are ready to recover $\mathbf{t}$, ω, and $\mathbf{n}$.

Note that we do not have to repeat the eigenvalue-eigenvector analysis, since $\mathbf{P}^{\circ}$ has the same eigenvectors as $\mathbf{P}''$, and its eigenvalues are merely shifted so as to make the middle one equal to zero. This follows from the fact that if $\mathbf{u}$ and λ are an eigenvector-eigenvalue pair of $\mathbf{P}''$, that is,

$$\mathbf{P}''\mathbf{u} = \lambda\mathbf{u},$$

then $\mathbf{u}$ and $(\lambda - 2l)$ are an eigenvector-eigenvalue pair of $\mathbf{P}^{\circ}$, since

$$\mathbf{P}^{\circ}\mathbf{u} = (\mathbf{P}'' - 2l\mathbf{I})\mathbf{u} = (\lambda - 2l)\mathbf{u}.$$

4. Recovering Motion and Structure

We now show how to compute the parameters of the translational motion and the plane orientation from the essential parameters. When we have done this, we will be able to also find the rotational parameters using

$$\mathbf{\Omega} = \mathbf{n}\mathbf{t}^{\mathbf{T}} - \mathbf{P}.$$

As we saw before

$$\mathbf{P}^{\circ} = \mathbf{P} + \mathbf{P}^T = \mathbf{t}\mathbf{n}^T + \mathbf{n}\mathbf{t}^T,$$

since Ω is skew-symmetric. Let us use the notation $\sigma = |\mathbf{n}|\,|\mathbf{t}|$, and $\tau = \hat{\mathbf{n}} \cdot \hat{\mathbf{t}}$, where

$$\hat{\mathbf{n}} = \frac{\mathbf{n}}{|\mathbf{n}|}, \quad \text{and} \quad \hat{\mathbf{t}} = \frac{\mathbf{t}}{|\mathbf{t}|},$$

are unit vectors in the directions of the surface normal and the translation vector, respectively. Then,

$$\text{Trace}(\mathbf{P}') = \text{Trace}(\mathbf{P}) + \text{Trace}(\mathbf{P}^T) = 2\mathbf{n} \cdot \mathbf{t} = 2\sigma\tau.$$

It turns out that $\hat{\mathbf{n}}$ and $\hat{\mathbf{t}}$ can be easily recovered from the eigenvectors of the matrix $\mathbf{P}'$. In the following lemma, we show that the eigenvectors of $\mathbf{P}'$ are combinations of the sought after vectors $\hat{\mathbf{n}}$ and $\hat{\mathbf{t}}$.

Lemma 1: Let $\mathbf{P}' = \mathbf{U}\mathbf{\Lambda}\mathbf{U}^{\mathbf{T}}$ be the eigenvalue decomposition of $\mathbf{P}' = (\mathbf{t}\mathbf{n}^T + \mathbf{n}\mathbf{t}^T)$. If $\mathbf{n}$ is not parallel to $\mathbf{t}$. Then,

$$\mathbf{\Lambda} = \text{Diag}\Big(\sigma(\tau - \mathbf{1})\ ,\ \mathbf{0}\ ,\ \sigma(\tau + \mathbf{1})\Big),$$

and,

$$\mathbf{U} = \left[\frac{\hat{\mathbf{t}} - \hat{\mathbf{n}}}{\sqrt{2(1-\tau)}} \quad \frac{\hat{\mathbf{t}} \times \hat{\mathbf{n}}}{\sqrt{1-\tau^2}} \quad \frac{\hat{\mathbf{t}} + \hat{\mathbf{n}}}{\sqrt{2(1+\tau)}} \right].$$

Proof: Note that

$$\mathbf{P}' = \sigma(\hat{\mathbf{t}}\hat{\mathbf{n}}^T + \hat{\mathbf{n}}\hat{\mathbf{t}}^T).$$

Now $(\hat{\mathbf{t}} \times \hat{\mathbf{n}})$ is the eigenvector with eigenvalue zero since

$$\mathbf{P}'(\hat{\mathbf{t}}\times\hat{\mathbf{n}}) = \sigma(\hat{\mathbf{t}}\hat{\mathbf{n}}^T + \hat{\mathbf{n}}\hat{\mathbf{t}}^T)(\hat{\mathbf{t}}\times\hat{\mathbf{n}}) = \sigma\hat{\mathbf{t}}\left[\hat{\mathbf{n}}\hat{\mathbf{t}}\hat{\mathbf{n}}\right] + \sigma\hat{\mathbf{n}}\left[\hat{\mathbf{t}}\hat{\mathbf{n}}\hat{\mathbf{t}}\right] = 0.$$

Since $\mathbf{P}'$ is real symmetric, it has 3 orthogonal eigenvectors. The other two eigenvectors must, therefore, be in the plane containing $\hat{\mathbf{t}}$ and $\hat{\mathbf{n}}$. Let $\mathbf{u} = \alpha\hat{\mathbf{t}} + \beta\hat{\mathbf{n}}$ and λ denote an eigenvector-eigenvalue pair for some α and β (to be determined). Then,

$$\sigma(\hat{\mathbf{t}}\hat{\mathbf{n}}^T + \hat{\mathbf{n}}\hat{\mathbf{t}}^T)(\alpha\hat{\mathbf{t}} + \beta\hat{\mathbf{n}}) = \lambda(\alpha\hat{\mathbf{t}} + \beta\hat{\mathbf{n}}),$$

that becomes

$$\sigma[\alpha(\hat{\mathbf{t}}\cdot\hat{\mathbf{n}}) + \beta(\hat{\mathbf{n}}\cdot\hat{\mathbf{n}})]\hat{\mathbf{t}} + \sigma[\alpha(\hat{\mathbf{t}}\cdot\hat{\mathbf{t}}) + \beta(\hat{\mathbf{t}}\cdot\hat{\mathbf{n}})]\hat{\mathbf{n}} = \lambda\alpha\hat{\mathbf{t}} + \lambda\beta\hat{\mathbf{n}}.$$

Since $(\hat{\mathbf{t}} \cdot \hat{\mathbf{n}}) = \tau$, we can write

$$\begin{pmatrix} \sigma\tau - \lambda & \sigma \\ \sigma & \sigma\tau - \lambda \end{pmatrix} \begin{pmatrix} \alpha \\ \beta \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}.$$

For this pair of homogeneous equations to have a non-trivial solution for α and β, the determinant of the 2x2 coefficient matrix must be zero, that is,

$$(\sigma\tau - \lambda)^2 - \sigma^2 = 0,$$

or

$$\lambda = \sigma(\tau \pm 1).$$

Substituting for λ into the earlier equations, we obtain

$$\alpha = \pm\beta.$$

Note that $\sigma(\tau - 1) < 0$ and $\sigma(\tau + 1) > 0$ because $|\tau| < 1$, as it is the cosine of the angle between $\hat{\mathbf{n}}$ and $\hat{\mathbf{t}}$. So one eigenvalue is negative and one is positive (This is why we choose to make the middle eigenvalue zero when constructing $\mathbf{P}'$ from $\mathbf{P}''$). We find that eigenvectors corresponding to the eigenvalues $\lambda_1 = \sigma(\tau - 1)$ and $\lambda_3 = \sigma(\tau + 1)$ are $\hat{\mathbf{t}} - \hat{\mathbf{n}}$ and $\hat{\mathbf{t}} + \hat{\mathbf{n}}$, respectively. If we normalize these, we obtain the unit vectors

$$\mathbf{u}_1 = \frac{\hat{\mathbf{t}} - \hat{\mathbf{n}}}{\sqrt{2(1-\tau)}} \quad \text{and} \quad \mathbf{u}_3 = \frac{\hat{\mathbf{t}} + \hat{\mathbf{n}}}{\sqrt{2(1+\tau)}}.$$

Note that we can determine $\sigma = |\mathbf{n}|\,|\mathbf{t}|$ from

$$\sigma = \frac{1}{2}(\lambda_3 - \lambda_1).$$

∎

The equations for $\mathbf{u}_1$ and $\mathbf{u}_3$ are linear in $\hat{\mathbf{t}}$ and $\hat{\mathbf{n}}$, and so can be easily solved for these vectors:

$$\hat{\mathbf{n}} = \sqrt{\frac{1}{2}(1+\tau)}\mathbf{u}_3 - \sqrt{\frac{1}{2}(1-\tau)}\mathbf{u}_1,$$

$$\hat{\mathbf{t}} = \sqrt{\frac{1}{2}(1+\tau)}\mathbf{u}_3 + \sqrt{\frac{1}{2}(1-\tau)}\mathbf{u}_1.$$

The sign of the eigenvectors are arbitrary. If we change the sign of $\mathbf{u}_1$, we obtain instead

$$\hat{\mathbf{n}} = \sqrt{\frac{1}{2}(1+\tau)}\mathbf{u}_3 + \sqrt{\frac{1}{2}(1-\tau)}\mathbf{u}_1,$$

$$\hat{\mathbf{t}} = \sqrt{\frac{1}{2}(1+\tau)}\mathbf{u}_3 - \sqrt{\frac{1}{2}(1-\tau)}\mathbf{u}_1,$$

where $\hat{\mathbf{n}}$ and $\hat{\mathbf{t}}$ are interchanged. This is the dual solution.

The signs of the two eigenvectors can be chosen independently. This might suggest that there are a total of four different solutions for $\hat{\mathbf{n}}$ and $\hat{\mathbf{t}}$. We show next that two of these solutions can be discarded because they correspond to viewing the planar surface "from behind". We assume that the visible part of the plane is the bounding surface of some solid object. We chose to define the orientation of the surface using the inward pointing normal $\mathbf{n}$. The equation of the plane is $\mathbf{R} \cdot \mathbf{n} = 1$, or $(\mathbf{r} \cdot \mathbf{n})(\mathbf{R} \cdot \hat{\mathbf{z}}) = 1$, since

$$\mathbf{R} = (\mathbf{R} \cdot \hat{\mathbf{z}})\mathbf{r}.$$

Now, $\mathbf{R} \cdot \hat{\mathbf{z}} = Z$ is positive for points in front of the viewer, and so $\mathbf{r} \cdot \mathbf{n}$ must be positive for points on the visible portion of the plane. The equation $\mathbf{r} \cdot \mathbf{n} = 0$ corresponds to a line in the image. Points on one side of this line, for which $\mathbf{r} \cdot \mathbf{n} > 0$, can be images of points on the plane defined by the inward pointing normal $\mathbf{n}$. Conversely, points on the other side of the line, where $\mathbf{r} \cdot \mathbf{n} < 0$, can not. They can be thought of as images of points on a parallel but oppositely oriented plane corresponding to the vector $-\mathbf{n}$. We are analyzing brightness gradients for a particular image region. If $\mathbf{r} \cdot \mathbf{n} > 0$ for points in this region, then $\mathbf{n}$ is a possible solution for the surface normal. If $\mathbf{r} \cdot \mathbf{n} < 0$ for points in this region, then $-\mathbf{n}$ is a possible solution. If $\mathbf{r} \cdot \mathbf{n} > 0$ for some points and $\mathbf{r} \cdot \mathbf{n} < 0$ for others, then we are not dealing with the image of a single planar surface.

Also, note that we can recover $\mathbf{t}$ and $\mathbf{n}$ up to a scale factor. We can let $\mathbf{t}$ to be a unit vector without loss of generality. Then, $\mathbf{n}$ can be found as follows:

$$\mathbf{n} = |\mathbf{n}|\,\hat{\mathbf{n}} = |\mathbf{n}|\,|\mathbf{t}|\,\hat{\mathbf{n}} = \sigma \mathbf{n},$$

using the known value of σ.

So far, we have assumed that $\mathbf{n}$ and $\mathbf{t}$ are not parallel. In the special case that $\hat{\mathbf{t}} \parallel \hat{\mathbf{n}}$, we have

$$\mathbf{P}' = \sigma(\hat{\mathbf{t}}\hat{\mathbf{n}}^T + \hat{\mathbf{n}}\hat{\mathbf{t}}^T) = 2\sigma\hat{\mathbf{n}}\hat{\mathbf{n}}^T.$$

This dyadic product has rank one, that is it only has one non-zero eigenvalue. This is easy to show since any vector perpendicular to $\hat{\mathbf{n}}$ is an eigenvector with zero eigenvalue. Also, $\hat{\mathbf{n}}$ is an eigenvector with eigenvalue 2σ.

So if we find that $\mathbf{P}''$ has two equal eigenvalues (that is $\mathbf{P}'$ has two zero eigenvalues), then we conclude that $\hat{\mathbf{n}}$ and $\hat{\mathbf{t}}$ are parallel and equal to the eigenvector corresponding to the remaining eigenvalue.

We then solve for the rotation parameters by substituting the solutions for $\mathbf{n}$ and $\mathbf{t}$ into the equation

$$\boldsymbol{\Omega} = \mathbf{n}\mathbf{t}^{\mathbf{T}} - \mathbf{P}.$$

Even though we gave a complete and compact proof of the dual solution in an earlier paper[15], it is intriguing to confirm those results with our closed form solution. There, we showed that the two solutions are related by

$$\mathbf{n}' = |\mathbf{n}|\,\mathbf{t}, \quad \mathbf{t}' = \frac{\mathbf{n}}{|\mathbf{n}|}, \quad \omega' = \omega + \mathbf{n} \times \mathbf{t},$$

where we have arbitrarily set $|\mathbf{t}| = 1$. The two solutions given earlier for $\mathbf{n}$ and $\mathbf{t}$ already satisfy the duality relationship given above. The identity

$$(\mathbf{n}\mathbf{t}^T - \mathbf{t}\mathbf{n}^T)\mathbf{x} = \mathbf{x} \times (\mathbf{n} \times \mathbf{t}),$$

holds for any vector $\mathbf{x}$. Using this in

$$\omega' \times \mathbf{x} = (\omega + \mathbf{n} \times \mathbf{t}) \times \mathbf{x} = \omega \times \mathbf{x} + (\mathbf{n} \times \mathbf{t}) \times \mathbf{x},$$

we arrive at

$$\omega' \times \mathbf{x} = \omega \times \mathbf{x} - (\mathbf{n}\mathbf{t}^{\mathbf{T}} - \mathbf{t}\mathbf{n}^{\mathbf{T}})\mathbf{x},$$

or

$$\boldsymbol{\Omega}'\mathbf{x} = (\boldsymbol{\Omega} - \mathbf{n}\mathbf{t}^{\mathbf{T}} + \mathbf{t}\mathbf{n}^{\mathbf{T}})\mathbf{x}.$$

If this is to be true for all vectors $\mathbf{x}$, we must have

$$\boldsymbol{\Omega}' = \boldsymbol{\Omega} - \mathbf{n}\mathbf{t}^{\mathbf{T}} + \mathbf{t}\mathbf{n}^{\mathbf{T}}.$$

So, we finally obtain

$$-\boldsymbol{\Omega}' + \mathbf{n}'\mathbf{t}' = -\boldsymbol{\Omega} + \mathbf{n}\mathbf{t}^{\mathbf{T}} - \mathbf{t}\mathbf{n}^{\mathbf{T}} + \mathbf{t}\mathbf{n}^{\mathbf{T}},$$

or,

$$-\boldsymbol{\Omega}' + \mathbf{n}'\mathbf{t}' = -\boldsymbol{\Omega} + \mathbf{n}\mathbf{t}^{\mathbf{T}} = \mathbf{P}.$$

We conclude that $\mathbf{n}'$, $\mathbf{t}'$, and ω', as defined above, constitute a second solution since they lead to the same set of essential parameters.

5. Summary

In this paper, we presented a closed form solution for recovering the motion of an observer with respect to a planar surface without having to compute the optical flow as an intermediate step. We need the image intensity gradients at a minimum of 8 points. However, in general, it is better to compute gradients in a larger portion of the image to reduce the noise effects. We first employed a constraint equation we developed for planar surfaces to compute 9 intermediate parameters, the elements of a 3x3 matrix. We referred to them as *essential parameters*. The special structure of this matrix allows us to compute the motion and plane parameters easily.

6. Acknowledgements

This report describes research done at the Artificial Intelligence Laboratory of the Massachusetts Institute of Technology and the Robotics Group of the University of Hawaii. Support for the MIT artificial intelligence research is provided in part by the Advanced Research Projects Agency of the Department of Defense under office of Naval Research contract N00014-75-C-0643 and in part by System Development Foundation. Support for the University of Hawaii's Robotics Group is provided in part by the National Science Foundation.

References

[1] Adiv, G., "Determining 3-D Motion and Structure from Optical Flow Generated by Several Moving Objects," COINS TR 84-07, Computer and Information Science, University of Massachusetts, Amherst, MA, April 1984.

[2] Aloimonos, J., Chou, P. B., "Detection of Surface Orientation and Motion from Texture: 1. The Case of Planes," TR 161, Department of Computer Science, University of Rochester, Rochester, NY, January 1985.

[3] Ballard, D.H., Kimball, O.A., "Rigid Body Motion from Depth and Optical Flow," TR 70, Computer Science Department, University of Rochester, Rochester, NY, 1981.

[4] Bruss, A.R., Horn, B.K.P., "Passive Navigation," *Computer Vision, Graphics, and Image Processing*, Vol. 21, pp. 3–20, 1983.

[5] Buxton, B.F., et al., "3D Solution to the Aperature problem," *Proceedings of the Sixth European Conference on Artificial Intelligence*, pp. 631–640, September 1984.

[6] Fennema, C.L., Thompson W.B., "Velocity Determination in Scenes Containing Several Moving Objects," *Computer Graphics and Image Processing*, 9, pp. 301–315, 1979.

[7] Hildreth, E.C., *The Measurement of Visual Motion*, MIT Press, 1983.

[8] Horn, B.K.P., Schunck, B.G., "Determining Optical Flow," *Artificial Intelligence*, Vol. 17, pp. 185–203, 1981.

[9] Kanatani, K., "Detecting the Motion of a Planar Surface by Line and Surface Integrals," *Computer Vision, Graphics, and Image Processing*, 29, pp. 13–22, 1985.

[10] Longuet-Higgins, H.C., Prazdny, K., "The Interpretation of a Moving Retinal Image," *Proceedings of the Royal Society of London*, Series B, Vol. 208, pp. 385–397, 1980.

[11] Longuet-Higgins, H.C., "A Computer Algorithm for Reconstructing a Scene from Two Projections," *Nature*, Vol. 293, pp. 131–133, 1981.

[12] Longuet-Higgins, H.C., "The Visual Ambiguity of a Moving Plane," *Proceddings of the Royal Society of London*, B 223, pp. 165–175, 1984.

[13] Nagel, H., "On the Derivation of 3D Rigid Point Configurations from Image Sequences," *Proceedings of Pattern Recognition and Image Processing*, Dallas, Texas, 1981.

[14] Negahdaripour, S., Horn, B.K.P., "Determining 3-D Motion of Planar Objects from Brightness Patterns," *Proceedings of Ninth Int. Joint Conf. on A.I.*, pp. 898–901, 1985.

[15] Negahdaripour, S., Horn, B.K.P., "Direct Passive Navigation ," to appear in *IEEE Trans. Pattern Analysis and Machine Intelligence* (also A.I. Memo No. 821, M.I.T. A.I. Lab, Cambridge, MA, February 1985).

[16] Roach, J.W., Aggarwal, J.K., "Determinig the Movement of Objects from a Sequence of Images," *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. PAMI-2, November 1980.

[17] Subbarao, M., Waxman, A.M., "On the Uniqueness of Image Flow Solutions for Planar Surfaces in Motion," CAR-TR-114, Computer Vision Laboratory, Center for Automation Research, University of Maryland, College Park, MD, April 1985.

[18] Sugihara, K., Sugie, N., "Recovery of Rigid Structure from Orthographically Projected Optical Flow," TR 8304, Department of Information Science, Nagoya University, Nagoya, Japan, October 1983.

[19] Tsai, R.Y., Huang, T.S., Zhu, W.L., "Estimating Three-Dimensional Motion Parameters of a Rigid Planar Patch, II: Singular Value Decomposition," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, Vol. ASSP-30, No. 4, August 1982.

[20] Tsai, R.Y., Huang, T.S., "Uniqueness and Estimation of Three-Dimensional Motion Parameters of Rigid Objects with Curved Surfaces," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-6, No. 1, January 1984.

[21] Waxman, A.M., Ullman, S., "Surface Structure and 3-D Motion from Image Flow: A Kinematic Analysis," CAR-TR-24, Computer Vision Laboratory, Center for Automation Research, University of Maryland, College Park, MD, October 1983.

[22] Waxman, A.M., Wohn, K. "Contour Evolution, Neighborhood Deformation and Global Image Flow: Planar Surfaces in Motion," CAR-TR-58, Computer Vision Laboratory, Center for Automation Research, University of Maryland, College Park, MD, April 1984.

DIRECT PASSIVE NAVIGATION: ANALYTICAL SOLUTION FOR PLANES

S. Negahdaripour
B. K. P. Horn

IEEE COMPUTER SOCIETY REPRINT

Reprinted from PROCEEDINGS OF THE IEEE INTERNATIONAL CONFERENCE ON ROBOTICS AND AUTOMATION, San Francisco, California, April 7-10, 1986

IEEE COMPUTER SOCIETY
1730 Massachusetts Avenue, N.W.
Washington, D.C. 20036-1903

IEEE COMPUTER SOCIETY PRESS

THE INSTITUTE OF ELECTRICAL AND ELECTRONICS ENGINEERS