An Image Based Approach to Pose Estimation

S. H. Or K. H. Wong T. K. Lao T. T. Wong

Department of Computer Science & Engineering, The Chinese University of Hong Kong, Shatin, NT, Hong Kong, China

{shor,khwong,tklao,ttwong}@cse.cuhk.edu.hk

Abstract

We developed an algorithm which incorporates the pose estimation algorithm into the image matching problem. First we formulate the warping function which maps the image coordinates of a point to its transformed counterpart. The Jacobian which relates the image coordinates with the motion information is then derived. Finally, using the sum of squared differences measure, we link up the motion information with the intensity gradients. The motion recovery can thus be proceeded as an iterative process which takes account of the intensity profile changes across frames. An important contribution here is that we showed that the classical *Longuet-Higgins* equation of image motion is actually another form of a full projective formulation of *Lowe*'s Gauss Newton method–another well known pose estimation algorithm. The Longuet–Higgins equation is applied to the pose estimation problem in the framework of Lowe's formulation. The result is significant in that the classical Lowe's algorithm can now be applied to the direct estimation problem, which implies more flexibility in tracking complicated object such as articulated ones. On the other hand, an improved formulation of the Longuet–Higgins equation in handling large displacement can thus be applied to various applications.

keyword : motion analysis, pose estimation, optic flow, Sum of Squared Differences, image warping

1 Motivation

Most object tracking systems are operated by first establishing the correspondences of all the feature points across the image frames. This process is usually performed without the help of global information such as object movement in mind [9]. The establishment of correspondences are usually based on some similarity measures in intensity patterns of two images [4]. The displacements of the feature points are then used by a motion estimation algorithm to obtain the result. Most researchers adopt this approach [5, 11, 15, 3, 16, 6]and the state-of-the-art techniques in pose estimation perform very well [11, 15, 3, 6, 1]. By segmenting the two stages apart, the difficulty of the tracking problem is reduced significantly. In particular, the correspondence problem is an open problem which cannot be solved without making further assumptions about the scene [4].

Various techniques can be used to locate the correspondences across two images [12]. One generally would divide those methods into two main categories, namely the *sparse feature-based* approach [12] and the *dense correspondences* [4] approach. In feature-based approach, a set of feature points which have the most salient image characteristics are chosen from the first image. The correspondences of these feature points are located by placing some kinds of matching template in a search window in the second image. The most popular template based method is the normalized correlation. The drawback of feature based approach is that the size of the searching window will have great impact on the matching result – too big a window would generate many possible candidates while too small the search window may result in wrong or no match. The optical flow method [4], which is another more widely used term for dense correspondences method, establishes the correspondences of all pixels in the image by minimizing the differences in gray levels of the displaced pixels with that in previous image. The optic flow technique has a drawback known as the *aperture problem* [8] and that the displacements of the feature points should not be large. More global information i.e. smoothness constraint is thus needed to make the objective function into convex one to find the solution [8]. To combat the unwanted smoothing-out effect at the flow discontinuities, robust objective functions are introduced and good results are obtained [4].

Nevertheless, this approach of separate operating stages in motion recovery works well provided that the tracking environment is a non-cluttered one and the captured image is not noisy. A noisy image and cluttered environment would generate a lot of false matches [9]. Without the global information of what geometric characteristics the feature point should share, correspondences methods we described would easily select the wrong matches. This in turn causes the motion estimation algorithm to report a wrong motion of the object. The wrong prediction of the object position would then be used by the feature tracker

in the subsequent images, resulting in a divergent behavior of the overall tracking system.

In this paper, we proposed a new framework which combines the 3D motion estimation and correspondence establishment into one integral process. By combining these two processes, stability of tracking should be guaranteed since it takes account of the global matching criteria during the correspondence establishment. In addition, we show that the full projective formulation [2] of the classic Guass-Newton object tracking algorithm proposed by Lowe [11] can be viewed as another formulation of the image motion equation derived by Longuet-Higgins and Prazdny [10]. The primary contribution here is that the framework proposed by Lowe with stabilization can thus be applied to image matching based on intensity. An advantage utilizing the framework proposed by Lowe is its generality. Lowe's method can be easily extended to estimate both the camera focal length as well as non-rigid objects. We would expect more powerful non-rigid tracking algorithms can be derived from the results presented in this paper.

Basically our idea is using the image based rendering techniques [14] to warp an image such that the resulting image is as closely resemble the matching image as possible. By reformulating the sum of squared differences(SSD) measure in optical flow based method to allow for recovery of 3D motion, we can determine the 3D motion parameters which best explain the changes in intensity values in the image. This may help as the last stage in motion analysis to refine the results obtained by existing pose estimation techniques [6, 11, 3, 15]. We believe this would bridge up the two stages and result in more stable tracking behavior.

Our paper is organized as follow. In section 2, we will briefly review the techniques related to our algorithm. In section 3, we derive the Jacobian that relates the changes in intensity values to the 3D motion parameters. Various experiments which evaluate the performance of the derived Jacobian are discussed in section 4. In section 5 we present an algorithm for image matching of 3D objects using our derived results. Finally some discussions will be presented in section 5.

2 Optic Flow using SSD

The sum of squared differences measure [12, 4] is widely used in intensity based matching method to determine image correspondence. In this method, one would try to minimize the differences between two image patches $I_1(\mathbf{p}')$ and $I_2(\mathbf{p})$ at image point $\mathbf{p}'$ and $\mathbf{p}$ respectively:

$$E(\mathbf{q}) = \sum_i [I_1(\mathbf{p}'_i) - I_2(\mathbf{p}_i)]^2$$

$$= \sum_i [g_i^{\mathrm{T}} J_i^{\mathrm{t}} \mathbf{q} + \mathbf{e}_i]^2$$

where

$$\mathbf{e}_i = I_1(\mathbf{p}_i) - I_2(\mathbf{p}_i), \quad (1)$$

$$\mathbf{g}_i^{\mathrm{T}} = \nabla I_1(\mathbf{p}'_i). \quad (2)$$

In the above, $\mathbf{q}$ is the parameter vector which defines the transformation between the two images, $\mathbf{g}_i^{\mathrm{T}}$ denotes the transpose of the image gradient vector $\mathbf{g}_i$, and $\mathbf{J}_i$ is the Jacobian, $\mathbf{J}_i = \left(\frac{\partial x}{\partial \mathbf{q}}, \frac{\partial y}{\partial \mathbf{q}}\right)^{\mathrm{T}}$, where x and y is the coordinates of the image point. previous iteration [14]. It is well known that the above equation has a simple solution given by [14]

$$\mathbf{A}\mathbf{q} = -\mathbf{b}, \quad (3)$$

where

$$\mathbf{A} = \sum_i \mathbf{J}_i \mathbf{g}_i \mathbf{g}_i^{\mathrm{t}} \mathbf{J}_i^{\mathrm{t}}, \quad (4)$$

$$\mathbf{b} = \sum_i \mathbf{e}_i \mathbf{J}_i \mathbf{g}_i. \quad (5)$$

3 Our Approach

Consider a point $\mathbf{P} = (X, Y, Z, 1)^{\mathrm{T}}$ in 3D space, which is being transformed by a 3×3 rotation matrix $\mathbf{R}$ followed by a 3×1 translation vector $\mathbf{T}$ yielding $\mathbf{P}'$, we can write their relationship in homogeneous coordinate [7] as

$$\mathbf{P}' = \begin{pmatrix} X' \\ Y' \\ Z' \\ 1 \end{pmatrix} = \begin{pmatrix} \mathbf{R} & \mathbf{T} \\ 0 & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} = \mathbf{D}\mathbf{P}. \quad (6)$$

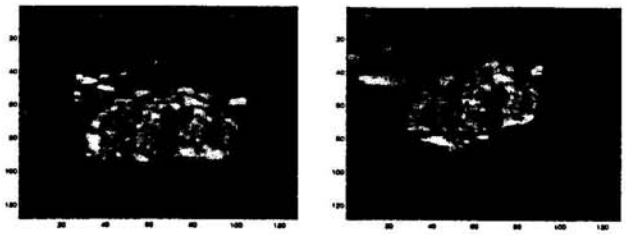
Assuming that the camera is placed at the origin, the 2D image projections of $\mathbf{P}$ and $\mathbf{P}'$ are given by

$$\begin{pmatrix} dx_i \\ dy_i \\ d \\ 1 \end{pmatrix} = \begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} = \mathbf{H}\mathbf{P}, \quad (7)$$

and

$$\begin{pmatrix} d'x'_i \\ d'y'_i \\ d' \\ 1 \end{pmatrix} = \mathbf{H}\mathbf{P}' = \mathbf{H}\mathbf{D}\mathbf{H}^{-1} \begin{pmatrix} dx_i \\ dy_i \\ d \\ 1 \end{pmatrix} = \mathbf{M} \begin{pmatrix} dx_i \\ dy_i \\ d \\ 1 \end{pmatrix}, \quad (8)$$

where f is the focal length of the camera, d and d' are the depth values of the original and transformed point respectively. The above equation relates the positional changes of an image point after a rigid transformation given its depth value. It is thus possible to generate a novel image from



(a) (b)

Figure 1: Image warping of a 128 by 128 image. a) Original dragon image, b) Warped image.

the 3D transformation specified. One should note that the above proposal is valid only when the inverse exists for the transfer matrix $\mathbf{M}$, i.e, $|\mathbf{M}| \neq 0$.

Fig. 1 illustrates some results of applying the above simple algorithm to a sample image with a given depth map. From the novel view generated, it can be seen that the details of the dragon is accurately predicted under the novel pose. Some occluded area are revealed when the original depth map is transformed to the novel pose, but they are being smoothed out by the super-sampling process. The above method showed the potential in application for solving the pose estimation problem – since we can generate an image of an object under different poses, it should also be possible to compare these 'warped' version with the now obtained image to see any significant difference exists. If the warped image agrees with the obtained image to a certain extent, it should be highly probable that the object in the scene has undergone a similar rigid transformation. The above reasoning led to our proposal in the subsequent section.

Consider Eq. 8 again, assuming a very small increment both in rotation ($\mathbf{R}_\omega$) and translation ($\Delta\mathbf{T}$), we have

$$\begin{pmatrix} d'x'_i \\ d'y'_i \\ d' \\ 1 \end{pmatrix} = \mathbf{H} \begin{pmatrix} (\mathbf{I}+\mathbf{R}_\omega)\mathbf{R} & \mathbf{T}+\Delta\mathbf{T} \\ \mathbf{0} & 1 \end{pmatrix} \mathbf{H}^{-1} \begin{pmatrix} dx_i \\ dy_i \\ d \\ 1 \end{pmatrix} \tag{9}$$

Since $\mathbf{H}$ is a diagonal matrix, we have

$$\mathbf{H}^{-1} = \begin{pmatrix} \mathbf{V}^{-1} & \mathbf{0} \\ \mathbf{0} & 1 \end{pmatrix} \quad \text{where} \quad \mathbf{V} = \begin{pmatrix} f & 0 & 0 \\ 0 & f & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{10}$$

Expanding the R.H.S of Eq. 9 and rearranging terms, we have

$$(\mathbf{I} + \mathbf{H} \begin{pmatrix} \mathbf{R}_\omega & \mathbf{0} \\ \mathbf{0} & 0 \end{pmatrix} \mathbf{H}^{-1}) \mathbf{M} \begin{pmatrix} dx_i \\ dy_i \\ d \\ 1 \end{pmatrix} + \begin{pmatrix} \delta X \\ \delta Y \\ \delta Z \\ 0 \end{pmatrix}, \tag{11}$$

where

$$\delta\mathbf{P} = \begin{pmatrix} \delta X \\ \delta Y \\ \delta Z \end{pmatrix} = \mathbf{V}(\Delta\mathbf{T} - \mathbf{R}_\omega\mathbf{T}). \tag{12}$$

We thus have the following compact form for a new $\mathbf{M}$ which due to small changes in $\mathbf{D}$

$$\begin{aligned} \mathbf{M}' &= (\mathbf{I} + \mathbf{D}_M)\mathbf{M}, \qquad (13) \\ \mathbf{D}_M &= \mathbf{H} \begin{pmatrix} \mathbf{R}_\omega & \mathbf{0} \\ \mathbf{0} & 0 \end{pmatrix} \mathbf{H}^{-1} + \begin{pmatrix} \mathbf{0} & \delta\mathbf{P} \\ \mathbf{0} & 0 \end{pmatrix} M^{-1}. \end{aligned}$$

When small rotation is assumed, we can reduce the parameter space by using the incremental rotation matrix in place of $\mathbf{R}_\omega$

$$\mathbf{R} \leftarrow \mathbf{R}(\mathbf{I} + \mathbf{R}_\omega), \tag{14}$$

$$\mathbf{R}_\omega = \begin{pmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{pmatrix}. \tag{15}$$

As a result, we use only a six parameters vector $\mathbf{q} = \{\omega_x, \omega_y, \omega_z, u, v, w\}$ to control the warping of the image where $\{u, v, w\}$ is the incremental translation vector, $\{\omega_x, \omega_y, \omega_z\}$ is the incremental rotation.

We need to determine the Jacobian of the image coordinates with respect to the parameter vector. Consider an image formed from an incremental rigid transformation, from eq. (13) we have

$$\begin{pmatrix} d'x'_i \\ d'y'_i \\ d' \\ 1 \end{pmatrix} = (\mathbf{I} + \mathbf{D}_M)\mathbf{M} \begin{pmatrix} dx_i \\ dy_i \\ d \\ 1 \end{pmatrix}. \tag{16}$$

Minimizing the differences between the two images can be interpreted as minimizing the differences between the second image frame and a warped version of the original image defined by M(we denote the warped image as $\tilde{I}_1$).

From Eq. 16, the warped coordinates $(d'x', d'y', d', 1)^{\mathrm{T}} = (\mathbf{I} + \mathbf{D}_M)(d''x'', d''y'', d'', 1)^{\mathrm{T}}$ are given by

$$\begin{pmatrix} x' \\ y' \end{pmatrix} = \begin{pmatrix} \frac{d''x'' - \omega_z d''y'' + f\omega_y d'' + f(u + \omega_z T_y - \omega_y T_z)}{-\frac{\omega_y d''x''}{f} + \frac{\omega_x d''y''}{f} + d'' + w + \omega_y T_x - \omega_x T_y} \\ \frac{\omega_z d''x'' + d''y'' - f\omega_x d'' + f(v - \omega_z T_x + \omega_x T_z)}{-\frac{\omega_y d''x''}{f} + \frac{\omega_x d''y''}{f} + d'' + w + \omega_y T_x - \omega_x T_y} \end{pmatrix}, \tag{17}$$

where (T_x, T_y, T_z) is the translation vector estimated in previous turn. In addition, we now sample in the warped image $(\tilde{I}_1)$ instead of the original image I_1.

To simplify the notation, we write all the symbols in the warped version of original image back into no double

prime format, i.e. d'' is written back as d, etc. The Jacobian is thus given by

$$\frac{1}{d}\begin{pmatrix} A & B & -dy+fT_y & f & 0 & -x \\ C & D & dx-fT_x & 0 & f & -y \end{pmatrix}. \quad (18)$$

where $A=-x(\frac{dy}{f}-T_y)$, $B=f(d-T_z)-x(-\frac{dx}{f}+T_x)$, $C=f(T_z-d)-y(\frac{dy}{f}-T_y)$, and $D=y(\frac{dx}{f}-T_x)$.

From this Jacobian, we have the following observations. Writing $(\tilde{X},\tilde{Y},\tilde{Z})^{\mathrm{T}}$ as the 3D coordinates of the point with only rotation applied, $\mathbf{T}=(T_x,T_y,T_z)^{\mathrm{T}}$ is the translation in the camera coordinate frame, we have

$$\begin{pmatrix} X' \\ Y' \\ Z' \end{pmatrix} = \begin{pmatrix} \tilde{X}+T_x \\ \tilde{Y}+T_y \\ \tilde{Z}+T_z \end{pmatrix} \quad (19)$$

Noting that d' in Eq. 18 is just Z' i.e. $\tilde{Z}+T_z$, in the Jacobian above, thus we have

$$\frac{1}{d'}\begin{pmatrix} -x(\tilde{Y}) & f\tilde{Z}+x\tilde{X} & f\tilde{Y} & f & 0 & -x \\ -f\tilde{Z}-y\tilde{Y} & y.\tilde{X} & -f\tilde{X} & 0 & f & -y \end{pmatrix} \quad (20)$$

Writing

$$(a,b,c)=(\tilde{X}+T_x,\tilde{Y}+T_y,\frac{1}{\tilde{Z}+T_z}), \quad (21)$$

we have the same format for the Jacobian in the fully projective formulation of Araújo et al. [2],

$$\begin{pmatrix} -fac^2\tilde{Y} & fc(\tilde{Z}+ac\tilde{X}) & -fc\tilde{Y} & fc & 0 & -fac^2 \\ -fc(\tilde{Z}+bc\tilde{Y}) & fbc^2.\tilde{X} & fc.\tilde{X} & 0 & fc & -fbc^2 \end{pmatrix} \quad (22)$$

Finally consider Eq. 20 again, writing each 3D coordinates in its 2D image form i.e. $\tilde{X}=\frac{\tilde{Z}\tilde{x}}{f}$ where $\tilde{x}$ represent the image projection of $\tilde{X}$, etc., we have

$$\frac{1}{Z'}\begin{pmatrix} \frac{-x\tilde{Z}\tilde{y}}{f} & f(\tilde{Z})+\frac{x\tilde{x}\tilde{Z}}{f} & -f\frac{\tilde{Z}\tilde{y}}{f} & f & 0 & -x \\ -f\tilde{Z}-\frac{y\tilde{Z}\tilde{y}}{f} & \frac{y\tilde{Z}\tilde{x}}{f} & \frac{f\tilde{Z}\tilde{x}}{f} & 0 & f & -y \end{pmatrix}. \quad (23)$$

Taking the approximation of $X'\cong\tilde{X}$, $Y'\cong\tilde{Y}$ and $Z'\cong\tilde{Z}$, we have the same format for the rotational and translational components of the classic Longuet-Higgins equation [10]

$$\begin{pmatrix} -\frac{xy}{f} & \frac{(f^2+x^2)}{f} & -y & \frac{f}{Z'} & 0 & -\frac{x}{Z'} \\ -\frac{(f^2+y^2)}{f} & \frac{xy}{f} & x & 0 & \frac{f}{Z'} & -\frac{y}{Z'} \end{pmatrix}. \quad (24)$$

It should be noted that the Jacobian derived Eq.18 can be viewed as another representation than that of the classical Longuet-Higgins equation (24) since it takes less approximation in evaluating the step increment in each iteration. It should therefore result in more accurate estimation result.

4 Performance Analysis of Derived Jacobian

It is in general difficult to assess the performance of image matching algorithm due to the complicated reflectance properties of real world objects and immense possibilities of the movements an object can assume. We have seen that the image motion equation used by Longuet-Higgins (Eq. 24) is a variation of our Jacobian (Eq. 20). In addition, since the derived formula is in the form of a Jacobian, it should thus be possible to apply it in the framework of pose estimation algorithm by Lowe where the change in parameter vector is calculated from the Jacobian also.

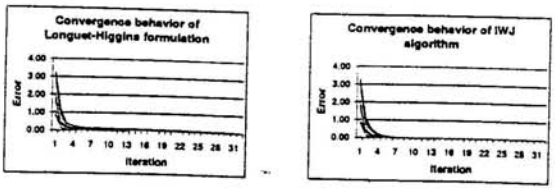
So, we make the following assertion:

The Longuet-Higgins equation for the image motion is a variation of the Jacobian of image displacement with respect to the parameter vector consists of small angle rotation and translation (Eq. 20). In addition, the Jacobian is of the same form as the full projective formulation of Lowe's pose estimation algorithm.

We have already showed that the approximation in obtaining the Longuet–Higgins equation is that we take the assumption that the coordinates of the *rotated* coordinates of a point is the same as the *actual transformed* coordinates, i.e. the translational component is treated as zero in the Jacobian. In general, when both the original and transformed points are just a small distance apart, this should not caused any problem in the estimates obtained. However one would expect wrong estimates would be obtained as the translational components rise to a certain threshold[1].

Effect of Translation on Estimate To confirm the above assertion on the behavior of the algorithm, we perform the following experiment. A number of eight randomly generated points in 3D space is transformed by the following parameters– rotation(RPY)$(0.2-0.10.2)$ and translation of increasing values$(10*i,-10*i,10*i)$ where i increase from 5 to 200 in step 5. This is a challenging data set since it involves both significant changes in rotation and translation. The focal length is set to 1 for convenience whereas the x and y values are chosen to within 100 times of focal length. The object has a z-dimension of 700 units and is placed at a distance of 400 units from the focal center. This particular set up is very challenging in that the object have significant depth variation within the view and thus would pose problem to formulations taking approximation such as weak perspective projection. The projected coordinates are then passed to the estimation algorithm. The squared distance between estimated point set and projected data is plotted against iteration for each experiment. We perform the test on our formulation(our algorithm is abbreviated as *IWJ*, which stands for Image Warp Jacobian) as well as the Longuet–Higgins equation. The

[1] This corresponds to the situation that the object or the camera move at quite a fast speed, or the sampling rate of the camera is not good enough.



(a) (b)

Figure 2: Convergence comparison of the derived Jacobian and the Longuet-Higgins equation. a) Longuet–Higgins equation, b) Derived Jacobian

results of all 40 tests are showed in Fig. 2.

As seen from the figure, the convergence behavior of the Longuet–Higgins equation deteriorates as the translational components become more significant. The wrong convergence behavior can be observed for those tests with large translations(i.e. those curves on the top). On the other hand, the pose estimates of *IWJ* are very accurate even at large translation. Another promising properties of our algorithm is that the convergence is very fast– there are nearly no changes after 7 to 10 iterations for all the tests.

Comparison between *IWJ*, Longuet-Higgins Equation and Lowe's Algorithm

As we have emphasized in previous section, the derived Jacobian as well as the Longuet-Higgins equation can be applied under Lowe's framework to estimate the pose of an image. At the same time, Lowe's algorithm can also be used to perform the task of image matching. For the ease of comparison, the expressions used by these three approaches are listed in the Table 4.

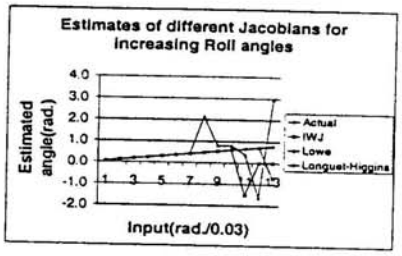
In the above table, (x, y, z) are the 3D coordinates of the estimated point whereas (x', y', z') designate the 3D rotated coordinates of the point, i.e., without translation added. (x_i, y_i) denote the image coordinates of the estimated point.

To determine the accuracy of the derived Jacobian, two methods can be used. The first is applying it to the problem of pose estimation with established correspondences. Another method is use testing images to check its performance under different intensity profiles. We used both methods in our testing.

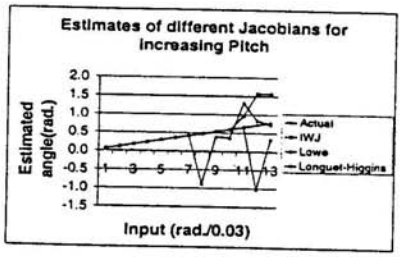
We test the performance of *IWJ* together with the other two algorithms under synthetic data for established correspondences. In fact, this testing procedure should better reflect the performance of the tested algorithms since the actual solutions are available for verification. In image sets testing, owing to the vast number of parameters in image formation, a complete performance characterization is almost impossible. In this experiment, a cube of length 30 (focal length of one unit) placed in the center of view and

Table 1: Comparison of Lowe, Longuet-Higgins and IWJ expression for Jacobian

	Lowe	Longuet-Higgins	IWJ
$\frac{\partial x}{\partial \omega_x}$	$-fc^2x'y'$	$\frac{-x_i y_i}{f}$	$\frac{-x_i y'}{z}$
$\frac{\partial x}{\partial \omega_y}$	$fc(z'+cx'^2)$	$\frac{(f^2+x_i^2)}{f}$	$\frac{fz'+x_i x'}{z}$
$\frac{\partial x}{\partial \omega_z}$	$-fcy'$	$-y_i$	$\frac{fy'}{z}$
$\frac{\partial x}{\partial t_x}$	1	$\frac{f}{z}$	$\frac{f}{z}$
$\frac{\partial x}{\partial t_y}$	0	0	0
$\frac{\partial x}{\partial t_z}$	$-fc^2x$	$-\frac{x_i}{z}$	$-\frac{x_i}{z}$
$\frac{\partial y}{\partial \omega_x}$	$-fc(z'+cy'^2)$	$\frac{-x_i y_i}{f}$	$\frac{-fz'-y_i y'}{z}$
$\frac{\partial y}{\partial \omega_y}$	$fc^2(x'y')$	$\frac{xy}{f}$	$\frac{yx'}{z}$
$\frac{\partial y}{\partial \omega_z}$	$-fcx'$	x_i	$-fx'$
$\frac{\partial y}{\partial t_x}$	0	0	0
$\frac{\partial y}{\partial t_y}$	1	$\frac{f}{z}$	$\frac{f}{z}$
$\frac{\partial y}{\partial t_z}$	$-fc^2y$	$-\frac{y_i}{z}$	$\frac{y_i}{z}$

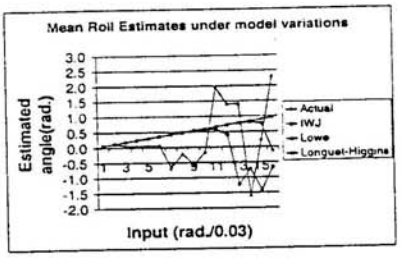


(a)

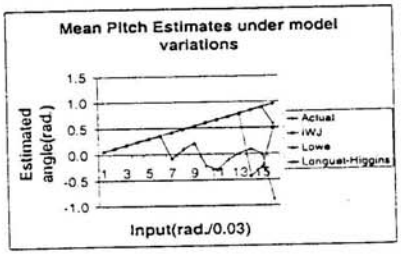


(b)

Figure 3: Convergence comparison of the derived Jacobian and the Longuet-Higgins equation under different motion. a) Roll, b) Pitch.



(a)



(b)

Figure 4: Convergence comparison of the derived Jacobian and Longuet-Higgins equation under model variation. a) Roll, b) Pitch.

at a distance 50 units from camera center. With an initial value of (0.03, 0.03, 0.03) in rotation angle (RPY format), a step increment of 0.03 is added to one of the angles after each iteration whereas other angles keep the same. The estimated results for roll and pitch angles are shown in Fig. 3 [2].

From the figure, it can be seen that the estimates of Lowe algorithm diverge at about 0.6 radian for roll and pitch angle, while *IWJ* and Longuet-Higgins equation still give correct results. This confirms our statement that the derived Jacboian is a full projective formulation in which more stable estimation could be achieved.

To further test the performance of each algorithm, we design the following set of experiment. A randomly generated point set of 8 points inside the volume of $50 \times 50 \times 1000$ times of focal length, at a distance of 50 times of focal length from optical center, are sampled and tested under the same condition as that in Fig. 3. This experiment tests the performance of different Jacobians under the effect of model variation. A number of 100 tests are performed on each rotation value and the mean estimates are plotted as shown in Fig. 4. As can be seen in the figure, both *IWJ* and Longuet equation have a bigger range of convergence for motion than Lowe. In addition, as seen from the figure, it is interesting to find that Longuet–Higgins' formulation still gives correct estimates in most cases, e.g. at around 0.7 rad. for Roll and Pitch angle whereas *IWJ* estimates start to deteriorate. This is a point which deserves future investigation. However, in other experiments in which the model size is of bigger value, say 100 times of focal length, the estimates of *IWJ* outperform the Longuet–Higgins estimates by having a bigger range of convergence.

We conclude that both *IWJ* and the Longuet-Higgins equation have better performance in terms of the range of convergence and model variations.

5 An Algorithm for Image Matching of 3D Objects

Our algorithm for intensity matching works as follows. We assumed that the depth map of the object is available which may come from previous iterations or by other methods such as stereo vision technique [7]. During each iteration, a number of feature points on the object are selected and a window (5×5 in our experiments) is sampled at each feature point both at the current and previous frame. The locations being sampled at both frames could be the same if the image capture rate is fast enough. The image gradients of the two image would then guide the estimation algorithm to find the correspondence. The incremental 3D rotation and translation parameters, which warp the sampled image so that the intensity differences (SSD) between the two windows are minimal, will then be estimated. The warped result will then replace the original image in the next iteration and the process is repeated until convergence established.

A set of examples illustrating the application of the algorithm to a synthetic image are shown in Fig. 5. In this experiment, a plane with a texture of repeating sine wave surface is translated horizontally (Fig. 5b), vertically (Fig. 5c), and in the direction of out of the picture (Fig. 5d) by a steady speed of 2 pixels per frame. We use a Gaussian filter to remove the high frequency components of the input images. A Gaussian kernel of size 1×5 is applied both in the spatial and time domain before the application of our algorithm. We apply the calculation to each pixel in the image and the estimated 3D motion is again projected back on the image plane to check for validation visually on the accuracy of the algorithm. As seen from the figure, the estimated flow is correct for most of the area in the image except near the object boundary. The instability of the result near the boundary is probably due to absence of texture outside the silhouette of the object to guide the image motion. It therefore implies that the tracking of the object by the algorithm should be better inside the textured object interior.

[2] Results for yaw angle are not shown since all algorithms estimated it satsifactorily. This also applies to the next experiment as well.

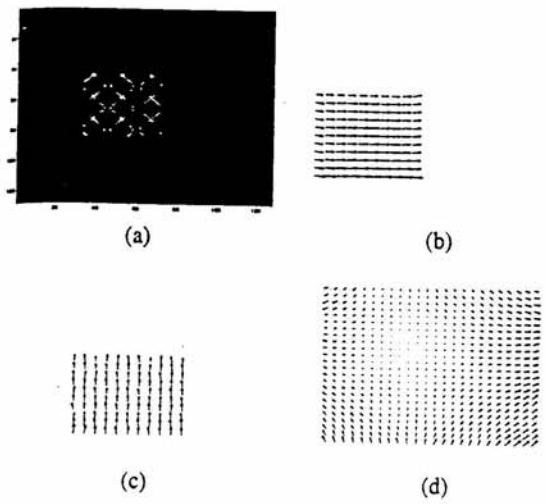


Figure 5: Synthetic image testing of 3D motion estimation algorithm. a) Original image, b) Projected flow of the part with the plane in image frame 8 of horizontal movement, c) Projected flow of vertical movement, d) Projected flow out of paper movement.

Another set of examples showing the result for estimating rotation further confirms the above implication. In Fig. 6, separate rotation in RPY format is applied to the same synthetic image in Fig. 5 and the estimated motion is projected onto the image plane. It can be seen that the estimated motion become much more unreliable near the boundary of the object.

For the real image experiment, we encountered a problem of depth map availability in real images. Since it is difficult to find a benchmark image set which has a depth map available with the data. We hope that with the advent of research on motion analysis as well as structure from motion, reliable performance analysis can be performed on the proposed motion analysis algorithm.

To demonstrate the feasibility of our approach, we used two frames from the rubic cube image sequence as the testing data. In the sequence, the rubic cube is rotating from left to right with an interframe rotation of about 1.44 degree [13]. Since a ground truth data for the depth map is not available, we choose only a face on the rubic cube(near the center of the image, 20 by 20 pixels window in size with image coordinates 120–140 in x-direction, 100–120 in y-direction) and assumed the same depth values (we chose a value of 100) for all pixels in this area. The focal length is assumed to have a value of 0.05. The assumption of the same depth values for all pixels is roughly valid in this case since the face of the rubic cube is nearly parallel with the image plane. For each pixel in the window, we applied the

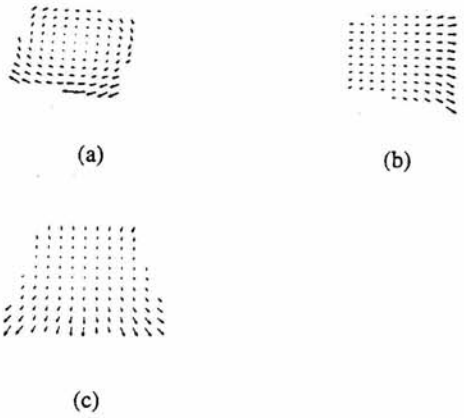


Figure 6: Estimation of rotation a) Roll, b) Pitch, c) yaw

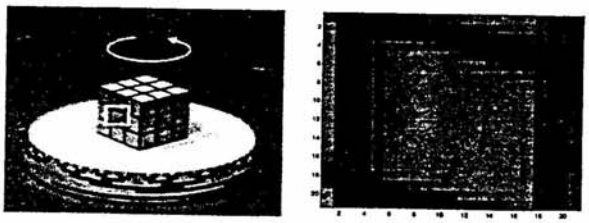


Figure 7: Real image experiment. a) Rubic cube image, b) Reprojected 2D flow overlay on cube image(size:20 by 20)

algorithm to estimate the motion parameters. To check the validity of the results, we used the estimated parameters to reproject the 2D image flow and overlay the results with the original image. The result is shown in Fig. 7b. From the figure, it can be seen that the estimated flow is having a tendency of coming out of the paper, which matched with the motion of the actual cube. We emphasized that a number of factors are affecting the accuracy of the estimated result. Firstly the depth values of each pixel is assumed to be the same and surely will contribute to the error in the estimation. Secondly we only apply the algorithm individually to the pixels in the selected region and a regularizing scheme is lacked to enforce the uniformity of the results. Finally outliers rejection scheme is not implemented so that noise in the image would probably create some wrong estimates. Nevertheless, the experiment clearly indicate the possibility of recovering the 3D motion directly from the image intensity profiles.

6 Summary

A framework for estimating the 3D motion from image intensity differences is formulated. The linkage between the motion parameters and feature tracking is built by formulating the solution to the sum of squared image differences through the Jacobian of motion parameters. Synthetic as well as real image experiments have confirmed the usefulness of our algorithm.

An important contribution here is that we showed that the classical *Longuet-Higgins* equation of image motion can actually be treated as a variation of a full projective formulation of *Lowe*'s Gauss Newton method–another well known pose estimation algorithm. The Longuet–Higgins equation is applied to the pose estimation problem in the framework of Lowe's formulation and its weakness on estimation of large translational displacement is confirmed. The presented results are significant in that the classical Lowe's algorithm can now be applied to the direct estimation problem, which implies more flexibility in tracking complicated object such as articulated ones. On the other hand, an improved formulation of the Longuet–Higgins equation in handling large displacement can also be applied to various applications.

References

[1] T. D. Alter. 3-d pose from 3 points using weak-perspective. *IEEE Trans. Pattern Anal. Machine Intell.*, 16(8):802–808, August 1994.

[2] H. Araujo, R. L. Carceroni, and C.M. Brown. A fully projective formulation to improve the accuracy of lowe's pose-estimation algorithm. *Comput. Vision and Image Understanding*, 70(2):227–238, May 1998.

[3] A. Azarbayejani and A. Pentland. Recursive estimation of motion, structure, and focal length. *IEEE Trans. Pattern Anal. Machine Intell.*, 17(6):562–575, June 1995.

[4] M. J. Black and A. Rangarajan. On the unification of line processes, outlier rejection, and robust statistics with applications in early vision. *Intl. Journal of Comput. Vision*, 19:57–91, 1996.

[5] T. J. Broida. Recursive 3-D motion estimation from a monocular image sequence. *IEEE Trans. Aerospace Electronic Systems*, 26(4):639–655, July 1990.

[6] D. F. Dementhon and L. S. Davis. Model-based object pose in 25 lines of code. *Intl. Journal of Comput. Vision*, 15:123–141, 1995.

[7] O. D. Faugeras. *Three-Dimensional Computer Vision: a geometric viewpoint*. MIT Press, 1993.

[8] B. K. P. Horn. *Robot Vision*. MIT/McGraw–Hill, New York, 1986.

[9] M. Isard and A. Blake. Icondensation: Unifying low-level and high-level tracking in a stochastic framework. *Proc. 5th European Conf. on Computer Vision, 1998*, 1998.

[10] H.C. Longuet-Higgins and K. Prazdny. The interpretation of a moving retinal image. *Proc. Roy. Soc. London B*, pages 385–397, 1980.

[11] D. G. Lowe. Robust model-based motion tracking through the integration of search and estimation. *Intl. Journal of Comput. Vision*, 8:113–122, 1992.

[12] J. Shi and C. Tomasi. Good features to track. *Proc. CVPR94*, pages 593–600, 1994.

[13] R. Szeliski. Shape from rotation. *Cambridge Research Laboratory Technical Report Series*, (9013), December 1990.

[14] R. Szeliski and H.Y.Shum. Creating full view panoramic image mosaics and texture-mapped models. *Proc. SIGGRAPH'97(Los Angeles)*, pages 251–258, August 1997.

[15] J. Weng, N. Ahuja, and T. S. Huang. Optimal motion and structure estimation. *IEEE Trans. Pattern Anal. Machine Intell.*, 15(9):864–884, September 1993.

[16] Z. Zhang and O. Faugeras. *3D Dynamic Scene Analysis*. Springer-Verlag, 1992.

PROCEEDINGS

1999 International Symposium on Signal Processing and Intelligent System (ISSPIS'99)

November 26-28, 1999
Guangzhou, China

Sponsor:
Circuit and System Society of CIE
National Natural Science Foundation of China
South China University of Technology
Co-Operation with
Circuit and System Society of IEEE

SOUTH CHINA UNIVERSITY OF TECHNOLOGY PRESS