EN.520.665

HW5 Solution

## Question 1

V. Cappellini and A. G. Constantinides, Eds. Amsterdam, The Netherlands: Elsevier, pp. 770–775.

[3] J. O'Rourke, "Motion detection using Hough techniques," in *Proc. Conf. Pattern Recognition and Image Processing*, Dallas, TX, 1981, p. 737.

[4] T. M. Silberberg, L. Davis, and D. Harwood, "An iterative Hough procedure for three-dimensional object recognition," *Pattern Recognition*, vol. 17, no. 6, pp. 621–629, 1984.

[5] H. Li, M. A. Lavin, and R. J. LeMaster, "Fast Hough Transform," in *Proc. 3rd Workshop Computer Vision: Representation and Control*, Bellair, MI, 1985, pp. 75–83.

[6] H. Li and M. A. Lavin, "Fast Hough Transform based on Bintree data structure," in *Proc. Conf. Computer Vision and Pattern Recognition*, Miami Beach, FL, 1986, pp. 640–642.

[7] R. Lumia, L. Shapiro, and O. Zuniga, "A new connected components algorithm for virtual memory computers," *Comput. Graphics Image Processing*, vol. 22, pp. 287–300, 1983.

[8] A. Bowyer and J. Woodwark, *A programmers geometry*. London: Butterworth, 1983.

[9] M. Cohen and G. T. Toussaint, "On the detection of structures in noisy pictures," *Pattern Recognition*, vol. 9, pp. 95–98, 1977.

[10] T. M. Van Veen and F. C. A. Groen, "Discretization errors in the Hough Transform," *Pattern Recognition*, vol. 14, pp. 137–145, 1981.

[11] R. Lumia, "A new three-dimensional connected components algorithm," *Comput. Vision, Graphics, Image Processing*, vol. 23, pp. 207–217, 1983.

# Least-Squares Fitting of Two 3-D Point Sets

K. S. ARUN, T. S. HUANG, AND S. D. BLOSTEIN

***Abstract*—Two point sets $\{p_i\}$ and $\{p_i'\}$; $i = 1, 2, \cdots, N$ are related by $p_i' = Rp_i + T + N_i$, where $R$ is a rotation matrix, $T$ a translation vector, and $N_i$ a noise vector. Given $\{p_i\}$ and $\{p_i'\}$, we present an algorithm for finding the least-squares solution of $R$ and $T$, which is based on the singular value decomposition (SVD) of a $3 \times 3$ matrix. This new algorithm is compared to two earlier algorithms with respect to computer time requirements.**

***Index Terms*—Computer vision, least-squares, motion estimation, quaternion, singular value decomposition.**

## I. Introduction

In many computer vision applications, notably the estimation of motion parameters of a rigid object using 3-D point correspondences [1] and the determination of the relative attitude of a rigid object with respect to a reference [2], we encounter the following mathematical problem. We are given two 3-D point sets $\{p_i\}$; $i = 1, 2, \cdots, N$ (here, $p_i$ and $p_i'$ are considered as $3 \times 1$ column matrices)

$$p_i' = Rp_i + T + N_i \tag{1}$$

where $R$ is a $3 \times 3$ rotation matrix, $T$ is a translation vector ($3 \times 1$ column matrix), and $N_i$ a noise vector. (We assume that the rotation is around an axis passing through the origin). We want to find $R$ and $T$ to minimize

$$\Sigma^2 = \sum_{i=1}^{N} \left\| p_i' - (Rp_i + T) \right\|^2. \tag{2}$$

Manuscript received July 2, 1986; revised April 9, 1987. Recommended for acceptance by S. W. Zucker. This work was supported by the National Science Foundation under Grant IRI-8605400.

The authors are with the Coordinated Science Laboratory, University of Illinois, Urbana, IL 61801.

IEEE Log Number 8715809.

An iterative algorithm for finding the solution was described in Huang, Blostein, and Margerum [3]; a noniterative algorithm based on quaternions in Faugeras and Hebert [4]. In this correspondence, we describe a new noniterative algorithm which involves the singular value decomposition (SVD) of a $3 \times 3$ matrix. The computer time requirements of the three algorithms are compared.

After the submission of our correspondence, it was brought to our attention that an algorithm similar to ours had been developed independently by Professor B. K. P. Horn, M.I.T., but not published.

## II. Decoupling Translation and Rotation

It was shown in [3] that: If the least-squares solution to (1) is $\hat{R}$ and $\hat{T}$, then $\{p_i')$ and $\{p_i'' \triangleq \hat{R}p_i = \hat{T}\}$ have the same centroid, i.e.,

$$p' = p'' \tag{3}$$

where

$$p' \triangleq \frac{1}{N} \sum_{i=1}^{N} p_i' \tag{4}$$

$$p'' \triangleq \frac{1}{N} \sum_{i=1}^{N} p_i'' = \hat{R}p + \hat{T} \tag{5}$$

$$p \triangleq \frac{1}{N} \sum_{i=1}^{N} p_i. \tag{6}$$

Let

$$q_i \triangleq p_i - p \tag{7}$$

$$q_i' \triangleq p_i' - p'. \tag{8}$$

We have

$$\Sigma^2 = \sum_{i=1}^{N} \left\| q_i' - Rq_i \right\|^2. \tag{9}$$

Therefore, the original least-squares problems is reduced to two parts:

(i) Find $\hat{R}$ to minimize $\Sigma^2$ in (9).
(ii) Then, the translation is found by

$$\hat{T} = p' - \hat{R}p. \tag{10}$$

In the next section, we describe an algorithm for (i) which involves the SVD of a $3 \times 3$ matrix.

## III. An SVD Algorithm for Finding $\hat{R}$

### A. Algorithm

*Step 1:* From $\{p_i\}$, $\{p_i'\}$ calculate $p$, $p'$; and then $\{q_i\}$, $\{q_i'\}$.

*Step 2:* Calculate the $3 \times 3$ matrix

$$H \triangleq \sum_{i=1}^{N} q_i q_i'^t \tag{11}$$

where the superscript $t$ denotes matrix transposition.

*Step 3:* Find the SVD of $H$,

$$H = U \Lambda V^t. \tag{12}$$

*Step 4:* Calculate

$$X = VU^t. \tag{13}$$

*Step 5:* Calculate, det $(x)$, the determinant of $X$.

If det $(x) = +1$, then $\hat{R} = X$.

If det $(x) = -1$, the algorithm fails. (This case usually does not occur. See Sections IV and V.)

### B. Derivation

Expanding the right-hand side of (9),

$$\Sigma^2 = \sum_{i=1}^{N} (q_i' - Rq_i)^t (q_i' - Rq_i)$$
$$= \sum_{i=1}^{N} (q_i'^t q_i' + q_i^t R^t R q_i - q_i'^t R q_i - q_i^t R^t q_i')$$
$$= \sum_{i=1}^{N} (q_i'^t q_i' + q_i^t q_i - 2 q_i'^t R q_i).$$

Therefore, minimizing $\Sigma^2$ is equivalent to maximizing

$$F = \sum_{i=1}^{N} q_i'^t R q_i$$
$$= \text{Trace}\left(\sum_{i=1}^{N} R q_i q_i'^t\right) = \text{Trace}\,(RH) \tag{14}$$

where

$$H \triangleq \sum_{i=1}^{N} q_i q_i'^t. \tag{11}$$

*Lemma:* For any positive definite matrix $AA^t$, and any orthonormal matrix $B$,

$$\text{Trace}\,(AA^t) \geq \text{Trace}\,(BAA^t).$$

*Proof of Lemma:* Let $a_i$ be the $i$th column of $A$. Then

$$\text{Trace}\,(BAA^t) = \text{Trace}\,(A^t BA)$$
$$= \sum_i a_i^t (Ba_i).$$

But, by the Schwarz inequality,

$$a_i^t (Ba_i) \leq \sqrt{(a_i^t a_i)(a_i^t B^t B a_i)} = a_i^t a_i.$$

Hence, Trace $(BAA^t) \leq \Sigma_i\, a_i^t a_i =$ Trace $(AA^t)$. Q.E.D.

Let the SVD of $H$ be:

$$H = U \Lambda V^t \tag{12}$$

where $U$ and $V$ are $3 \times 3$ orthonormal matrices, and $\Lambda$ is a $3 \times 3$ diagonal matrix with nonnegative elements. Now let

$$X = VU^t \quad \text{(which is orthonormal)}. \tag{13}$$

We have

$$XH = VU^t U \Lambda V^t$$
$$= V \Lambda V^t \tag{15}$$

which is symmetrical and positive definite. Therefore, from Lemma, for any $3 \times 3$ orthonormal matrix $B$,

$$\text{Trace}\,(XH) \geq \text{Trace}\,(BXH) \tag{16}$$

Thus, among all $3 \times 3$ orthonormal matrices, $X$ maximizes $F$ of (14). And if det $(X) = +1$, $X$ is a rotation, which is what we want.

However, if det $(X) = -1$, $X$ is a reflection, which is not what we want. Fortunately, this degenerate case usually does not occur. We shall discuss the situation in some detail in the next two sections.

## IV. Degeneracy: Noiseless Case

Assume $N_i = 0$ in (1) for all $i$. Then, obviously there is a solution $\hat{R}$ (which is a rotation, i.e., det $(\hat{R}) = +1$) for which $\{q_i'\}$ and $\{\hat{R} q_i\}$ are congruent and hence $\Sigma^2 = 0$. From geometrical considerations, it is easy to see that there are three possibilities.

*1) $\{q_i\}$ are not coplanar*—Then, the rotation solution is unique. Furthermore, there is no reflection $X$ which can make $\Sigma^2 = 0$. Therefore, the SVD algorithm will give the desired solution.

*2) $\{q_i\}$ are coplanar but not colinear*—There is a unique rotation as well as a unique reflection which will make $\Sigma^2 = 0$. Therefore, the SVD algorithm may give either. We shall see presently that this situation can be easily resolved.

*3) $\{q_i\}$ are colinear*—There are infinitely many rotations and reflections which will make $\Sigma^2 = 0$.

Now we come back to the coplanar case. From examining the elements of the $3 \times 3$ matrix $H$, it can readily be shown that the points $\{q_i\}$ are coplanar, if and only if one of the three singular values of $H$ is zero. Let the three singular values be $\lambda_1 > \lambda_2 > \lambda_3 = 0$. Then

$$H = \lambda_1 u_1 v_1^t + \lambda_2 u_2 v_2^t + 0 \cdot u_3 v_3^t \tag{17}$$

where $u_i$ and $v_i$ are columns of $U$ and $V$, respectively. Note that changing the sign of $u_3$ or $v_3$ will not change $H$. Therefore, if $X = VU^t$ minimizes $\Sigma^2$, so does $X' = V'U^t$ where

$$V' = [v_1, v_2, -v_3]. \tag{18}$$

If $X$ is a reflection, then $X'$ is a rotation, and vice versa. Thus, if the SVD algorithm gives a solution $X$ with det $(X) = -1$, we form $X' = V'U^t$ which is the desired rotation.

We mention, in passing, that the points $\{q_i\}$ are colinear, if and only if, two of the three singular values of $H$ are equal.

## V. Degeneracy: Noisy Case

If either $\{q_i\}$ or $\{q_i'\}$ are coplanar, then it can readily be shown that the discussion on the coplanar case in Section IV is still valid, except of course now the minimum of $\Sigma^2$ is no longer zero. Hence, if the SVD algorithm gives a reflection $X = VU^t$, we can form the desired rotation $X' = V'U^t$. A special case of interest is when $N = 3$. Then both $\{q_i\}$ and $\{q_i'\}$ are coplanar point sets.

The situation we cannot handle is when the SVD algorithm gives a solution $X$ with det $(X) = -1$, and none of the singular values of $H$ is zero. This means that neither $\{q_i\}$ nor $\{q_i'\}$ are coplanar; yet there is no rotation which yields a smaller $\Sigma^2$ then the reflection $x$. This can happen only when the noise $N_i$ are very large. In that case, the least-squares solution is probably useless anyway. A better approach would be to use a RANSAC-like technique (using 3 points at a time) to combat against outliers [5].

## VI. Summary of Algorithm

Using the procedure of Section III-A, we obtain

$$X = VU^t.$$

1) If det $(X) = +1$, then $X$ is a rotation which is the desired solution.

2) If det $(X) = -1$, then $X$ is a reflection.

a) one of the singular values ($\lambda_3$, say) of $H$ is zero. Then, the desired rotation is found by forming

$$X' = V'U^t$$

where $V'$ is obtained from $V$ by changing the sign of the 3rd column.

b) None of the singular values of $H$ is zero. Then, conventional least-squares solution is probably not appropriate. We go to a RANSAC-like technique.

## VII. Computer Time Requirements

Computer simulations have been carried out on a VAX 11/780 to compare the three algorithms (SVD, quaternion, iterative) with respect to time requirements. In each simulation, a set of 3-D points $\{p_i\}$ were generated. They are randomly distributed in a cube of size $6 \times 6 \times 6$ with center at $(0, 0, 0)$. Then $\{p_i'\}$ were calculated by rotating $\{p_i\}$ by an angle of 75° around an axis through the origin with direction cosines (0.6, 0.7, 0.39) followed by a translation of (80, 60, 70), and finally by adding to each coordinate of the resulting points Gaussian random noise with mean zero and standard deviation 0.5. Then the algorithms were used to estimate $\hat{R}$ and $\hat{T}$. The CPU times used are listed in Table I. For the iterative algorithm, the numbers of iterations are given in parentheses. The programs were written in C. The IMSL subroutine package was

# Question 2)

## a)

The first edge in the left image can be matched to n edges in the right image. The second edge in the left image to (n-1) edges in the right image and so on. So overall we get n!

## b)

If the edges are ordered the same way from left to right, there is only one way to match them.

## c)

There are n! ways of matching up left images with right images and the (n-m) null edges. The null edges are all equivalent, so it does not matter which way we match up to them. So, we must divide by (n-m)! to get n!/(n-m)!. (Check that if n=m, we get the result in (a) above)

## d)

There are n!/(n-m)! ways of matching the left edges with right edges and null edges, except that out of m! ways of matching against right edges, only one preserves the ordering. So we get n!/(n-m)!m!

# Question 3)

See Section 2.

# New Algorithms For Reconstruction Of A 3-D Depth Map From One Or More Images*

M. Shao, T. Simchony and R. Chellappa

Signal and Image Processing Institute
Phe 324 , Dept of EE-Systems
University of Southern California
Los Angeles , CA90089-0272 .

**Abstract**

New algorithms are developed to recover the depth and orientation maps of a surface from its image intensities. They combine the advantages of stereo vision and shape-from-shading (SFS) methods. These algorithms generate dense surface depth and orientation maps accurately and unambiguously. Previous SFS algorithms can not be directly extended to combine stereo images because the recovery of surface depth and that of orientation are separated in these formulations. A new SFS algorithm is proposed to couple the generation of the depth and orientation maps. The new formulation also ensures that the reconstructed surface depth and its orientation are consistent. The SFS algorithm for a single image is next extended to utilize stereo images. The correspondence over stereo images is established simultaneously with the generation of surface depth and orientation. An alternative approach is also suggested for combining stereo and SFS techniques. This approach can be used to combine needle maps which are directly available from other sources such as photometric stereo. Finally, we discuss the use of embedding techniques to combine sparse depth measurements.

## 1 Introduction

One of the goals of a computer-vision system is to recover the three dimensional shape of a surface from its image intensities. There exist several approaches to this problem. Conventional stereo methods [1] are characterized by matching certain feature points in stereo images. As stereo vision can determine only a sparse set of surface depths, it is often followed by a surface interpolation process [2]. The surface orientation can not be recovered directly from matching. The most difficult problem with this method is that of identifying corresponding feature points. Algorithms for recovery of shape from shading have been investigated extensively [3, 4, 5, 6]. Shape-from-shading (SFS) techniques explore the information contained in image intensities by reconstructing a surface that is consistent with observed image intensities. The SFS techniques are formulated for a single image so correspondence is not necessary. The results of SFS algorithms for a single image may not be accurate or robust. Sometimes it is ambiguous to recover a surface from a single image [7]. Thus stereo images are often needed to recover a surface accurately and unambiguously.

*Partially supported by the NSF Grant MIP-84-51010 and matching funds from IBM and AT&T Information Systems.

In this paper new algorithms are proposed to fill the gap between conventional stereo and SFS techniques. They combine the advantages of stereo vision and SFS methods. SFS techniques are used to recover the depth and orientation maps of a surface from its stereo images which are taken from different viewing directions with fixed light source. Uniform matching is performed over these stereo images in order to obtain dense depth and orientation maps. Both the stereo correspondence and surface depth are established simultaneously under two constraints. The first is the geometrical constraint of stereo vision. The second constraint is provided by the irradiance equations so that the reconstructed surface is everywhere consistent with observed image intensities. These algorithms ensure full use of shading information and recover both surface depth and orientation. Because stereo images are used in recovering shape from image intensities the solution is accurate, and can not be ambiguous as it can be when a single image is used.

Stereo vision and irradiance equations in SFS problem provide constraints on surface depth and orientation respectively. In order to combine these two methods we need a natural way of incorporating surface depth into orientation constraints. Existing SFS algorithms can not be used for this purpose because all these algorithms recover the orientation map in a separate step, prior to recovering the surface depth. Also some of them do not even enforce the integrability constraint so that the reconstructed surface orientation and depth may not be consistent [6]. Consistent surface reconstruction requires that the reconstructed needle map always correspond to the orientation map of the reconstructed surface depth. In this work we first suggest a new SFS algorithm which provides a novel solution to these two difficulties. It allows a natural incorporation of geometric stereo into SFS methods. The SFS problem is formulated here as that of solving a coupled set of first-order partial differential equations. The first one of these equations is the irradiance equation. The other two enforce the consistency between the reconstructed surface depth and orientation. Because surface depth and orientation are reconstructed simultaneously, stereo images can be easily incorporated.

An alternative algorithm is also suggested to generate surface depth and orientation maps from stereo images by combining stereo method and SFS techniques. In this approach, different needle maps are generated for each image, using the SFS technique proposed in [7]. The depth map can be generated from these needle maps by establishing the correspondence so that the disparity over these needle maps is minimized. The



CH2605-4/88/0000/0530$01.00 © 1988 IEEE

integrability constraint is also enforced. This approach is related to Ikeuchi's work of combining needle maps [8]. However Ikeuchi's formulation of combining needle maps has some errors. First of all, the orientation of a surface is different with respect to different coordinate systems. The needle maps reconstructed from different images can not be compared directly. Instead they should be mapped into the same coordinate system. Secondly the derivatives of the needle maps used in the formulation should be obtained before the transformation of the needle maps. Although the transformation of needle maps between different coordinates is independent of the position of surface points, the transformation of the derivatives of needle maps is a non-linear function of the surface heights. We will show how to combine needle maps correctly.

Photometric stereo seems to be the only practical solution for obtaining an orientation map for surface with varying albedo. The result may not be accurate because of modeling problems associated with the imaging process. One can improve in accuracy by using sparse depth measurements from geometric stereo . In this paper we show how to combine photometric stereo and geometric stereo information to improve the reconstruction of the surface height.

The organization of the paper is as follows:

Section 2 introduces a new method to recover the depth and orientation maps of a surface simultaneously and consistently. In Section 3 we show how the needle maps of a surface viewed from different directions are related, and extend the results in Section 2 and the algorithm presented in [7] to combine stereo images. The use of sparse information to recover surface is discussed in Section 4, followed by a summary in Section 5.

# 2 Recovery of Depth and Needle Maps From A Single Image

The SFS problem is to extract the shape information from image intensities. Formally, given a 2-D intensity distribution $E(x,y)$, and a reflectance map $R(p,q)$ with constant albedo, it may be regarded as a problem of recovering a surface, $Z(x,y)$ , satisfying the *image irradiance equation* :

$$E(x,y) = R(p,q) \tag{1}$$

where

$$p = Z_x \tag{2}$$

$$q = Z_y \tag{3}$$

and $(-p(x,y), -q(x,y), 1)$ is the orientation of the surface at $(x, y, Z(x,y))$.

Almost all SFS algorithms recover the needle map $(p,q)$ in a separate step, prior to recovering the depth map. The needle map $(p,q)$ is obtained by minimizing the brightness error under the constraint that the surface is smooth. Then the depth $Z$ is recovered from $p$ and $q$ [5]. As $p$ and $q$ are treated as independent variables, the recovered surface needle map $(p,q)$ may not correspond to the orientation of the underlying surface. The integrability constraint [6, 9, 7] is needed to ensure the solution to the SFS problem is the correct one. This constraint is often expressed in terms of the gradient space as

$$p_y = q_x \tag{4}$$

Horn and Brooks [6] used the constraint as a penalty term in their formulation to enforce the integrability constraint. This formulation is not satisfying for our problem. Mathematically (2) and (3) do not always imply (4). Most of all, the depth information contained in stereo images is not coupled into the recovery of a needle map. This makes it difficult to generalize the algorithm to combine stereo images.

Instead of using the integrability constraint $p_y = q_x$ as a penalty term, we formulate the SFS problem in a different way so that surface depth and needle maps are coupled and the recovered needle map is always consistent with the reconstructed surface depth.

The above equations (1), (2) and(3) can be considered as a coupled set of first-order differential equations of independent unknown functions $p(x,y)$, $q(x,y)$ and $Z(x,y)$. Equation (1) is the irradiance equation which enforces the recovered surface to correspond to the given image intensities. Equations (2) and (3) are called *consistency constraints*. Consistency constraints ensure that the reconstructed needle map $(p,q)$ always corresponds to the orientation map of the reconstructed surface depth. Thus the SFS problem is now reduced to solving equations (1), (2) and (3) for orientation $p$ and $q$, and depth $Z$.

Instead of directly solving these nonlinear equations, we reformulate the problem as one of finding $p$, $q$ and $Z$ such that equations (1), (2) and (3) are satisfied under some criterion. Specifically $p$, $q$ and $Z$ should be chosen to minimize the error functional:

$$\int\int_\Omega [(E(x,y) - R(p,q))^2 + (Z_x - p)^2 + (Z_y{'} - q)^2] dxdy \tag{5}$$

Solving for $Z$, $p$ and $q$ is still an ill-posed problem in the sense of Hadamard [10] as there is no unique solution. To overcome this difficulty we regularize it by assuming that the surface is smooth. According to Ikeuchi and Horn [5], the measure of "lack of smoothness" is given by

$$\int\int (p_x^2 + p_y^2 + q_x^2 + q_y^2) dxdy \tag{6}$$

Adding this term to the error functional term, one has the following functional to be minimized with respect to $p$, $q$ and $Z$:

$$\begin{aligned}\int\int_\Omega [(E(x,y) - R(p,q))^2 \; &+ \; (Z_x - p)^2 + (Z_y - q)^2 \\ &+ \; \lambda(p_x^2 + p_y^2 + q_x^2 + q_y^2)] dxdy\end{aligned} \tag{7}$$

Here $\lambda$ is a weighting factor for the smoothness term.

Using Euler formula [11] one obtains the following equations:

$$\begin{cases} \nabla^2 p = -\frac{1}{\lambda}[(Z_x - p) + (E - R(p,q))R_p] \\ \nabla^2 q = -\frac{1}{\lambda}[(Z_y - q) + (E - R(p,q))R_q] \\ \nabla^2 Z = p_x + q_y \end{cases} \tag{8}$$

where $R_p$ and $R_q$ are the partial derivatives of $R(p,q)$ with respect to $p$ and $q$. And

$$\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$$

is the Laplacian operator.

Thus we get a coupled set of non-linear Poisson equations. In order to solve these equations, boundary conditions for $p$, $q$ and $Z$ are needed. The boundary conditions can be obtained in two steps. First the surface depth along the zero crossing boundaries can be found by Marr-Poggio-Grimson [2] stereo algorithm. Then derivatives of the depth and image intensities can be used to find the surface orientation along the boundaries [8, 12].

Because of the non-linear nature of the equations one can not get a closed-form solution. They are solved by using Jacobi Picard iterations :

$$\begin{cases} \nabla^2 p^{n+1} = -\frac{1}{\lambda}[(Z_x^n - p^n) + (E - R(p^n, q^n))R_p(p^n, q^n)] \\ \nabla^2 q^{n+1} = -\frac{1}{\lambda}[(Z_y^n - q^n) + (E - R(p^n, q^n))R_q(p^n, q^n)] \\ \nabla^2 Z^{n+1} = p_x^n + q_y^n \end{cases} \tag{9}$$

The iterations can be continued until there is little change in $p$, $q$ and $Z$ between two consecutive iterations.

At each step of the iterations three Poisson equations have to be solved. We used direct methods [7] to solve these Poisson equations. Experiments with synthetic images show that the direct methods are fast and accurate and that they work on both rectangular and irregular regions. Due to space limitations, we have not included these results.

In this formulation of the SFS problem the integrability constraint is enforced implicitly. The reconstructed surface depth and orientation are always consistent. As the depth $Z$ and the orientation $p$ and $q$ are coupled at every step, it is easy and natural to generalize this algorithm to combine stereo images as described in the following section.

# 3 Combining Stereo Images

Recovering a surface from a single image is sometimes ambiguous [7]. And the solution of SFS problem is not accurate and robust. Stereo methods use multiple images to overcome these difficulties. They do not make full use of the shading information. Instead they establish correspondence at certain feature points and use geometric relation over stereo images to recover the surface depth. Thus only a sparse set of surface depths can be recovered. In the following we show how SFS techniques can be combined with stereo methods to overcome the above difficulties. The coupling between surface depth and orientation in the SFS formulation given in Section 2 enables us to combine the geometric constraint on surface depth and the irradiance constraint on orientation. Thus global correspondence can be established over stereo images. Furthermore the surface orientation and depth are recovered simultaneously with the global correspondence.

In the following we first show how the needle maps from stereo images are related. Then a method is presented to combine multiple needle maps to obtain an accurate surface depth. It is found that this technique can be applied to recover surface orientation and depth directly from stereo images by combining the SFS algorithm presented in [7]. As this formulation is highly non-linear we present a simple and elegant algorithm for combining stereo images by extending the results in Section 2.

For simplicity all the formulas are derived for the case of two images. They can be easily extended to the case when more than two images are available.

## 3.1 Camera Set-up and Needle Map Transformation

We use the same camera set-up as in Ikeuchi [13], see Figure 1. The left image plane is perpendicular to the spatial $z$ axis, while the right image plane is inclined with respect to $z$-axis so that the two optical axes intersect with each other at the origin of the global coordinate system, which is fixed on the surface. Let $(u^l, v^l, w^l)$ and $(u^r, v^r, w^r)$ be the left and right camera coordinate systems and $(x, y, z)$ be the global coordinate system. Assume the object is far away from the cameras so that orthographic projection can be used. Using the parameters in Figure 1, one has the following coordinate transformations:

$$\begin{bmatrix} u^l \\ v^l \\ w^l \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ -d^l \end{bmatrix} \tag{10}$$

$$\begin{bmatrix} u^r \\ v^r \\ w^r \end{bmatrix} = \begin{bmatrix} cos\theta & 0 & -sin\theta \\ 0 & 1 & 0 \\ sin\theta & 0 & cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ -d^r \end{bmatrix} \tag{11}$$

$$\begin{bmatrix} u^r \\ v^r \\ w^r \end{bmatrix} = \begin{bmatrix} cos\theta & 0 & -sin\theta \\ 0 & 1 & 0 \\ sin\theta & 0 & cos\theta \end{bmatrix} \begin{bmatrix} u^l \\ v^l \\ w^l \end{bmatrix} + \begin{bmatrix} -d^l sin\theta \\ 0 \\ -d^r + d^l cos\theta \end{bmatrix} \tag{12}$$

By defining a$=cos\theta$, b$=sin\theta$, c$=-d^l\, sin\theta$, d$=-d^r + d^l\, cos\theta$, one obtains

$$\begin{bmatrix} u^r \\ v^r \\ w^r \end{bmatrix} = \begin{bmatrix} a & 0 & -b \\ 0 & 1 & 0 \\ b & 0 & a \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ -d^r \end{bmatrix} \tag{13}$$

and

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \begin{bmatrix} a & 0 & b \\ 0 & 1 & 0 \\ -b & 0 & a \end{bmatrix} \begin{bmatrix} u^r \\ v^r \\ w^r \end{bmatrix} + \begin{bmatrix} bd^r \\ 0 \\ ad^r \end{bmatrix} \tag{14}$$

Suppose the underlying surface can be expressed in the global coordinate system as:

$$Z = Z(x, y)$$

and the gradient map is

$$p = Z_x(x, y)$$
$$q = Z_y(x, y)$$

Then the orientation of the surface at $(x, y, Z)$ is $(-p, -q, 1)$.

As the left camera coordinate system is identical to the global coordinate system except the translation along the $z$-axis, it is easy to see that the gradient in the left camera coordinate system is

$$p^l = p \tag{15}$$

$$q^l = q \tag{16}$$

In the right coordinate system, the orientation at the same point corresponds to different $p$ and $q$ because of the relative rotation of the coordinate systems. The relation between $(p^r, q^r)$

# Question 4)

Our points are $[A1, A2, B1, B2]$ and it corresponds to $x = [1, 1, -1, -1]^T$
Therefore we have:

$$W = \begin{bmatrix} 0 & 3 & 2 & 2 \\ 3 & 0 & 0 & 2 \\ 2 & 0 & 0 & 3 \\ 2 & 2 & 3 & 0 \end{bmatrix}$$

$$D = \begin{bmatrix} 7 & 0 & 0 & 0 \\ 0 & 5 & 0 & 0 \\ 0 & 0 & 5 & 7 \\ 0 & 0 & 0 & 5 \end{bmatrix}$$

Our eigenvalue problem is:

$$D^{-\frac{1}{2}}(D - w)D^{-\frac{1}{2}}z = \lambda z$$

Doing the calculation we get:

$$D^{-\frac{1}{2}}(D - w)D^{-\frac{1}{2}} = \begin{bmatrix} 1 & -0.5071 & -0.3381 & -0.2857 \\ -0.5071 & 1 & 0 & -0.3381 \\ -0.3381 & 0 & 1 & -0.5071 \\ -0.2857 & -0.3381 & -0.5071 & 1 \end{bmatrix}$$

After calculating the eigenvectors and eigenvalues we have:

$$\lambda = [0, 0.9215, 0.364, 1.7143]$$

$$z1 = [0.5401, 0.4564, 0.4564, 0.5401]^T$$
$$z2 = [-0.2977, -0.6414, 0.6414, 0.2977]^T$$
$$z3 = [-0.6414, 0.2977, -0.2977, 0.6414]^T$$
$$z4 = [-0.4564, 0.5401, 0.5401, -0.4564]^T$$

The second smallest corresponding eigenvector is $z2 = [-0.2977, -0.6414, 0.6414, 0.2977]^T$. And therefore the classes are $[-1, -1, 1, 1]$ and hence the result groups are $[A1, A2]$ and $[B1, B2]$ as we expected from the ground truth labels.