An integrated approach to stereo matching, surface reconstruction and depth segmentation using consistent smoothness assumptions

Liang-Hua Chen and Terrence E. Boult
Columbia University Department of Computer Science
New York City, NY 10027 tboult@cs.columbia.edu

Abstract

This paper presents a new algorithm for stereo matching which makes use of simultaneous matching, surface reconstruction, and segmentation of world surfaces. By integrating these three phases, which are traditionally temporally separated, the algorithm can make use of the current surface information to help disambiguate the potential matches.

After discussing the required mathematical background, the paper describes the integrated process of matching, reconstruction and segmentation. Unlike past attempts at integrating these processes, the presented algorithm uses a single smoothness criterion for both matching, reconstruction and segmentation. The segmentation part of the process is based on estimates of surface bending energy, and is significantly different from previous segmentation algorithms. Examples are presented showing results on both synthetic images and camera acquired images. The camera-based examples include both a traditional type scene with two objects, and a scene with transparent objects.

1 Introduction

Stereopsis is a technique for computing depth from two disparate images of a scene. This section discusses the background the problem which caused us to adopt our integrated approach. The processes of feature detection, matching and surface reconstruction and their inter-relationship are also discussed.

1.1 Background

As stereo vision is very important in many areas, the central task in stereo is to solve the correspondence problem, i.e., identify features in two images that are projections of the same entity in the three-dimension world. Once this is done, one can compute the distance to this entity. Ideally, one would like to find the correspondences of every individual pixel in both images of a stereo pair. However, it is obvious that the information content in the intensity value of a single pixel is too low for unambiguous matching. In practice, therefore, coherent collection of pixels are matched. These collections are determined and matched in two distinct ways ([Barnard and Fischler 82]):

(1) Area-based matching tries to match an area of pixels in one image to another image. A small window is chosen as the matching unit. A window in one image is matched with a range of windows in the other image using cross-correlation or similar measure of the similarity between two windows.

(2) Feature-based matching attempts to match some specific points, individual edge points, or linear edge segments which consists of chains aligned edge points. However, the feature matching necessarily leads to a sparse depth map and the rest of surface must be reconstructed by approximation.

Feature-based matching has been more effective in stereo (see [Medioni and Nevatia 85]), and the remainder of the paper will concentrate on an algorithm that uses this approach.

1.2 Motivation

Traditionally, there are a number of constraints that can be used to prune the possible search space for candidate matches. For example, many algorithms use the sign of zero crossing, the "orientation" of the feature, epipolar geometry, etc.. These constraints are heuristically-based on assumptions about the imaging system and feature detectors. Yet, in general, these constraints are insufficient to remove the matching ambiguity for all features, and systems must employ more potent assumptions.

Among the most common classes of powerful assumptions is the imposition of a smoothness constraint. Even before the advent of computer vision, it was noted in [Gibson 50] that depth usually varies "smoothly" across surfaces. Thus, the disparity values derived from matching should also vary "smoothly".* This smoothness constraint can then be used to further constrain the matching process, and thus resolve most of the ambiguities. While this is an important observation, it leaves the meaning of "smoothly" up to the reader.

In vision research there have been many stereopsis models proposed with different "smoothly" varying disparities, such as, the continuity constraint of [Marr and Poggio 79], figural continuity in [Mayhew and Frisby 81] (see also [Kim and Bovik 86]), the disparity gradient limit of [Koenderink and vanDoorn 76], (see also [Pollard, Mayhew and Frisby 85]), the analytic disparity fields of [Eastman and Waxman 85], and the local planar/quadric patches

*Note that this constraint does not hold at the boundary of three dimensional objects and therefore the disparity along projections of such discontinuities need not be smooth.

of [Hoff and Ahuja 87]. All the constraints are intended to enforce a model of surface smoothness. However, they only partially capture the desired model. There are several problems in the above models:

1. It is difficult to translate surface smoothness constraints into disparity smoothness constraints. Depth is a nonlinear function of camera geometry, pixel position, and disparity. Therefore, smoothness assumptions are different in disparity space and real world. (Most of the above references model smoothness in disparity space). While it may be possible to define a realistic smoothness assumption in disparity space, it seems more likely to be able to do so for world surfaces.

2. Obviously, the matching process provides constrains for surface reconstruction. One interpretation of the smoothness constraint is to impose conditions on the matching phase such that the resulting reconstructed surface is generally smooth.. Traditionally, matching and surface reconstruction are two separate and time-sequential processes. Thus, matching could not make use of information from the reconstruction stage.

3. There may be multiple surfaces in the image. An edge segment may crosses different surfaces (e.g. when the contrast between the boundary of surfaces is not strong enough), and the disparity will not vary smoothly along the edge segment. Thus, algorithms that try to use a disparity smoothness over a window to locate the correct matcher will fail if the window cross several surfaces. This implies that surface segmentation must be incorporated into the matching process.

As to the surface segmentation problem, the most common approach is to determine the "discontinuity boundaries" in surface depth, surface orientation and/or surface curvature. This approach usually requires some reconstruction of the surface, and this presents numerous problems. First, in order to correctly reconstruct a surface, knowledge of the data segmentation is generally required. This results in a difficult chicken-and-egg problem. To make matters worse, the quality of the reconstruction in the neighborhood of an unmarked (i.e. as yet undetected) discontinuity is generally poor. Thus the localization of the discontinuity of iterative reconstruct/segment approaches, see e.g. [Terzopoulos 84] or [Hoff and Ahuja 87], will be questionable. Furthermore, any boundary-based segmentation approach will require considerable post processing to handle extended multiply connected objects (say behind a picket fence) and may never be able to handle transparent surfaces where locally there are only a few points on any one surface.

A final remark about traditional segmentation is related to the definition of "boundaries". It is well known that the perceived "boundaries" of surfaces in depth share many characteristics with subjective contours, see [Julesz 71], [Marr 81]. This suggests that a definition of "boundaries" in depth might be accomplished by some secondary processing which is shared with "boundary" detection from other visual modalities.

To ameliorate the above mentioned problems with boundary-based segmentation, this paper proposes that segmentation of 3D information should simply classify points as belonging to the same surface. The determination of boundaries will be relegated to some secondary process which will not be discussed here. [1]

In computer vision, as well as other domains, researchers have used minimal surface bending-energy as an assumption to aid in surface recovery, i.e., the acceptability of a "smooth surface" is inversely related to the energy (in term of a regularizing functional) of the surface, for example, see [Grimson 81], [Terzopoulos 84], [Wahba 84], [Franke 82], [Hoff and Ahuja 85], [Lee 85], [Choi and Kender 85], [Blake and Zisserman 86], [Boult 86], and [Lee and Pavlidis 87]. Thus, bending energy appears to be a natural choice for a "measure" determining if a group of points belong to the same surface. The bending energy of a surface f is given by:

$$\left\{ \int\int_{\mathbb{R}2} \left(\left(\frac{\partial^2 f}{\partial x^2} \right)^2 + 2 \cdot \left(\frac{\partial^2 f}{\partial x \partial y} \right)^2 + \left(\frac{\partial^2 f}{\partial y^2} \right)^2 \right) dx\, dy \right\}^{\frac{1}{2}} \qquad (1)$$

Of course, the above measure can only be the basis for a practical measure for segmentation. Other issues that must be addressed by a practicable measure include:

- Determination of the threshold for separation of a group or alternatively defining the tradeoff between the number of surfaces and sum of the energy of these surfaces to keep the system from segmenting the data into a large number of planar patches (which have zero energy). (The algorithm presented herein follows the first approach.)
- Careful determination of how to handle surface size or equivalently, the area over which the energy is measured.
- Relation of the energy to the number of data points,
- Determination of which point(s) in a group are the cause of a surface energy which is too high, i.e., the credit assignment problem.
- Relation of "depth" discontinuities and "orientation" discontinuities and how they effect the energy measure.

The authors acknowledge that there are numerous other measures of "surface smoothness" as might be implemented by parametric surface patches (e.g. [Allen 85]) or volumetric models (e.g. see [Rao, Nevatia and Medioni 87], [Boult and Gross 87], or [Bajcsy and Solina 87]. These approaches deserve careful consideration in future research efforts.

2 Mathematical Model and Tools

From the above discussion, what we need is a model of smooth world surfaces. By using such a model and some mathematical tools, we can simultaneously do matching, surface reconstruction and segmentation.

2.1 Definition of the Model of World Surfaces

The assumed model of world surface is intimately related to techniques for regularized surface reconstruction, see [Boult 86]. The

[1] Of course this view cannot be taken too far, there must be some limit to the number of possible "transparent" surfaces and some limit to the extent of any "disconnected" object which will be recognized as connected.

class of surfaces used is defined as those functions (distributions) with their second derivative (in a distributional sense) in $H^{\frac{1}{2}}$, where $H^{\frac{1}{2}}$ is the Hilbert space of functions such that their tempered distributions ν in $\mathbb{R}^2$ have Fourier transform $\tilde{\nu}$ that satisfy

$$\int\int_{\mathbb{R}^2} \left(|\tau| \cdot |\tilde{\nu}(\tau)|^2 \, d\tau \right) \quad < +\infty.$$

This class of functions, referred to as $D^{-2}H^{\frac{1}{2}}$, is equipped with the second Sobolev semi-norm,

$$\| \cdot \|_{D^{-2}H^{\frac{1}{2}}} = \left\{ \int\int_{\mathbb{R}^2} \left(\left(\frac{\partial^2 f}{\partial x^2} \right)^2 + 2 \left(\frac{\partial^2 f}{\partial x \partial y} \right)^2 + \left(\frac{\partial^2 f}{\partial y^2} \right)^2 \right) dx \, dy \right\}^{\frac{1}{2}} \tag{2}$$

which makes it a semi-Hilbert space.

Intuitively these functions are smooth (almost everywhere) up to derivatives of order approximately 1.5, i.e., they are significantly smoother than membrane surfaces but are not as smooth as thin-plate splines. The motivation for this choice is this "intermediate" level of smoothing assumed, and is supported by the results of [Boult 87].

2.2 The Definition of Reproducing Kernel-Based Spline

An essential ingredient of the current algorithm, at least from the point of view of efficient serial implementation, is the use of the reproducing kernel-based spline reconstruction as described in [Boult 86]. This section introduces some aspects of that algorithm necessary for later discussions.

Among all functions in the above class, the surface reconstruction aspect of the segmentation algorithm is required to find the surface which minimizes

$$\lambda \cdot \sum_{i=1}^{n} \frac{(\sigma(x_i, y_i) - z_i)^2}{\delta_i} + \|\sigma\|_{D^{-2}H^{\frac{1}{2}}}$$

where the data z_i at point (x_i, y_i), $i = 1, \ldots, n$ is assumed to be on one surface. The *global smoothing parameter*, λ, should depend on the overall error in the initial data, and the factors δ_i allow for individual points to have greater "noise"; the factor λ effects the overall tradeoff between surface smoothness (as measured by the norm $\| \cdot \|_{D^{-2}H^{\frac{1}{2}}}$) and the fidelity to the data points z_i while the factors δ_i effects the contribution of a single data point so as not to penalize the surface as much (or to penalize it more, depending on the value of δ_i) for not closely approximating the data at that point. Techniques for choosing these parameters have been discussed by other researchers, see [Bates and Wahba 82].

One solution to the above reconstruction problem is a reproducing kernel-based spline. It has already been shown, see [Meinguet 83], that for the above model of world surfaces, the appropriate reproducing kernel here is

$$K(x, y; u, v) = \gamma((x-u)^2 + (y-v)^2)^{\frac{3}{2}}$$

for a known constant γ

Given the above kernel, the spline which approximates the information

$$z = z_1, \ldots, z_k = \{f(x_1, y_1), \ldots, f(x_k, y_k), \quad i = 1, \ldots, k\}$$

can be expressed as:

$$\sigma_z = \sum_{i=1}^{k} \alpha_i K(x, y; x_i, y_i) + \beta_1 + \beta_2 x + \beta_3 y \tag{3}$$

where the constants α_i and β_i are the solution to the system of linear equations:

$$\begin{matrix} \mathcal{A}_{1,1} & \ldots & \mathcal{A}_{1,k} & \mathcal{B}_{1,1} & \mathcal{B}_{1,2} & \mathcal{B}_{1,3} & & z_1 \\ \vdots & \ddots & \vdots & \vdots & \vdots & \vdots & & \vdots \\ \mathcal{A}_{k,1} & \ldots & \mathcal{A}_{k,k} & \mathcal{B}_{k,1} & \mathcal{B}_{k,2} & \mathcal{B}_{k,3} & = & z_k \\ \mathcal{C}_{1,1} & \ldots & \mathcal{C}_{1,k} & \mathcal{D}_{1,1} & \mathcal{D}_{1,2} & \mathcal{D}_{1,3} & & 0 \\ \mathcal{C}_{2,1} & \ldots & \mathcal{C}_{2,k} & \mathcal{D}_{2,1} & \mathcal{D}_{2,2} & \mathcal{D}_{2,3} & & 0 \\ \mathcal{C}_{3,1} & \ldots & \mathcal{C}_{3,k} & \mathcal{D}_{3,1} & \mathcal{D}_{3,2} & \mathcal{D}_{3,3} & & 0 \end{matrix}$$

where (4)

$$\begin{aligned} \mathcal{A}_{i,i} &= \tfrac{\alpha_i \lambda \delta \pi}{\delta_i} \quad i = 1, \ldots, k; \\ \mathcal{A}_{i,j} &= \alpha_j (K(x_j, y_j; x_i, y_i)), \quad i, j = 1, \ldots, k, \; i \neq j; \\ \mathcal{B}_{i,j} = \mathcal{C}_{j,i} &= \beta_j p_j(x_i, y_i) \quad i = 1, \ldots, k, \; j = 1, \ldots, 3; \\ \text{and} \quad \mathcal{D}_{i,j} &= 0 \; i = 1, \ldots, 3 \; j = 1, \ldots, 3; \end{aligned}$$

The important properties of the above solution to the surface reconstruction problem are:

1. The algorithm is efficient for very sparse data (anything more than 3 non-colinear points will do, and the fewer the number of points, the faster the surface can be computed).
2. The surface is defined by the solution to a linear system which depends only on the location of the data. If the solution to this system can be updated quickly, the surface can also be updated quickly.
3. The surface is given in a functional form, thus the evaluation of derivatives is trivial, and bounds on the energy of the surface can be computed analytically.
4. The surfaces are independent of the "boundaries" of discontinuities, and depend only on the data values. However, the actual surface will change if the number/value of data points on the boundary are changed.

2.3 Definition of the Energy Measure

The basic form of the energy measure is given by equation 2 except that the region of integration may be different than that expressed therein.

The energy of the surface will depend on the size of the region in $\mathbb{R}^2$ over which the energy norm is computed. The two most natural choices are $\mathbb{R}^2$ itself, and the convex hull of the data defining the "current" surface. Unfortunately, neither of these is appropriate. For the above class of functions, the integral over $\mathbb{R}^2$ is not necessarily finite. Although the energy norm over the convex hull of the data defining the "current" surface is obviously finite, this choice has two difficulties:

1. The convex hull would be continuously changing as new data points were added to a surface.
2. The use of a domain which ends near the data points will allow the addition of new points to actually lower the surface energy, thus the energy will no longer be monotonicly

increasing and segmentation could not proceed with a region growing method.

Because of the above difficulties with the "natural" choices for the domain of integration, the algorithm uses the following heuristic: given the starting basis, the algorithm computes the energy of the surface over a square region which is centered around the centroid of the data (including the points not yet considered) with the length of the side of the rectangle 100 times larger than the larger dimension of the rectangle bounding all data points.

2.4 Derivation of Bounds on Energy of Surface

Given the definition of the spline as in equation 3, one can symbolically compute bounds on the energy. To begin, the exact form of the energy integral is manipulated to explicitly expand the squaring operation and move the differentiation and integration inside the sum, to wit:

$$
\begin{aligned}
&\|\sigma\|_{D^{-1}H^{\frac{1}{2}}} = \\
&\quad \left\{ \int_{X_l}^{X_u} \int_{Y_l}^{Y_u} \left(\left(\frac{\partial^2 \sum_{i=1}^{k} \alpha_i K(x,y;x_i,y_i)}{\partial x^2} \right)^2 \right.\right. \\
&\qquad +2 \cdot \left(\frac{\partial^2 \sum_{i=1}^{k} \alpha_i K(x,y;x_i,y_i)}{\partial x \partial y} \right)^2 \\
&\qquad \left.\left. + \left(\frac{\partial^2 \sum_{i=1}^{k} \alpha_i K(x,y;x_i,y_i)}{\partial y^2} \right)^2 \right) \, dx \, dy \right\}^{\frac{1}{2}} \\
&= \left\{ \int_{X_l}^{X_u} \int_{Y_l}^{Y_u} \left(\left(\sum_{i=1}^{k} (\alpha_i \frac{\partial^2 K(x,y;x_i,y_i)}{\partial x^2}) \right)^2 \right.\right. \\
&\qquad +2 \cdot \left(\sum_{i=1}^{k} (\alpha_i \frac{\partial^2 K(x,y;x_i,y_i)}{\partial x \, \partial y}) \right)^2 \\
&\qquad \left.\left. + \left(\sum_{i=1}^{k} (\alpha_i \frac{\partial^2 K(x,y;x_i,y_i)}{\partial y^2}) \right)^2 \right) \, dx \, dy \right\}^{\frac{1}{2}} \qquad (5) \\
&= \left\{ \begin{array}{l} \sum_{i=1}^{k} \sum_{j=1}^{k} \left(\int_{X_l}^{X_u} \int_{Y_l}^{Y_u} (\alpha_i K_{xx}(x,y;x_i,y_i)) \right. \\ \qquad \left. \cdot (\alpha_j K_{xx}(x,y;x_j,y_j)) \, dx \, dy \right) \\ + \; 2 \cdot \sum_{i=1}^{k} \sum_{j=1}^{k} \left(\int_{X_l}^{X_u} \int_{Y_l}^{Y_u} (\alpha_i K_{xy}(x,y;x_i,y_i)) \right. \\ \qquad \left. \cdot (\alpha_j K_{xy}(x,y;x_j,y_j)) \, dx \, dy \right) \\ + \; \sum_{i=1}^{k} \sum_{j=1}^{k} \left(\int_{X_l}^{X_u} \int_{Y_l}^{Y_u} (\alpha_i K_{yy}(x,y;x_i,y_i)) \right. \\ \qquad \left. \cdot (\alpha_j K_{yy}(x,y;x_j,y_j)) \, dx \, dy \right) \end{array} \right\}^{\frac{1}{2}}
\end{aligned}
$$

While it would be most appropriate to symbolically integrate the terms in the last of the above equations, the authors (and MACSYMA) have been unable to obtain a solution. Fortunately, a symbolic solution can be obtained for bounds on the above equations. First note that if $V_i \geq 0, i = 1, \ldots, n$ then

$$
\begin{aligned}
&\sum_{i=1}^{k} \sum_{j=1}^{k} a_i \cdot a_j \cdot (min(V_i, V_j))^2 \\
&\quad \langle \sum_{i=1}^{k} \sum_{j=1}^{k} a_i \cdot a_j \cdot V_i \cdot V_j \\
&\quad \langle \sum_{i=1}^{k} \sum_{j=1}^{k} a_i \cdot a_j \cdot (max(V_i, V_j))^2.
\end{aligned} \qquad (6)
$$

In fact, the upper bound is trivially true even if some of the V_i's are negative. Thus an upper bound on the energy integral above can be written in terms of similar to

$$
\sum_{i=1}^{k} \sum_{j=1}^{k} max \left[\begin{array}{l} \left(\int_{X_l}^{X_u} \int_{Y_l}^{Y_u} (K_{xx}(x,y;x_i,y_i))^2 , dx \, dy \right), \\ \left(\int_{X_l}^{X_u} \int_{Y_l}^{Y_u} (K_{xx}(x,y;x_j,y_j))^2 dx \, dy \right) \end{array} \right]
$$

While the general energy integral has not been computable in closed form, the above simpler integral is computable in closed form. In particular, one can derive:

$$
\begin{aligned}
&\int_{X_l}^{X_u} \int_{Y_l}^{Y_u} (K_{xx}(x,y;s,t))^2 dx \, dy = \\
&\left\{ \begin{array}{l}
9 \cdot \left[tan^{-1}(\frac{X_u - s}{Y_u - t}) \cdot (\frac{X_u}{4} - s \cdot X_u^3 \right. \\
\qquad \left. + \frac{3}{2} s^2 \cdot X_u^2 - s^3 \cdot X_u) \right. \\
\qquad + \; tan^{-1}(\frac{Y_u - t}{X_u - s}) \cdot (\frac{Y_u}{4} - t \cdot Y_u^3 \\
\qquad\quad + \frac{3}{2} t^2 \cdot Y_u^2 - t^3 \cdot Y_u + \frac{1}{4}(t^4 - s^4)) \\
\qquad + \; (Y_u - t) \cdot (X_u \cdot \frac{1}{12} \cdot (6 \cdot t \cdot Y_u - 3 \cdot (Y_u^2 - t^2)) \\
\qquad\quad + \frac{1}{36} \cdot (3 \cdot X_u^3 + 9 \cdot (s \cdot X_u^2 - s^2 \cdot X_u))) \\
\qquad + \; X_u \cdot (3 \cdot Y_u^3 + 9 \cdot (t \cdot Y_u^2 - t^2 \cdot Y_u)) \\
\qquad + \; 3 \cdot Y_u \cdot (3 \cdot X_u^3 + 9 \cdot (s \cdot X_u^2 - s^2 \cdot X_u)) \Big] \\
-9 \cdot \left[tan^{-1}(\frac{X_l - s}{Y_u - t}) \cdot (\frac{X_l}{4} - s \cdot X_l^3 \right. \\
\qquad \left. + \frac{3}{2} s^2 \cdot X_l^2 - s^3 \cdot X_l) \right. \\
\qquad + \; tan^{-1}(\frac{Y_u - t}{X_l - s}) \cdot (\frac{Y_u}{4} - t \cdot Y_u^3 \\
\qquad\quad + \frac{3}{2} t^2 \cdot Y_u^2 - t^3 \cdot Y_u + \frac{1}{4}(t^4 - s^4)) \\
\qquad + \; (Y_u - t) \cdot (X_l \cdot \frac{1}{12} \cdot (6 \cdot t \cdot Y_u - 3 \cdot (Y_u^2 - t^2)) \\
\qquad\quad + \frac{1}{36} \cdot (3 \cdot X_l^3 + 9 \cdot (s \cdot X_l^2 - s^2 \cdot X_l))) \\
\qquad + \; X_l \cdot (3 \cdot Y_u^3 + 9 \cdot (t \cdot Y_u^2 - t^2 \cdot Y_u)) \\
\qquad + \; 3 \cdot Y_u \cdot (3 \cdot X_l^3 + 9 \cdot (s \cdot X_l^2 - s^2 \cdot X_l)) \Big] \\
+9 \cdot \left[tan^{-1}(\frac{X_l - s}{Y_l - t}) \cdot (\frac{X_l}{4} - s \cdot X_l^3 \right. \\
\qquad \left. + \frac{3}{2} s^2 \cdot X_l^2 - s^3 \cdot X_l) \right. \\
\qquad + \; tan^{-1}(\frac{Y_l - t}{X_l - s}) \cdot (\frac{Y_l}{4} - t \cdot Y_l^3 \\
\qquad\quad + \frac{3}{2} t^2 \cdot Y_l^2 - t^3 \cdot Y_l + \frac{1}{4}(t^4 - s^4)) \\
\qquad + \; (Y_l - t) \cdot (X_l \cdot \frac{1}{12} \cdot (6 \cdot t \cdot Y_l - 3 \cdot (Y_l^2 - t^2)) \\
\qquad\quad + \frac{1}{36} \cdot (3 \cdot X_l^3 + 9 \cdot (s \cdot X_l^2 - s^2 \cdot X_l))) \\
\qquad + \; X_l \cdot (3 \cdot Y_l^3 + 9 \cdot (t \cdot Y_l^2 - t^2 \cdot Y_l)) \\
\qquad + \; 3 \cdot Y_l \cdot (3 \cdot X_l^3 + 9 \cdot (s \cdot X_l^2 - s^2 \cdot X_l)) \Big]
\end{array} \right. \qquad (7)
\end{aligned}
$$

$$
\left.
\begin{aligned}
-9 \cdot \Big[& tan^{-1}\left(\frac{X_u - s}{Y_l - t}\right) \cdot \left(\frac{X_u}{4} - s \cdot X_u^3 \right. \\
& \left. + \tfrac{3}{2} s^2 \cdot X_u^2 - s^3 \cdot X_u\right) \\
+ \; & tan^{-1}\left(\frac{Y_l - t}{X_u - s}\right) \cdot \left(\frac{Y_l}{4} - t \cdot Y_l^3 \right. \\
& \left. + \tfrac{3}{2} t^2 \cdot Y_l^2 - t^3 \cdot Y_l + \tfrac{1}{4}(t^4 - s^4)\right) \\
+ \; & (Y_l - t) \cdot \left(X_u \cdot \tfrac{1}{12} \cdot (6 \cdot t \cdot Y_l - 3 \cdot (Y_l^2 - t^2)) \right. \\
& \left. + \tfrac{1}{36} \cdot (3 \cdot X_u^3 + 9 \cdot (s \cdot X_u^2 - s^2 \cdot X_u))\right) \\
+ \; & X_u \cdot (3 \cdot Y_l^3 + 9 \cdot (t \cdot Y_l^2 - t^2 \cdot Y_l)) \\
+ \; & 3 \cdot Y_l \cdot (3 \cdot X_u^3 + 9 \cdot (s \cdot X_u^2 - s^2 \cdot X_u))\Big]
\end{aligned}
\right\}
$$

Similar derivations exists for the two integrals

$$\int_{X_l}^{X_u} \int_{Y_l}^{Y_u} (K_{xy}(x, y; s, t))^2 dx\, dy$$

and

$$\int_{X_l}^{X_u} \int_{Y_l}^{Y_u} (K_{yy}(x, y; s, t))^2 dx\, dy$$

(The latter can, in fact, be obtained by a change of variables in equation 8.)

Combining equations 6 and evaluating the formulas as in 8 one can obtain closed form equations for the upper bound on the energy of a reproducing kernel-based spline. The lower bound is a bit more difficult. If the terms $K_{xx}(x, y; x_i, y_i)$, $K_{xy}(x, y; x_i, y_i)$, and $K_{yy}(x, y; x_i, y_i)$ were all nonnegative, then the bound from equation 6 would apply. Unfortunately, the terms may be negative.

For the segmentation process, it is considerably more convenient to use a single number (i.e. if energy $\leq$ threshold) rather than developing some technique to handle both upper and lower bounds. While the upper bound alone could be used, this seems to produce too conservative an estimate. Thus, throughout this paper, the phrase "energy" of a surface refers to the heuristic estimate given by the average of the upper bound and the "lower" bound from equation 6 divided by the number of points defining the surface. While this is theoretically a meaningless number, the results in later sections support this as a reasonable heuristic. When we derive a true lower bound, we believe that the same heuristic will be appropriate, only it will be more robust.

3 The Integration of Matching, Surface Reconstruction, and Segmentation

Matching, surface reconstruction and segmentation work "cooperatively" in this stereo algorithm. The first pass determines the potential matches for features, and the uniquely matching features determine initial depth data which is used for surface reconstruction (and segmentation). The current surface reconstruction provides the surface smoothness constraint which is used to disambiguate the remaining potential matching features, updates the surface reconstruction/segmentation as it goes.

The algorithm has five phases:

1. image acquisition and camera calibration,
2. feature detection,
3. determination of the potential matches, and the amount of ambiguity for each feature,
4. initial reconstruction and segmentation of surfaces,
5. disambiguation of the remaining ambiguous features with continual refinements of the segmented surface reconstructions.

Each of these phases is described in turn.

3.1 Image Acquisition and Camera Calibration

The stereo images were taken using a single camera at two different positions. Because of the rotation of the camera and lens distortion, it is difficult to have a horizontal epipolar line. However, we still can estimate the non-horizontal epipolar geometry (see the feature detection phase).[‡]

The aim of calibration is to calculate the perspective transformation matrix between 3-D world coordinates and image coordinates. The algorithm used by the system is based on a procedure in [Duda and Hart 73], see also [Ballard and Brown 82]. Given the measured 3-D world coordinates of a number of non-coplanar calibration points and the corresponding 2-D image coordinates, the coefficients in the perspective transformation matrix can be computed by least square solution. Given the perspective transformation matrix, the camera parameter can be calculated if needed (e.g., see [Ganapathy 84]), and the depth value of features can also be computed after the matching phase completes.

3.2 Feature Detection

The features used are an interest operator based on [Moravec 79], and the zero crossings of the Laplacian of the Gaussian (e.g. seee [Marr and Hildreth 80]). The reasons for using multiple features are:

1. Since the feature points of interest operator are very sparse, most of the points are uniquely matched. We can make use of the already matched pairs to estimate the non-horizontal epipolar geometry (mainly, the vertical disparity). This is needed to match the zero-crossings which cannot be easily distinguished vertically.
2. The zero crossing of the image provide a large number of features for matching algorithm, unfortunately the localization of these features is not highly accurate (especially if there are errors in vertical disparity). The features from the interest operator are not very dense, however, they provide very accurate localization of the features. By combining the two different types of features, we can avoid the problem that features of the stereo system are too sparse, have poor localization, or are sensitive to noise.

[‡] Although it is not difficult in principle to calculate non-horizontal epipolar geometry from camera parameters and imaging geometry, most stereo systems would rather use a parallel camera model to allow the use of the more efficient scan-line coherency constraint.

3. A final reason, possibly unique to our approach, is that we need a number of "unique matches" to build out initial surface reconstruction. The features from the interest operator generally produce a unique match, and thus supply numerous points for our initial surface reconstruction.

Additionally, to reduce error due to digitization and early processing, the zero crossing are thresholded based on the gradient magnitude. A quantitative argument about the threshold value was described in [Kim and Bovik 86]. If necessary, the output of interest operator also can be thresholded to ensure unique matching.

Since the number of points provided by two feature detectors are different, in building the resulting surface, each data point must give a weight. Otherwise, the zeroing crossing (with 1000-2000 points) would totally dominate the point's generated by interest operator (with 100-200 points).

3.3 Determination of Possible Matches and Feature Ambiguity

First, match the points produced by interest operators. Because these specific features are very sparse, the searching space can be expanded vertically, and the result will have little ambiguity. As mentioned above, after matching these features, the epipolar geometry of the image can be calculated. It is also assumed that some information about the experimental environment is known, e.g. the maximum and minimum depth, then by the perspective transformation matrix, the location and width of searching window (vergence window) can be estimated.

Secondly, for every non-horizontal zero crossing in the left image, a search is performed along the corresponding epipolar line. Assume the width of the searching window is L. A feature can match only those features in the window with similar features. For zero-crossings, "similar" is defined as having the same sign, and an orientation within $\pm 30°$ of the other feature.

Each possible matching feature within the window in the right image is considered for a given feature in the left image. If there is only one point, the match is considered unique, otherwise the number of potential matches ($< \frac{L}{2}$) characterizes the degree of ambiguity. Currently, the algorithm will reject any point which has more than $\frac{L}{2}$ possible matches.

3.4 Surfaces Reconstruction and Segmentation

After the determination of potential matches, those matches which were determined to be unique are converted into depth data. Surface(s) are reconstructed incrementally using reproducing kernel-based spline(s). Surface reconstruction and segmentation are two concurrent processes. The algorithm proceeds by building an initial approximation of a surface from the depth data.** Points are added to a surface as long as the addition does not cause the energy of said surface to exceed a certain threshold. When multiple surfaces exist, the different surfaces are tried, and the surface with the lowest energy will accept the point. If no surface can accept the point, a "new" surface is created and the point is added to that surface. This process is continued until all data points have been processed.

**For the current implementation, this is 4 data-points. The points are chosen as a local cluster, although this is not critical to the performance and may cause problems when transparent surfaces are considered.

3.5 Disambiguation of Ambiguous Features

After the initial surface(s) are "reconstructed" from the unique matches, they are used to disambiguate the remaining potential matches. The ambiguous matches are considered in increasing order of ambiguity, and within a given level of ambiguity, the reconstructed world-surface is incrementally updated by considering the matches from left to right and from top to bottom. For each ambiguous match, the algorithm uses the information from the calibration phase to compute the possible three dimensional coordinates of the "feature" for each of the possible matches. The potential match which corresponds to the three dimensional point "closest" to any of the existing surfaces is considered the correct match. The point is then added to the reconstructed world-surface, using the level of ambiguity to adjust the associated parameter δ_i.

4 The Good points and the Bad Points of the Approach

This section critically reviews the algorithm described in this paper pointing out some of the major advantages +, major problems −, and some aspects which can be viewed as either a good or bad ±, depending on ones point of view.

+ The segmentation algorithm can handle transparent and occluded objects with few problems.

+ The segmentation process is based on surfaces having low bending energy, a heuristic which can be directly related to the physical process of surface formation.

+ The functional form of the reproducing kernel-based spline allows for direct estimation of the surface energy, thus making the segmentation process reasonably computationally efficient.

+ In the algorithm, multiple features are used. This reduces the need for the severe scan-line coherency constraint, while still allowing a large number of reliable features.

± In the camera calibration phase, only the perspective transformation matrix is calculated. This is flexible, allowing one to use a single camera to acquire stereo images. However, when base line information cannot be obtained correctly and there is a certain amount of vertical disparity, the depth value is best computed by a least square solution (i.e. not by the traditional triangulation techniques). Since we use least square method twice (the first time used is in calibration phase), one might expect the error to be large, however, the experimental results show the error of this procedure is still acceptable, partially because the reproducing kernel-based spline allows individual points to have greater "noise".

± The algorithm does not recover "boundaries" for the segmented data. This is advantageous because it allows for transparent and/or occluding surfaces, and because data is generally

sparse (and often noisier) near the boundary resulting in a poor boundary definition. This is a disadvantage because it requires a secondary process (possibly using ideas borrowed from work on subjective contour perception or Gestalt psychology) to determine the actual boundary. It is also is a disadvantage because depth discontinuities induce a region where no potential matches can exist. Because the algorithm does not develop boundaries, it cannot make use of this observation.

± The algorithm can easily be adapted to different measures of surface smoothness. This is advantageous because it allows for greater flexibility, but disadvantageous because determination of the most appropriate measure is difficult. The measure used in the experiments presented herein has proved to be a reasonable one.

± The algorithm uses reproducing kernel-based splines which are essentially a global surface reconstruction algorithm and provide for efficient serial implementation for sparse data (say < 1000 points per surface). If there are more points then the algorithm can be extended to use local reproducing kernel-based splines (loosely based on [Franke 82]), at the cost of making the surface definition localized to patches. The local method has been evaluated and performs reasonably well on large data sets but very poorly on sparse data (probably because some of the patches may have little or no data).

± Since the current algorithm segments the depth data by one pass, the order of processing of points will effect the resultant segmentation, especially when two surfaces come into direct contact and join in a rather smooth fashion (e.g. the wedge example above). This may actually be used to help in the segmentation process by processing the data in multiple orders and using any difference in data labeling to suggest a refined segmentation.

± The linear systems which define the splines are known to be moderately ill-conditioned, see [Boult 86]. This problem is exacerbated when the data used to define the splines is nearly linearly dependent. Unfortunately, because of the smoothness assumptions implied by the model, if two points are very close in x, y, (relative to the size of the area x, y being considered), and have similar z values, the information becomes more linearly dependent (if the Z values are different they will almost assuredly be segmented). Fortunately, and directly because it is almost linearly dependent, such information does not significantly effect the reconstructed surface. Thus, the algorithm can determine that such information is redundant and discard that information. Currently, this is done heuristically but future work will investigate the usefulness of such information in modifying the confidence of those points which are maintained by the system.

– The algorithm currently uses a heuristic approximation of the surface energy divided by the number of data points in the surface as a threshold for the segmentation. This is a hack, and future work must attempt to redress this issue. Luckily, this threshold for energy-based segmentation does not seem as sensitive as say thresholds for segmentation of an image based on intensity.

– The algorithm assumes one is interested in smooth surfaces and will most likely fail when this assumption is not satisfied. Unfortunately, the algorithm cannot even determine if the assumptions are satisfied (For example, consider a rough surface similar to a plane covered with a large number of small densely packed cones. If the data supplied to the algorithm are points on the background and the peak values of the cones, the algorithm is hopelessly doomed to predict two planar surfaces.)

– The algorithm is surfaced based, and cannot deal with data from multiple views of a volumetric object. Additionally, it will often fail if noise is such that a single x, y location is assigned multiple data values (of the same type).

5 Experimental Result

One set of synthetic images and two set of real images are presented to illustrate the performance of the algorithm. Both are 512 by 512 with 8 bits of grey scale. The purpose of the synthetic images was to enable us to obtain some estimates of the error of the system. The other scene poses more realistic problems.

We first comment on the graphical display of surfaces. The reproducing-kernel splines (like almost any approximation algorithm) are not extremely good at extrapolation and display of the surface far from any data would be misleading. Since the reconstruction does not determine boundaries, there are no "clean" edges for display. Thus, the graphical display shows only the portion of the surface in the convex-hull of the data. The display of occluding or transparent surface is also difficult (with resorting to a ray-tracer) and thus, some of the surfaces are presented "floating" in space.

Figure 1 shows two planes synthetic image, the x, y value are in the range $[-.1, 2.0]$ with two synthetic camera locations at a distance of approximately 40 units. The equations of the underlying planes are $z = 0$ when $x < 1$, and $z = 2x$ when $x > 1$. The reconstructed surfaces can be seen in figure 2.

The system uses 2 synthetic images, first for calibration and then for the stereo processing. In the calibration phase the z value for the center of each square is assumed known, and the image coordinates of each square are obtained by thresholding and computing the centroid.

After stereo processing, the estimated depth values were compared with the underlying plane equations, assuming the error was in the direction of left-camera position. The RMS of error is 0.049 with variance 0.045. The maximum value of the error was .111 on the plane $z = 0$ and .45 on $z = 2x$.

Figure 3 shows a stereo pair of images of a cup and a playing card. The range of depth values is 90 mm to 140 mm (measured in z direction), and the camera location was at about 500mm (left camera on the z axis). Figure 4 shows some of the output of two feature detectors. In this example, the algorithm found 264 unique matches and 923 ambiguous points which could be disambiguated. The remaining 230 points were rejected (declared unmatchable) either because they had too many potential matches or no potential matches with similar features. Figure 5 shows reconstructions after the algorithm has successfully performed matching, reconstruction and segmentation on the two surfaces in the scene.

The third example shows that the algorithm can work well

even with transparent (or extended, multiply connected) objects. In figure 6 the reader can see a few square labels on a wall behind a glass plate with triangular labels on it.

The range of depth value in the scene is -5 mm to 140 mm, and again the camera was at about 500mm. In this example, the algorithm found 155 unique matches and 786 ambiguous points. The other 204 points were rejected. Figure 8 shows the reconstructed surfaces.

Currently the algorithm requires about 20min (wall clock time) on a Vax750 when processing a 512 by 512 image.[††] This is an unacceptable time requirement for practical problems, and future work on optimization and possible parallel implementation will address this issue.

6 Conclusion

This paper has presented an integrated approach for stereo matching,visual surface reconstruction and segmentation. The algorithm uses a model of surface smoothness which can be based on physical properties, and which provide for a flexible choice of different smoothness measures. The segmentation algorithm does not determine boundaries between segmented surfaces which allows it to handle extended objects occluded by other objects and transparent objects. The algorithm has been successfully demonstrated one synthetic image and two real images examples, but further testing on more complex images is needed.

One possible disadvantage of the approach is that the overall reliability of the stereo system heavily depends on that of the first pass of matching. If the first pass of matching cannot be achieved reliably, the overall approach cannot succeed. If the unique (unambiguous) matching cannot always result in correct matching, then in the segmentation phase when the energy of a certain surface exceeds the threshold, there is no way to know whether a point should be on a different surface or if it is just a mismatched point (which should be discarded). Future work will investigate, on the assumption that the mismatched points are few, a way of identifying these points (probably using a local energy measure). In particular, we will examine the direct use of the energy measure to determine the best match from the potential match set.

The second fault is due to the fact that the segmentation algorithm cannot predict boundary contours. Thus, when the ambiguous point is close to the occluded boundary, it is difficult to decide which surface the point should be on, and cannot choose the correct match. Currently the algorithm chooses the potential match which corresponds to a 3 dimensional point "closest" to "any" of the existing surfaces. Future work will involve the development of an algorithm which can take the "clusters" of data points determined by the energy-based segmentation algorithm, and compute the subjective contours that determine their occluding boundaries.

Finally, the current approach depends on a global thresholding technique to realize the segmentation. Such a process is doomed to be troublesome unless a systematic determination of the threshold is possible. Future work will address this issue and will also investigate the use of adaptive thresholding (depending on the actual data) and the use of other properties, say rate of change of energy, as the means of realizing segmentation.

[††] Admittedly, like most code developed for research purposes there has been very little attempt to optimize the implementation of the algorithm. The major point of this research to date has been to show that the algorithms are reasonable and that they can be computed with moderate time complexity.

Acknowledgments

This work was supported in part by Darpa contract #N00039-84-C-0165.

References

[Allen 85] P. K. Allen. *Object Recognition Using Vision and Touch.* PhD thesis, University of Pennsylvania, Department of Computer Science., 1985.

[Bajcsy and Solina 87] R. Bajcsy and F. Solina. Three dimensional object representation revisted. In *Proceedings of the IEEE Computer Society International Conference on Computer Vision*, pages 231–240, IEEE, June 1987.

[Ballard and Brown 82] D.H. Ballard and C.M. Brown. *Computer Vision.* Prentice-Hall, Inc., Englewood Cliffs, NJ, 1982.

[Barnard and Fischler 82] S.T. Barnard and M.A. Fischler. Computational stereo. *Computer Surveys*, 14(4), December 1982.

[Bates and Wahba 82] D. Bates and G. Wahba. Computational methods for generalized cross-validation with large data sets. In C.T.H. Baker and G.F. Miller, editors, *Treatment of Integral Equations by Numerical Methods*, pages 283–296, Academic Press, New York, 1982.

[Blake and Zisserman 86] A. Blake and A. Zisserman. Invariant surface reconstruction using weak continuity constraints. In *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 62–68, IEEE, 1986.

[Boult 86] Terrance E. Boult. *Information Based Complexity in Non-Linear Equations and Computer Vision.* PhD thesis, Department of Computer Science, Columbia University. 1986.

[Boult 87] T.E. Boult. What is regular in regularization? In *Proceedings of the IEEE Computer Society International Conference on Computer Vision*, pages 457–462, IEEE, June 1987.

[Boult and Gross 87] T.E. Boult and A.D. Gross. Recovery of superquadrics from depth information. In *Proceedings of the AAAI Workshop on Spatial-Reasoning and Multisensor Integration*, AAAI, October 1987.

[Choi and Kender 85] D.J. Choi and J. R. Kender. Solving the depth interpolation problem with adaptive chebyshev acceleration method on a parallel computer. In *Proceedings of the DARPA Image Understanding Workshop*, pages 219–223, DARPA, 1985.

[Duda and Hart 73] R. O. Duda and P.E. Hart. *Pattern Classification and Scene Analysis.* Wiley, NYC, NY, 1973.

[Eastman and Waxman 85] R.D. Eastman and A.M. Waxman. Disparity functionals and stereo vision. In *Proceedings of the DARPA Image Understanding Workshop*, pages 245–254, DARPA, 1985.

[Franke 82] R. Franke. Smooth interpolation of scattered data by local thin plate splines. *Comp. & Math. with Applications*, 8(4):273–281, 1982.

[Ganapathy 84] S. Ganapathy. Decomposition of transformation matrices for robot vision. In *International Conference on Robotics and Automation*, pages 130–139, 1984.

[Gibson 50] J.J. Gibson. *The Perception of the Visual World.* Houghton-Mifflin, Boston, 1950.

[Grimson 81] W. E. L. Grimson. *From Images to Surfaces: A Computational Study of the Human Visual System.* MIT Press, Cambridge, MA, 1981.

[Hoff and Ahuja 85] W. Hoff and N. Ahuja. Surfaces from stereo. In *Proceedings of the DARPA Image Understanding Workshop*, pages 98–106, DARPA, 1985.

[Hoff and Ahuja 87] W. Hoff and N. Ahuja. Extracting surfaces from stereo images: an integrated approach. In *Proceedings of the IEEE Computer Society International Conference on Computer Vision*, pages 284–294, IEEE, 1987.

[Julesz 71] B. Julesz. *Foundations of Cyclopean Perception.* University of Chiage Press, Chiago, IL, 1971.

[Kim and Bovik 86] N.H. Kim and A.C. Bovik. A solution to the stereo correspondence problem using disparity smoothness constraints. In *Proceegings of the IEEE conference on Systems, Man, and Cybernetics*, October 1986.

[Koenderink and vanDoorn 76] J. J. Koenderink and A. J. van Doorn. Geometry of binocular vision and a model for stereopsis. *Biological Cybernetics*, 21:29–35, 1976.

[Lee 85] D. Lee. *Contributions to Information-based Complexity, Image Understanding, and Logic Circuit Design.* PhD thesis, Department of Computer Science, Columbia University, 1985.

[Lee and Pavlidis 87] D. Lee and T. Pavlidis. One-dimensional regularization with discontinuities. In *Proceedings of the IEEE Computer Society International Conference on Computer Vision*, pages 572–577, IEEE, June 1987.

[Marr 81] D. Marr. *VISION.* Freeman, San Francisco, 1981.

[Marr and Hildreth 80] D. Marr and E. C. Hildreth. Theory of edge detection. *Proceeding Royal Society of London.*, B(207):187–217, 1980.

[Marr and Poggio 79] D. Marr and T. Poggio. A computational theory of human stereo vision. *Proceeding Royal Society of London.*, B(204):301–328, 1979.

[Mayhew and Frisby 81] J.E.W. Mayhew and J. P. Frisby. Psychological and computational studies towards a theory of human stereopsis. *Artificial Intelligence*, 17:349–385, 1981.

[Medioni and Nevatia 85] G. Medioni and R. Nevatia. Segment-based stereo matching. *Computer Vision, Graphics, and Image Processing*, 31:2–18, July 1985.

[Meinguet 83] J. Meinguet. Surface spline interpolation: basic theory and computational aspects. *Institut de Mathematique Pure dt Appliquee, Universite Catholique de Louvain*, 35, 1983.

[Moravec 79] H.P. Moravec. Visual mapping by a robot rover. In *Proceedings of the Sixth International Joint Conference on Artificial Intelligence*, pages 598–600, August 1979.

[Pollard, Mayhew and Frisby 85] S.B Pollard, J.E.W. Mayhew, and J.P. Frisby. Pmf: a stereo correspondence algorithm using a disparity gradient limit. *Perception*, 14:449–470, 1985.

[Rao, Nevatia and Medioni 87] K. Rao, R. Nevatia, and G. Medioni. Issues in shape description and an approach for working with sparse data. In *Proceegings of the AAAI Workshop on Spatial Reasoning and Multi-sensor Fusion*, pages 168–177, St. Charles. IL, October 1987.

[Terzopoulos 84] D. Terzopoulos. *Multiresolution Computation of Visible-Surface Representations.* PhD thesis, MIT, 1984.

[Wahba 84] G. Wahba. Surface fitting with scattered noisy data on euclidean d-space and on the sphere. *Rocky Mountain Journal of Mathematics*, 14(1):281–299, 1984.

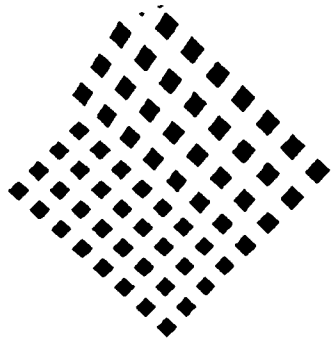

Figure 1: Synthetic image of two planes

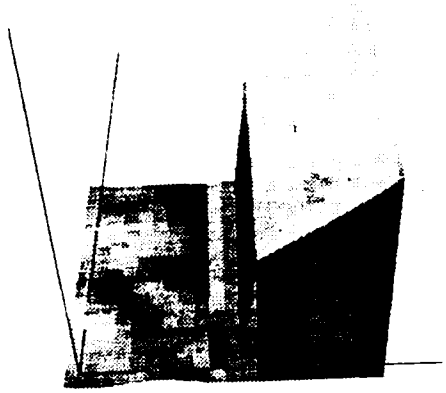

Figure 2: Reconstruction of two segmented planes

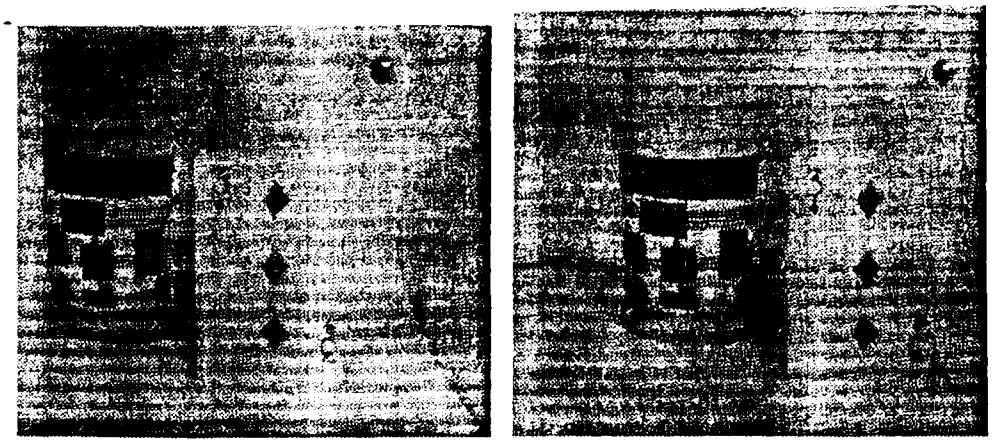

Figure 3: Left and right image of cup and poker card

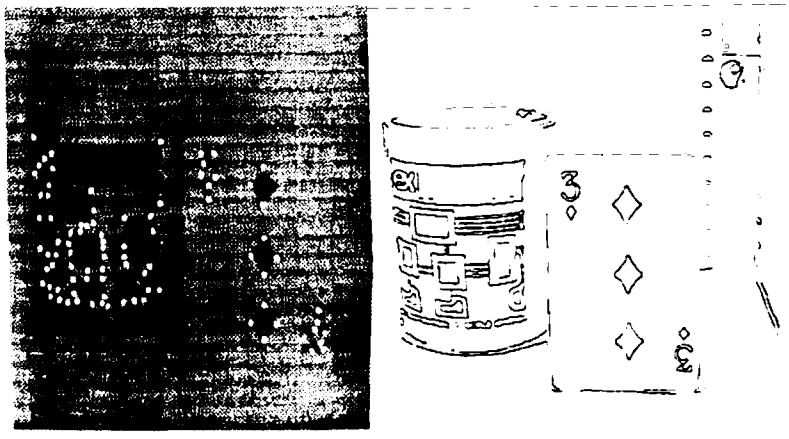

Figure 4: The left image is the output of interest operator, and the right image is the zero crossing of the Laplacian of the Gaussian.

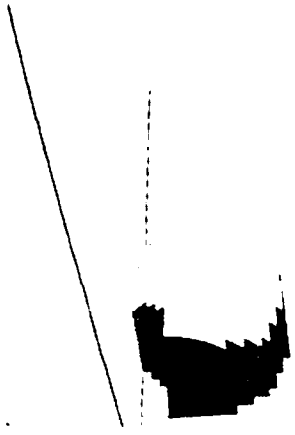

Figure 5: Reconstruction of two segmented surfaces

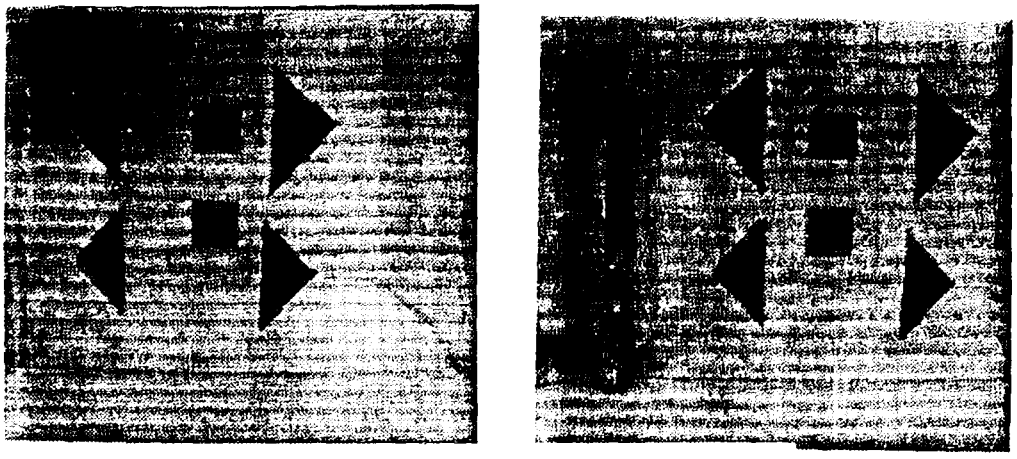

Figure 6: Left and right image of glass

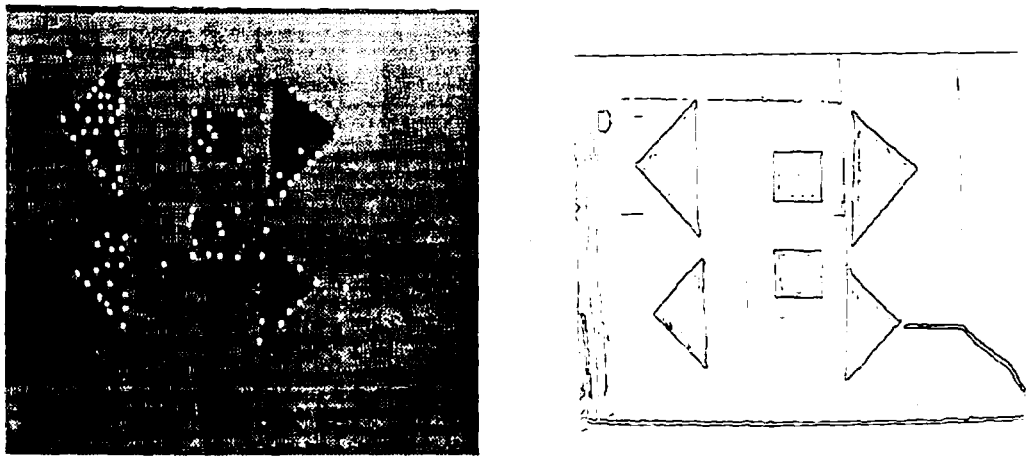

Figure 7: The left image is the output of interest operator, and the right image is the zero crossing of the Laplacian of the Gaussian.

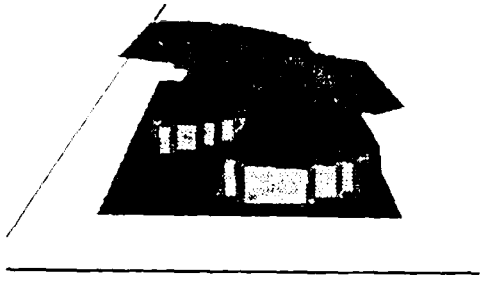

Figure 8: Reconstruction of two segmented surfaces