Computer Vision and Image Understanding, Special Issue on CAD Based Vision, March 1998

Implicit Surface Based Geometric Fusion

A. Hilton, A.J.Stoddart, J. Illingworth and T.Windeatt
Centre for Vision, Speech and Signal Processing,
University of Surrey, Guildford. GU2 5XH. U.K.
a.hilton@surrey.ac.uk

October 1996, Revised May/September 1997

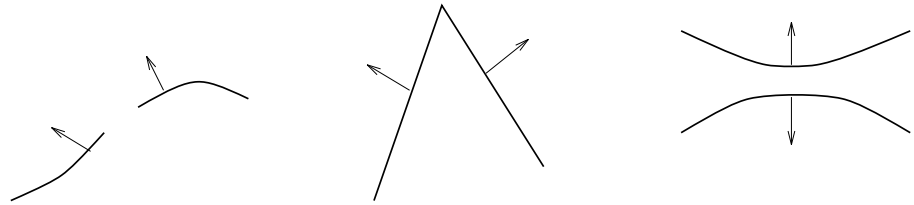
Abstract

This paper introduces a general purpose algorithm for reliable integration of sets of surface measurements into a single 3D model. The new algorithm constructs a single continuous implicit surface representation which is the zero-set of a scalar field function. An explicit object model is obtained using any implicit surface polygonisation algorithm. Object models are reconstructed from both multiple view conventional 2.5D range images and hand-held sensor range data. To our knowledge this is the first geometric fusion algorithm capable of reconstructing 3D object models from noisy hand-held sensor range data.

This approach has several important advantages over existing techniques. The implicit surface representation allows reconstruction of unknown objects of arbitrary topology and geometry. A continuous implicit surface representation enables reliable reconstruction of complex geometry. Correct integration of overlapping surface measurements in the presence of noise is achieved using geometric constraints based on measurement uncertainty. The use of measurement uncertainty ensures that the algorithm is robust to significant levels of measurement noise. Previous implicit surface based approaches use discrete representations resulting in unreliable reconstruction for regions of high curvature or thin surface sections. Direct representation of the implicit surface boundary ensures correct reconstruction of arbitrary topology object surfaces. Fusion of overlapping measurements is performed using operations in 3D space only. This avoids the local 2D projection required for many previous methods which results in limitations on the object surface geometry that is reliably reconstructed. All previous geometric fusion algorithms developed for conventional range sensor data are based on the 2.5D image structure preventing their use for hand-held sensor data. Performance evaluation of the new integration algorithm against existing techniques demonstrates improved reconstruction of complex geometry.

1 Introduction

Reverse engineering is recognised as an important goal for recovering accurate and realistic 3D models of real objects. Application domains include computer graphics [2, 10, 21, 30, 41, 28, 42, 43], machine vision [5, 7, 19, 33, 36, 39] and medical imagery [40, 26, 27, 29]. Reconstruction of an integrated 3D surface model from multiple 2.5D range images has received considerable interest [7, 20, 30, 33, 36, 41, 39]. Extraction of the correct surface topology is recognised as a fundamental step in building 3D models [20, 33, 36, 41, 39, 42, 43]. Recent research has resulted in the independent publication of several algorithms that aim to reconstruct triangulated 3D models of unknown objects



(a) Hole (b) Crease Edge (c) Thin Object

Figure 1: Examples of Complex Surface Geometry

[10, 19, 21, 33, 36, 41]. These algorithms aim to automatically reconstruct a 3D model which preserves the object surface topology estimated from the 2.5D range images. However, existing integration algorithms have a variety of inherent limitations many of which are undocumented. In practice this limits the reliability of the reconstructed surface model. Several previous range image integration algorithms require local projection of surface measurements to 2D sub-planes [33, 36, 41]. This imposes a limitation on the object surface geometry that is reliably reconstructed. Examples of complex object geometries that may cause failure of previous approaches are small holes, crease edges and thin object regions as illustrated in Figure 1. The integration algorithm introduced in this paper addresses the issue of reliable reconstruction for complex objects. A summary of the terminology used throughout this paper is presented in Table 1.

1.1 Problem Statement

The generic goal of surface reconstruction is to estimate a manifold surface, S', that approximates an unknown object surface, S, from a set of N sample points, $X = \{\vec{x}_0, ..., \vec{x}_{N-1}\}$, in R^3 3D Euclidean space, $\vec{x} = (x, y, z)$. In addition information may be available about the sampling resolution $\Delta\vec{x}$ and the measurement uncertainty, $u(\vec{x}_i)$. Correct approximation of the object surface, S, requires that the model, S', has the same topology and is an accurate approximation of the surface geometry. Given a method we want to be able to specify conditions on the original surface, S, and sample, X, that allow S' to be a correct approximation. This requires an explicit statement of the assumptions inherent in a particular approach. Performance characterisation of a particular method must establish criteria for reliable reconstruction of both surface geometry and topology given a sample X.

Surface reconstruction from multiple 2.5D range images is a more constrained problem. The measurement samples for each range image lie on a regular 2D grid, $X_k = \{\vec{x}_k(0,0), .., \vec{x}_k(i,j), ..., \vec{x}_k(n-1, m-1)\}$. The 2D grid coordinates (i, j) define the adjacency between surface measurements. This information greatly simplifies the problem of estimating the surface topology from the measurement samples. This is illustrated in Figure 4 and discussed further in section 3.1.

1.2 Overview

A new approach to geometric fusion is introduced based on the construction of an intermediate continuous implicit surface representation. Throughout this paper the term implicit surface refers to an iso-surface of a field function in 3D Euclidean space constructed from the surface measurements. The implicit surface representation is based on surface measurements as in previous work [10, 21]. This representation allows modelling of unknown objects with arbitrary topology and geometry. This is distinct from implicit shape function such as quadrics [6] which assume a fixed topology and require a shape fitting procedure.

Point	Vector displacement from origin in R^3: $\vec{x} = (x, y, z)$.
Surface	A simple manifold $S(\vec{x})$ in R^3 which does not self-intersect, is locally continuous, has arbitrary topology and is open or closed.
Geometry	The metric shape of surface $S(\vec{x})$ in R^3.
Topology	Property of the surface $S(\vec{x})$ unaffected by a continuous change of shape or size.
Continuity	A function $S(\vec{x})$ which has a smooth variation in the r^{th} derivative has G^r geometric continuity.
Boundary	A discontinuity in the G^0 positional continuity of surface $S(\vec{x})$. By definition a manifold must be open to have a boundary.
Triangular Mesh	Piece-wise planar surface $M(V, T)$ with G^0 continuity composed of a set of vertices, $V = \{\vec{v}_0, ..., v_{N_V - 1}\}$, and a set of triangular elements $T = \{T_0, ..., T_{N_T - 1}\}$. Each element T_{ijk} is composed of three vertices $(\vec{v}_i, \vec{v}_j, \vec{v}_k)$ and three edges (e_{ij}, e_{jk}, e_{ki}).
Connectivity	Defines the continuity and topology of the mesh M. If two elements are adjacent along an edge e_{ij} the mesh has G^0 continuity across the edge. If element T_{ijk} has no adjacent element along edge e_{ij} then it is a boundary edge.
Field function	Continuous scalar function, $f(\vec{x})$, which is defined for any point $\vec{x}$.
Implicit surface	Iso-surface of the field function $f(\vec{x}) = constant$ which is a simple manifold in R^3.

Table 1: Glossary of terminology

Individual 2.5D range images are triangulated to obtain a 3D mesh which approximates the local surface topology and geometry. Measurement integration is performed by combining multiple overlapping meshes into a single field function based on the signed distance to each mesh. Reliable integration of overlapping measurements for complex geometry is achieved using geometric constraints and statistical tests based on measurement uncertainty. Examples of complex geometries which cause failure of previous integration algorithms are illustrated in Figure 1. The resulting implicit representation is the zero-set of the integrated field function which is a piece-wise continuous surface. An explicit integrated representation is obtained using an implicit surface polygonisation algorithm. The principal differences of this approach to previous work is the integration of measurements using operations in 3D space only and the construction of a continuous implicit surface representation. This approach is shown to give improved performance for reconstruction of complex geometry compared to previous integration algorithms.

In section 2 we review related work on reconstruction of surface models from 3D measurements. The new geometric fusion algorithm is presented in section 3. Section 4 discusses important issues related to algorithm implementation. Limitations and complexity of the proposed approach compared to previous methods is discussed in section 5. Results of 3D model reconstruction and performance characteristics are presented in section 6. Conclusions are presented in section 7.

2 Previous Work

Reconstruction of 3D models from surface measurements is a problem that has been widely addressed. Algorithms can be categorised into three classes depending on the generality of the problem addressed:

1. Unorganised 3D points sets.

2. 3D medical images.

3. Multiple 2.5D range images.

This paper addresses the third class of problem. However, it is useful to review related work in all three classes. In the general problem of reconstruction of a manifold surface model from a set of unorganised measurements no information is available about measurement connectivity. Surface reconstruction from 3D and 2.5D images are sub-problems which can also be addressed as unorganised points. The other key distinction which can be drawn between reconstruction techniques is the use of an explicit (parametric) or implicit representation. A comprehensive review of surface representation schemes is presented by Bolle and Vemuri [6]. In general a priori assumptions on the representation restrict the class of objects that can be modelled while enabling more efficient reconstruction.

2.1 Unorganised 3D Point Sets

Graph based approaches have been used to estimate the neighbourhood relations between unorganised points. The Euclidean distance between measurements is used as the basis for establishing their adjacency on the surface. Neighbourhood relations enable approximation of surface topology and continuity. Boissonnat [5] used the 3D Delaunay triangulation to construct a neighbourhood graph for a set of N point measurements on a manifold surface. A triangulated model is then extracted as a sub-set of the faces in the Delaunay triangulation using local geometric constraints. This approach is in general prohibitively expensive for large point sets requiring $O(N^2 \log N)$ computations. Hoppe et al. [21] use the Euclidean minimum spanning tree to construct an implicit surface representation based as the zero-set of a signed field function. A local planar fit to a measurement neighbourhood is used to estimate the surface orientation. The cost of constructing the Euclidean minimum spanning tree is $O(N^2)$. Marching Cubes implicit surface polygonisation algorithm is used to reconstruct a triangulated surface model. Their use of a signed field function to represent an implicit surface is similar to the approach used in this paper. However, their representation is a discrete implicit surface with no information on continuity or boundaries. This approach is not reliable for reconstruction of complex surface geometries. Results are presented for 3D model reconstruction from both unorganised points and range images. Mencl [28] recently proposed a rule based algorithm for joining leaf nodes in the Euclidean minimum spanning tree for surface approximation. The objective is to obtain an improved estimate of the manifold surface geometry. Results are only demonstrated for simple objects. Graph based approaches have two problems in common. Firstly, the use of neighbourhood relations to estimate surface topology requires uniform dense surface sampling. Model accuracy is limited to an order of magnitude greater than the sampling resolution. Secondly, reconstruction of a neighbourhood graph is computationally expensive for a large number of points.

An alternative to graph based approaches is the use of particle systems [39]. This enables integration of scattered data into a representation of arbitrary topological type. However, the integration does not preserve surface connectivity information and does not provide a final object surface model.

2.2 3D Medical Images

Segmentation of 3D structures from medical images has been the focus of extensive research. Medical images consist of discrete samples on a regular 3D grid. Treating the images as samples of a scalar field function $f(\vec{x})$ has lead to the development of contour tracing algorithms for implicit surface extraction, $f(\vec{x}) = constant$ [3, 9, 25]. The approach presented in this paper constructs an analogous scalar field functions from sets of surface measurements. Implicit surface polygonisation algorithms enable reconstruction of closed manifold iso-surface of arbitrary topology and geometry [25].

Deformable models provide an alternative approach to medical image segmentation which have been used extensively [27]. Models are based on an explicit parametric representation such as superquadrics [29] or finite elements [26]. Deformable surfaces which model physical properties are particularly suited to the extraction of temporal shape variation. The principal limitation of these representations for reconstruction of object models is the a priori assumption of known object topology. Recent research on volumetric segmentation has lead to the development of deformable models with adaptive topology [11, 40]. This addresses the problem of reconstructing an unknown object model from an implicit surface. These techniques are of considerable interest for model reconstruction from the implicit surface representation of multiple range images introduced in this paper.

2.3 2.5D Range Images

Integration of multiple 2.5D range images is required for 3D model reconstruction. Due to self occlusion individual range images only provide a partial model of the object surface. Techniques for 3D model reconstruction from range images can be separated into two classes according to the surface representation. Firstly, deformable models with an explicit parametric representation [8, 14]. These techniques are analogous to the deformable surface models used for segmentation of 3D medical image. The surface measurements are treated as an unstructured 3D point cloud and an inflation force is applied to the deformable model to fit the data. These models are of fixed topological genus and do not allow reconstruction of unknown objects.

The second class of algorithm reconstructs a triangulated mesh representation of the object surface [5, 10, 19, 31, 33, 36, 41]. The use of a triangulated model allows the representation of surfaces of arbitrary geometry and topology. However, limitations inherent in the integration algorithms restrict the surface geometry that is reliably reconstructed. The triangulation based integration algorithms all share a common first stage. Individual range images are converted to a triangular mesh which approximates the local surface geometry and topology. Step discontinuity constrained triangulation is used to estimate the local surface continuity as described in section 3.1. Boissonnat [5] and Rutishauser et al. [33] use a graph-based approach to determine correspondences between overlapping meshes. Retriangulation of two overlapping meshes to form a single mesh is then performed using local 2D constraints on triangle shape. This approach may fail to correctly reconstruct the surface in regions of high curvature and thin object sections. Soucy et al. [36] integrate triangulated meshes using canonic subsets of the Venn diagram. The canonic subsets each represent the overlap between a subset of the 2.5D range images and are associated with a 2D viewpoint reference frame. The 2D reference frames are used to eliminate redundant data and merge intersecting regions. Turk et al. [41] integrate overlapping triangulated meshes using a 'zippering' approach. Overlapping meshes are eroded and the boundary correspondences found by operations in 3D space. A local 2D constrained triangulation is then used to join the overlapping mesh boundaries. This approach may fail for complex geometries if incorrect boundary correspondences are found. This can occur in regions of high surface curvature. Each of these algorithms requires the 3D surface measurements to be projected to a 2D plane. For correct surface reconstruction it is required that the 3D to 2D projection is injective (i.e. it respects the spatial ordering of the surface measurements). This requirement results in an inherent limitation on the maximum surface curvature for correct reconstruction.

Recently, several authors including ourselves have introduced implicit surface based geometric fusion of multiple overlapping triangulated meshes [10, 19]. A scalar field function is evaluated as the signed-distance from any point in 3D space to the nearest point on the measured surface triangulation. The object surface is represented by an implicit surface which is the zero-set of the field function. Integration based on triangulated meshes uses estimates of the local surface continuity to avoid

limitations of previous implicit surface based techniques for fusion of unstructured 3D point sets [21]. Overlapping meshes are combined using operations in 3D space only. This avoids limitations on object surface geometry of previous mesh based integration algorithms which require local 2D projection [33, 36, 41]. Curless et al. [10] introduced a discrete implicit surface representation. This approach uses spatial data structures for computationally efficient geometric fusion. However, the discrete implicit surface representation does not enable reliable geometric fusion in regions of high curvature or thin surface sections. A hole filling algorithm was introduced to eliminate errors in the reconstructed 3D model. This approach is suitable for applications in computer graphics where geometric accuracy is not essential. Reliable reconstruction is essential for accurate reverse engineering of objects for CAD and manufacture. Hilton et al. [19] introduced a continuous implicit surface representation for reliable integration using local geometric constraints. Explicit representation of surface continuity and boundary information enables reliable geometric fusion for complex object geometries. This approach forms the basis of the work presented in this paper. The use of a discrete representation for efficient implementation is discussed in section 4.3. A comparison of results for implicit surface based geometric fusion using discrete and continuous representations is presented in section 6.4.

3 New Integration Algorithm

In this section we present a new range image integration algorithm based on an intermediate implicit surface representation. A 'continuous implicit surface' is proposed that represents both the surface topology and geometry. The new technique has several advantages over previous approaches:

1. Geometric fusion of surface measurements based on operations in 3D space only. This avoids limitations on object geometry associated with previous techniques [33, 36, 41].

2. Geometric constraints are used to ensure correct reconstruction of complex geometry.

3. Statistical tests based on measurement uncertainty are used to prevent combination of measurements from different surface regions in the presence of noise.

4. A continuous implicit surface is used for accurate representation of surface geometry and topology. Previous discrete implicit surface representations fail for complex object geometry [10, 21].

An overview of the implicit surface based geometric fusion algorithm is presented in Figure 2. The input to the reconstruction is a set of range images taken from multiple viewpoints. Triangulation of the range images to obtain a set of meshes that approximate the surface topology is discussed in section 3.1. The triangulated meshes are then fused into a single continuous implicit surface representation, $f(\vec{x}) = 0$, which is the zero-set of a scalar field function, $f(\vec{x})$. Construction of an implicit representation for a single mesh and multiple overlapping meshes is presented in sections 3.2 and 3.3. The use of measurement uncertainty as a constraint in implicit surface based geometric fusion is discussed in section 3.4. The output of the reconstruction algorithm is a 3D triangulated surface model. Section 3.5 discusses algorithms for polygonisation of the fused implicit surface representation. The input and output of the implicit surface based geometric fusion algorithm are both 3D triangulated meshes. Therefore, it is possible to dynamically integrate new surface measurements into an existing partial object model.

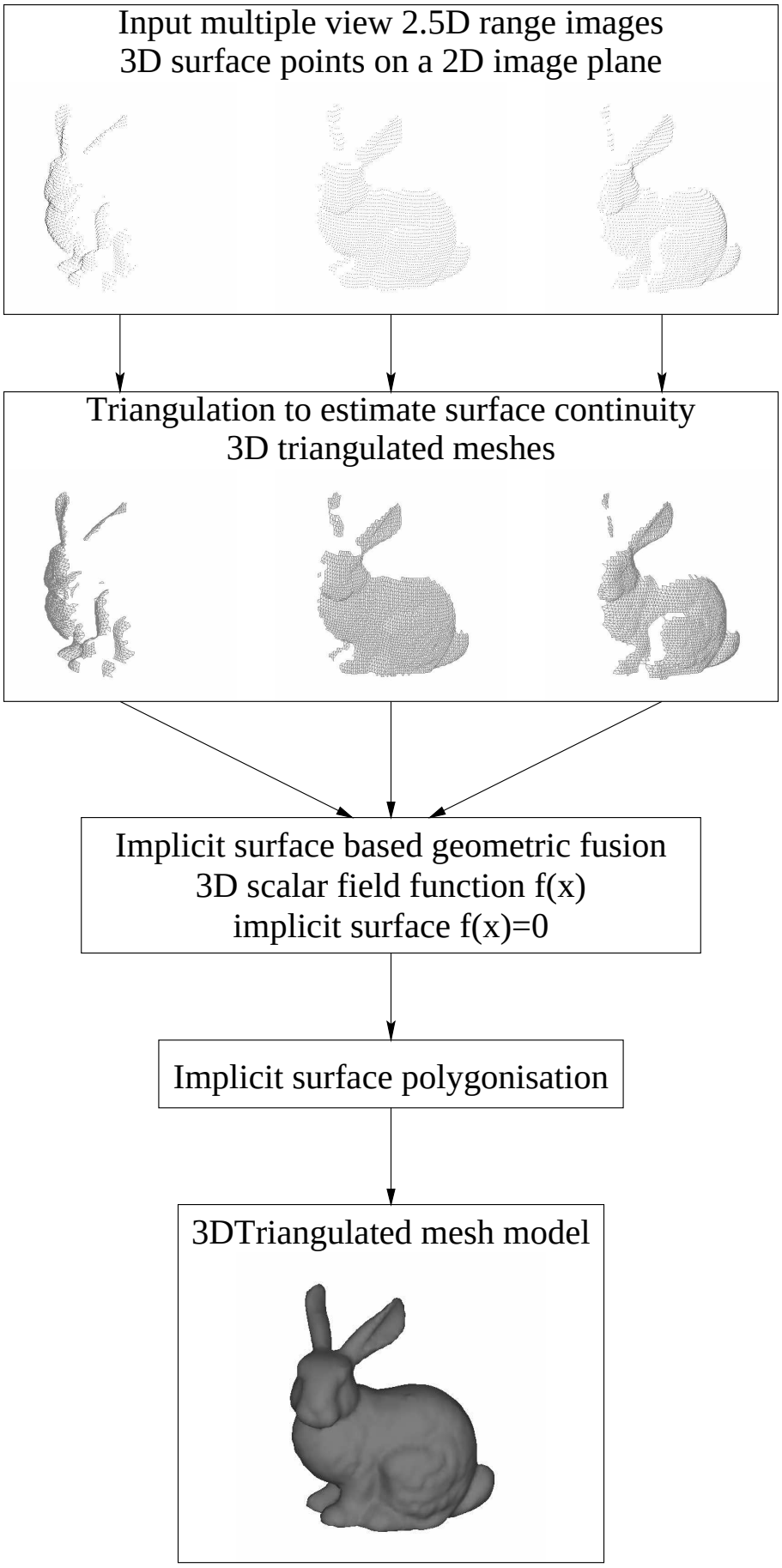


Figure 2: Overview of 3D model reconstruction from multiple range images

Input the range image R_k with $N_1 \times N_2$ pixels

For each pixel $\vec{x}(n,m)$ in the range image R_k where $n = 0...N_1 - 2$ and $m = 0...N_2 - 2$:

1. For each pair of adjacent pixels in the 4-neighbourhood $\vec{x}(n+r, m+s)$ and $\vec{x}(n+p, m+q)$ where $r, p, s, q = 0, 1$ and $r \neq p$ if $s = q$:
   (a) Evaluate the Euclidean distance between each pair of neighbours: $d_{rspq} = |\vec{x}(i+r, j+s) - \vec{x}(i+p, j+q)|$.
   (b) If $d_{rspq} \leq t_d$ the adjacent pixels can be connected.
2. For all pairs of adjacent pixels that can be connected evaluate the set of 0,1 or 2 triangles that give the smallest sum of distances: $min \sum d_{rspq}$

Output the step discontinuity constrained triangulated mesh, M_k.

Figure 3: Algorithm for Step Discontinuity Constrained Triangulation

3.1 Step Discontinuity Constrained Triangulation

Occlusion in the 2.5D range image results in step discontinuities between adjacent surface measurements. The objective of constrained triangulation is to estimate the surface continuity from the range image. Step discontinuity constrained triangulation estimates the local surface continuity from the the distance between adjacent measurements. A continuous surface between adjacent measurements is assumed if their distance is less than a constant threshold. Several previous range image integration algorithms [35, 41, 33] have used the 2D image structure to construct a step discontinuity constrained triangulated mesh. An algorithm for estimating step discontinuities based on the distance between adjacent measurements is presented in Figure 3. This procedure takes a 2.5D range image, R_k, as input and outputs a 3D triangulated mesh, M_k, according to the distance threshold, t_d.The resulting mesh 'connectivity' approximates the continuity of the measured object surface for a single range image. This is illustrated in Figure 4.

Threshold setting should take into account two factors: measurement resolution, $\Delta\vec{x}$, and the measurement uncertainty, $u(\vec{x})$. The geometric fusion algorithm presented in this paper reconstructs a locally continuous surface from a set of meshes if any of the meshes are locally connected. Correct reconstruction requires that measurements are only connected if there is a high probability of them originating from the same surface. As in previous work it is assumed that a sufficiently small threshold exists that allows the detection of discontinuities. This implies that the measurement error must be an order of magnitude less than the measurement resolution, $u(\vec{x}) \ll \Delta\vec{x}$. A constant distance threshold, t_d, can be defined from the sampling resolution, $t_d = n\Delta\vec{x}$ [35, 41]. An equivalent angle threshold, t_θ, can be defined in terms of the angle between the range image optical axis and the triangle normal: $t_\theta \approx cos^{-1}(\Delta x/t_d) = cos^{-1}(1/n)$ [33]. Throughout this work a constant distance threshold has been used ($n = 3$). Geometric limitations resulting from this assumption are discussed in section 5.2.

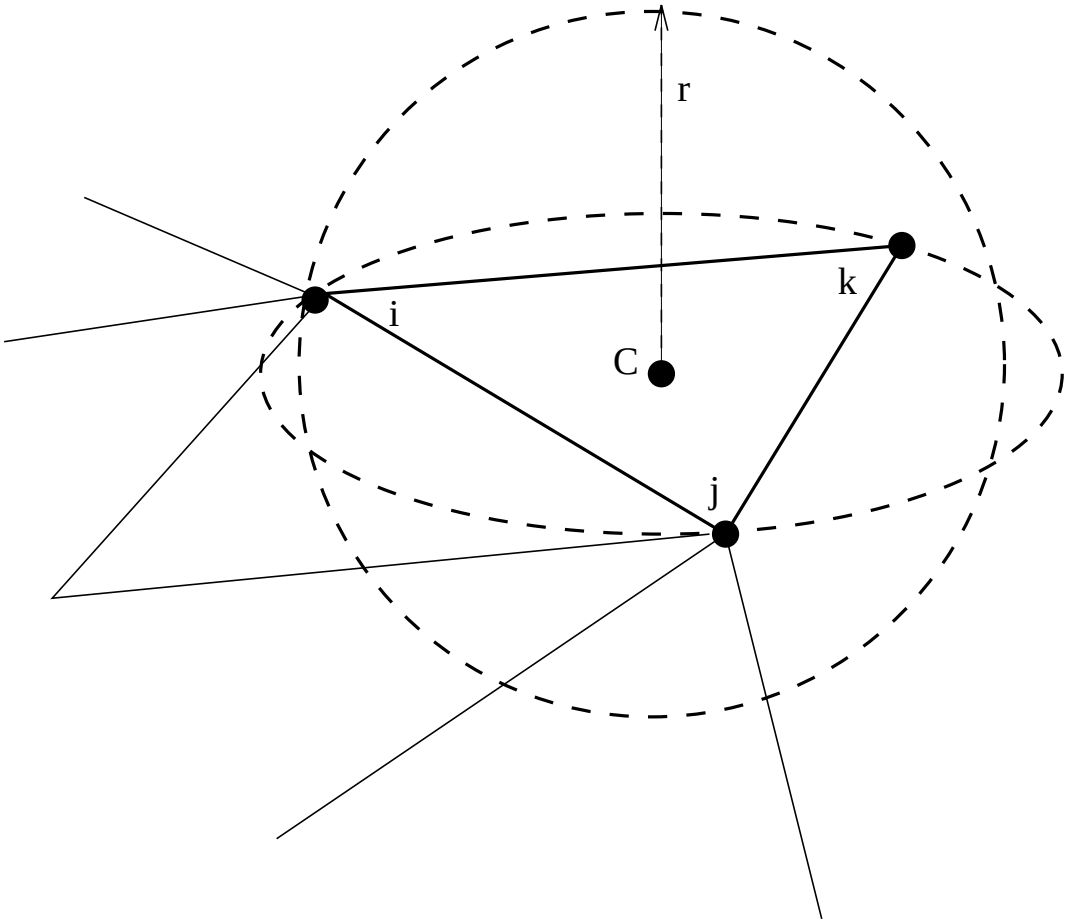


Figure 4: Step discontinuity constrained triangulation of a 2.5D Range Image Measurement

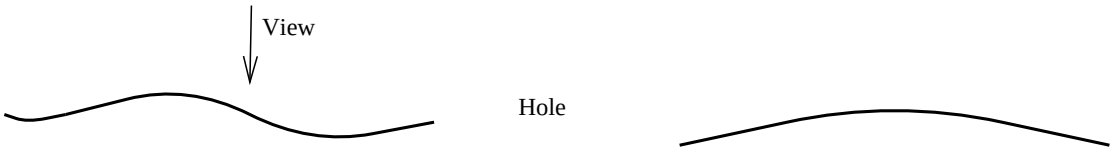
3.2 Implicit Surface Construction for a Single Range Image

This section introduces a procedure for representing a single range image triangulation, M_k, as an implicit surface. The implicit surface is defined as the zero-set of a scalar field function $f(\vec{x}) = 0$. The aim of representing a mesh, M_k, as an implicit surface is to construct a smooth field function, $f(\vec{x})$, such that the zero-set approximates the mesh as closely as possible. For a single mesh, M_k, there is an exact correspondence between the implicit surface and the explicit mesh representation. The implicit representation enables integration of multiple meshes as discussed in section 3.3. The implicit surface representation introduced here is based on surface measurements as in previous work [10, 21]. This allows representation of unknown objects with arbitrary geometry and topology. This representation is distinct from implicit parametric shape functions such as quadrics which assume a known topology and require a fitting procedure [6].

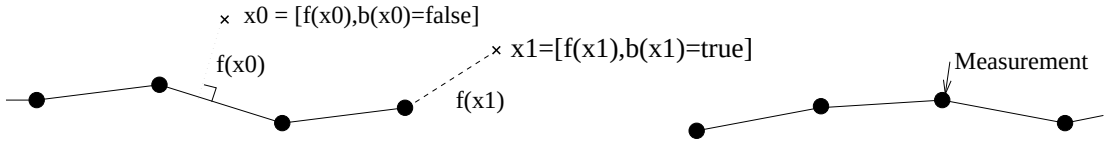
Meshes, M_k, are initially constructed from each range image using a step discontinuity constrained triangulation as described in section 3.1. For a single mesh M_k the field function, $f_k(\vec{x})$, is constructed as the signed distance to the nearest point on the mesh, $\vec{p}$. A binary function, $b_k(\vec{x}) = [\text{'}true\text{'},\text{'}false\text{'}]$, is used to explicitly label field function values, $f_k(\vec{x})$, with nearest points on the mesh boundary. If the nearest point, $\vec{p}$, is not on the mesh boundary, $b_k(\vec{x}) = \text{'}false\text{'}$, and the signed distance is the dot product of the vector to the nearest point with the surface normal, $\vec{n}_p$, at the nearest point: $f_k(\vec{x}) = (\vec{x} - \vec{p}).\vec{n}_p$. Alternatively, if the nearest point, $\vec{p_b}$, is on the mesh boundary, $b_k(\vec{x}) = \text{'}true\text{'}$, the signed distance function is evaluated as the sign of the dot product of the vector to the nearest points, $(\vec{x} - \vec{p_b})$, with the nearest point normal, $\vec{n}_{p_b}$, multiplied by the Euclidean distance: $f_k(\vec{x}) = sign[(\vec{x} - \vec{p_b}).\vec{n}_{p_b}] \times |(\vec{x} - \vec{p_b})|$. Surface normals are estimated from the mesh orientation for points internal to the triangular elements and by an average of adjacent element orientation at edges and vertices. The bounded implicit surface function, $[f_k(\vec{x}) = 0, b_k(\vec{x}) = \text{'}false\text{'}]$ for a single mesh, M_k, is thus a geometrically G^0 continuous function with the same geometry and topology as the mesh. Note that the mesh, M_k, is a surface with G^0 continuity in position, the nearest point, $\vec{p}$, can be any point on this surface (not just the vertices). Therefore the resulting implicit surface $f_k(\vec{x}) = 0$ has G^0 continuity and corresponds exactly in both geometry and topology to the original mesh M_k.

The implicit surface reconstruction from an object surface containing a hole is illustrated for a 2D cross section in Figure 5. Evaluation of the field function, $f_k(\vec{x})$, and boundary function, $b_k(\vec{x})$, from the nearest point on the mesh is illustrated in Figure 5(b). The resulting boundary, $b_k(\vec{x})$, and field, $f_k(\vec{x})$, functions are illustrated in Figure 5(c) and (d) respectively. The partition between points, $\vec{x}$, with nearest mesh points inside, $b_k(\vec{x}) = \text{'}false\text{'}$, and on the boundary of the mesh,$b_k(\vec{x}) = \text{'}true\text{'}$, is a surface normal to the mesh boundary, see Figure 5(c). In regions of space where the nearest point is inside the mesh, $b_k(\vec{x}) = \text{'}false\text{'}$, the field function is continuous. Conversely in regions where the nearest point is on the boundary, $b_k(\vec{x}) = \text{'}true\text{'}$, the field function $f_k(\vec{x})$, is discontinuous. These two cases are illustrated in Figure 5(e) which shows the field function for a cross section through the mesh. Explicit labelling of boundary regions enables correct representation of the mesh topology. In non-boundary regions the field function is continuous with zero-set, $f_k(\vec{x}) = 0$, corresponding to the geometry of the mesh, M_k.

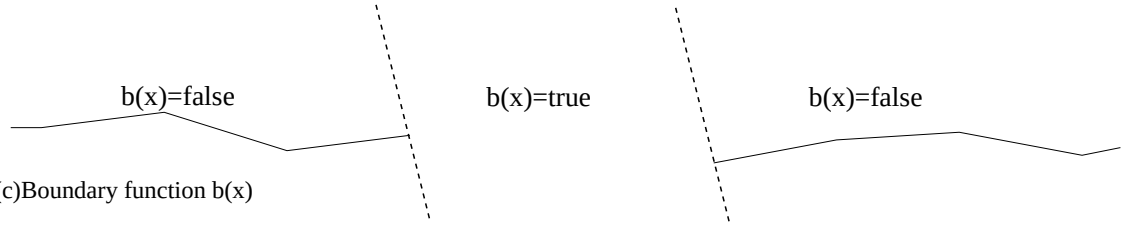
An example of the volumetric field function for range data of a real object is shown in Figure 6(a). The right hand side of the figure shows a rendered version of the triangulated mesh, M_a, with a plane cutting through the volume. The left hand side of the figure shows a cross-section on the cutting plane through the volumetric field function for all point where $|f(\vec{x})|$ is less than a constant distance threshold d_{max}.The colour indicates the magnitude of the field function red corresponds to $f(\vec{x}) = -dmax$, blue $f(\vec{x}) = dmax$ and the yellow/green boundary $f(\vec{x}) = 0$. Notice the continuity of the implicit surface representation $f(\vec{x}) = 0$.



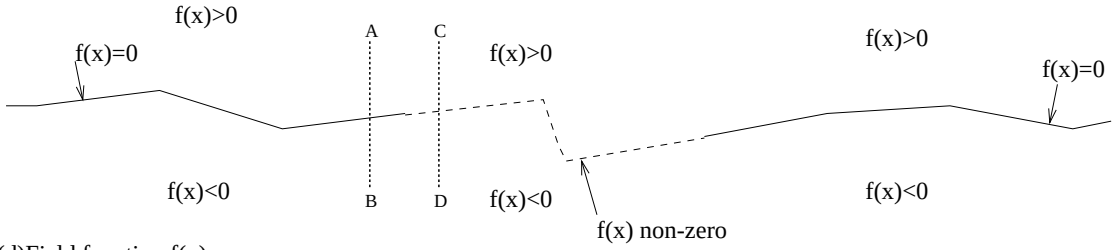
(a) Cross-section through the object surface S(x)



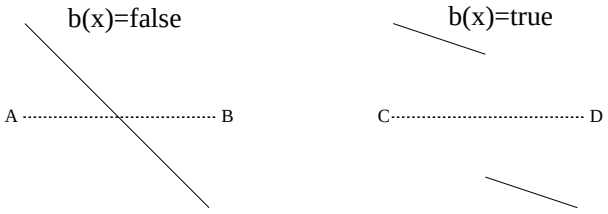
(b)Step discontinuity constrained triangulated mesh M



(c)Boundary function b(x)



(d)Field function f(x)



(e)Field function f(x) for a cross-section through the mesh M

Figure 5: Implicit surface function $[f_k(x), b_k(x)]$ for a single mesh M_k.

3.3 Implicit Surface Construction for Multiple Range Images

In this section we introduce an algorithm for integrating the field functions, $[f_k(\vec{x}), b_k(\vec{x})]$, from multiple range images triangulations, M_k for $k = 0...m-1$, into a single implicit surface representation, $[f(\vec{x}) = 0, b(\vec{x}) = \text{'}false\text{'}]$. A field function $f_k(\vec{x})$ is evaluated for each mesh, M_k, as described in section 3.2. The individual field functions are then combined using a simple set of rules based on the local surface geometry. The algorithm for mesh integration is presented in Figure 7. The objective is to combine field functions corresponding to measurements of the same surface region without combining field functions for different surface regions. This is necessary for correct reconstruction of complex object geometries such as regions of high curvature and thin object sections.

The rule based algorithm enables the integration of field functions from multiple overlapping meshes to define a continuous implicit surface, $f(\vec{x}) = 0$, in all regions where one or more mesh is continuous. Geometric constraints determine if overlapping meshes which are in close spatial proximity correspond to the same or different surface regions. Constraints are based on surface proximity, surface orientation and measurement uncertainty. The algorithm accounts for the case of multiple overlapping meshes with different orientations.

Given a set of triangulated meshes, M_k $k = 0...m-1$, the algorithm presented in Figure 7 evaluates an integrated field function, $[f(\vec{x}), b(\vec{x})]$ for an arbitrary point $\vec{x}$. Steps (1) and (2) evaluate the field function for each mesh, M_k, and determine the nearest boundary and non-boundary points. Step (3) combines the boundary information from individual field functions to generate an integrated boundary representation, $[f(\vec{x}), b(\vec{x}) = \text{'}true\text{'}]$. Steps (4—6) use surface orientation to eliminate overlapping field functions which do not correspond to the same surface region. Step (7) performs a statistical test based on measurement uncertainty to determine if the remaining field functions correspond to the same surface region. Finally, step (8) combines overlapping field functions corresponding to the same surface region to obtain an integrated field function, $[f(\vec{x}), b(\vec{x}) = \text{'}false\text{'}]$. Overlapping field functions are combined using a weighted average based on estimates of measurement confidence or blending functions [44].

The resulting implicit surface representation is the zero-set of the integrated field function, $[f(\vec{x}) = 0, b(\vec{x}) = \text{'}false\text{'}]$. This is a piecewise continuous representation of both surface geometry and topology. Discontinuities only occur between overlapping and non-overlapping regions. In practice the magnitude of discontinuities is less than the measurement error which is an order of magnitude smaller than the sampling resolution, Δx. Previous discrete implicit surface representations [10, 21] have discontinuities between each sample point and therefore do not represent surface topology. The surface geometry is piecewise planar in non-overlapping regions and may be non-planar in overlapping regions due to the weighted average.

Figure 8 illustrates the resulting implicit surface representation. Figure 8 (a) shows the case of a single mesh with holes. The implicit surface, $[f(\vec{x}) = 0, b(\vec{x}) = \text{'}false\text{'}]$, exactly matches the input mesh. Figure 8 (b) and (c) show the cases of two overlapping meshes with the same orientation and with opposite orientation. Overlapping meshes with the same orientation are combined to form a single implicit surface. Overlapping meshes with opposite orientation are not combined and form two overlapping implicit surfaces. Figure 8(d) illustrates the case of three overlapping meshes where the middle one has opposite orientation. In this case the meshes are not combined resulting in three overlapping implicit surfaces.

An example of the fused volumetric field function for range data of a real object is shown in Figure 6(c). Figures 6 (a) and (b) are the field functions for the two input triangulated meshes shown on the right hand side. Field function are shown for all point where $|f(\vec{x})|$ is less than a constant distance threshold d_{max}. The left hand side of Figure 6(c) shows a cross-section through the integrated field

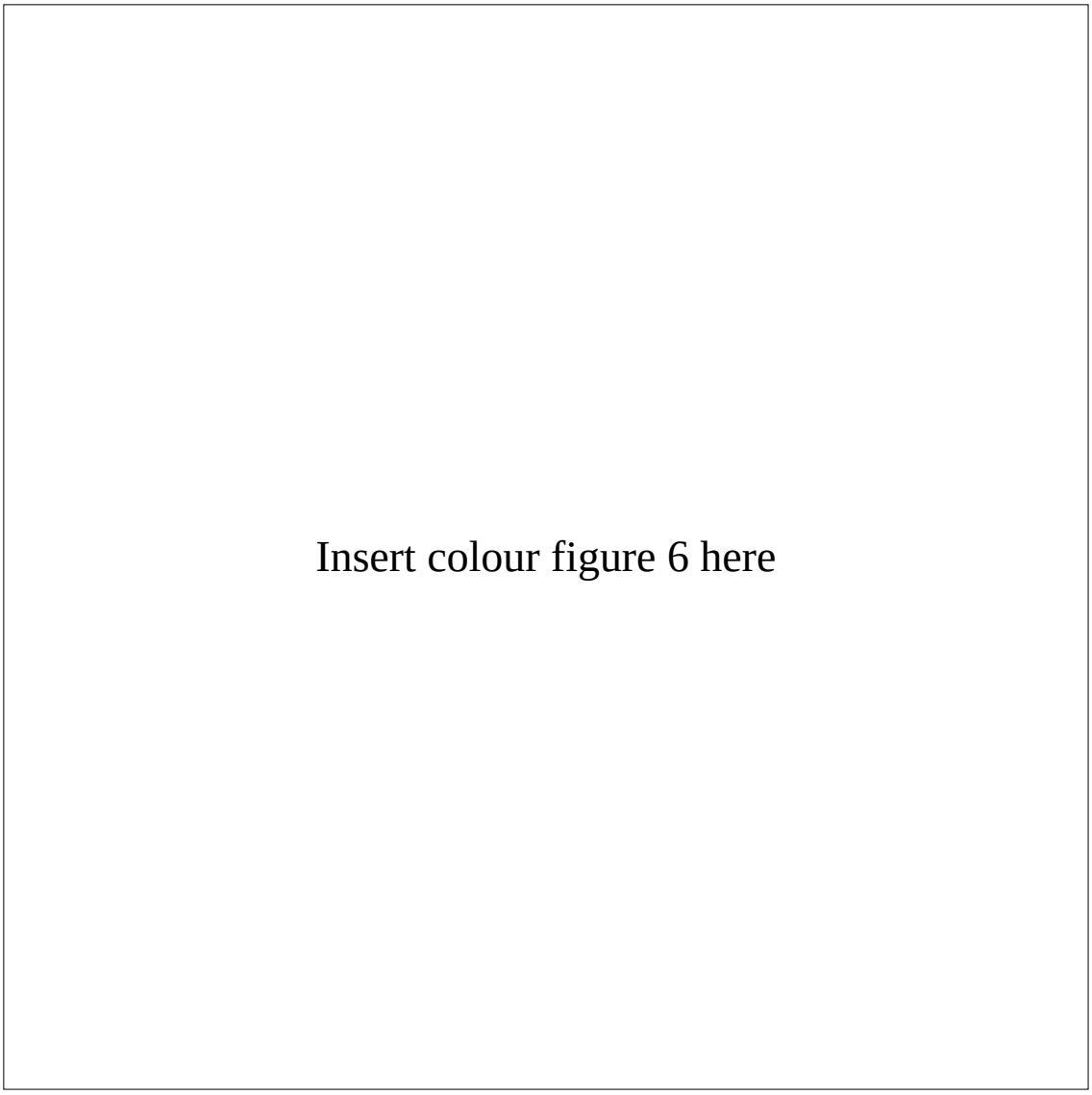


Figure 6: Example of implicit surface based geometric fusion for real data. Right: Rendered mesh and cutting plane. Left: Cross-section through field function on cutting plane for $|f(\vec{x})| < d_{max}$.The colour indicates the magnitude of the field function red corresponds to $f(\vec{x}) = -dmax$, blue $f(\vec{x}) = dmax$ and the yellow/green boundary $f(\vec{x}) = 0$.

function. Notice that in the region where the two meshes overlap the field function is continuous. The right hand side of Figure 6(c) shows the resulting fused model from the two input meshes.

3.4 Geometric Fusion based on Measurement Uncertainty

The use of measurement uncertainty in the integration process is necessary to ensure accurate reconstruction of complex geometry. Knowledge of measurement uncertainty is used for two purposes in the integration algorithm. Firstly, to test if field function values, $f_i(\vec{x})$ and $f_j(\vec{x})$ are statistically likely to correspond to measurements of the same surface region in step 7 of Figure 7. Secondly, for the integration of multiple measurements from the same surface region to obtain an improved estimate of the surface geometry in step 8 of Figure 7.

Measurement uncertainty for a triangulation based range sensor can be modelled as an ellipsoid with the principal axis aligned with the line of sight of the light source [33, 41]. The noise variance, σ_N^2, for a particular sensor can be estimated by the distribution of measurement errors on a planar surface. Given a range image we can approximate the average distribution of measurement error by a zero mean Gaussian distribution, $N[0, \sigma_N^2]$. The validity of assuming a Gaussian distribution of measurement error is verified in [17]. The measurement noise distribution can be used to estimate the variance, $\sigma_i^2(\vec{x})$, of individual range image measurements based on the surface normal. Measurement uncertainty, $u_i(\vec{x})$, is approximately proportional to the magnitude of the dot product of the object surface unit normal, $\vec{n}_i(\vec{x})$, with a unit vector in the range image viewpoint direction $\vec{r}_i$: $u_i(\vec{x}) = \vec{r}_i \cdot \vec{n}_i(\vec{x})$ [33, 36, 41]. The measurement variance, $\sigma_i^2(\vec{x})$, provides an error metric for the i^{th} measurement point $\vec{x}_i$. This is obtained by scaling the measurement uncertainty, $u_i(\vec{x})$, by the variance of the sensor noise, σ_N^2: $\sigma_i^2(\vec{x}) = u_i(\vec{x})\sigma_N^2$. Linear interpolation of the measurement uncertainty can be used to estimate the uncertainty for all points on a range image triangulation [33]. For multiple overlapping range images it is assumed that the registration error is an order of magnitude less than the measurement error as reported in [12, 41].

Testing of measurement overlap is required to prevent the combination of measurements from different surface regions. The hypothesis that two (or more) measurements correspond to the same object surface can be tested using statistical decision theory. If we assume that two samples $f_i(\vec{x})$ and $f_j(\vec{x})$ are approximately normally distributed, $N[m_i(\vec{x}), \sigma_i^2(\vec{x})]$ and $N[m_j(\vec{x}), \sigma_j^2(\vec{x})]$, and that they have the same orientation, $\vec{n}_i(\vec{x}) \cdot \vec{n}_j(\vec{x}) > 0$, then the variance of the joint sample distribution is given by: $\sigma_{ij}^2 = \sigma_i^2 + \sigma_j^2$. The confidence statistic for the hypothesis that two samples $f_i(\vec{x})$ and $f_j(\vec{x})$ correspond to measurements of the same point on the surface, $m_i(\vec{x}) = m_j(\vec{x})$, is given by: $z = |f_i - f_j|/\sigma_{ij}$ [38]. The significance of this test statistic is defined by the area under the standard normal curve. Thus we can test the overlap hypothesis to the required confidence level. Throughout this work a confidence level of 95% is used corresponding to $z = \pm 1.96$.

Accurate surface geometry can be estimated by combining measurements from the same surface region according to their relative uncertainty [33, 36, 41]. Given a set of overlapping field function values corresponding to the same region of the object surface, $f_k(\vec{x})$ for $k = 0...m-1$, we use a weighted average to estimate an integrated field function value, $f(\vec{x})$. The weights are based on the measurement confidence $w_k(\vec{x}) = u_k(\vec{x}) / \sum_{k=0}^{n} u_k(\vec{x})$ where $\sum_{k=0}^{n} w_k(\vec{x}) = 1$. The confidence of the resulting weighted average measurement is given by $u(\vec{x}) = \sum_{k=0}^{n} u_k(\vec{x})$.

3.5 Implicit Surface Polygonisation

An explicit triangulated mesh representation can be obtained using any implicit surface polygonisation algorithm. This section discusses polygonisation algorithms in the context of implicit surface

Input multiple meshes, M_k for $k = 0...m-1$.

To evaluate the integrated field function $[f(\vec{x}), b(\vec{x})]$ at any point $\vec{x}$:

1. Evaluate the signed field function,$f_k(\vec{x})$, and boundary function, $b_k(\vec{x})$, from the nearest point on each mesh M_k for $k = 0...m-1$.

2. Find the nearest boundary and non-boundary mesh points from the set $f_k(\vec{x})$:
   $f_{min_{bound}}(\vec{x}) = min|f_k(\vec{x})|$ where $b_k(\vec{x}) =$'*true*'
   $f_{min}(\vec{x}) = min|f_k(\vec{x})|$ where $b_k(\vec{x}) =$'*false*'

3. If a non-boundary point, $f_{min}(\vec{x})$, does not exist or $f_{min_{bound}}(\vec{x}) < f_{min}$ and $|f_{min}(\vec{x}) - f_{min_{bound}}(\vec{x})| < t_d$ then return the nearest boundary point, $b_k(\vec{x}) =$'*true*': $f(\vec{x}) = f_{min_{bound}}(\vec{x})$ and $b(\vec{x}) =$'*true*'.
   Terminate the integrated field function evaluation.

4. Else find all non-boundary points, $b_k(\vec{x}) =$'*false*', with the same orientation as the nearest point, $f_{min}(\vec{x})$: $F = \{f_{same_i}(\vec{x})\}$ $i = 0...N_{same}$, where $n_{min}.n_{same_i} > 0$

5. Find the nearest non-boundary point with opposite orientation, $f_{opposite}(\vec{x})$, where $n_{min}.n_{opposite} < 0$.

6. Eliminate all points in F, where $f_{same_i}(\vec{x}) > f_{opposite}(\vec{x})$.

7. Eliminate all point in F that do not correspond to the same surface as $f_{min}(\vec{x})$ based on the distribution of measurement error.

8. Evaluate the nearest point as a weighted average of all points in F: $f(\vec{x}) = \sum_k w_k f_k(\vec{x})$ where $\sum w_k = 1$ and $b(\vec{x}) =$'*false*'.
   Terminate the integrated field function evaluation.

Output the implicit surface for all points $\vec{x}$ where $[f(\vec{x}) = 0, b(\vec{x}) =$'*false*'$]$

Figure 7: Algorithm for integration of multiple meshes into a single implicit representation

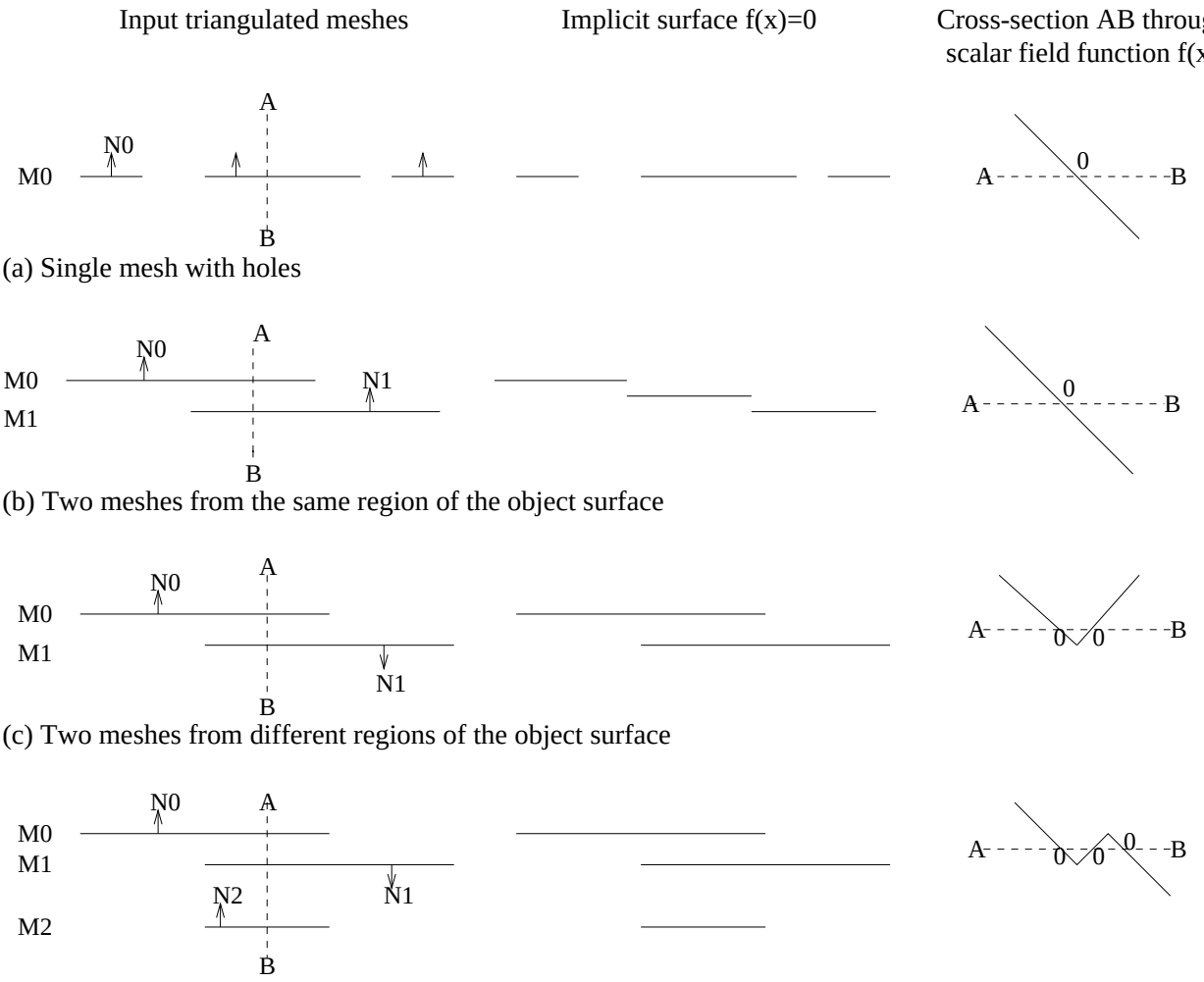


Figure 8: Construction of a single continuous implicit surface function $f(\vec{x}) = 0$ from multiple overlapping triangulated meshes, M_k

based geometric fusion of measured data. Volume based implicit surface polygonisation algorithms such as the 'Marching Cubes' [4, 25, 44] are widely used for visualisation in computer graphics and medical imaging. A polygonal representation is constructed by determining the intersection of the implicit surface with volumetric cells (cubes or tetrahedra). However, the surface to volume intersection results in a highly non-uniform distribution of triangle shape and size. Mesh optimisation is required to obtain an efficient representation [13, 22, 23, 34, 37].

Recently an alternative *surface based* approach to implicit surface polygonisation has been proposed [15, 18] called 'Marching Triangles'. The surface based approach allows mesh vertices to be placed according to the local surface geometry. Resulting in an efficient implicit surface triangulation with uniform triangle shape and size. The Marching Triangles approach has two additional advantages over volume-based approaches for the application of implicit surface based geometric fusion. Firstly, it allows accurate polygonisation of bounded manifold surfaces. Volume-based polygonisation algorithms were designed to polygonise closed manifold surface and do not accurately polygonise bounded manifolds [21]. Secondly, Marching Triangles allows dynamic integration of new surface measurements into an existing mesh. This enables iterative refinement of a partial object model through the sequential addition of new measurements. A complete description of the Marching Triangles algorithm is beyond the scope of this paper. A full description is presented in [15]. Throughout this work the Marching Triangles implicit surface polygonisation algorithm has been used to obtain a triangulated mesh model for visualisation.

4 Implementation

Computationally efficient implementation of the implicit surface based geometric fusion algorithm requires efficient evaluation of the field function $f(\vec{x})$ and boundary function $b(\vec{x})$. The key problem is efficient evaluation of the nearest point, $\vec{p}$, on a mesh , M_k, to an arbitrary point $\vec{x}$. This section presents three techniques for nearest point computation and discusses their relative performance: Vertex based; Surface based and Volume based. Vertex based techniques have been widely used in previous range image integration [36, 41] and surface registration [1, 8, 12, 41] algorithms. A volume based approach was recently introduced in [10] which enables efficient nearest point evaluation. However, neither vertex or volume based approaches guarantee correct nearest point evaluation. This may result in incorrect reconstruction of surface geometry. A surface based approach is introduced in this paper which is guaranteed to give the correct nearest point for an arbitrary triangulated mesh. This approach is computationally expensive compared to the volume based approach.

4.1 Vertex Based Nearest Point Evaluation

The nearest point on a mesh M_k with vertices $V = \{\vec{v}_0 ... \vec{v}_{N-1}\}$ is implemented by first evaluating the nearest vertex, $\vec{v}_i$, to search point $\vec{x}$. The nearest point, $\vec{p}$, is then estimated for the mesh elements adjacent to the vertex, $\vec{v}_i$. Direct evaluation of a single nearest point requires $0(N)$ operations. This can be reduced to constant time using an efficient spatial data structure [21].The 3D space containing the mesh vertices, V, is divided into voxels of size, Δc. The spatial data structure stores a list of the vertices inside each voxel. The nearest vertex search to an arbitrary point $\vec{x}$ then proceeds by evaluating the distance to all vertices in voxels $\vec{x} \pm n\Delta c$. The computational cost is proportional to the number of voxels searched and the number of vertices per voxel. Efficient storage of the voxel data structure is achieved using a hash table of size $O(N)$.

This approach is guaranteed to correctly evaluate the nearest vertex $\vec{v}_i$ to an arbitrary point in 3D

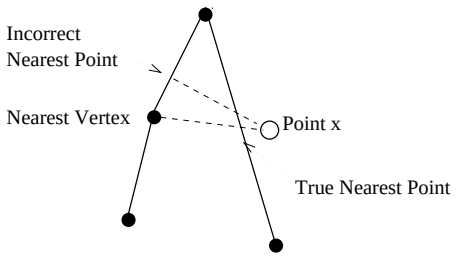


Figure 9: Failure of vertex based nearest point evaluation

space $\vec{x}$. However, this does not guaranteed correct evaluation of the nearest point on the mesh. An example of incorrect nearest point evaluation is illustrated in Figure 9. Due to the close proximity between two surface regions the nearest mesh vertex is not on the mesh region closest to the search point $\vec{x}$. Incorrect nearest point evaluation may cause errors in range image integration [36, 41] and surface registration [1, 8, 12, 41].

4.2 Surface Based Nearest Point Evaluation

A surface based scheme is introduced in this section which guarantees correct nearest point evaluation. Initially the mesh M_k is pre-processed to build a bucket structure [24, 36]. Let us define a plane, P, which is partitioned by a uniform grid of resolution, Δg. For each grid square (p, q) we define an empty bucket $b(p, q)$ where $p = 0...N_1$ and $q = 0...N_2$. The bucket structure is constructed by orthogonal projection of each triangle T_i, in mesh, M_k, onto the plane, P. For all squares, (p,q), that the projection of triangle T_i intersects we add a triangle label, i, to the bucket b(p,q). Projection of a triangle and labelling of the bucket structure is illustrate in Figure 11(a). Having constructed the bucket structure for mesh, M_k, each bucket, $b(p, q)$, contains a set of labels for all triangles that intersect the grid square, (p, q).

The nearest point on the mesh, M_k, to an arbitrary point, $\vec{x}$, is evaluated by a recursive search through the bucket structure in the plane P. Point $\vec{x}$ is orthogonally projected to a grid square (r, s) on the plane P. The nearest point, $\vec{x}_{nearest}$ is then evaluated for the set of triangles in bucket, $b(r, s)$. Adjacent buckets, $b(r+n, s+n)$, are then recursively searched for $n = 1, 2, ...$ to evaluate the nearest point, $\vec{x}_{nearest}$, for all triangles in buckets $b(r \pm n, s \pm n)$. The search is terminated when the distance to the nearest point is less than the minimum distance between grid square (r, s) and $(r \pm n, s \pm n)$ in the plane P: $|\vec{x} - \vec{x}_{nearest}| \leq n\Delta g$. Nearest point evaluation using a recursive search in the projection plane P is illustrated in Figure 11(b). The algorithm for surface based nearest point evaluation is summarised in Figure 10.

Surface based nearest point evaluation is guaranteed to give the correct result for an arbitrary mesh, M_k, and any search point $\vec{x}$. This follows from the property of orthogonal projection that the distance between two points $\vec{x}_1$ and $\vec{x}_(2)$ in R^3 is greater than or equal to the distance between their orthogonal projections p_1 and p_2 on a plane P in R^2: $|p_1 - p_2| \leq |\vec{x}_1 - \vec{x}_2|$. Thus by searching all buckets corresponding to grid squares whose projection is inside a distance $|\vec{x} - \vec{x}_{nearest}|$ of $\vec{x}$ in the plane P the correct nearest point $\vec{x}_{nearest}$ will always be found.

Nearest point evaluation is performed in constant time with a cost dependent on the distribution of mesh vertices in the projection plane. A reasonable projection plane is given by either the 2D range image plane or the two principal eigenvectors of the mesh vertices. Implementation of the bucket structure with grid resolution, Δg, equal to the range image sampling resolution, Δx, requires $O(N)$ storage cost.

SETUP BUCKET STRUCTURE $b(p,q)$ FOR $p = 0...N_1$ AND $q = 0...N_2$:

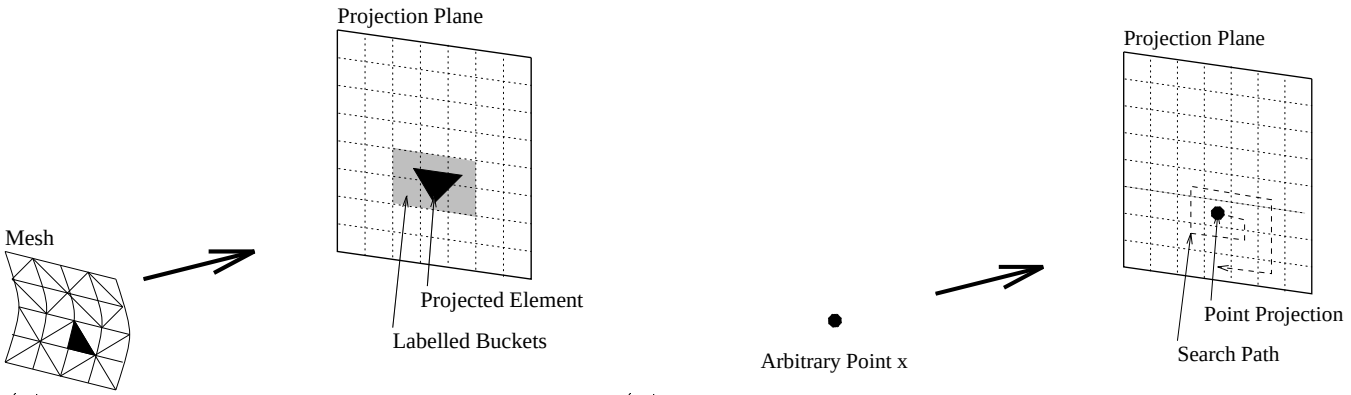
1. Define a 2D projection plane P
2. Define grid resolution Δg for partitioning plane P
3. For each triangle T_i on mesh M_k:
   (a) Orthogonally project T_i onto plane P.
   (b) For all grid squares (p,q) intersected by the projection of T_i add a label to bucket $b(p,q)$

INITIALISE NEAREST POINT SEARCH FOR POINT $\vec{x}$:
Find the grid square, (r,s), corresponding to the orthogonal projection of point $\vec{x}$ onto plane, P.

RECURSIVE SEARCH STARTING WITH $n = 0$:

1. Find the nearest point, $\vec{x}_{nearest}$, on a triangle for the set of triangles in bucket $b(r \pm n, s \pm n)$
2. If $|\vec{x} - \vec{x}_{nearest}| \leq n\Delta g$ return $\vec{x}_{nearest}$ as the nearest point and terminate.
3. Else increment $n = n + 1$ and repeat steps 1 and 2.

Figure 10: Algorithm for surface based evaluation of the nearest point on mesh M_k to point $\vec{x}$



(a) Mesh projection to bucket structure. (b) Search for nearest point.

Figure 11: Surface based nearest point evaluation

4.3 Volume Based Nearest Point Evaluation

Curless et al. [10] recently introduced an approximate volumetric implicit surface representation for integration of multiple range images. The integrated field function is pre-computed and stored in a discrete volumetric grid representation. The advantage of this representation is that field function evaluation is reduced to a single lookup operation. This eliminates the search procedure required for both vertex and surface based nearest point evaluation. Resulting in an order of magnitude reduction in computational cost.

The volume based integration algorithm [10] has two principal disadvantages over the approach presented in this paper. Firstly, fusion of overlapping measurements is not based on geometric constraints. This allows the integration of measurements from different surface regions resulting in unreliable reconstruction of complex geometry. Secondly, the discrete volumetric representation is an approximation of the measured surface geometry. The accuracy of the representation is determined by the resolution of a volumetric grid.

In this section we extend the volume based nearest point evaluation [10] to achieve accurate reconstruction with reduced computational cost. Further details of discrete volumetric surface representation are presented in [16]. The space is partitioned into uniform cubic voxels of size Δc with centres $\vec{c}_{ijk}$. For each mesh M_k the volumetric representation is pre-computed for all voxels with centres $\vec{c}_{ijk}$ less than a distance d_{max} from the mesh. In each voxel the signed distance function, $f(\vec{c}_{ijk})$, the measurement confidence, $w(\vec{c}_{ijk})$, the surface orientation, $\vec{n}(\vec{c}_{ijk})$, and the boundary function, $\vec{b}(\vec{c}_{ijk})$, is stored. For multiple meshes M_k $k = 0...m-1$ the local planar approximations are sequentially integrated according to the algorithm presented in section 3.3. This gives a discrete planar approximation for a volumetric envelope around the implicit surface, $[f(\vec{x}) = 0, b(\vec{x}) = \text{'}false\text{'}]$, for all points $\vec{x}$ where $f(\vec{x}) \leq d_{max}$.

The accuracy of the discrete volumetric representation is dependent on the voxel resolution and the order in which meshes, M_k, are introduced. This gives improved reconstruction of complex geometry compared to previous discrete volumetric representations [10]. Incorrect reconstruction of complex geometry may still occur as sequential integration does not allow discrimination between multiple sets of overlapping measurements, this is shown in Figure 8(d). The cost of volume based nearest point evaluation is a single lookup operation which is an order of magnitude less than either vertex or surface based approaches.

5 Algorithm Performance

This section evaluates the time complexity and geometric limitations of the mesh based implicit surface integration algorithm introduced in this paper. The results enable performance comparison with previous integration algorithms. A full description of the performance characterisation of previous integration algorithms is beyond the scope of this paper. For further details on performance evaluation of integration algorithms see [19].

5.1 Time Complexity

The general form for the computational complexity of any integration algorithm is a function of the particular image set. This depends on the number of images, m, the number of points in each image, N_k, the proportion of redundancy between images and the length of the boundary between overlapping images. In this section we derive the worst-case computational complexity for integrating m images of N points. Worst-case complexity occurs where all m images are fully overlapping. This

Integration Algorithm	Worst-case Complexity for m Images	Complexity of Integrating 2 images
I Point Based Implicit Surface [21]	$O(m^2N)$	$6N$
II Mesh Based Implicit Surface (this paper)	$O(m^2N)$	$12N$
III Volume Based Implicit Surface [10]	$O(mN)$	$2N$
IV Canonic Views [36]	$O(m^2N)$	$4N + 2\sqrt{N}$
V Mesh Zippering [41]	$O(m^2N)$	$N + \sqrt{N}$
VI Mesh Growing [33]	$O(mN \log N)$	$20N \log N (K = 10)$

Table 2: Integration Algorithm Complexity

enables qualitative comparison with the worst-case complexity of previous integration algorithms [19]. Quantitative comparison is enabled by considering a special case: the cost of integrating two images of N points with 50% overlap.

Computation of the mesh based implicit surface function for a single mesh requires nearest point evaluation. Vertex and surface based nearest point evaluation are approximately constant time operations. For m overlapping meshes the vertex and surface based schemes require m nearest point evaluations resulting in $O(m)$ complexity. The volumetric scheme integrates m overlapping meshes into a single representation during pre-computation. Volume based nearest point evaluation therefore requires only a single lookup operation.

The computational complexity of the implicit surface polygonisation depends on the number of implicit function evaluations. For the Marching Triangles algorithm this is directly proportional to the area of the surface [18]. This is equal to the lower bound of the computational complexity for the Marching Cubes approach. If the surface is approximately uniformly sampled then the number of samples, N, is proportional to the surface area. For a single range image the cost of implicit surface polygonisation using the Marching Triangles approach is $O(N)$. The worst-case complexity will occur for m range images where there is no redundancy. This cost of implicit surface polygonisation is then $O(mN)$. For vertex and surface based nearest point evaluation the cost is $O(m)$ resulting in $O(m^2N)$ overall complexity of the integration algorithm. For volume based nearest point evaluation the cost is a single operation resulting in $O(mN)$ computational complexity.

The computational complexity of this and previous integration algorithms are shown in Table 2. Results for previous integration algorithms were evaluated in [19]. Comparison of these results shows that the worst-case complexity of the mesh based implicit surface is the same as that of previous approaches, $O(m^2N)$. The approximate volume based implicit surface representation reduces the computational complexity to $O(mN)$ at the expense of geometric reliability.

5.2 Geometric Limitations

In this section the geometric limitations inherent in the implicit surface based integration algorithm introduced in this paper are defined. Assumptions about object surface geometry result in limitations for all integration algorithms. An analysis of limitations of previous integration algorithms was presented in [19]. Results of this analysis are summarised in Table 3 for comparison. Geometric limitations inherent in the volume based implicit surface representation are also discussed. Limitations on correct reconstruction of surface geometry are identified by considering three cases: minimum feature size, maximum surface curvature and minimum surface separation. Examples of the complex

surfaces corresponding to these cases are illustrated in Figure 1.

Minimum Feature Size: Step discontinuity constrained triangulation is used to estimate the local surface topology from individual range images, section 3.1. The constant distance threshold $t_d = n\Delta x$ limits the minimum feature size that is reliably reconstructed to $n\Delta x$. A second potential limitation on the feature size is the resolution of the implicit surface polygonisation algorithm. This will not impose an additional limitation provided the triangle size for Marching triangles is less than the distance threshold as is the case throughout this work. The volumetric implicit surface without boundary information may fail to reliably reconstruct holes smaller than the distance threshold d_{max}. Addition of boundary information into the volumetric representation enables accurate representation of all features greater than the cube size $\Delta c = \Delta x$.

Minimum Crease Angle: Due to the use of a continuous implicit surface function the algorithm will correctly reconstruct the surface in regions of high curvature (i.e. edges). This is guaranteed provided at least one of the range image meshes, M_k, correctly defines the local surface continuity. The requirement that a single mesh should define the local continuity across an edge imposes a practical limitation on the maximum edge angle. This is a compromise between the sampling resolution, Δx and the step discontinuity distance threshold $n\Delta x$. The minimum crease angle, Figure 1(b), which can be reliably reconstructed is approximately $\theta_{min} \approx 2sin^{-1}(\Delta x/n\Delta x) = 2sin^{-1}(1/n)$ (for $n = 4 \;\; \theta_{min} \approx 30°$). The discrete volumetric implicit surface [10] fails to reliably reconstruct the surface for crease angles less than 90°. This is due to the integration of field functions from different surface regions close to the edge. Addition of surface orientation information into the discrete representation allows geometric constraints to prevent combination of overlapping field functions from different surfaces. This enables improved reconstruction of crease edges. However, errors may still occur for multiple overlapping meshes due to the discrete volumetric approximation.

Minimum Surface Separation: Adjacent surfaces with opposite orientations (Figure 1(c)) will be correctly represented in the integrated implicit surface provided the minimum surface separation is less than the maximum measurement error ε_{max}. Marching Triangles correctly reconstructs arbitrarily thin surface sections by local tracking of the implicit surface. Marching Cubes polygonisation imposes an additional limitation that the minimum surface separation is greater than the spatial subdivision Δc. The discrete volumetric implicit surface may fail to reconstruct surfaces in close proximity if their field functions overlap. This is similar to the failure of this approach at crease edges. If the field function is approximated for a maximum distance $d_{max} = n\Delta x$ either side of the surface then errors may occur for surface separations less than $2d_{max}$. Addition of orientation information into the volumetric representation eliminates this problem. This reduces the minimum surface separation to be equal to the voxel cube size, Δc.

Geometric limitations for previous integration algorithms [19] together with the new approach are presented in Table 3. This shows that the mesh based implicit approach enables reconstruction which is as good or better than all previous approaches. In particular other implicit surface based approaches (I,III) [10, 21] have much more severe limitations on object surface geometry.

6 Results

This section presents results of 3D model reconstruction using the new implicit surface based approach. Results for multiple range images are presented in section 6.1 and for multiple surface patches from a hand-held range sensor in section 6.2. The computational cost for model reconstruction using vertex, surface and volume based nearest point computation are presented in section 6.3. Improved reliability for model reconstruction using the new approach is demonstrated in section 6.4.

Integration Algorithm	Minimum Feature Size	Minimum Crease Angle	Minimum Surface Separation
I Point Based Implicit Surface [21]	$> 3n\Delta\vec{x}$	140°	$n\Delta\vec{x}$
II Mesh Based Implicit Surface (this paper)	$n\Delta\vec{x}$	30°	ε_{max}
III Volume Based Implicit Surface [10]	$n\Delta\vec{x}$	90°	$n\Delta\vec{x}$
IV Canonic Views [36]	$n\Delta\vec{x}$	30°	ε_{max}
V Mesh Zippering [41]	$2n\Delta\vec{x}$	90°	$n\Delta\vec{x}$
VI Mesh Growing [33]	$2n\Delta\vec{x}$	90°	ε_{max}

Table 3: Integration Algorithm Limitations for Reliable Reconstruction

6.1 3D Model Reconstruction from Multiple Range Images

Results of the reconstruction process using the mesh based implicit surface for multiple range images of four objects are shown in Figure 12. Novel views of the 3D object models are shown which were not in the input 2.5D range image data sets. Nearest point evaluation is implemented using the surface based approach introduced in section 4.2. Implicit surface polygonisation is performed using the Marching Triangles approach with triangle size equal to the magnitude of the range image sampling resolution $|\Delta\vec{x}|$. The bunny [1] and telephone[1] data sets are composed of ten range images taken from multiple 3D views. This data was previously used to demonstrate the mesh zippering algorithm, [41]. The teapot [2] and soldier[2] data sets are composed of eight range images taken in a horizontal plane around the objects. These data sets do not contain any measurements for horizontal surfaces. The Canonic subsets integration algorithm was previously applied to this data, [36].

Results demonstrate that the new integration algorithm correctly reconstructs complex surface features larger than the sampling resolution $\Delta\vec{x}$. Holes in the original data are correctly preserved in the reconstructed model. For example holes in the input data that occur on horizontal surfaces in the soldier data set are preserved in the integrated 3D model. Surface regions of high curvature are correctly reconstructed as continuous surfaces. An example of correct reconstruction in a region of high curvature is the top of the bunny ear. Failure of previous integration algorithms in regions of high curvature is demonstrated in section 6.4. Correct reconstruction for thin object sections is demonstrated for the bunny ear, teapot handle and soldiers shield. These results show that the new geometric fusion algorithm overcomes several limitations on object geometry inherent in previous integration methods [10, 21, 33, 41]. Results indicate that the new implicit surface based geometric fusion algorithm enables reliable reconstruction of complex object geometry and topology.

6.2 3D Model Reconstruction from Hand-Held Sensor Range Data

Hand-held range sensors for object surface measurement are a recent development. This technology consists of a laser stripe based range sensor combined with a six degree-of-freedom position sensor. Unlike previous range sensors measurements are not in an image format. Range measurements are on a series of consecutive stripes which do not form a regular grid pattern. Constrained triangulation can be performed on consecutive range data stripes to obtain triangulated surface meshes. In general all triangles on a mesh will not be visible from a single view-point. This prevents the use of all previous integration techniques based on 2D projection [33, 36, 41] or range image view-point [10]. The implicit surface based geometric fusion algorithm introduced in this paper enables the integration of arbitrary

[1] Cyberware scanner range data registered using ICP [41]

[2] NRCC range data [32] registered using InnovMetric software [36]

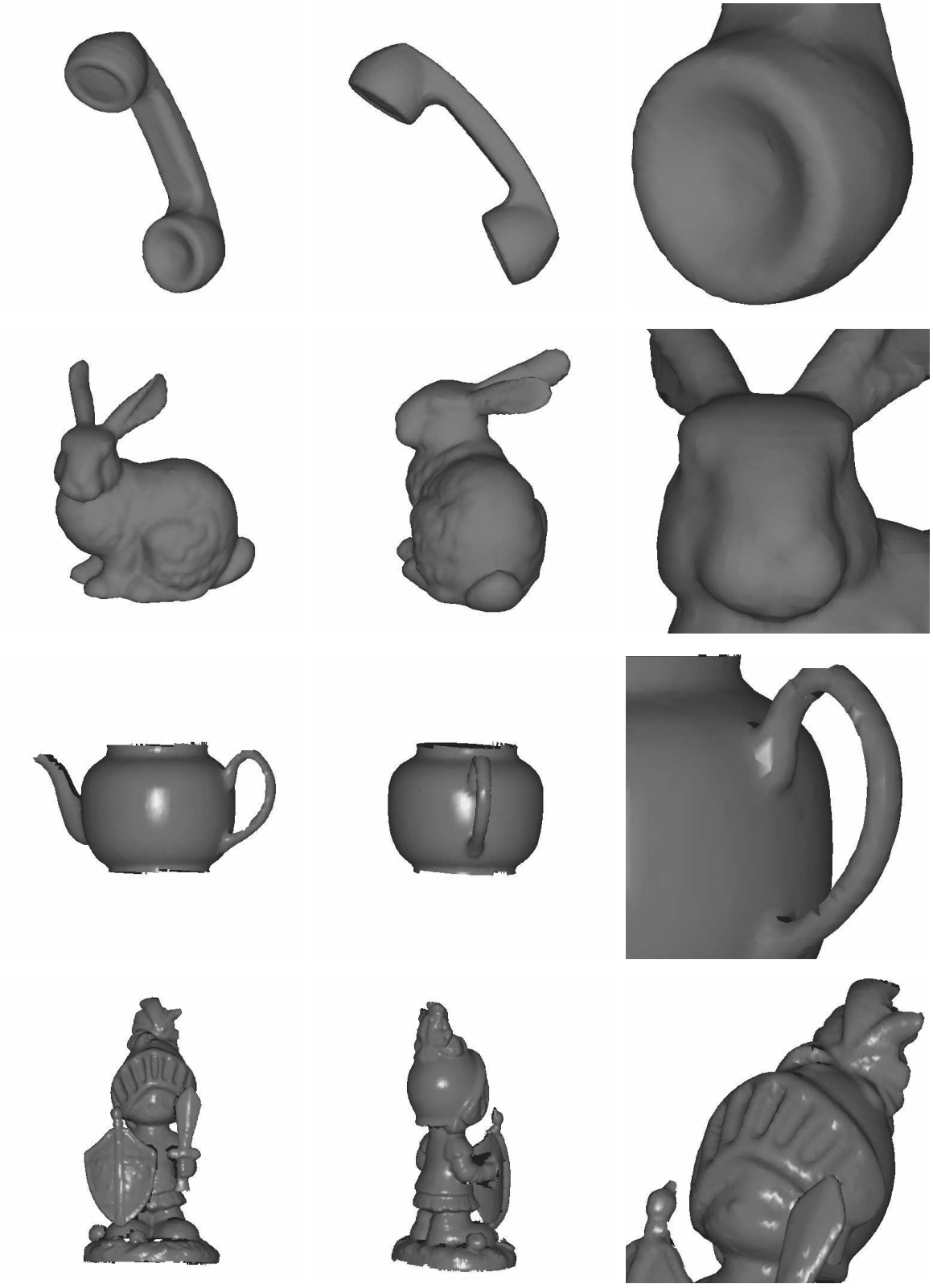


Figure 12: Novel views of 3D models reconstructed by fusing data from multiple range images

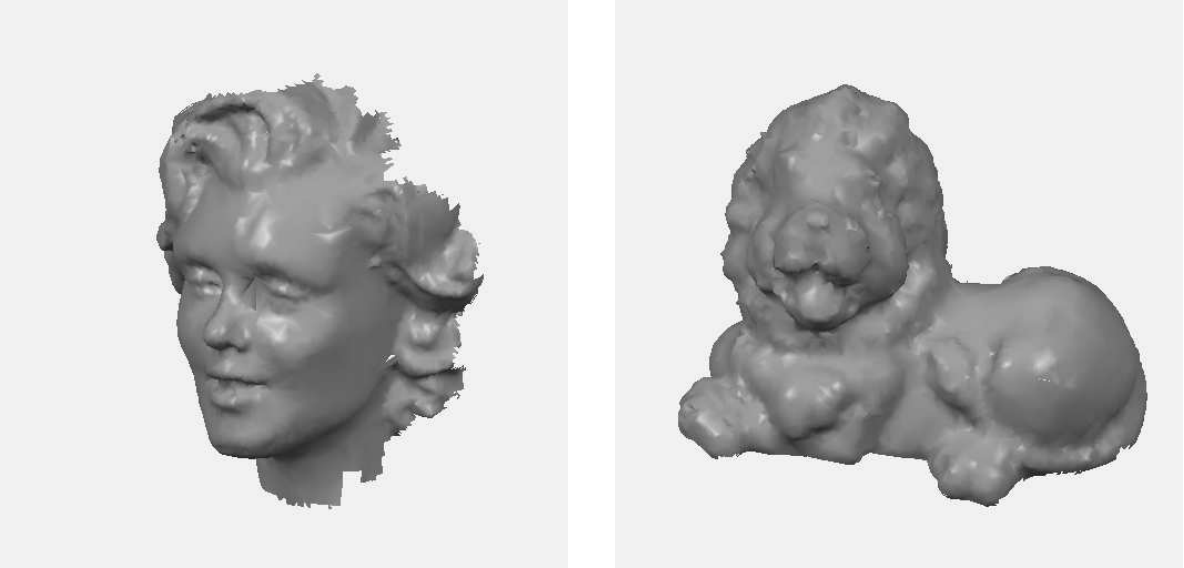


Figure 13: Reconstructed 3D models from a hand-held range sensor

overlapping meshes using operations in 3D space only. Therefore, the new algorithm can be used to integrate surface measurements from a hand-held sensor into a single implicit surface representation. This is the first geometric fusion algorithm capable of integrating surface measurements from a hand-held range sensor. Fusion of measurements from a hand-held sensor presents the additional challenge that the sensor accuracy is currently an order of magnitude lower than conventional range image sensor. This is due to the requirement for a low cost six degree-of-freedom device to measure the spatial position of the sensor.

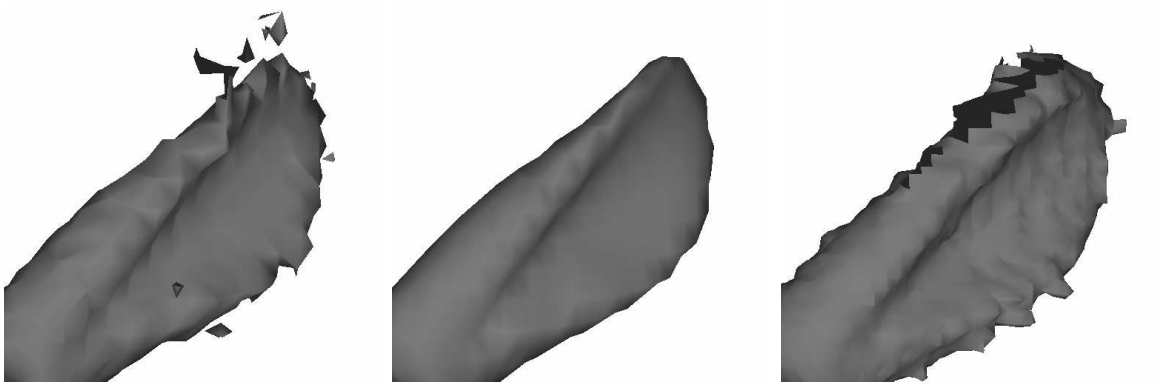
Figure 13 shows results of the geometric fusion for surface measurements of two objects obtained using the 3D Scanners ModelMaker hand-held sensor. These data sets are both composed of approximately thirty surface patches. Volume based nearest point computation was used for efficient implementation of the fusion algorithm. Results demonstrate correct reconstruction of the object surface despite the increased noise levels. Correct reconstruction for hand-held sensor data demonstrates the robustness of the new geometric fusion algorithm to measurement noise.

6.3 Comparison of Nearest Point Implementations

Table 4 gives a comparison of the computational cost on a SUN Sparc10 for nearest point computation using the three schemes presented in section 4. The computation time given for the volume based approach is for the nearest point field function evaluation only. The cost of pre-computing the volumetric representation depends on the resolution and the maximum distance from the surface, d_{max}. Fast schemes for pre-computation of the volumetric representation are discussed in [10]. Comparison of the computation time shows that the surface based approach has the worst performance. However, this approach is the only one which guarantees correct nearest point evaluation. This approach is preferable to the vertex based approach used in previous work [41, 36]. Volume based nearest point evaluation has the lowest computational cost. However, this approach does not enable reliable fusion for complex object geometry as discussed in section 4.3. The choice of nearest point implementation therefore requires a tradeoff between reliable reconstruction and computation cost.

Object	Nearest Point Scheme		
	Vertex	Surface	Volume
Telephone	43	61	8
Bunny	106	138	16
Teapot	795	957	28
Soldier	1087	1579	64

Table 4: Nearest point computation time(s) (SUN Sparc10)



(a) Point based [21] (b) Mesh based (this paper) (c) Volume based [10]

Figure 14: Comparison of implicit surface based reconstruction

6.4 Performance Comparison of Implicit Surface Fusion Techniques

Three implicit surface based geometric fusion algorithms have been proposed point based [21], mesh based (this paper) and volume based [10]. Figure 14 shows the results of each of these approaches for a high curvature region of an object surface. Results show that both the point and volume based techniques fail to reliably reconstruct the continuous object surface. Both of these techniques use a discrete implicit surface representation resulting in incorrect combination of overlapping measurements. The mesh based approach introduced in this paper uses a continuous implicit surface representation. This enables correct reconstruction of complex geometries such as regions of high curvature and thin object sections. In addition the mesh based implicit surface has an explicit representation of the surface boundary. This allows correct reconstruction of holes in the object surface. Point and volume based approaches do not represent boundary information and therefore can not accurately reconstruct open manifold surfaces.

7 Conclusion

In this paper a new implicit surface based technique for reliable geometric fusion of surface measurements has been introduced. The principal advantages of the new approach over previous approaches are:

- Geometric fusion using operations in 3D space only. Previous approaches [33, 36, 41] required 2D projection resulting in limitations on object geometry for reliable reconstruction.

- Integration of overlapping measurements based on geometric constraints and measurement uncertainty. This enables reliable geometric fusion of complex object geometry in the presence of measurement noise.

- Reconstruction of objects of arbitrary unknown topology. No a priori assumptions are made about object topology. However, estimation of surface topology using constrained triangulation limits the minimum hole size to a multiple of the sampling resolution $n\Delta x$. This limitation also applies to previous methods [10, 33, 36, 41].

- A continuous implicit surface function. Other implicit surface schemes [10, 21] use discrete representations which do not enable reliable reconstruction for regions of high curvature or thin sections.

- Reliable reconstruction of object geometry and topology. All previous integration approaches [10, 21, 33, 36, 41] have greater restrictions on the object surface geometry that is reliably reconstructed.

- A general algorithm for geometric fusion of surface measurements from both conventional range image sensors and unstructured hand-held range sensor data. Previous algorithms based on the range image structure [10, 33, 36, 41] are unable to integrate surface measurements in a non-image format.

- Robust geometric fusion of noisy surface measurements. The new approach correctly reconstructs the object surface from hand-held range sensor data which is an order of magnitude less accurate than conventional range image sensors.

Results have been demonstrated for geometric fusion of surface measurements from multiple range images and hand-held sensor range data. Reliable fusion is achieved for complex geometries such as holes, regions of high curvature and thin object sections. The new implicit surface based approach is the first general purpose algorithm for geometric fusion of multiple sources of surface measurement. In addition this algorithm gives improved reliability for reconstruction of complex geometry and is insensitive to measurement noise.

A surface based approach to nearest point computation has been introduced. This approach is guaranteed to compute the correct nearest point on a triangulated mesh. Correct nearest point computation is important for both surface registration [1, 8, 12] and geometric fusion. The surface based approach has the same order of complexity as previous vertex based approaches [36, 41]. Volume based nearest point evaluation [10] has been extended to include surface orientation and boundary information. This gives reliable reconstruction based on a partial implementation of the integration algorithm introduced in this paper. The volume based approach gives an order of magnitude reduction in computational cost compared to the surface based approach. However, the discrete volumetric approximation of the implicit surface may fail for complex geometries. Implementation of implicit surface based geometric fusion therefore requires a compromise between reliability and computational cost.

References

[1] P. Besl and N. McKay. A method for registration of 3-d shapes. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 14(2):239—255, 1992.

[2] E. Bittar, N. Tsingos, and M-P. Gascuel. Automatic reconstruction of unstructured 3d data: Combining a medial axis and implicit surface. In *EUROGRAPHICS 14(3)*, pages 457—468, 1995.

[3] J. Bloomenthal. Polygonization of implicit surfaces. *Computer Aided Geometric Design*, 5:341—355, 1988.

[4] J. Bloomenthal. An implicit surface polygonizer. *Graphics Gems ed. Heckbert,P.S.*, 4:324—350, 1994.

[5] J.D. Boissonnat. Geometric structures for three-dimensional shape representation. *ACM Transactions on Graphics*, 3(4):266—286, 1984.

[6] R.M. Bolle and B.C. Vemuri. On three-dimensional surface reconstruction methods. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 13(1):1—13, 1991.

[7] Y. Chen and F. Medioni. Surface description of complex objects from multiple range images. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 153—158, 1994.

[8] Y. Chen and G. Medioni. Object modeling by registration of multiple range images. *Image and Vision Computing*, 10(3):145–155, 1992.

[9] H.E. Cline, W.E. Lorensen, S. Ludke, C.R. Crawford, and B.C. Teeter. Two algorithms for the three-dimensional reconstruction of tomograms. *Medical Physics*, 15(3):320—327, 1988.

[10] B. Curless and M. Levoy. A volumetric method for building complex models from range images. In *Computer Graphics Proceedings, SIGGRAPH*, 1996.

[11] D. DeCarlo and D. Metaxas. Adaptive shape evolution using blending. In *International Conf. on Computer Vision*, pages 834—839, 1995.

[12] H. Gagnon, M. Soucy, R. Bergevin, and D. Laurendeau. Registration of multiple range views for automatic 3d model building. In *Proceedings IEEE Conference on Computer Vision and Pattern Recognition*, pages 581–586, 1994.

[13] A. Guziec and D. Dean. Simplification of triangulated surfaces for crest line extraction and surface registration. In *Proceedings of Leeds Annual Statistics Research Workshop, Image Fusion and Shape Variability Techniques*, pages 103—114, 1996.

[14] A. Hilton and J. Goncalves. 3d scene representation using a deformable surface. In *IEEE Workshop on Physics Based Modelling*, pages 24—30. IEEE, 1995.

[15] A. Hilton and J. Illingworth. Marching triangles: Surface based implicit surface polygonisation. In *Submitted to IEEE Computer Graphics and Applications*. The draft submission is available from ftp://ftp.ee.surrey.ac.uk/pub/vision/papers/hilton-tr1-97.ps.Z, 1997.

[16] A. Hilton and J. Illingworth. Multi-resolution geometric fusion. In *International Conference on Recent Advances in 3D Digital Imaging and Modeling*. IEEE, 1997.

[17] A. Hilton, J. Illingworth, and T. Windeatt. Statistics of surface curvature estimates. *Pattern Recognition*, 28(8):1201—1221, 1995.

[18] A. Hilton, A.J. Stoddart, J. Illingworth, and T. Windeatt. Marching triangles: Range image fusion for complex object modelling. In *International Conf. on Image Processing*, pages 381—384. Lusanne, 1996.

[19] A. Hilton, A.J. Stoddart, J. Illingworth, and T. Windeatt. Reliable surface reconstruction from multiple range images. In *4th European Conference on Computer Vision*, pages 117—126. Springer, 1996.

[20] H. Hoppe, T. DeRose, T. Duchamp, M. Halstead, J. Hin, H. McDonald, J. Schweitzer, and W. Stuetzle. Piecewise smooth surface reconstruction. In *Computer Graphics Proceedings, SIGGRAPH*, pages 295—302, 1994.

[21] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle. Surface reconstruction from unorganised points. *Computer Graphics*, 26(2):71—77, 1992.

[22] H. Hoppe, T. DeRose, T. Duchamp, J. McDonald, and W. Stuetzle. Mesh optimization. In *Computer Graphics Proceedings, SIGGRAPH*, pages 19—26, 1993.

[23] A. Kalvin and R. Taylor. Superfaces: polygonal mesh simplification with bounded error. *IEEE Computer Graphics and Applications*, 16(3):64—77, 1996.

[24] D. Knuth. *Sorting and Searching.* The Art of Computer Programming Volume 3, Addison-Wesley, 1975.

[25] W.E. Lorensen and H.E. Cline. Marching cubes: A high resolution 3d surface construction algorithm. *Computer Graphics*, 21(4):163—169, 1987.

[26] T. McInerney and D. Terzoupoulos. A finite element model for 3D shape reconstruction and nonrigid motion tracking. In *International Conf. on Computer Vision*, pages 518—523, 1993.

[27] T. McInerney and D. Terzoupoulos. Deformable models in medical image processing. *Medical Image Analysis*, 1(2):91—108, 1996.

[28] R. Mencl. A graph-based approach to surface reconstruction. In *EUROGRAPHICS 14(3)*, pages 446—456, 1995.

[29] D. Metaxas and D. Terzoupoulos. Shape and nonrigid motion estimation through physics-based synthesis. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 15, 1993.

[30] S. Muraki. Volumetric shape description of range data using blobby model. *Computer Graphics*, 25(4):227—235, 1991.

[31] R. Pito. Mesh integration based on co-measurements. In *International Conf. on Image Processing*, pages 397—400, 1996.

[32] M. Rioux. Laser range finder based on synchronized scanners. *Applied Optics*, 23(21):3837—3844, 1984.

[33] M. Rutishauser, M. Stricker, and M. Trobina. Merging range images of arbitrarily shaped objects. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 573—580, 1994.

[34] W.J. Schroeder, J.A. Zarge, and W.E. Lorensen. Decimation of triangular meshes. In *SIGGRAPH 26(2)*, pages 65—70, 1992.

[35] M. Soucy and D. Laurendeau. Multi-resolution surface modelling from multiple range images. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, pages 348–353, 1992.

[36] M. Soucy and D. Laurendeau. A general surface approach to the integration of a set of range views. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 14(4):344–358, 1995.

[37] M. Soucy and D. Laurendeau. Multiresolution surface modeling based on hierarchical triangulation. *Computer Vision and Image Understanding*, 63(1):1—14, 1996.

[38] M.R. Spiegel. *Statistics: Theory and Problems.* Schaum's Outline Series, 1992.

[39] R. Szeliski and D. Tonnesen. Surface modeling with oriented particle systems. *Computer Graphics*, 26(2):185–194, 1992.

[40] H. Tek and B.B. Kimia. Volumetric segmentation of medical images by three-dimensional bubbles. In *IEEE Workshop on Physics-Based Modelling in Computer Vision*, 1995.

[41] G. Turk and M. Levoy. Zippered polygon meshes from range images. In *Computer Graphics Proceedings, SIGGRAPH*, 1994.

[42] W. Welch and A. Witkin. Free-form shape design using triangulated surfaces. In *Computer Graphics Proceedings, SIGGRAPH*, pages 247–255, 1994.

[43] A.P. Witkin and P.S. Heckbert. Using particles to sample and control implicit surfaces. In *Computer Graphics Proceedings, SIGGRAPH*, pages 269–277, 1994.

[44] G. Wyvill, C. McPheeters, and B. Wyvill. Data structure for soft objects. *The Visual Computer*, 2:227–234, 1986.