Closing Gaps by Clustering Unseen Directions

G. Impoco, P. Cignoni, R. Scopigno
Istituto di Scienza e Tecnologie dell'Informazione (ISTI)
Consiglio Nazionale delle Ricerche
Via Moruzzi 1, 56125 - Pisa, ITALY.
e-mail: impoco | cignoni | scopigno @ isti.cnr.it

Abstract

Although in recent years the 3D-scanning field has reached a good level of maturity, it is still far from being perceived by common users as a 3D-photography approach, as simple as standard photography is. The main reason for that is that obtaining good 3D models without human intervention is still very hard. In particular, two problems remain open: automatic registration *of single shots and* planning *of the acquisition session. In this paper we address the second issue and propose a solution to improve the coverage of automatically acquired objects. Rather than searching for the* next-best-view *in order to minimise the number of acquisitions, we propose a simple and easy-to-implement algorithm limiting our scope to closing gaps (i.e. filling unsampled regions) in roughly acquired models. The idea is very simple: detect holes in the current model and cluster their estimated normals in order to determine new views. Some results are shown to support our approach.*

1. Introduction

Scanning 3D objects is not an easy task, even if the accuracy and speed of current scanning technology improved significantly. The *accuracy* of modern scanning devices (50 microns or better for triangulation-based scanners), their *spatial resolution* (i.e. the mean distance between sampled points, usually in the range of 0.1-0.5 mm) and finally the *sampling speed* (100K-300K samples per second) are considered to fulfill the requirements of most applications. However, there are still bottlenecks slowing down the scanning process: the time and user effort required to *process* the *raw data* in order to convert a set of range maps into an accurate triangle mesh [1], and the difficulty of obtaining a *complete sampling* of the requested object surface. The first bottleneck can be reduced by designing systems which automatise the post-processing phase by tracking the scanner position (usually, using either computer-controlled gantries or vision-based technologies) or by enhancing the post-processing tools with a nearly automatic alignment process [2]. The second bottleneck has been reported, among others, by Marc Levoy on the base of his experience while scanning a large subset of the Michelangelo's statues [13]. He noted that it is often hard to obtain a complete coverage of complex objects' surfaces, and the progressive coverage of models comes with an effort which is inversely proportional to the sampled fraction.

The aim of the research presented in this paper is to automatise the scanning process, helping the user in the selection of the views to be taken. Planning the acquisition of a complex object is a very hard task, assuming that no previous knowledge on the shape is known. We cannot leave this duty to the operator, since planning and optimising the acquisition of 100-200 range maps can be difficult even for an expert operator. Our goal is to design a system able to scan efficiently (in terms of time required and completeness of the surface sampling) any complex object, by making a joint use of the basic intuition of the operator and the finest planning and refining capabilities of a computer-based supporting tool.

This topic has been extensively studied in literature and it is usually referred to as *Next Best View (NBV)* selection. The goal of most of the previous NBV research is to determine, given a single scan or a set of already scanned intermediate results, which is the next *best* location and orientation of the scanner. While looking for the NBV, a preliminary choice is how do we define the optimality of a given view. Although many authors select the *minimal number* of views as an optimality criterion, in our opinion this is a non-critical parameter due to the acquisition speed of contemporary scanners and the quality of current post-processing software (which can manage hundreds of range maps). Moreover, data redundancy can be used to reduce the effect of noise and sampling inaccuracies. Therefore, the *optimality* of a scanning set is mainly related to the *completeness* of the scanning (percentage of surface sampled), and the mean *glanc-*

ing angle used to scan the surface, since most scanners give high quality samples only when the acquisition direction is nearly parallel to the surface normal.

Some of the previous approaches have also focused on how to determine a first set of views, to be further improved in a second stage. To accomplish this task we can either use already known information on the object's shape (e.g. the available CAD model in quality inspection of manufactured pieces), or by taking first an approximate model of the object. The second approach is often implemented by using a different acquisition device having a wider field of view, increasing both the HW cost and the operative complexity for the operator (he should be able to manage different systems). We believe that designing a preliminary acquisition plan (which could be refined in a second stage) is very easy even for a modestly-skilled operator. In practice, most of the scanning tasks can be solved by choosing one of two different approaches.

1. **X-Y sweep**. This is the typical setup used to scan a bas-relief or a car body section. The characteristics of the scanner are assumed to be known. Given the extent of the object in a given projection plane, we can easily partition the scanning space into $n \times m$ range maps, to be acquired by a regular sweeping of the scanner along the two directions on the projection plane. Scanner displacement depends on the inter-scan overlapping factor (usually 0.7-0.8 the scan height and width).

2. **Cylindrical sweep**. The common approach to scan an all-round object consists in the acquisition of a set of cylindrical scans, obtained by rotating the scanner around the object or by using a rotating platform, which allows to move the object while the scanner is steady. The number of shots in a complete rotation depends on the mean diameter of the object and the inter-scan overlap factor (usually 20-30%) and can be easily computed. The vertical sweep factor and the number of cylindrical acquisitions again depend on the inter-scan overlapping factor and the object's height.

In both the above cases, the initial view planning is a matter of a few seconds. Obviously, very rarely this ends up in a 100% complete acquisition, since it does not take into account surface characteristics to guide further planning. In fact, even if the above scanning strategies give a rather well distributed set of views, the presence of self-obstructing regions on the surface can produce *unseen regions* (see Figure 1). However, an initial rough planning is a good starting point.

The approach proposed in this paper tries to improve over this simple initial planning, by detecting the surface regions that have not been covered by the initial scans. For each of these potential holes, we estimate a set of surface normal vectors (e.g. considering the supposed orientation of holes' faces). A set of views which will improve considerably the coverage factor is automatically generated on the basis of a quantisation of the viewing direction space.

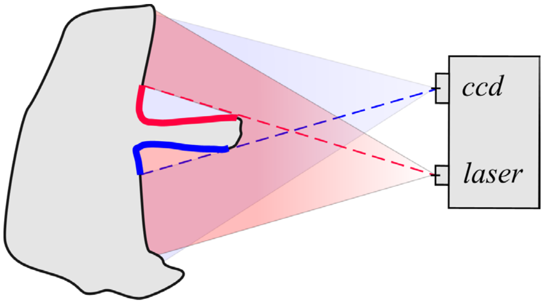


Figure 1. An example of unseen regions originated from the presence of a self-obstructing component of the shape. Thick lines indicate unsampled surface sections.

The remainder of this paper is structured as follows. After a brief overview of the some well-known planning strategies (Section 2) and some initial considerations (Section 3), our approach is presented in Section 4. An extension that allows to manage full sets of range maps is then discussed in Section 5, and some results are presented in Section 6. Finally, some considerations and directions for future work are drawn in Section 7.

2. Related Work

Sensor planning can be defined as the problem of determining a sequence of sensing operations to view the surface of a given scene. Each sensing operation is a set of *sensor parameters* such as *viewing pose* (i.e. position and orientation of the sensor), camera focus and field of view, and *environment parameters* such as controlled lighting conditions (e.g. in BRDF acquisition). Most of the techniques in literature do not take into account environment parameters and assume fixed most sensor parameters. The focus is mainly on the determination of a set of viewing directions, optimising some given criteria and subject to a number of constraints. A common target criterion is the minimisation of the number of sensing operations that allow the scene to be completely covered.

Sensor planning has been addressed mainly by the computer vision community as the problem of determining the minimum set of viewing directions covering the whole object's surface. Following the approach of Maver and Bajcsy [14] planning strategies can be classified according to how much a-priori knowledge about the scene is available. If a complete model of the scene is known in advance, the plan can be computed off-line using the reference model. On the other side, if no geometrical information is available prior to the acquisition phase, the best we can do is to build up the plan on-line step by step. This approach, known as the Next Best View (NBV) problem, is more popular in the computer graphics community in the context of surface acquisition. A third class of algorithms lies midway between the first two in the sense that only partial information about the scene is accessible.

When planning the acquisition of a 3D shape no prior knowledge is given about the surface to be acquired. Sometimes the object to be acquired is assumed to be bounded by a (often spherical or cylindrical) volume. The scanning system is often constrained to lie onto the surface of the bounding volume, thus reducing the number of degrees-of-freedom (DOFs) of the viewing pose from six to two. This may cause some viewable surface areas to become not measurable by the system. The same may happen discretising the set of viewpoints.

A common situation in which a model of the scene is known a-priori is the case of industrial inspection of mechanical parts, where a CAD reference model is used to verify the compliance of manufactured objects. Tarbox and Gottschlich's algorithm [25] generates a plan based on the reference model and then verifies it using a model of the data acquisition system. Despite this off-line verification phase, when the plan is executed it can still be incomplete due to imperfections in the model of the acquisition system. A further on-line planning phase is then performed to fill in the holes. Three different planning algorithms are proposed. The first one constructs the plan incrementally by choosing the sensing operation that is able to sense the largest portion of unseen surface area. A quality constraint must be satisfied in order to consider a surface point measurable from a given viewpoint. Namely, if the angle between the normal to the surface in the given point and the viewing direction (*glancing angle*) exceeds a given threshold the point is marked as non measurable from the current viewpoint. A second procedure weights each point with respect to its glancing angle in order to balance between a short plan and a high quality reconstructed model. A third algorithm employs a simulated annealing scheme to perform a local search on the space of viewpoints.

One of the first algorithms in the class of approaches with no a-priori knowledge was proposed by Connolly [5]. He partitions a spherical bounding volume using an octree representation. All octree nodes have a label initialised as *unseen*. When a portion of the surface is scanned, the nodes containing the sampled surface are labelled as *seen* and the regions in the portion of the conoid between the scanner and the surface are labelled as *empty*. Two algorithms are presented to compute the NBV. In the "planetarium" algorithm the bounding sphere surface is sampled in a number of candidate viewpoints. Visibility of the surface for each candidate position is evaluated and the NBV is selected as the viewpoint maximising the amount of unseen nodes. This is a burdensome algorithm because computing the visibility information requires ray casting through each octree node. The "normal" algorithm counts the area shared by the faces of nodes which separate empty and unseen regions. Consequently, only six directions are possible. The NBV is the direction maximising the shared area. Although this algorithm is computationally much cheaper with respect to planetarium, it is a fairly naive approach.

Maver and Bajcsy [14] proposed a planning algorithm tailored to a light stripe range sensor constrained to translations and limited rotations in a plane above the object. Occluded regions are represented as polygons. Viewpoint visibility constraints are computed from the polygon boundaries. This algorithm cannot be easily used with other scanning configurations.

Whaite and Ferrie [26] developed a model-based approach in which a parametric model (superquadric) is fitted to the currently sensed data. At each step the model can be refined minimising the uncertainty in the model itself. Since the uncertainty is strictly tied to how well the sensed data fits the current model, the next operation is to scan the region where data fits the model worst. The main limit of this algorithm is the inability of simple parametric models to accurately represent surface detail.

A general framework for viewpoint planning is presented by Pito [18]. His algorithm assumes that the scanning volume is enclosed by a surface parameterised using a representation called *positional space*, consisting of two bi-dimensional scalar fields. This representation allows to avoid the heavy memory requirements of an octree structure. The positional space encodes the visibility information for each candidate viewpoint. The viewpoint which maximises the unseen volume is chosen as the NBV. Although Pito's framework can account for every possible bounding volume, since the scanner is constrained to move on its boundary, the volume shape should be tailored to every object. As a consequence, in practice it is very difficult to obtain a full six DOF sensor planning algorithm in this framework.

All the approaches described above, in order to avoid collisions between the scanner and the object, assume that the scanner never enters a given bounding volume. Papadopoulos-Orfanos and Schmitt [16] presented an approach which incorporates a path planning algorithm for collision avoidance. They use a laser stripe scanner mounted on a robot with three translational DOF. The acquired data is used to guide both the sensor planning and path planning algorithms. They also exploit the geometry of the laser stripe scanner to obtain a more efficient space carving strategy, based on direct and indirect shadowing (i.e. occlusions from the light source and to the camera). The main drawback of this approach is that the scanning system is not allowed to rotate around the optical centre. This constrains severely the shape of scannable objects.

Reed and Allen [19] encoded three planning constraints as operations on sets. They consider the workspace volume as a set of points. The constraints are enforced onto the workspace by applying set operators. Viewpoints are regarded as constraints. Since set operations may be computationally expensive, this approach is inefficient.

Scott et alii [21] worked out a multi-stage approach in which a coarse model acquired in a stage is used to guide scene exploration for the fine modelling of the next stage. An a-priori base model is needed at the first stage. It can be as simple as a bounding volume. Virtually any number of stages can be used but in practice a choice of two should suffice. The algorithm partitions the rough model into cavities, holes and planar patches and for each patch a set of candidate viewpoints is generated. A user defined *measurability* function is computed for each patch of the rough model. A simple approximated set covering algorithm is employed to select the views that together give the best sampling of the whole surface. This algorithm does not constrain viewpoint space, hence full six DOF plans can be generated if a suitable discretisation of the viewpoint space is done. However, finding a good discretisation is not straightforward, since we cannot assume that the scanning system can sense all surface data in a given direction. Namely, a slope along the direction of optical axis may be so steep that some points would not be imaged at all. Hence, some points' measurability from a given direction may be misrepresented. Figure 2 depicts an example in which this situation occurs. The authors do not suggest any method to cope with this problem.

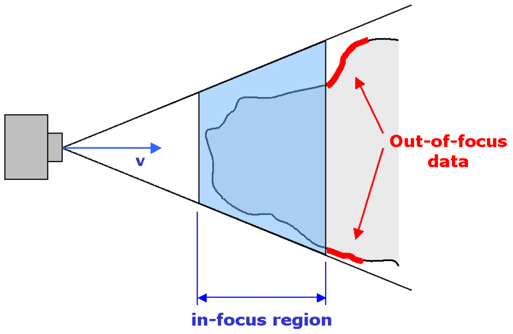


Figure 2. A candidate NBV. Some planning algorithms do not exclude out-of-focus surface points when computing measurability from a viewing direction.

Model-based object recognition and localisation is an example of planning problem with partial a-priori knowledge. Given a model of the object's shape, its pose must be determined. Here an approximation of the shape is known in advance, but not the object's pose. Most approaches in this class follow a common scheme [10, 11]. A search is performed in the space of poses employing the *hypothesis-and-verify* paradigm. Hypotheses are formed regarding the objects' identities and poses, and assessed in compliance to some metrics. New sensing operations are performed accordingly until a halting condition is met.

A good survey of view planning algorithms can be found in [24]. For a recent survey see [22] which also provides an alternative and effective classification.

3. Assumptions

Complex objects may have many protrusions and partially occluded concavities. In order to fulfil the *scanning requirement* (i.e. all the "viewable" surface should be scanned [18]), a planning algorithm should allow the scanner to be positioned and oriented in every possible way. Planning algorithms that constrain the scanner to move on a predetermined surface (e.g. a sphere centred on the object centre of mass), may have forbidden positions in the sixth-order pose space that are dependent on the algorithm itself, rather than on objects' features. That is why they are doomed to fail in meeting the scanning requirement.
Our algorithm, being guided by *occluding patches* (see Figure 3), does not have any restrictions or forbidden regions and, in principle, can sample all the surface viewable by the scanning hardware. Actually, a quantisation of the pose space is carried out, but since the pose space is quantised uniformly the "missing" surface patches will be smaller and smaller as the quantisation step becomes finer. In the limit, no missing patches will be left.
Despite many of the methods proposed in literature keep an eye on quality and completeness of the sampled model, research in this field has put the stress mainly on the generation of short plans. As a result, surface coverage is not always satisfactory. Moreover, many proposed solutions are complex and memory-consuming. On the contrary, our approach tries to maximise quality and coverage, rather than minimising plan length.

In the following we will make use of the expressions *occluding contour* and *occluding surface* (refer to Figure 3) to indicate, respectively, the border of an occluding patch, and the surface which spans between an occluded contour and its projection onto the back surface.

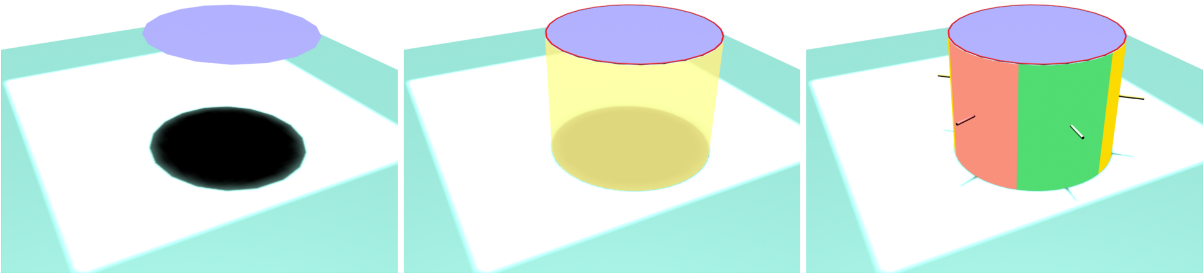


Figure 3. ***Occluding Surfaces*** **- A scan is taken from a direction orthogonal to the surface. In the left image an** ***occluding patch*** **casts a shadow onto the back surface. The corresponding** ***occluding contour*** **and** ***occluding surface*** **are shown in the middle. On the right the resulting clusters are displayed together with the associated normals.**

4. Improving the coverage factor of a single range map

The proposed approach falls in the class of algorithms with no a-priori knowledge. It attempts to build a plan to sample unseen surface by finding a small number of views covering the occluding surface. A minimum-quality constraint is enforced. This section deals with a *local* approach, which can sample holes in a single range map. It is then extended to a *global* method to scan more complex models in Section 5. Figure 4 shows the steps common to both local and global algorithms. The *hole location* phase is the only step which differs between the two algorithms.

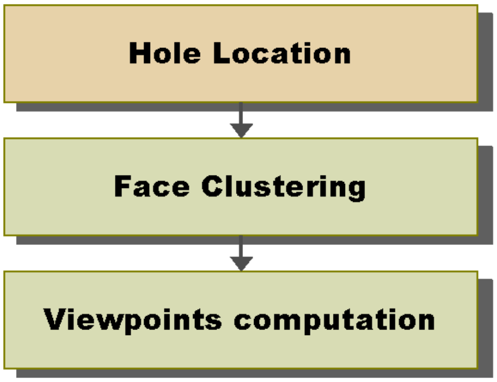


Figure 4. Pipeline of the proposed algorithm. The ***Hole Location*** **step differs between local and global solutions.**

The **local method** relies on the following observation. When a range map is acquired, the data occluded to the CCD sensor lies inside a conoid bounded by occluding contours. Thus, the occluding surface separates unseen volume from volume known to be empty. Since, we know nothing about the unseen volume except its boundary, our idea is to exploit this minimal knowledge and use occluding surfaces as a guide for planning.

In order to compute occluding surfaces, unsampled regions must be located in the model. The performance of hole location is critical for the generation of good plans. In our local method, hole location is view-dependent as it operates in $2\frac{1}{2}$D. Namely, the boundaries of occluding surfaces are detected by searching for strong *height jumps* in the range map. The range map is triangulated and triangles associated to high-quality sampled regions are filtered out, i.e. all but very skewed triangles are removed. The resulting mesh is the dual of a "good-looking" surface patch. In this way we obtain a sort of *hole map*, consisting of all the faces that are to be removed from the original mesh patch. Notice that the faces of this hole map are nearly orthogonal to the viewing direction.

The hole detection phase is preceded by a smoothing operation on the raw data, regarded as a height field. This step is fundamental in order prevent overly scattered normals in the hole surface. Since the hole surface is made up of skewed triangles the distribution of triangle normals is usually very noisy. Smoothing the raw data also attenuates noise in boundary data, which could cause artifacts in the hole surface. Smoothing is further discussed in Section 6.

Once holes have been located and a mesh representation has been computed for them, mesh faces are grouped in clusters with respect to their normals. Each cluster represents all the faces that are measurable from a single direction. A representative direction is associated to every cluster. A face belongs to a cluster if its distance from the representative direction is within a given threshold. The distance measure we employ is simply the glancing angle. The threshold value depends on a quality function of the scanning system, which is usually directly related to the glancing angle. Notice that here we are assuming that faces pointing in the same direction can be acquired in a single shot, i.e. they are inside the viewing cone of the scanner optics (Figure 2). This condition is met if we assume that the whole object lies inside the scanning volume. Anyway, it can be avoided if a constraint on maximum euclidean distance is embodied in the distance measure used for clustering. A method to account for this will be presented later in this section.

The clustering algorithm we employed is in the spirit of Octree Quantisation [9]. This is a simple method used for colour quantisation in palette-based colour images. Suppose you are given a colour table that can contain at most k entries. When a new colour is added to the table, if there is a free slot in the table then you are done. Otherwise you have to make room for the new entry. This can be done by merging some close neighbours into one cluster. A common colour is assigned to the cluster. An octree is used to represent the RGB space. Colour components are the coordinates within the octree. Exact colours are represented as leafs of maximum depth, while intermediate nodes represent clusters. The deeper the level of a node, the fewer the colours it represents. Every time the number of leaves exceeds k, some leaves are merged. In order to generate an equally distributed partition, the leaves of deepest level which represent the fewest pixels are chosen. Once every pixel has been added to the octree, the colour table is given by the leaves. The mapping from colour indices to table entries can be done simply by traversing the octree structure; when a leaf is found its colour is returned. Since unit normals have only two degrees of freedom, we implemented a sort of “quadtree quantisation” in 2D mapping azimuth and elevation angles to colours.

We chose this solution in place of more widely used clustering methods, such as k-means [23], because of its simplicity. Even if k-means might give better results, we do not need to be very accurate since our knowledge of the missing data (i.e. the occluding surfaces) is imprecise. Moreover, the gain in accuracy would be at the expenses of running-time. Face clustering algorithms have also been proposed in the computer graphics community [12, 8]. However, even if they are simple and fast as well as Octree Quantisation, they are too tied to mesh topology and can give incorrect results for our application (see Figure 5).

In order to adapt this algorithm to our purposes, we have slightly modified the termination condition and the criterion to choose the next merging. Namely, we choose the nodes to be merged taking into account not only the number of normals represented (encoded as colours), but also the glancing angle with respect to the representative view direction. Since we are interested in finding the minimum number of clusters that cover the whole normal distribution, the number of leaves k must be as little as possible. The merging is thus stopped when no more merging can be carried out without violating a constraint on the maximum glancing angle.

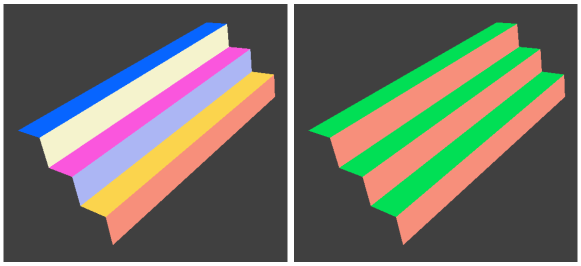


Figure 5. Face clustering methods: *topology-driven* methods (on the left) generate a cluster for each face, while *octree quantisation* correctly gives two clusters (clusters are shown in different colours - see colour plate).

Once we have clustered face normals, the directions associated to the clusters constitute a small set of viewing directions covering the holes in the current model. To compute the viewing positions, we take the centroid of the vertices belonging to each cluster and displace it along the viewing direction. The displacement is determined in order to allow every face of the cluster to lie inside the imaging cone of the scanner (Figure 2).

Now that a small number of viewpoints has been determined, their quality can be ranked with respect to the area of unknown surface viewed. This is done by summing up the area of the faces in each cluster. The viewpoint associated with a given cluster gets a rank proportional to its area. A minimum area constraint can be enforced to discard low-ranked views.

A method to get rid of the assumption that the object must lie inside the viewing cone of the scanner is as follows. For each vertex in the cluster the distance from the optical centre can be computed. A histogram of the distances is then built. By examining the distance histogram distribution the cluster is split in as many sub-clusters as needed, in order to scan each sub-cluster in a single shot. However, this method is tricky to implement and may produce some unnecessary viewpoints.

Better results can be obtained by integrating a constraint on maximum Euclidean distance in the distance measure used for clustering. We do that simply by using the full Octree Quantisation algorithm in three dimensions. Namely, when the algorithm tries to merge neighbouring nodes it must check not only their orientation but also their linear distance along the candidate viewing direction. This can be thought as constraining the clusters to lie in a volume bounded by two planes, that is exactly what we want to do.

5. Closing Gaps on a Set of Range Maps

In Section 4 we showed how to estimate a set of suitable viewing poses to cover reasonably well unseen areas of a range map, using only local information. In this section we extend this approach, to find new views which allow to improve the coverage factor of a given set of range maps. The main idea remains the same: first, we identify those areas of the intermediate reconstructed surface that do not correspond to real surface patches (or to well-sampled surface); then, we cluster the normals of these *fake* portions of mesh; finally, we produce a new set of viewpoints and directions, based on the information gathered. The main difference with respect to the local approach is the *hole location* phase. The first step is to build a closed mesh starting from a set of registered range maps. This is a well known problem, for which many solutions have been proposed [6, 20, 7, 4, 15]. For our experiment we have chosen a variant of the Curless and Levoy's volumetric reconstruction approach [6], with volumetric diffusion hole filling [7]. It is worth noting that we are not interested in obtaining a highly accurate model at this stage, but only an approximate reconstructed mesh appropriate for the detection of the un-sampled portions of the original mesh. For this reason we can reconstruct the volume with a low sampling rate (i.e. with a voxel width much larger than the range maps inter-sampling distance). Working with a low resolution voxel set allows a more efficient execution of the volumetric diffusion of the discretised distance field. Volumetric diffusion allows to close holes by expanding the distance field on the unseen region as proposed in [7]. We take care of computing a *confidence* value during the volumetric diffusion and reconstruction processes that helps us to identify the surface parcels that are originated by the field expansion from those that originate form real data. Figure 7(a) shows a model before fusion (image on the left) and the resulting hole-free mesh reconstructed at low resolution (image on the right). The colour on the reconstructed surface maps the confidence value using the usual colour ramp: red indicates to surface portions that were acquired with a low confidence, are sampled in an insufficient way, or are associated to holes.

The unsampled portion of the mesh can be simply detected by defining a threshold on the confidence value. The remainder of the algorithm proceeds as explained in Section 4: the normals of faces with an insufficient confidence value are clustered and a set of new viewpoints is generated from clusters' representatives. Figure 7(b) shows the hole map and directions resulting from clustering the low-confidence faces of the mesh shown in Figure 7(a). The final model is shown in Figure 7(c).

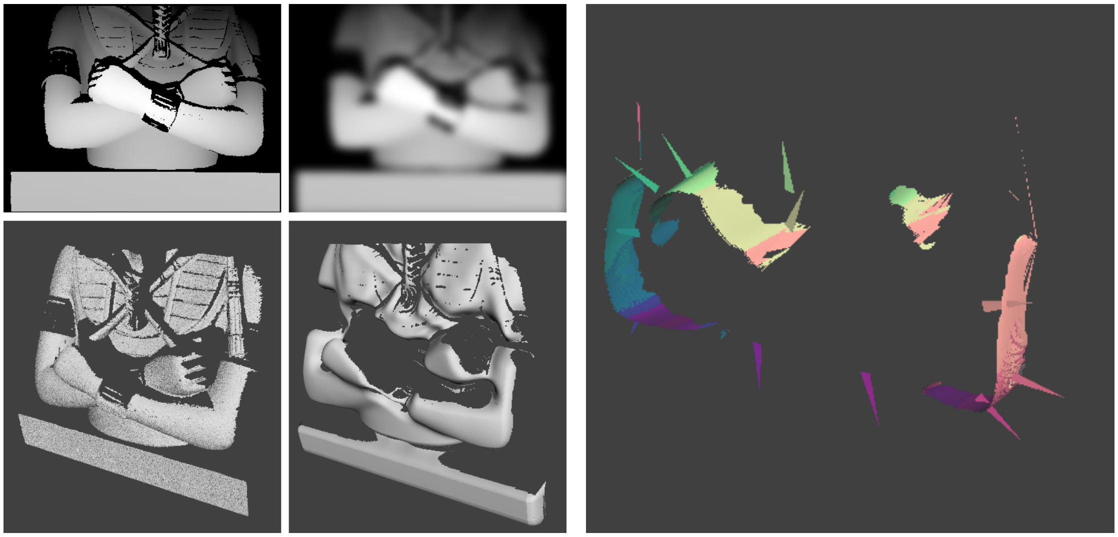
6. Experimental Results

We have tested our algorithm with both synthetic and real data. In both cases the presented method has been able to "suggest" good viewpoints to the user, thus obtaining a better sampling of the objects' surface. Figure 6 shows some snapshots of each step of the local algorithm. Runs of the global algorithm are depicted in Figure 7 and Figure 8.

As can be observed from Figure 6, some gaps were missed by the local algorithm. This depends on the smoothing phase. Namely, some holes were not detected due to the view-dependent nature of the local hole location method (i.e. only hole surfaces nearly orthogonal to the viewing direction can be detected). Moreover, as discussed in Section 4, strong height jumps in the range map are associated to hole boundaries. Smoothing range data attenuates height jumps. In order to avoid to smooth out jump edges we tried to employ an edge-preserving smoothing filter [17]. This did not work well since preserving jump edges was obtained at the expenses of an unsatisfactory removal of noise in proximity of those edges, that is exactly what we are interested in filtering out. Hence the only chance to remove noise around edges, still being able to detect holes correctly, is to adjust the threshold used to detect badly-shaped triangles with respect to the smoothing parameters. In conclusion, the hole detection step needs an accurate steering of the user and therefore makes this phase of the *local* approach not easily executable in an unattended fashion.

This problem is not present in the *global* approach, as can be observed from Figure 7. The second approach is thus a better choice to design a semi-automatic scanning system. On the other hand, this approach is slower than the local one and is also more memory-demanding. The burdensome component is the volumetric diffusion of the volumetric distance field. However since we can run it at a reduced resolution the overall timings are affordable on modern PCs. The examples reported in Figure 7 and Figure 8 require a couple of minutes, which is a fairly affordable time in the framework of a complex acquisition session. For that reason, we did not take care of keeping track of timings more precisely.

A comparison in terms of performances with other approaches is hard to carry out, since authors use different models and the meshes processed (the initial incomplete mesh and the final result) are usually unavailable. Moreover, timings and memory requirements are often not presented. We believe that our approach can run faster than many previous approaches since, for example, we do not have to fit parametric models to fairly complex meshes [26] or to compute polygonal regions as in [14]. Moreover, our approach is more memory-efficient than other approaches as we encode the un-sampled surface with just a set of normal vectors, rather than using a volumetric model [5] or any other representation involving the storage of dense matrices [18, 21].



(a) Range data and relative meshes. To the right the range data are smoothed using a gaussian filter with $\sigma = 15$. Jump edges are highly attenuated in the smoothed mesh.

(b) Hole maps: different colours refer to different representative normals (i.e. viewpoints). A spike is shown for each viewpoint computed (see colour plate).

Figure 6. Some snapshots showing each step of the local method on real data.

7. Concluding Remarks

We have presented an algorithm that exploits some properties of what we called *occluding surface* to make full six DOF sensor plans. Since a discretisation of the viewpoint space must be carried out, a good sampling has to be found in order to fulfil the scanning requirement. A-priori constraining the viewpoint space is not a good choice, since it does not take into account object's surface features and can fail to cover complex shapes. This can be easily managed if some a-priori knowledge of the object's shape is available, but it is a fairly difficult problem if only incomplete or no information at all is given. Our approach tries to guess "good" sensor poses with respect to the available information, i.e. a single scan or a set of already scanned range maps. The space of normals to the surface parcels separating void and unseen volumes (*occluding surface*) is quantised. New views covering unsampled regions are chosen by partitioning the space of normals using a quality threshold.

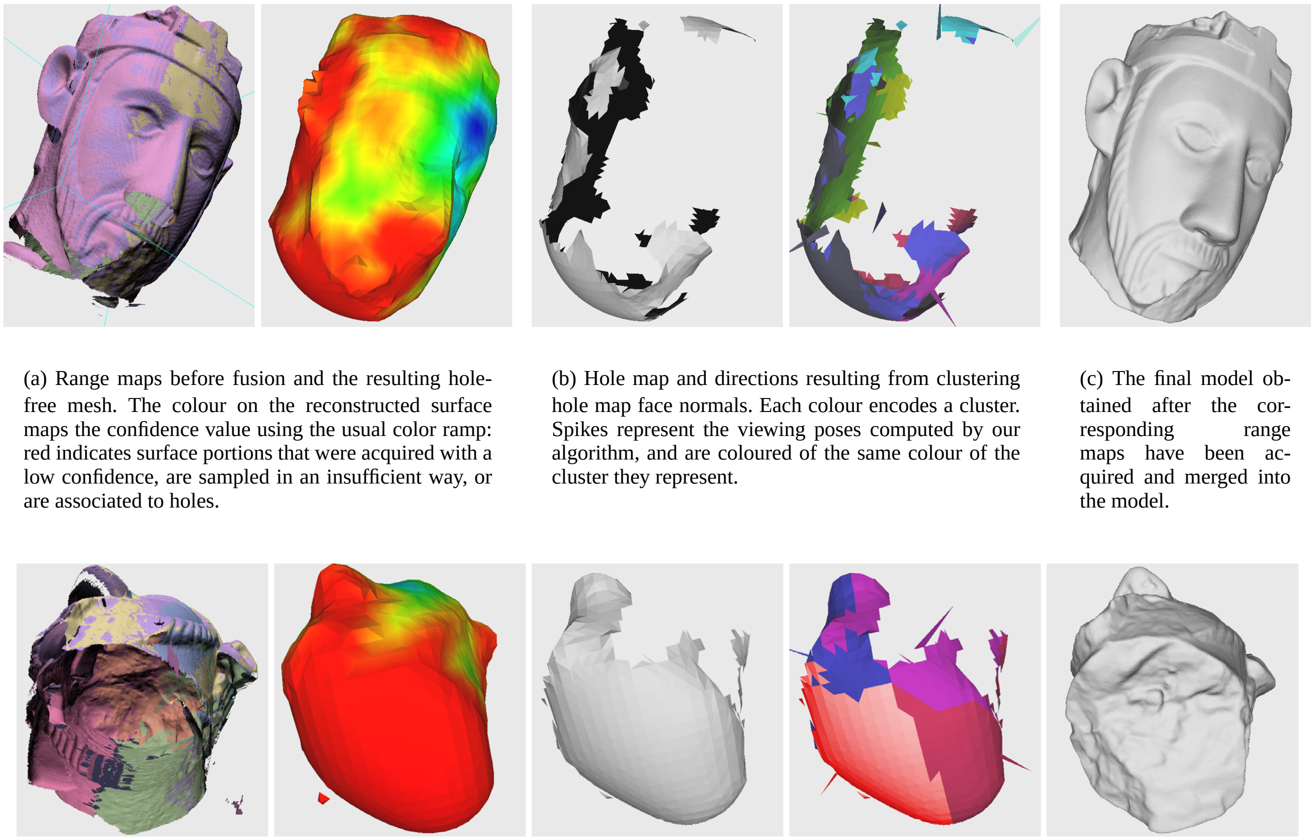
The approach presented is going to be included in the design of a semi-automatic robotised scanner (*Digital Sculptor Project*, in cooperation with Scienzia Machinale s.r.l.) [3]. The aim of this project is to automatise the scanning process using a robotic arm. The software system will support: the easy setup of any of the two scanning strategies described in the introduction (*X-Y* or *cylindrical* sweeps), the selection of further scans needed to get a satisfactory coverage factor, and finally the automatic alignment of the range maps and reconstruction of the final model. The added cost of the robotic arm is amortised by the fact that the overall system is designed to support first the semi-automatic acquisition, and then the automatic reproduction on stone or other materials (by replacing the scanner with a drilling tool, and computing the required drilling paths on the reconstructed 3D model).

The main drawback of our algorithm is that it does not take into account self-occlusions; occlusions are supposed to be generated only by protrusions and not by deep concavities. This depends on our current implementation, which does not consider self-intersection of the surface while determining the best line of sight. Although some visibility constraint could be enforced during the clustering procedure (e.g. adding some line-of-sight occlusion detection based on ray-tracing), we were not yet able to find a good way to do so without undermining the simplicity of our method. It is a topic for future research.

We are also experimenting a GPU-based approach which takes advantage of common rendering techniques in order to evaluate the quality of candidate viewpoints [3].

Acknowledgements

This research was supported by the projects EU IST-2001-32641 "ViHAP3D" and FIRB - "MACROGeo".



(a) Range maps before fusion and the resulting hole-free mesh. The colour on the reconstructed surface maps the confidence value using the usual color ramp: red indicates surface portions that were acquired with a low confidence, are sampled in an insufficient way, or are associated to holes.

(b) Hole map and directions resulting from clustering hole map face normals. Each colour encodes a cluster. Spikes represent the viewing poses computed by our algorithm, and are coloured of the same colour of the cluster they represent.

(c) The final model obtained after the corresponding range maps have been acquired and merged into the model.

(d) Some snapshots showing the same model viewed from the bottom. A large hole has been detected and split into a number of distinct clusters. The computed views are shown as spikes. The final model is also shown.

Figure 7. Snapshots showing each step of the global method. This model was acquired by means of a rotating platform (see colour plate).

References

[1] F. Bernardini and H. E. Rushmeier. 3D Model Acquisition. In *Eurographics 2000, State of the Art Reports Proceedings*, pages 41–62. Eurographics Association, Aug. 24–25 2000.

[2] M. Callieri, P. Cignoni, F. Ganovelli, C. Montani, P. Pingi, and R. Scopigno. Vclab's tools for 3d range data processing. In A. C. D. Arnold and F. Niccolucci, editors, *VAST 2003 and EG Symp. on Graphics and Cultural Heritage*, page (in press), Bighton, UK, Nov. 5-7 2003. Eurographics.

[3] M. Callieri, A. Fasano, G. Impoco, P. Cignoni, R. Scopigno, G. Parrini, and G. Biagini. Roboscan: an automatic system for accurate and unattended 3d scanning. Technical report, ISTI-CNR, 2004 (submitted paper).

[4] J. C. Carr, R. K. Beatson, J. B. Cherrie, T. J. Mitchell, W. R. Fright, B. C. McCallum, and T. R. Evans. Reconstruction and representation of 3D objects with radial basis functions. In *SIGGRAPH 2001, Computer Graphics Proceedings*, Annual Conference Series, pages 67–76. ACM Press / ACM SIGGRAPH, 2001.

[5] C. Connolly. The determination of next best views. In *CRA85*, pages 432–435, 1985.

[6] B. Curless and M. Levoy. A volumetric method for building complex models from range images. In *Comp. Graph. Proc., Annual Conf. Series (SIGGRAPH 96)*, pages 303–312. ACM Press, 1996.

[7] J. Davis, S. R. Marschner, M. Garr, and M. Levoy. Filling holes in complex surfaces using volumetric diffusion. In *First International Symposium on 3D Data Processing, Visualization, and Transmission, Padua, Italy*, June 2002.

[8] M. Garland, A. Willmott, and P. Heckbert. Hierarchical face clustering on polygonal surfaces. In *Proceedings of the ACM Symposium on Interactive 3D Graphics*, pages 49–58. ACM Press, 2001.

[9] M. Gervautz and W. Purgathofer. A Simple Method for Color Quantization: Octree Quantization. In N. Magnenat-

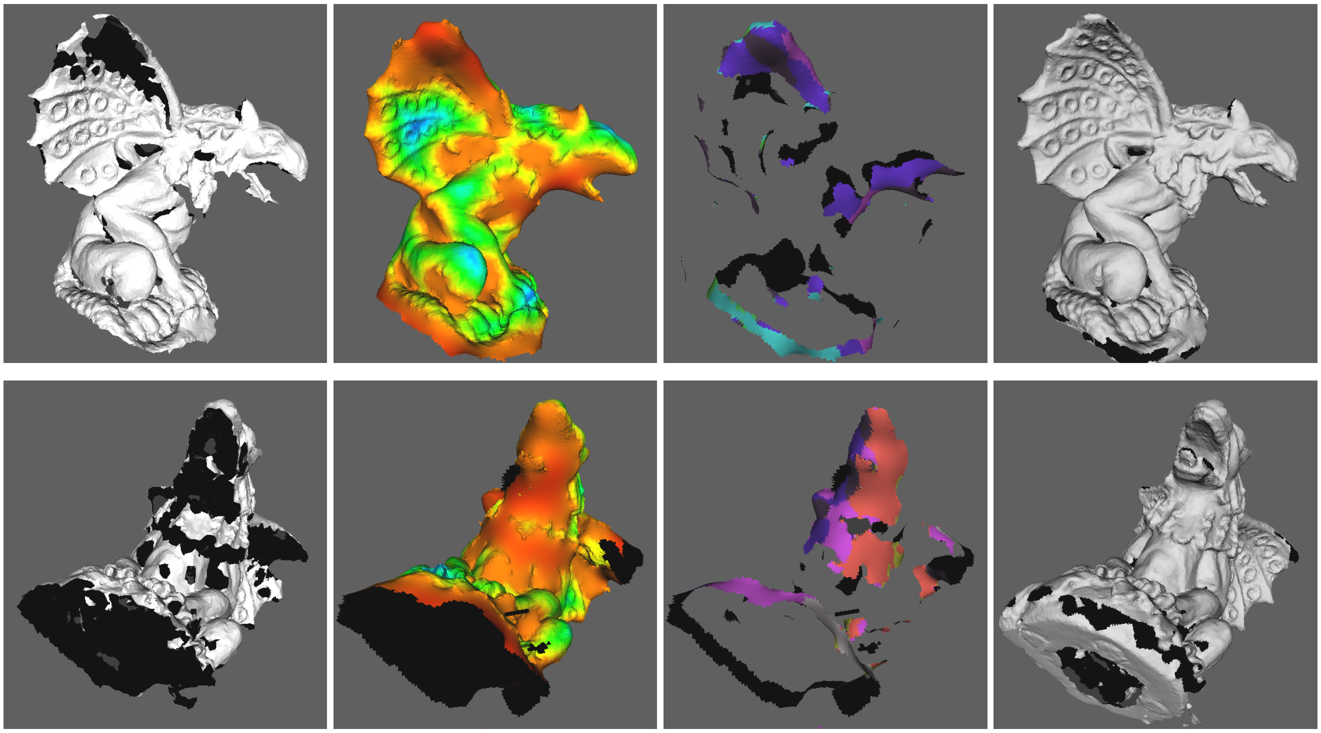


Figure 8. Gap closing of the *Gargoyle* model (see colour plate). One more iteration of the proposed algorithm is needed to cover the remaining holes.

Thalmann and D. Thalmann, editors, *New Trends in Computer Graphics*, pages 219–231. Springer-Verlag, New York, NY, 1988.

[10] W. Grimson. Sensing strategies for disambiguating among multiple objects in known poses. In *MIT AI Memo*, 1985.

[11] S. Hutchinson and A. Kak. Planning sensing strategies in robot work cell with multi-sensor capabilities. *RA*, 5(6):765–783, December 1989.

[12] A. D. Kalvin and R. Taylor. Superfaces: Poligonal mesh simplification with bounded error. *IEEE C.G.&A.*, 16(3):64–77, 1996.

[13] M. Levoy, K. Pulli, B. Curless, S. Rusinkiewicz, D. Koller, L. Pereira, M. Ginzton, S. Anderson, J. Davis, J. Ginsberg, J. Shade, and D. Fulk. The Digital Michelangelo Project: 3D scanning of large statues. In *SIGGRAPH 2000, Computer Graphics Proceedings*, Annual Conference Series, pages 131–144. Addison Wesley, July 24-28 2000.

[14] J. Maver and R. Bajcsy. Occlusions as a guide for planning the next view. *PAMI*, 15(5):417–433, May 1993.

[15] Y. Ohtake, A. Belyaev, M. Alexa, G. Turk, and H.-P. Seidel. Multi-level partition of unity implicits. *ACM Transactions on Graphics*, 22(3):463–470, July 2003.

[16] A. Papadopoulos-Orfanos and F. Schmitt. Automatic 3d digitization using a laser rangefinder with a small field of view. In *3DIM97*, pages 3 – View Planning, 1997.

[17] P. Perona and J. Malik. Scale space and edge detection using anisotropic diffusion. *PAMI*, 12(7):629–639, July 1990.

[18] R. Pito. A solution to the next best view problem for automated surface acquisition. *PAMI*, 21(10):1016–1030, October 1999.

[19] M. Reed and P. Allen. Constraint-based sensor planning for scene modeling. *PAMI*, 22(12):1460–1467, December 2000.

[20] R. Sagawa, K. Nishino, and K. Ikeuchi. Robust and adaptive integration of multiple range images with photometric attributes. In *2001 Conference on Computer Vision and Pattern Recognition (CVPR 2001)*, pages 172–179, dec 2001.

[21] W. Scott, G. Roth, and J.-F. Rivest. Performance-oriented view planning for automatic model acquisition. In *Proceedings of the 31st International Symposium on Robotics*, pages 314–319, 2000.

[22] W. R. Scott, G. Roth, and J.-F. Rivest. View planning for automated three-dimensional object reconstruction and inspection. *ACM Computing Surveys (CSUR)*, 35(1):64–96, 2003.

[23] H. Späth. *Cluster analysis algorithms*. Ellis Horwood, Chichester, UK, 1980.

[24] K. Tarabanis, P. Allen, and R. Tsai. A survey of sensor planning in computer vision. *RA*, 11(1):86–104, February 1995.

[25] G. Tarbox and S. Gottschlich. Planning for complete sensor coverage in inspection. *CVIU*, 61(1):84–111, January 1995.

[26] P. Whaite and F. Ferrie. Autonomous exploration: Driven by uncertainty. In *McGill*, 1994.