DOCUMENTATION OF ARTIFACTS
Using of Laser and Digital Camera Based Systems for 3D Object Documentation  

K. Pavelka  

Volume Based Reconstruction of Archaeological Artifacts  

M. Kampel, R. Sablatnig, S. Tosovic  

Surface Model Generation by the Relics from Slice Images, and the Trial to the Automatic Restoration  

Y. Watanabe, K. Tanaka, N. Abe, H. Taki, Y. Kinoshita
USING OF LASER AND DIGITAL CAMERA BASED SYSTEMS FOR 3D OBJECT DOCUMENTATION

K. Pavelka

Czech Technical University in Prague, Faculty of Civil Engineering, Dept. Mapping and Cartography,
Laboratory of Photogrammetry, Thákurova 7, 166 09, Prague 6, Czech Republic
Tel.: +42024354951, Fax: +42024310774, E-mail: pavelka@fsv.cvut.cz

Commission CIPA, WG 6

KEY WORDS: digital photogrammetry, 3D documentation, laser scanning

ABSTRACT:

Examples of systems based on simple digital camera and laser are discussed in the paper. At the Czech Technical University in Prague, photogrammetric measurement is performed in the Laboratory of Photogrammetry (Faculty of Civil Engineering). Based on a grant of the Czech Grant Agency, a large project on historical monument documentation and presentation by using a new digital technology has been started. In the framework of co-operation between the Laboratory of Photogrammetry and the Laboratory of Quantitative Methods of Monuments Research (Faculty of Nuclear Physics and Physical Engineering), new methods of 3D objects documentation as a part of project are tested. Two devices for 3D object co-ordinates capturing are being developed at present. The first device uses a rotating platform developed for small objects, a laser for point or profile marking (on the object) and a digital camera (for image sequence saving) on theodolite. Such a 3D scanner can be used for small compact objects, such as small sculptures, vessels, models and so on. The second device uses rotating stable base equipped with a digital camera and a laser for point marking. This type is suitable for profiling of tunnels for example. The expected outputs are not only the 3D co-ordinates of the object, but also the experience with a new technology based on using laser and automatic image processing.

1. NEW DEVICES FOR 3D OBJECT MEASURING

In the framework of co-operation between the Laboratory of Photogrammetry and the Laboratory of Quantitative Methods of Monuments Research (Faculty of Nuclear Physics and Physical Engineering), new methods of 3D objects documentation as a part of project are tested. Two devices for 3D object co-ordinates capturing are being developed at present. The aim of the project is to develop a small inexpensive device for special purposes of 3D documentation. By combining several electronic parts such as CCD camera, laser marker, computer and distance measuring device a new laser sensor has been developed. There are only few possibilities how to construct laser based 3D sensors. The principle of these devices is the same: the laser beam is used as an object point marker (single point or line on object) and the laser track is recorded by using of a small CCD camera. The camera and laser position are convergent to the object, 3D co-ordinates can be computed from laser-camera basis.

2. LASER SYSTEM WITH ROTATING PLATFORM

For small objects such as small sculptures, vessels or models a system with rotating platform has been constructed. A laser beam optically modified to a thin line on the object is recorded from a basis with CCD camera. A maximum of 25 frames per second can be used. The measured object is situated on a rotating platform with a possibility to change the rotating velocity. All the images are stored on a PC and processed by using of special software. From the image coordinates of marked object points the real 3D coordinates are computed. The scanning process is demonstrated on Fig.1.

---

Fig.1: Rotating platform system
3. LASER SYSTEM WITH ROTATING LASER

For the second system, a new setting of the elements has been developed. There is a rotating base with a convergent laser marker and a CCD camera. This model is used for making profiles for example in tunnels. The frame CCD camera is connected with a notebook and the images are post-processed by using special software. The centre of laser track on the images is detected with a sub-pixel resolution and the centre of laser trace (in image co-ordinates) represents a horizontal parallax. The first image with a laser trace is used as a base measurement and, however, the distance $y_p$ between the base and the object is known by using self-reflecting distance meter. For each image the rotating angle is recorded. Further, the distances to the object point are computed from parallax and the final 3D co-ordinates are determined from rotating angle. The system is fixed on a platform and the platform position must be observed by using a total station. For this reason three reflecting prisms are added to the platform. The scheme is illustrated on fig.2. From a technical reasons it is better, when the camera is stationary and the rotating device is equipped by a prism.

Mathematically, the method is based on measurement of horizontal parallax of laser track centre. The first image is used as a base measurement. The difference between a laser track centre on the first image and the next images gives the parallax. The $b$ is the known base distance and it is know, $y_p$ must be measured at the beginning of the experiment. The camera axis is perpendicular to the base. In this case we can use an equation for normal case of terrestrial photogrammetry. For this method the relation to terrestrial photogrammetry is evident.

\[
\frac{y_0}{b} = \frac{dy}{p}, \quad dy = \frac{y_0}{b} p \quad (1)
\]

From equation (1) it is clear, that it is not necessary to know the camera constant. Nevertheless, for output precision reason it is recommended to use an objective with maximum focus distance. The precision of this can be obtained by derivation of (1):

\[
dy = \frac{Y_0}{b} dp + \frac{p}{b} dy_0 - \frac{y_0 p}{b^2} db \quad (2)
\]

The precision is given by the element $y_0 / b$. For example by using a CCD with the resolution 640x480 pixels, the object distance about 4m and basis 40cm, the precision in $dy$ is better than 10mm.

Fig.2: Theodolite with CCD camera

Fig.3: Laser

Fig.4: Optical device (for line track)

Fig.5: Laser point track on the wall
4. CONCLUSION

Examples of inexpensive systems based on simple digital camera and laser are discussed in the paper. Both systems are under construction on Czech Technical University in Prague and they are used for the technology testing.

5. REFERENCES


REFERENCES FROM WEBSITES:

Laboratory of Photogrammetry, 2002, CTU Prague Web Site: http://gama.fsv.cvut.cz/k153/

6. ACKNOWLEDGEMENTS

This project is sponsored by grant Czech Grant Agency Nr.205/00/1360 and 103/02/0357.
VOLUME BASED RECONSTRUCTION OF ARCHAEOLOGICAL ARTIFACTS

Martin KAMPFL, Robert SABLATNIG, Srdjan TOSOVIC

Vienna University of Technology, Institute of Computer Aided Automation, Pattern Recognition and Image Processing Group Favoritenstr. 9, 183-2, A-1040, Vienna, Austria, {kampel,sab}@prip.tuwien.ac.at

KEY WORDS: 3D, scanning, close range, archaeology, surface models.

ABSTRACT:

An algorithm for the automatic construction of a 3d model of archaeological vessels using two different 3d algorithms is presented. In archeology the determination of the exact volume of arbitrary vessels is of importance since this provides information about the manufacturer and the usage of the vessel. To acquire the 3d shape of objects with handles is complicated, since occlusions of the object's surface are introduced by the handle and can only be resolved by taking multiple views. Therefore, the 3d reconstruction is based on a sequence of images of the object taken from different viewpoints with different algorithms; shape from silhouette and shape from structured light. The output of both algorithms are then used to construct a single 3d model. Images for both algorithms are acquired by rotating the object on a turntable in front of a stationary camera. Then an octree representation of the object is built incrementally, by performing limited processing of all input images for each level of the octree. Beginning from the root node at the level 0 a rough model of the object is obtained quickly and is refined as the processed level of the octree increases. Results of the algorithm developed are presented for both synthetic and real input images.

1. INTRODUCTION

The combination of the Shape from Silhouette method with the Shape from Structured Light method presented in this paper was performed within the Computer Aided Classification of Ceramics [11,6] project, which aims to provide an objective and automated method for classification and reconstruction of archaeological pottery. The final goal is to provide a tool which helps archaeologists in their classification process.

Pottery was made in a very wide range of forms and shapes. The purpose of classification is to get a systematic view of the material found, to recognize types, and to add labels for additional information as a measure of quantity [15]. In this context, decoration of pottery is of great interest. Decoration is difficult to illustrate since it is a perspective projection of an originally spherical surface. This fact induces distortions that can be minimized by 'unwrapping' the surface. In order to be able to unwrap the surface it is necessary to have a 3d representation of the original surface. Furthermore, the exact volume of the vessel is a great interest to archaeologists too, since the volume estimation allows also a more precise classification [15]. Since pottery is manufactured on a turntable we use a turntable based method for the 3d-reconstruction of the original. To acquire images from multiple views we put the archaeological vessel on a turntable which rotates in front of a stationary camera.

Shape from Silhouette is a method of automatic construction of a 3D model of an object based on a sequence of images of the object taken from multiple views, in which the object's silhouette represents the only interesting feature of the image [19,16]. The object's silhouette in each input image corresponds to a conic volume in the object real-world space. A 3D model of the object can be built by intersecting the conic volumes from all views, which is also called Space Carving [8].

Shape from Silhouette is a computationally simple algorithm (it employs only basic matrix operations for all transformations). It can be applied on objects of arbitrary shapes, including objects with certain concavities (like a handle of a cup), as long as the concavities are visible from at least one input view. This condition is very hard to hold since most of the archaeological vessels do have concavities (like a cup for instance) that have to be modeled. Therefore a second, active shape determination method has to be used to discover all concavities. The acquisition method used for estimating the 3d-shape of objects is shape from structured light, based on active triangulation [2]. Both methods requires only a camera and illumination devices as equipment, so they can be used to obtain a quick initial model of an object which can then be refined stepwise.

There have been many works on construction of 3D models of objects from multiple views. Baker [1] used silhouettes of an object rotating on a turntable to construct a wire-frame model of the object. Martin and Aggarwal [10] constructed volume segment models from orthographic projection of silhouettes. Chien and Aggarwal [4] constructed an object's octree model from its three orthographic projections. Veenstra and Ahuja [22] extended this approach to thirteen standard orthographic views. Potmesil [16] created octree models using arbitrary views and perspective projection. For each of the views he constructs an octree representing the corresponding conic volume and then intersects all octrees. In contrast to this, Szeliski [19] first creates a low resolution octree model quickly and then refines this model iteratively, by intersecting each new silhouette with the already existing model. The last two approaches project an octree node into the image plane to perform the intersection between the octree node and the object's silhouette. Srivastava and Ahuja [18] in contrast, perform the intersections in 3D-space. The work of Szeliski [19] and Niem [13] were used as a basis for the shape from silhouette approach presented in this paper. For the active triangulation method we use an approach bis Liska developed for a next view planning strategy using structured light [9].

The paper is organized as follows. Section 2 describes the equipment used. Section 3 describes the strategy for and
Section 4 presents experimental results. Finally, in Section 5 conclusions are drawn and future work is outlined.

2. ACQUISITION SYSTEM

The acquisition system (Figure 1) consists of the following devices:

- a turntable (Figure 1a) with a diameter of 50 cm, whose desired position can be specified with an accuracy of 0.05°. The turntable is used to obtain multiple views of the object observed.
- two monochrome CCD-cameras (Figure 1b and 1c) with a focal length of 16 mm and a resolution of 768x576 pixels. One camera (Camera-1 in Figure 1) is used for acquiring the images of the object's silhouettes and the other (Camera-2 in Figure 1) for the acquisition of the images of the laser light projected onto the object.
- a laser (Figure 1d) used to project a light plane onto the object. The laser is equipped with a prism in order to span a plane out of the laser beam. The color of the projected light is red.
- a lamp (Figure 1e) used to illuminate the scene for the acquisition of the silhouette of the object. The object should be clearly distinguishable from the background independent from the object's shape or the type of its surface. For that reason back-lighting [5] is used. A large (approx. 50x40 cm) rectangular lamp is put behind the turntable (as seen from the camera). In addition, a white piece of paper, larger than the lamp, is put right in front of the lamp, in order to make the light more diffuse.

The whole system is protected against the ambient light by a thick black curtain.

![Acquisition System](image)

Figure 1. Acquisition System

The geometrical setup of the acquisition devices is shown in Figure 2. Both cameras are placed about 50 cm away from the rotational axis of the turntable. Ideally the optical axis of the camera for acquiring object's silhouettes (Camera-1 in Figure 2) lies nearly in the rotational plane of the turntable, orthogonal to the rotational axis. The camera for acquiring the projection of the laser plane (Camera-2 in Figure 2) onto the object views the turntable from an angle of about 45° (α in Figure 2). The laser is directed such that the light plane it projects contains the rotational axis of the turntable. Camera-2 in Figure 2 views the light plane also from an angle of about 45° (α in Figure 2). The relative position of the two cameras to one another is not important, since the acquisition of the silhouettes and the acquisition of the laser light projection are independent from one another.

![Geometrical setup of acquisition system](image)

Figure 2. Geometrical setup of acquisition system

Prior to any acquisition, the system is calibrated in order to determine the inner and outer orientation of the camera and the rotational axis of the turntable. The calibration method used was exclusively developed for the Shape from Silhouette algorithm presented and it is described in detail in [20] and [7,21].

3. MODEL REPRESENTATION

There are many different model representations in computer vision and computer graphics used. Here we will mention only the most important ones. Surface-based representations describe the surface of an object as a set of simple approximating patches, like planar or quadric patches [12]. Generalized cylinder representation [17] defines a volume by a curved axis and a cross-section function at each point of the axis. Overlapping sphere representation [14] describes a volume as a set of arbitrarily located and sized spheres. Approaches such as these are efficient in representing a specific set of shapes but they are not flexible enough to describe arbitrary solid objects.

Two of the most commonly used representations for solid volumes are boundary representation (B-Rep) and constructive solid geometry (CSG) [17]. The B-Rep method describes an object as a volume enclosed by a set of surface elements, typically sections of planes and quadratic surfaces such as spheres, cylinders and cones. The CSG method uses volume elements rather than surface elements to describe an object. Typical volume elements are blocks, spheres, cylinders, cones and prisms. These elements are combined by set operations into the modeled object. The B-Rep and CSG method suffer from quadratic growth of elemental operations as the complexity of the modeled object increases.

An octree [3] is a tree-formed data structure used to represent 3-dimensional objects. Each node of an octree represents a cube subset of a 3-dimensional volume. A node of an octree which represents a 3D object is said to be:

- black, if the corresponding cube lies completely within the object
- white, if the corresponding cube lies completely within the background, i.e., has no intersection with the object
• gray, if the corresponding cube is a boundary cube, i.e., belongs partly to the object and partly to the background. In this case the node is divided into 8 child nodes (octants) representing 8 equally sized sub-cubes of the original cube.

All leaf nodes are either black or white and all intermediate nodes are gray. An example of a simple 3D object and the corresponding octree is shown in Figure 3.

This octree contains binary information in the leaf nodes and therefore it is called a binary octree and it is suitable for representation of 3D objects where the shape of the object is the only object property that needs to be modeled by the octree. Non-binary octrees can contain other information in the leaf nodes, e.g., the cube color in RGB-space. For the Shape from Silhouette algorithm presented, a binary octree model is sufficient to represent 3D objects, and in the remainder of this paper the term octree will always refer to a binary octree.

![Figure 3. A simple object (a) and the corresponding octree (b)](image)

3.1 Construction of an octree

The algorithm builds up a 3D model of an object in the following way: first, all input images are transformed into binary images where a "black" pixel belongs to the object observed and a "white" one to the background (Figure 4a). In the implementation, black means background and white means object, but it is more intuitive to describe an object pixel as "black" and a background pixel as "white". Then, the initial octree is created with a single root node (Figure 4b) representing the whole object space, which will be "carved out" corresponding to the shape of the object observed.

![Figure 4. Algorithm overview](image)

Then, the octree is processed in a level-by-level manner: starting from level 0 (with root node as the only node), all octree nodes of the current level marked as "black", i.e., belonging to the object, are projected into the first input image (Figure 4c) and tested for intersection with the object's image silhouette. Depending on the result of the intersection test, a node can remain to be "black", it can be marked as "white" (belonging to the background) or in case it belongs partly to the object and partly to the background, it is marked as "grey" and divided into 8 black child nodes of the next higher level (Figure 4d). The remainder of the black nodes of the current level is then projected into the next input image where the procedure of intersection testing with the object's silhouette is repeated. Once all input images have been processed for the current octree level, the current level is incremented by one and the whole procedure (starting from the projection of the black nodes of the current level into the first input image) is repeated, until the maximal octree level has been reached. The remaining octree after the processing of the last level is the final 3D model of the object (Figure 4e).

We define the following coordinate systems relevant for the projection of an octree node into the image plane:

- **octree coordinate system**: rooted at the intersection of the rotational axis of the turntable and its rotational plane. For the first input image, it is identical to the object coordinate system and it rotates with the object observed.
- **object coordinate system**: rooted at the same point as the octree coordinate system, but it is static, i.e., it doesn't rotate with the object. The y axis is the rotation axis of the turntable and the z axis is orthogonal to the image plane.
- **image coordinate system**: lies in the image plane and it is rooted at the image center point.

Figure 5 depicts these coordinate systems and their relative positions to one another.

![Figure 5. The octree-, object- and image coordinate system](image)

3.2 Determination of Octree node/s

An octree node is projected into the image plane in the following way: as a preprocessing step, the translation and rotation matrices and are multiplied for all possible view angles α, and the resulting matrices of these multiplications are stored in a lookup table. This is done before any processing of octree nodes starts. Once it starts, all vertices of the current node are projected into the image plane by multiplying their octree coordinates with the matrix from the lookup table corresponding to the current view angle and then multiplying the result with the appropriate scaling matrix. The result of the projection of an octree node into the image plane are image coordinates of all of the vertices of the node's corresponding cube. In the general case, the projection of a node looks like a hexagon, as depicted in Figure 6(a). To find the hexagon corresponding to the eight projected vertices is a costly task, because it requires to deter-
mine which points are inside and which outside the hexagon, and there can be hundreds of thousands of octree nodes that need to be processed. It is much simpler (and therefore faster) to compare the bounding box of the eight points. Figure 6 shows a projected octree node and the corresponding bounding box.

![Figure 6. Projection of a node (a) and its bounding box (b)](image)

The bounding box is tested for intersection with the object's silhouette in the current input (binary) image. All image pixels within the bounding box are checked for their color, whether they are black or white. The output of the intersection testing procedure is percentage of the black pixels of the bounding box, i.e., the percentage of pixels belonging to the object. If this percentage is equal or higher than a user definable threshold for black nodes, the node is marked as black. If the percentage is smaller than or equal with a user definable threshold for white nodes, the node is marked as white. Otherwise, the node is marked as gray and it is divided into eight child nodes representing eight sub-cubes of finer resolution.

The calculated image coordinates of the cube's vertices can lie between two image pixels, and a pixel is the smallest testable unit for intersection testing. Which pixels are considered to be "within" the bounding box? Figure 7 illustrates our answer to this question. We decided to test only pixels that lie completely within the bounding box (Figure 7a), because that way the number of pixels that need to be tested is smaller than testing all pixels that are at least partly covered by the bounding box. The pixels at the border of the bounding box are excluded, because most of them do not lie within the hexagon approximated by the bounding box. In the special case if there are no pixels that lie completely within the bounding box (Figure 7b) the pixel closest to the center of the bounding box is checked for the color.

![Figure 7: Selection of pixels for the intersection test](image)

The octree representation has several advantages [3]: for a typical solid object it is an efficient representation, because of a large degree of coherence between neighboring volume elements (voxels), which means that a large piece of an object can be represented by a single octree node. Another advantage is the ease of performing geometrical transformations on a node, because they only need to be performed on the node's vertices. The disadvantage of octree models is that they digitize the space by representing it through cubes whose resolution depend on the maximal octree depth and therefore cannot have smooth surfaces.

4. COMBINATION OF ALGORITHMS

An input image for Shape from Silhouette defines a conic volume in space which contains the object to be modeled (Figure 8a). Another input image taken from a different view defines another conic volume containing the object (Figure 8b). Intersection of the two conic volumes narrows down the space the object can possibly occupy (Figure 8c). With an increasing number of views the intersection of all conic volumes approximates the actual volume occupied by the object better and better, converging to the 3D visual hull of the object. Therefore by its nature Shape from Silhouette defines a volumetric model of an object.

![Figure 8. Two conic volumes and their intersection](image)

An input image for Shape from Structured Light using laser light defines solely the points on the surface of the object which intersect the laser plane (Figure 9a). Using multiple views provides us with a cloud of points belonging to the object surface (Figure 9b), i.e. with the surface model of the object.

![Figure 9. Laser projection and cloud of points](image)

The main problem that needs to be addressed in an attempt to combine these two methods is how to adapt the two representations to one another, i.e. how to build a common 3D model representation. This can be done in several ways:

- Build the Shape from Silhouette's volumetric model and the Shape from Structured Light's surface model independently from one another. Then, either convert the volumetric model to a surface model and use the intersection of the two surface models as the final representation or convert the surface model to a volumetric model and use the intersection of the two volumetric models as the final representation.
- Use a common 3D model representation from the ground up, avoiding any model conversions. That means either
design a volume based Shape from Structured Light algorithm or a surface based Shape from Silhouette algorithm.

With the former method both underlying algorithms would build their "native" model of the object. However, conversion and intersection of the models would not be a simple task. While conversion of the Shape from Silhouette's volumetric model to a surface model is straightforward --- one only has to find 3D points of the volume belonging to the surface --- an intersection of two surface models can be rather complex. One could start from the points obtained by Shape from Structured Light (because they really lie on the object's surface, whereas points on the surface of the volume obtained by Shape from Silhouette only lie somewhere on the object's visual hull) and fill up the missing surface points with points from the Shape from Silhouette model.

There are several problems with this approach. There could be many "jumps" on the object surface, because the points taken from the Shape from Silhouette model might be relatively far away from the actual surface. The approach would also not be very efficient, because we would need to build a complete volumetric model through Shape from Silhouette, then intersect it with every laser plane used for Shape from Structured Light in order to create a surface model, and then, if we also want to compute the volume of the object, we would have to convert the final surface model back to the volumetric model.

Another possibility would be converting the surface model obtained by Shape from Structured Light to a volumetric model and intersect it with the Shape from Silhouette's model. In this case the intersection is the easier part - for each voxel of the space observed one would only have to look up whether both models "agree" that the voxel belongs to the object - only such voxels would be kept in the final model and all others defined as background. Also the volume computation is simple in this case - it is a multiplication of the number of voxels in the final model with the volume of a single voxel. But the problem with this approach is the conversion of the Shape from Structured Light's surface model to a volumetric model - in most cases, the surface model obtained using laser plane is very incomplete (see the model of an amphora in Figure 9(b) because of the light and camera occlusions (Figure 10), so one would have to decide how to handle the missing parts of the surface.

When building a 3D volumetric model of an object based on a number of its 2D images, there are two possibilities regarding the decision whether a certain voxel is a part of the object or belongs to the background. Therefore, our approach proposes building a single volumetric model from the ground up, using both underlying methods in each step (illustrated in Figure 11):

1. Binarize the acquired images for both Shape from Silhouette and Shape from Structured Light in such a way that the white image pixels possibly belong to the object and the black pixels for sure belong to the background. This is shown in Figure 11a.

2. Build the initial octree, containing one single root node marked "black". (Figure 11b). This node is said to be at the level 0. Set the current level to 0.

3. All black nodes of the current level are assumed to be in a linked list. Set the current node to the first node in the list. If there are no nodes in the current level, the final model has been build so jump to Step 8. Otherwise, continue with Step 4.

4. Project the current node current level into all Shape from Silhouette binarized input images and intersect it with the image silhouettes of the object (by simply counting the percentage of white pixels within the projection of the node). As the result of the intersection the node can remain "black" (if it lies within the object) or be set to "white" (it lies outside the object) or "grey" (it lies partly within and partly outside the object). This is illustrated in Figure 11c. If at least one image says "this node is white", it is set to white. Otherwise, if at least one image says "this node is grey", it is set to grey and only if all images agree that the node is black, it stays black.

5. If the current node after Step 4 is not white, it is projected into two binarized Shape from Structured Light images representing two nearest laser planes to the node - one plane is the nearest among the images acquired with the turntable's rotation angle between 0° and 180° and the other the nearest among images taken with the angle between 180° and 360°. The separation into 2 intervals is done because if we use a single nearest plane, it could happen that the projection of the node lies completely in the occluded part of the image. Two nearest planes defined this way are almost identical, because they both contain the rotational axis of the turntable (because of the way we set the laser plane, see Figure 2, so if the nearest plane in the range 0° - 180° was with the angle α, then the nearest plane in the range 180° - 360° will be with the angle α +
180°. This way we increase the chance that the node does not lie in the occluded area in at least one of the planes. The projected node is now intersected with both images in the same way as in Step 4 (Figure 11c). If at least one image says “this node is white”, it is set to white. Otherwise, if at least one image says “this node is grey”, it is set to grey and only if both images agree that the node is black, it stays black. The intersection with the object in the image is performed in the same way as the intersection of a node with object’s silhouettes in Shape from Silhouette input images.

6. If the node is set to grey it is divided into 8 child nodes of the current level + 1, all of which are marked “black”

7. Processing of the current node is finished. If there are more nodes in the current level set the current node to the next node and go back to Step 4. If all nodes of the current level have been processed, increment the current level and go to Step 3.

8. The final octree model has been built (Figure 11d).

![Shape from Silhouette](image1)

![Shape from Structured Light](image2)

(a) Binarization of input images
(b) Initial octree
(c) Intersection testing
(d) Final model

Figure 11. Algorithm overview

5. RESULTS

For tests with synthetic objects we can build a model of a virtual camera and laser and create input images in such way that the images fit perfectly into the camera model. This way we can analyze the accuracy of the constructed models without impact of camera calibration errors. The parameters and the position of the camera and the laser are arbitrary, so we choose realistic values. We assume having a virtual camera with focal length $f = 20 \text{ mm}$, placed on the $y$ axis of the world coordinate system, 2000 mm away from its origin (Figure 12). We set the distance between two sensor elements of the camera to $d = 0.01 \text{ mm}$. The laser is located on the $z$ axis of the world coordinate system, 850 mm away from its origin, and the turntable 250 mm below the $x$-$y$ plane of the world coordinate system, with its rotational axis identical to the $z$ world axis, as shown in Figure 12.

We build input images with size 640 x 480 pixels, in which 1 pixel corresponds to 1 mm in the $x$-$z$ plane of the world coordinate system.

Having built the camera model and the input images we can test our 3D modeling algorithms with varying modeling parameters. As the measure of the accuracy of the models we compare the size (width, height and length) and the volume of the model with the size and the analytical volume of the object.

5.1 Synthetic object

As the synthetic object we create a sphere with radius $r = 200 \text{ mm}$, shown in Figure 13a. If we place the center of the sphere in the origin of the world coordinate system (see Figure 12), the sphere will look the same from all possible views. For our virtual acquisition system we can assume having neither camera nor light occlusions and we can construct perfect input images of the sphere (Figure 13b and c) which can be used for any view.

![Figure 12. Virtual acquisition system](image3)

![Figure 13. Synthetic sphere (a) and an input image for Shape from Silhouette (b) and Shape from Structured Light (c)](image4)

Note that the image from Figure 13c can not be obtained using the laser from Figure 12. Instead, we assume seeing the complete profile of the sphere, in order to be able to reconstruct the complete object using Shape from Structured Light only. Since the sphere does not contain any cavities, Shape from Silhouette can also reconstruct it completely. Therefore, we can measure the accuracy of each of the methods independently, as well as of the combined method.

In the first test we build models using 360 views with the constant angle of $1^\circ$ between two views, while increasing octree resolution. It turned out that the shape from Silhouette method performed best with an octree resolution of 128, where the approximation error was +0.83% of the actual volume, the
structured light method with a resolution of 256³ and +0.29% error (the other method produced there an error of -1.42%).

In the second test we build models with constant octree resolution of 256³ and increasing number of views. The angle between two neighboring views is always constant. Voxel size is 2 mm for all models. The models computed are shown in Figure 14.

![Figure 14. 3D models of synthetic sphere with increasing number of views](image)

In the tests with synthetic sphere we compared Shape from Silhouette and Shape from Structured Light against one another. With respect to octree resolution there was no significant difference in the behavior of the two methods -- the accuracy of the models built was approximately the same, with exception of octrees 256³ and 512³, where the volume and size of the Shape from Silhouette model started being smaller than the analytical values, while the Shape from Structured Light model truly converged to the analytical one. This can be explained through the intricacy of building a Shape from Silhouette based octree -- each node is projected into all 360 input images by projecting its 8 vertices, which means 2880 world-to-image projections of points per node. In the Shape from Structured Light method, a node is projected into two nearest images only, i.e. there are 16 world-to-image projections per octree node. Therefore, when dealing with octree nodes of finer resolution (when the projection of the node has approximately the size of a pixel), errors due to numerical instabilities are more likely to happen in Shape from Silhouette, especially when using a large number of views.

Regarding number of views, there was also no significant difference between the two methods. Using 20 instead of 360 input views was sufficient for both methods to create models less than 1% different from the models built using 360 views.

As second test object we used a synthetic cone where the relative error of the computed volume of models built with increasing octree resolution was much larger than the error of the models of the sphere built with same parameters. The reason for this is that the cone has a large number of octree nodes belonging to the surface, and these nodes are the ones contributing mostly to the error. The models of the cone built with different number of views showed the same behavior as the models of the sphere -- starting from 20 input views the volume error relative to the volume of the object built with 360 views falls to 1% or less.

If an object only needs to be visualized, without calculating its volume, a model built with the octree resolution 32³ and 10 input views can give a satisfactory result. However, it should be noted that the sufficient octree resolution as well as the sufficient number of view depend on the properties of the camera, the geometry of the acquisition system and the properties of the object observed. In the tests with synthetic data we dealt with rotational symmetric objects only. For more realistic cases with more complex data sets tests with real objects are necessary.

### 5.2 Real objects

For tests with real objects we use 7 objects shown in Figure 15: a metal cuboid, a wooden cone, a coffee mug, two archaeological vessels and two archaeological sherds.

![Figure 15. Real objects used for tests](image)

The cuboid and the cone have known dimensions so we can calculate their volumes analytically and compare them with the volumes of their reconstructed models. Using these two objects we can also measure the impact of ignoring camera lens distortion on the accuracy of the models. The other objects have unknown volume, so we will just show the models constructed.

All models shown in this section are built using 360 views, with constant angle of 1° between two neighboring views.

<table>
<thead>
<tr>
<th>Object</th>
<th>Voxel size (mm)</th>
<th>Measured dimensions (mm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vessel #1</td>
<td>0.74 mm</td>
<td>141.2 x 84.8 x 93.7</td>
</tr>
<tr>
<td>Vessel #2</td>
<td>0.53 mm</td>
<td>114.2 x 114.6 x 87.4</td>
</tr>
<tr>
<td>Sherd #1</td>
<td>0.84 mm</td>
<td>51.8 x 67.0 x 82.2</td>
</tr>
<tr>
<td>Sherd #2</td>
<td>0.76 mm</td>
<td>76.0 x 107.3 x 88.5</td>
</tr>
<tr>
<td>Cup</td>
<td>0.66 mm</td>
<td>113.3 x 80.0 x 98.9</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Object</th>
<th>Volume (mm³)</th>
<th>Calculated dimensions (mm)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vessel #1</td>
<td>336131</td>
<td>139.2 x 83.2 x 91.4</td>
</tr>
<tr>
<td>Vessel #2</td>
<td>263696</td>
<td>113.0 x 111.9 x 86.4</td>
</tr>
<tr>
<td>Sherd #1</td>
<td>35911</td>
<td>51.0 x 66.0 x 79.4</td>
</tr>
<tr>
<td>Sherd #2</td>
<td>38586</td>
<td>74.9 x 103.9 x 86.2</td>
</tr>
<tr>
<td>Cup</td>
<td>276440</td>
<td>111.6 x 79.0 x 98.3</td>
</tr>
</tbody>
</table>

Table 1. Reconstruction of two vessels, two sherds and a cup

The exact volumes of these objects are unknown and therefore the accuracy of the volume calculated through reconstruction...
can not be estimated. However, we can measure the bounding cuboid and compare it with the dimensions of the model. Table 1 summarizes the results. The resulting models, shown from three views, are depicted in Figure 16. All models are built with an octree resolution of 256^3 and using 360 views.

Figure 16. 3D models of two vessels, two sherds and a cup

6. CONCLUSION

Many In this paper a combination of a Shape from Silhouette method with a shape from Structured Light method was presented, which create a 3D model of an object from images of the object taken from different viewpoints. It showed to be a simple and fast algorithm, which is able to reconstruct models of arbitrarily shaped objects, as long as they do not have hidden concavities, i.e., concavities not visible in any of the input images. For concavities visible to the active system, results show that the computed volumes provide the correct model. The algorithm is simple, because it employs only simple matrix operations for all the transformations and it is fast, because even for highly detailed objects, a high resolution octree (256^3 voxels) and a high number of input views (36), the computational time hardly exceeded 1 minute on a Pentium II. Already for a smaller number of views (12) the constructed models were very similar to the ones constructed from 36 views and they took less than 25 seconds of computational time.

For archaeological applications, the object surface has to be smoothed in order to be applicable to ceramic documentation, for classification, however, the accuracy of the method presented is sufficient since the projection of the decoration can be calculated and the volume estimation is much more precise than the estimated volume performed by archaeologists.

7. REFERENCES


8. ACKNOWLEDGEMENTS

This work was partly supported by the Austrian Science Foundation (FWF) under grant P13385-INF, the European Union under grant IST-1999-20273 and the Federal Ministry of Education, Science and Culture.
SURFACE MODEL GENERATION BY THE RELICS FROM SLICE IMAGES, AND THE TRIAL TO THE AUTOMATIC RESTORATION

Y. Watanabe a,*, K. Tanaka a, N. Abe a, H. Taki b, Y. Kinoshita c

a The Faculty of Computer Science and System Engineering, Kyushu Institute of Technology, 680-4, Iizuka, Fukuoka, 820-8502, Japan - watanabe@scin.mse.kyutech.ac.jp, (kazuki, abe)@msec.kyutech.ac.jp
b The Faculty of Systems Engineering, Wakayama University, 930, Sakaedani, Wakayama, 640-8510, Japan
c Munakata Suikokai General Hospital, 341-1, Fukuuma, Munakata, Fukuoka, 811-3298, Japan

Commission VI, WG VI/4

KEY WORDS: X-ray CT, Relic, Intermediate point, Grid unit, Face tension algorithm, Normal vector, Cross section & point, Automatic restoration

ABSTRACT:

The relics that unearthed from the monument break well. The restoration work is necessary to the research, the analysis and the exhibition. The restoration task uses relics directly, and it takes very terrible time at the expert. Moreover, we have the possibility to inflict damage on the relics because we glue together and restore fragments. Therefore, we generate 3-D shape models in the computer and the restoration task is done. The measurement doesn't use the laser measurement that excels in getting the small shape of the surface, and uses the X-ray CT that measures the thickness of the fragments. We state the method that generates surface models automatically using the X-ray CT, and the restoration task is done in the computer.

The restoration task in the computer never inflicts damage on the relics. However, because the restoration task is manual operation, the restoration time doesn't change. That we use as the restoration simulator that returns the value of the fragment is very inefficient. Then, the automatic restoration is desired. Because X-ray CT maintains the cross section shape information of the fragment that is difficult in the laser measurement, we aim at the automatic restoration using the information. We find out normal vectors in the cross section and show an automatic restoration result.

1. INTRODUCTION

A relic excavated from remains appears as a collection of smaller fragments. For the research of the culture or technique of the age when the relic was produced or the exhibition of the original shape, re-constructing task is necessary to have these fragments joined together. Such a restoration task is taken place using excavated fragments directly up to now. But this restoration task is very complicated generally, and there are many cases that the restoration succeeds as a result of thinking error. Further there is the problem that fragments can't be returned to the original states after the restoration because they are adhered together with glue. Consequently, a re-constructed relic will fairly receive breakdowns compared with the original one. Further we can't examine an individual fragment in excavation after the restoration task. On the other hand, the development of 3-D measurement technique makes it possible to measure correct 3-D shapes of fragments. Further, the development of computers makes it possible to display data of high capacity. So we can measure the shape of each fragment in excavation, and practice restoration without using genuine fragments because a computer successfully reproduce fragments using computer graphics.

So far a laser measurement device has been principally used for measuring each fragment, but it is difficult to get the backside and thickness of a fragment although the device can get the close shape and color information of each face. So an X-ray CT is begin to use for measuring the internal shape of an object by acquiring a slice image (profile image) as shown in the Figure 1. Further because it has the transitivity, research on a relic or remains will have the broad possibility. Besides, for the restoration of a sophisticated model with a computer, a measurement with an X-ray CT is indispensable. Though a measurement with an X-ray CT can get a close internal shape, it becomes a problem that a connection between slice images becomes discontinuous. The image measured with the CT is modeled with voxels, but the data volume becomes so big that a strong machine power is necessary. So the surface model making the data volume comparatively small becomes necessary.

1. Figure 1. Difference in measurement methods

A surface model consists of a set of surfaces or boundary surfaces. Any surfaces of a 3-D object completely separate the outside from the inside of it, and must intersect with neither it nor any other surfaces. Besides, it is a very complicated problem to decide the surface including an arbitrary 3-D object

* Corresponding author. This is useful to know for communication with the appropriate person in cases with more than one author.
from voxel data of the object with a computer instead of the data of surfaces.

Because various interpretation in determining a surface is possible, many different surface construction algorithms are proposed, but needs to intervene with a man’ hand for complicated shapes. So the aim of this research is to generate automatically a complete surface model from slice images of very complicated shape measured with an X-ray CT.

So far all the restoration task is done by the manual operation at present. As long as we manipulate the relic fragment in virtual space, efficiency in this restoration task is not improved. Then, the automatic restoration is desired.

The precise shape information of a cross section of each fragment, which cannot be acquired with the laser measurement, can be easily got with our method using CT. Therefore, we aim at the automatic restoration using the information.

2. RELIC MEASUREMENT BY USING X-RAY CT

The laser measurement can measure only the shape of the irradiation range, the measurement doesn’t suit to reproduce a cross section and the back of relics. In this research, we used X-ray CT to measure a complicated shape. X-ray CT is the measurement device that measures 3-D space by the slice image. Because CT measures 3-D space, we don’t have to place fragments in a plane. A resolution of CT is not inferior compared with the laser measurement.

We fill fragments into the box (15cm × 15cm × 30cm) with the sponge that the CT value is different from the fragment. X-ray CT needs 20 seconds to measure 30 cm distance. And, the time, which produces a slice image from the CT exclusive use data, is 30 minutes. X-ray CT can measure the luster surface that it is difficult to measure by the laser measurement. However, the texture in the surface cannot be measured.

3. SURFACE MODEL GENERATION BY THE RELICS FROM SLICE IMAGES

The general model generation method using slice images is marching cube method. The method forms a triangular polygon based on the pattern of picture elements that are within eight neighborhoods of an element on the contour of an image. A surface model of the high quality can be generated with the method. There is, however, the danger that a different shape may be formed if several polygons are erroneously set up. If a shape includes intense changes between two levels of slice, wrong faces are patched there. As a result, the resultant shape is wrong because portions to be originally connected one another are torn to pieces.

For modeling of increasing relics, we must avoid manual operation. In this chapter, we state an automatic modeling. Not to be influenced by the size and the complexity of the slice image, we do process of grid unit. By introducing an intermediate point, we can do process of grid unit even if the gap between two slice images with intense change. We show the procedure in the followings.

3.1 Pre-processing

An X-ray CT image is processed before setting up faces. The image that is provided with an X-ray CT for each slice image is expressed with gray shaded picture elements of monochrome. Figure 2 is slice image taken with X-ray CT. This image sequence is slice images of 1-mm interval but is actually measured in 0.2-mm interval. Because gray shaded images cannot be expressed in polygons, they must be binaries. It is called threshold process. A threshold value is set at an intense place of alteration. And, we extract contours.

![Figure 2. Slice image measured with X-ray CT](image)

3.2 Approximation of a contour using a set of grid points

A salient characteristic of this research is a face tension with a grid unit. The finer a grid unit becomes, the more precise the approximation is. Points that a contour and the grid cross are selected to approximate the contour as shown in the Figure 3.

![Figure 3. Approximation of a contour using a set of grid points](image)

3.3 Intermediate points

A salient characteristic of this research is an intermediate point. An intermediate point is a point on the image obtained by taking difference of one slice image and another one. A detailed procedure is described using the Figure 4 as an example.

**AND information:** An AND collection is an intersection of two pieces of slice image. This portion is the region which polygon isn't set. Using this information, wrong selection of points nearby is avoidable even if the gap between two slice images with intense changes is interpolated.

**Intermediate information:** The difference information obtained by subtracting the AND information from the OR one mediates between a contour of lower slice from that of an upper one. In other words we don't need to look for corresponding points between adjacent contours. Intermediary points is obtained by taking grid points included this difference information. The intermediate points are completely separated.
from the list structure mentioned above.

![Diagram](image1)

**Figure 4. Procedure for generating intermediate points**

### 3.4 Process of grid unit

Because making correspondence between contours is difficult, intermediate points are exploited in this research. Without the search of corresponding points between contour, faces filling a gap between adjacent slice images are successfully set as shown in the Figure 5 using intermediate points.

![Diagram](image2)

**Figure 5. Face extension for each unit grid**

### 3.5 Labeling

There is the face that should be distinguished as shown in the Figure 6 when faces are dealt with grid unit. In other words it is a remaining portion obtained by removing both ‘AND portion’ and ‘Exception of OR portion’. We don’t set up face on this portion. We have only to perform face tension particularly.

![Diagram](image3)

**Figure 6. Areas in which face extension is not performed**

### 3.6 Face tension algorithm

The face tension is performed with respect to both grid unit and label unit. Tracing picture elements according to the direction of list structure, surfaces are set up as shown in the Figure 7. Because the direction of the face (normal) vector is the direction of the slice, we define the list structure of the upper slice as the clockwise direction, and we define the list structure of the lower slice as the counter clockwise direction. This allows every surface to be set up smoothly. A twisted portion can be patched up without any problem. Further for a set of grids where only intermediate points exist, the direction of a face can be easily determined from relationship between the top and bottom image.

![Diagram](image4)

**Figure 7. Face extension for a unit grid**

### 3.7 Surface model generation

Pasting the surface of all grids, a surface model of an object is generated as shown in the Figure 9. Because the model becomes a stair stepping when the intermediate point is constant, we specify the height of the intermediate point using the ratio with distance to the upper and lower slice image. Repeating this process over the consecutive pair of contours, a surface model is completed.

![Diagram](image5)

**Figure 8. The direction of the normal vector**

–86–
4. RESTORATION SYSTEM USING VR

After generating a model at the relics, we work in the restoration. A restoration system configuration is shown in Figure 10. We see virtual space at the stereo using HMD (Head Mounted Display). We hold object with the 2-D mouse and we work in the restoration with 3-D magnetic sensor.

5. RESTORATION RESULT

5.1 Relic model generation

The used relics are the KOIMARI bowl as shown in Figure 11. Surface models generated from the given relic fragments using this algorithm are shown in Figure 12. Because the surface of the KOIMARI bowl is a luster surface, it is difficult to get with a laser measurement. But we can confirm that detail of the model is reproduced with an X-ray CT (see Figure 13).

Figure 11. Original fragment of KOIMARI bowl

5.2 Restoration of relics

The original shape of a relic is restored using fragments restored with this algorithm. In the restoration task, the original restoration system (see 4. restoration system using VR) is used. Figure 14 shows a result of restoration.

Owing to the lack of some fragments, the restored relics include broken part. By using the tool of the restoration system, the fragment can be duplicated. The lack part is buried using the duplicated fragment (see Figure 15).

Figure 12. Surface model of KOIMARI bowl

Figure 13. The generation comparison in the aspect of the luster

Figure 14. A relic restored with proposed method

Figure 15. Virtual restoration
6. APPROACH TO THE AUTOMATIC RESTORATION

At present, though it is possible to restore a relic from fragments in virtual space, the restoration task is only moved from real space to virtual space. Though no damage is given to virtual fragments at all, the restoration task is still done by the manual operation. Under the present circumstances, the increase of the restoration task efficiency cannot be expected. Then, the automatic restoration is desired. As an approach for the automatic restoration, followings are regarded:

- The expectation of the relics by the nature of the soil and the find site.
- The utilization of the curvature of the model.
- The resemblance between cross sections of two fragments.

This knowledge is important with the restoration work but it is difficult for the computer to use this knowledge. The restoration task is usually done by exploiting, features obtained from the cross sections of fragments in the real space. And, much more precise information on the cross section of each fragment can be obtained with X-ray CT rather than the laser measurement. Therefore, we aim at the automatic restoration based on the former strategy.

6.1 Cross section & point extraction

It is necessary to extract the cross section of a fragment in order to find a counter part. As extracting method, normal vectors of the cross section are used. The normal vector is a perpendicular vector to the surface. In comparison with the normal vectors of the neighbors by extracting one with different angle, the cross section can be found as shown in Figure 16.

![Figure 16. Cross section definition](image)

However, this method extracts the cross section as a cluster of points. To create list structure automatically from the point group at present is a difficult problem. Because this problem seems to be solved in the other field, we create list structure by the manual operation.

If a counter part is selected among the all other fragments using the cross section information, the system will suffer from the inefficiency. Then, to aim only at the cross section information, we define a rapidly changing point as a cross point. Figure 17 shows a cross section & cross points of the fragment.

![Figure 17. The cross section & point](image)

6.2 Search of adaptation of the cross section

When we restore the relic, the tendency that relic is restored by the pair is strong. When restoring more than two as shown in Figure 18, a pair isn’t often made. Therefore, it is possible to discover a pair efficiently using the cross point. By re-forming cross point after assembling fragments, the new pair can be discovered.

![Figure 18. The order of a counter part](image)

Using the normal vector, we search a corresponding cross section with pair. When the amount of the normal vector in the mutual cross section becomes 0, we define as the corresponding cross section.

6.3 Automatic restoration example using the cross point

We do automatic restoration using the cross point. We measure the curve distance value of the cross section which is caught in the cross point as shown in Figure 19. The numerical value of the Figure 19 is the curve distance value of the cross section. Because we cannot search the right cross section, we cut off a curve distance below 100 pixels. We compute the amount of the normal vector in the cross section with near curve distance value (±5%).

![Figure 19. The automatic restoration example by using a point: 1](image)

The search in the cross section with near curve distance value was 19 patterns. By worker’s excluding fragments with different thickness, we chose 5 patterns as shown in Figure 20. Moreover, we got the result as shown in the Figure 21 by more searching. Then, we find that the [2,7] pair of the Figure 20 is a mistaken result. Because the computer can handle overlapping fragments, the trial and error is easy. In the last, we got a result as shown in Figure 22.

The error is born whenever continuing restoration, and cracks are built. However, by this automatic restoration result, we can imagine the whole shape of relic. By getting a relic number from this automatic restoration result, this system is useful as the restoration simulator.
7. CONCLUSION

A new approach is proposed in this paper that automatically restores a surface model of an object with a complicated shape from the CT slice images. Model generation from CT images so far requires not only complicated CAD operation but also intervention of a man. The method proposed makes it possible to automatically restore surface models of objects with complicated shapes. Compared with the thin model restored with a laser measurement, it becomes easy to catch the shape of a fragment by leaps and bounds. Further efficiency of a restoration task is improved by using the thickness of each fragment.

Problems to be solved include the improvement in the smoothness of a curved surface and the reduction of data volume. There are often cases where the unevenness is conspicuous because all shading is currently set to the same value. Taking a proper normal vector can be more smooth model. Even for the portion of little inclination the size of a grid is established in the same value. This is the cause that data volume increases idly. This is also solvable if different values are given to grids included in the areas of intense changes.

We showed the possibility of the automatic restoration by extracting a section and setting a point. The system cannot search all corresponding cross section. However, this system is more very efficient than the manual operation. By getting a relic number from this automatic restoration result, this system is useful as the restoration simulator.

References:


But, the system cannot search all corresponding cross section. In the case, we must restore by using the original restoration system. However, this system is more very efficient than the manual operation.