AN INTEGRATED MULTI-SENSORY SYSTEM FOR PHOTO-REALISTIC 3D SCENE RECONSTRUCTION*

Kia Ng,[1] Vítor Sequeira,[2] Stuart Butterfield,[1] David Hogg[1] and João G.M. Gonçalves[2]

[1] School of Computer Studies, University of Leeds, Leeds LS2 9JT, UK.
[2] European Commission - Joint Research Centre, TP 270, 21020 Ispra (VA), Italy.

E-mail: kia@scs.leeds.ac.uk, vitor.sequeira@jrc.it, stuart@scs.leeds.ac.uk, dch@scs.leeds.ac.uk, joao.goncalves@jrc.it

Commission V, Working Group 3

KEY WORDS: Laser-range, video, texture, multiple-view, integration, fusion.

ABSTRACT

In this paper, we describe an integrated approach to the construction of textured 3D scene models from laser range data and visual images. This approach has been realised in a collection of algorithms and sensors, within a prototype device for 3D reconstruction known as the AEST (Autonomous Environmental Sensor for Telepresence) -- an autonomous mobile platform carrying a scanning laser range finder and video camera. The AEST is intended to fully automate the creation of models from building interiors by navigating automatically between positions at which range data and video images are captured. Embedded software performs several functions, including triangulation of the range data and registration of video texture, registration and integration of data acquired from different capture points, and optimal selection of these capture points, ensuring that range data and video texture is acquired for all surfaces at the required resolution. This latter function is a principal novelty in the approach.
We describe the major components of the AEST and present preliminary results obtained from the prototype. The final model, including texture mapping information, is encoded in VRML format. It is then possible to access the model via the World Wide Web, with an appropriate browser. Potential applications include facilities management for the construction industry and within the general area of virtual reality, for example, virtual studios, virtualised reality for content-related applications (e.g., CD-ROMs), social tele-presence, architecture and others.

ABSTRAKT

In diesem Bericht beschreiben wir eine integrierte Methode zur Konstruktion von 3D Modellen aus Laserentfernungsbildern und Videobildern. Diese Methode wurde mit einer Reihe von Algorithmen und Sensoren in dem Projekt AEST (Autonomous Environmental Sensor for Telepresence) implementiert. AEST ist eine autonome, mobile Plattform mit einem Laserentfernungsmesser und einer Videokamera. Das Ziel ist es, die Konstruktion von 3D Modellen von Gebaeudeinneren vollstaendig zu automatisieren indem der Roboter selbststaendig zwischen Position navigiert, an denen Entfernungs- und Videodaten aufgenommen werden. Die Software erfuellt mehrere Aufgaben, unter anderem die Triangulation der Entfernungsdaten und die Registrierung mit den Videobildern, die Registrierung und Integration von Daten, die von verschiedenen Positionen aufgenommen wurden, sowie die optimale Auswahl der Aufnahmepositionen. Letzteres ist die wesentliche Neuheit unserer Methode und stellt sicher, dass Entfernungs- und Videodaten fuer alle Flaechen in der benoetigten Aufloesung zur Verfuegung steht.
Wir beschreiben die wesentlichen Komponenten von AEST und zeigen die vorlaeufigen Ergebnisse des Prototyps. Das resultierende Model mit der integrierten Videoinformation ist in VRML Format codiert und ermoeglicht somit den Zugriff ueber das World Wide Web. Moegliche Anwendungen sind unter anderem Gebaeudemanagement fuer das Baugewerbe sowie auf dem Gebiet der Virtual Reality, Virtual Studios, soziale Telepresaenz und Architektur.

1. INTRODUCTION

With advances in computational capabilities, realistic 3D models are becoming increasingly important to many computer applications, particularly in architecture, design, simulators and entertainment applications.

There have been attempts to record the spatial and visual complexity of real-world environments. Most of these works are based on creating first a graphical three-dimensional representation of the environment, and then have it "painted" to give the best possible approximation to the natural visual appearance. Other works have concentrated on the 3D sensation as "seen" from a specific point of view (e.g., holography, stereo-vision, etc.). However, the "feeling of being there" is limited by either the huge amount of work in creating realistic graphical representations of the environment, or by forcing the user to have a pre-defined point of view.

The work described in this paper attempts to create realistic 3D representations of real environments. To achieve this, we have implemented an integrated approach to the construction of textured 3D scene models from laser range data and digital images. The approach has been realised within a prototype system known as the AEST (Autonomous Environmental Sensor for Telepresence). The AEST is designed to fully

* This work has been carried out as part of the EU-ACTS project RESOLV, involving partners from four countries.

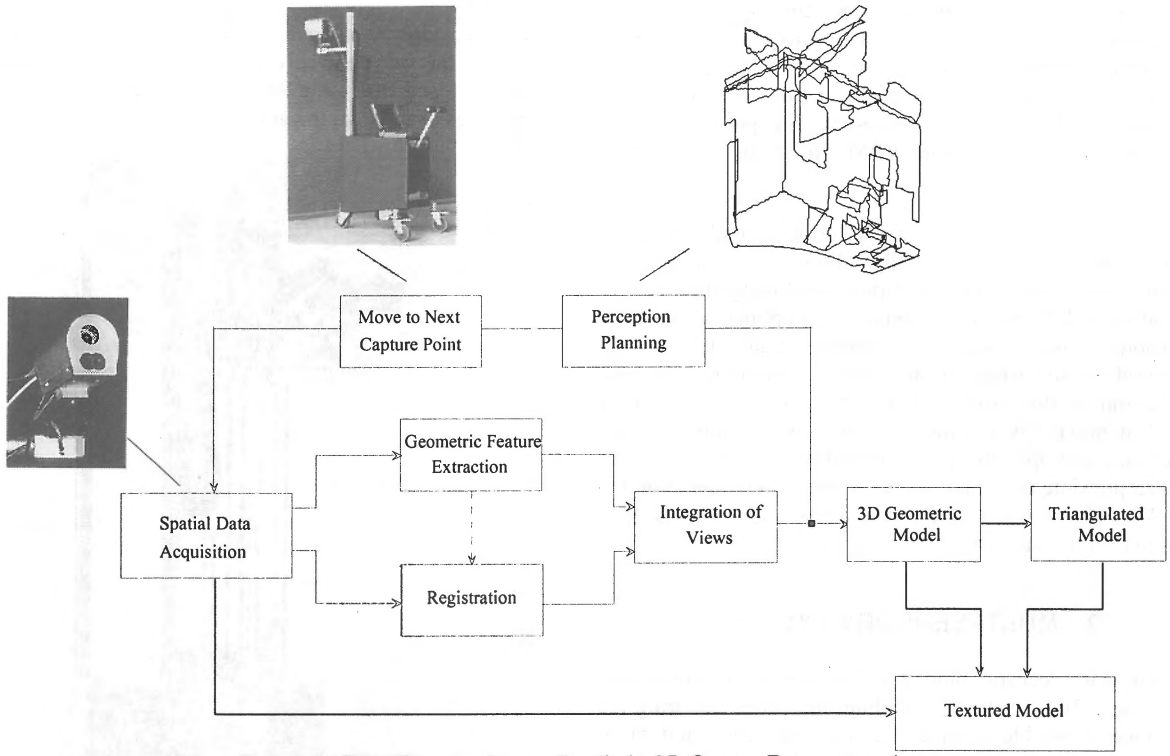


Figure 1: Paradigm for Photo-Realistic 3D Scene Reconstruction.

automate the model construction of building interiors by navigating automatically between locations at which range data and video images are captured.

An important feature of this work is the ability to resolve occlusions in the 3D representation of an environment. It should be noted that in order to achieve this, spatial data must be acquired at different capture points, and thus there is the need to move the acquisition device from one location to another. The end-user can see the environment from whatever viewpoint she/he selects and is unaware of the capture points from which the 3D model was reconstructed.

This paper describes the major components of the AEST and the earlier prototype (a manually-positioned trolley), as well as the main software modules. We also present some recent results. Further information about the project and example reconstructions, including VRML data for all the models shown in this paper, can be found on the RESOLV web page (http://www.hhdc.bicc.com/resolv).

2. PARADIGM FOR PHOTO-REALISTIC 3D RECONSTRUCTION

The 3D reconstruction paradigm (see Figure 1) begins with a data acquisition session, followed by the creation of a geometric model of the scene. The model is then analysed to detect occlusions. The best capture point for the next data acquisition is thus computed and instructions given to move the sensor head.

A second data acquisition session takes place. Raw data from this view is registered to that of previous acquisition sessions. The geometric model for the new view is extracted and merged with the existing one. If new or unresolved occlusions are detected, further acquisition sessions are required. When this process is complete, the geometric model is triangulated, and each triangle textured with data from the video camera. The main issues for consideration are discussed below:

- ***Acquisition of spatial data*:** To construct a 3D image it is required to scan the laser beam across the scene, either using lightweight scanning mirrors (AEST) or by directly steering the laser source (EST). A series of colour images, covering the field-of-view of the laser scan, are taken using the on-board video camera. These images are to be fused onto the reconstructed 3D model to create photo-realistic 3D model.
- ***Extraction of geometric features:*** Feature extraction for geometric modelling is required to provide accurate localisation, especially for those edges corresponding to surface discontinuities, and to guarantee that edges are correctly classified (i.e. jump, crease).
- ***Detection and resolution of occlusions:*** If a complete volumetric model is to be achieved, range data acquired from multiple viewpoints is required to solve all the ambiguities due to occlusions.
- ***Registration:*** The problem is registering two partially overlapping surfaces that may lack significant features. Registration is required to be as precise as possible.
- ***Integration of multiple range images:*** A complete model is more than the sum of individual models, and should integrate geometric features from single views. The integration approach depends on the final representation (e.g. triangular meshes, surfaces).
- ***Triangulation:*** Long and thin triangles, especially horizontally orientated ones, are not desirable due to the projection effects, and do not generate good texture maps.
- ***Perception Planning:*** The range acquisition system must enter the scene to collect data. Thus, the algorithm for planning the next view takes into account the environment that is incrementally being built and the associated constraints: topological (imposed by the objects being

scanned) and operational (imposed by the already reconstructed environment and by the acquisition system).

- ***Textured Model:*** To add realism, the triangulated VRML file produced by the 3D reconstruction module is then fused with the colour still-video-images captured earlier to generate a texture-mapped VRML file with one or more texture-images.

In short, the embedded software performs several functions, including data acquisitions, triangulation of the range data, registration of video texture, registration and integration of data acquired from different capture points, and optimal selection of these capture points, ensuring that range data and video texture is acquired for all surfaces at the required resolution. All these modules run on the host-PC, and communicate to each others via a Host Server (HS) module. In addition to controlling the flow of data and operations between software modules, the HS is also responsible for communication and remote-operation via the WWW. This allows a user to start a new scan and download the results via the Internet.

3. MULTI-SENSORY SYSTEM

To reconstruct realistic models, a mobile robot capable of conducting a 3D survey of a building, including capturing the appearance of visible surfaces has been designed and built. Figure 2 shows the latest prototype system known as the AEST. It can be divided into three main parts, namely:

- sensor head
- tower unit
- mobile platform

The sensor head consists of a laser range finder with rotating mirror for capturing 3D structure and a video camera for capturing texture information. The laser range finder and the video camera sit on a pan-and-tilt unit (PTU) which is, in turn, fixed on a moveable platform whose vertical movement is controlled by a belt-driven device.

There are two computers in the AEST. The first one, referred to as the host-PC, is situated in the tower unit, which stands on top of the mobile platform. This is responsible for the processing and acquisition of laser and video data. It also runs a Web server with a wireless network transmitter for remote access and control to the robot. The second computer is a Motorola 68040 which is situated in the mobile platform. Its main functionality is the control of the AEST movement. The two computers communicate via a parallel link.

At the base of the AEST, is the mobile platform which carries the Motorola computer, batteries and other components, including a ring of ultra-sonic sensors surrounding the mobile platform. These detect any obstacles in its path. Additional sensors detect unexpected changes in ground level, for example steps. The mobile platform is supported by three wheels, two of which are motorised.

Figure 3(a) shows an earlier version of the prototype system in the form of a manual trolley, and Figure 3(b) shows its sensor head design.

Since the aim is for the reconstructed model to convey the feeling of being present in the real scene, the data acquisition is performed at eye level. Thus the models look best when viewed from a virtual walking or seated position.

These considerations have dictated the size, geometry and design of the vehicle. The two devices mentioned above have been developed within the RESOLV project and are now in the final stages of software integration.

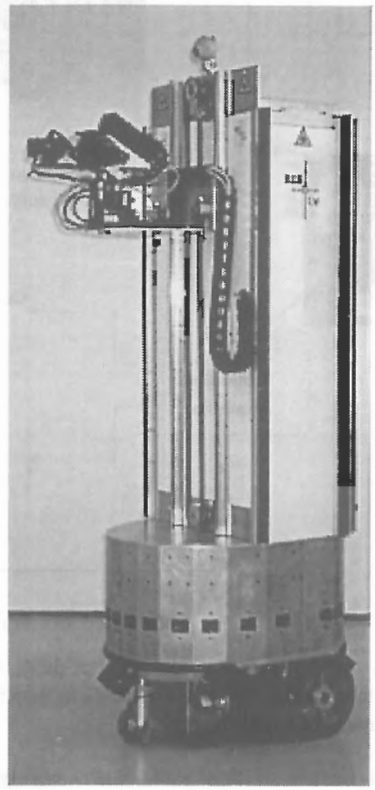
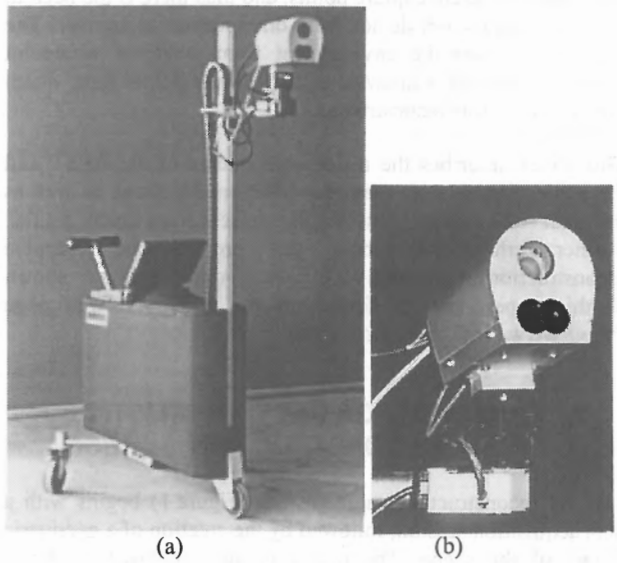


Figure 2: A picture of the AEST.



(a) (b)

Figure 3: A picture of the EST and a close-up view of its sensor head.

4. THREE DIMENSIONAL SCENE RECONSTRUCTION

An important issue for man made environments is the choice of the sensor type from which raw range information from the scene is obtained. Range image acquisition can be defined as

the process of determining the distance (or depth) from a given observation point to all points of consideration in a scene. Our system uses time-of-flight Laser Range Finders (LRF) and algorithms were developed for efficient 3D data acquisition.

4.1 Multiple Viewpoint Range Acquisition

The technology of laser range finders is not new and has been used for many years in military and airborne remote sensing survey applications. LRFs measure distance as a direct consequence of the propagation delay, i.e., the LRF emits a laser beam and detects the echoed beam. LRFs provide good distance precision with the possibility of increasing accuracy by means of longer measurement integration times. The *integration time* is related to the number of samples in each measurement. The final measurement is normally an average of sample measures, thus decreasing the noise associated with each single measure. *Spatial resolution* (i.e., the ability to distinguish two targets at different distances when placed side by side) is guaranteed by the small aperture and low divergence of the laser beam. The LRF sensor also provides a measure of the reflectance of the object being sensed. Reflectance measurements correspond, however, to the reflectance of the target at the wavelength of the laser beam, in this case infrared.

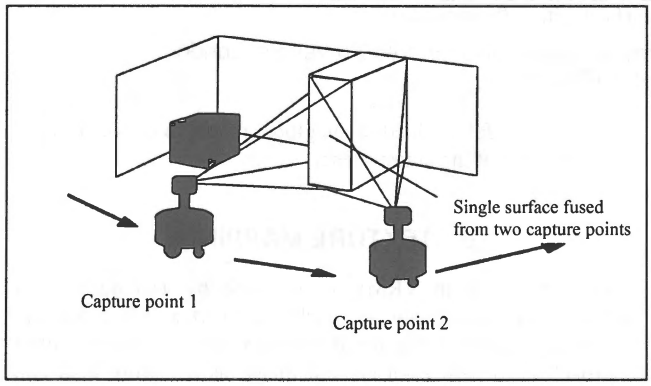


Figure 4: Multiple range views.

It should be noted that normally it is not possible to have complete 3D representations from data acquired at a single viewpoint. Indeed, to resolve occlusions in the scene or to reconstruct large scenes it is necessary to have 3D data acquired from multiple viewpoints (Sequeira et al., 1996 and 1996b) as represented in Figure 4. The following operations are needed:

- planning for the next view (i.e., the position and orientation of the LRF at the next viewpoint).
- registration and integration of the data acquired from different viewpoints (i.e., how to merge different range images together).

The quality of the final model is highly dependent on the quality of the registration between the different views. The efficiency of the system depends on the planning strategy. The number of viewpoints at which data is acquired should be minimised to reduce the time needed for a complete scene reconstruction.

Figure 6 (a) and (b) shows a sequence of four range and reflectance images taken from the lobby of the Royal Institute of Chartered Surveyors (RICS), London.

4.2 Scene Modelling

The scanning system provides an image containing explicit 3D range information for the scene. This 3D data, in itself, constitutes a 3D representation of the scene, but a higher level representation of is required to interpret it. It is desirable that the reconstructed surface description be as simple as possible while preserving its precision. If a piece-wise surface representation is used, the number of reconstructed patches should be as small as possible, and if polynomial patches are used the degree of the polynomial should be as low as possible. The final representation should not depend on how the different views were taken and a single model for the whole scene should be built irrespective of the number of range images used to characterise the environment.

The way a reconstructed model is described depends crucially on the purpose for which the descriptions are intended. Two categories can be identified:

- those based on shape primitives, e.g. Constructive Solid Geometry (CSG), which are suitable for CAD-style applications, and
- those composed of polygonal meshes, more suitable for graphical and visualisation applications.

This paper concentrates on the extension of previous work to build triangular piecewise planar meshes of arbitrary topology for situations where the visual appearance rather than the surface structure is required (Sequeira et al., 1995 and 1998, http://mortimer.jrc.it/3DRecSBA.html). The algorithm starts by building a triangular mesh connecting all the valid 3D points that are in adjacent rows and columns. To avoid joining portions of the surface that are separated by depth discontinuities (jump edges), a first test is performed, based on the triangle aspect ratio, t, between the radius of the circumscribed circle, R, and the inscribed circle, r (see Figure 5):

$$t = \frac{r}{R} \tag{1}$$

All triangles joining depth discontinuities are then discarded.

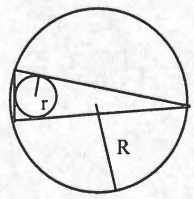
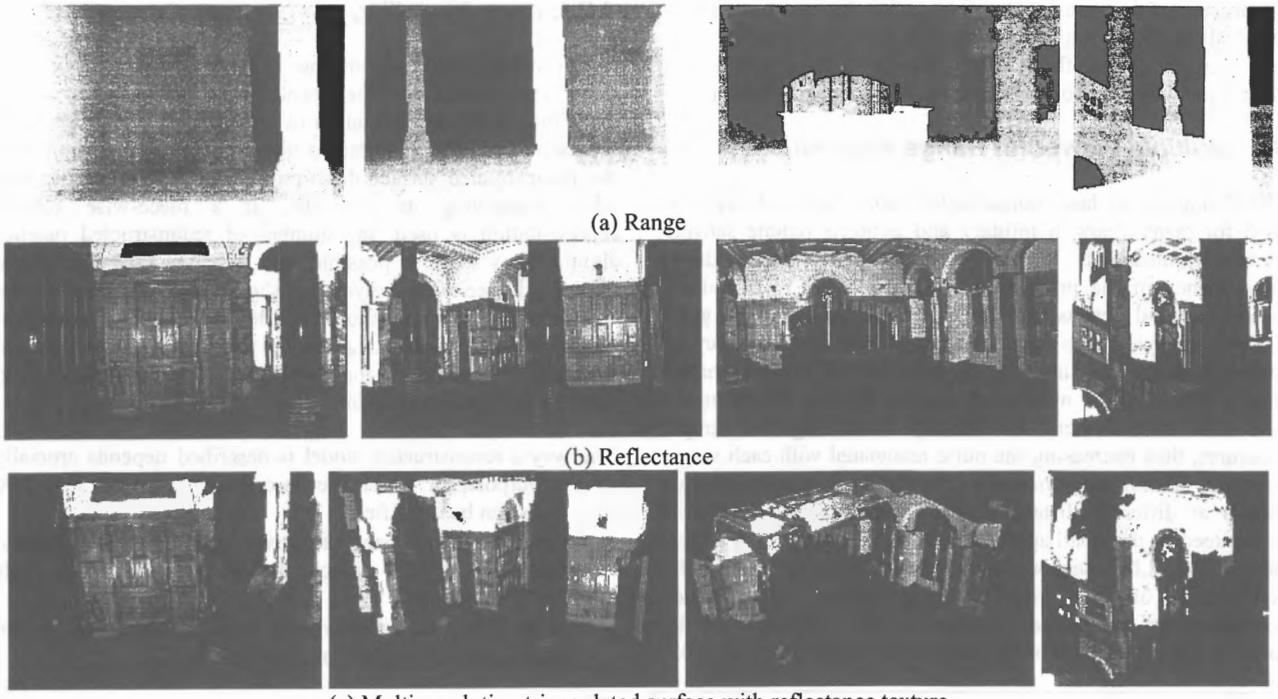


Figure 5: Triangle aspect ratio.

By applying this procedure it may be necessary to discard triangles that do not join depth discontinuities, but have been acquired with a small angle to the surface being scanned. These triangles are not reliable (the measurement noise increases exponentially with the angle between the surface normal and laser beam (Sequeira, 1996b)) and should also be discarded, data being acquired from another scanning position.

An iterative fitting procedure tries to reconstruct a surface mesh based on the initial fine triangulation, by enlarging the triangles where they fit the data to within a pre-specified tolerance. The result is a multi-resolution mesh of triangular surfaces, where edges are preserved, giving a very realistic representation of the



(a) Range

(b) Reflectance

(c) Multi-resolution triangulated surface with reflectance texture

Figure 6: 3D Reconstruction of the lobby at the Royal Institute of Chartered Surveyors, London. (http://mortimer.jrc.it/3DRoyal.html)

3D surface when textured with the laser reflectance or digital image data. Some results of using adaptive triangular meshes are shown in Figure 6(c).

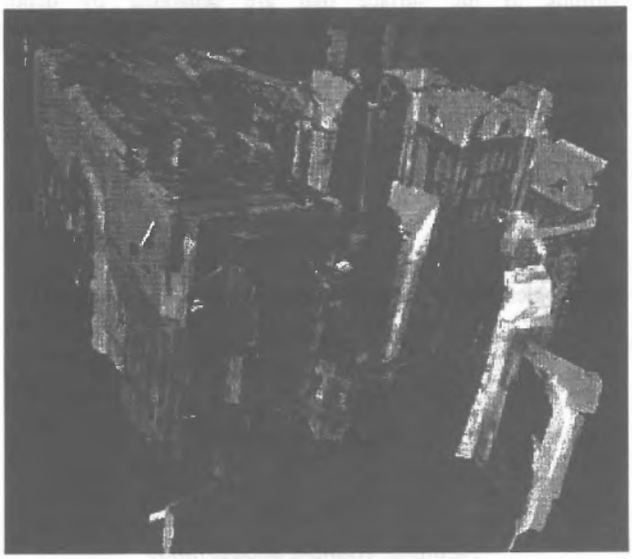


Figure 7: Model resulting from the integration of the four views (viewed from outside).

After each iteration, the integration module takes the triangulated mesh of the current viewpoint and transforms it into a global coordinate frame, using the transformation matrix provided by the registration process (see Figure 7). Next, the algorithm detects the overlapping areas between the current and global models and removes the corresponding triangles from the global model. The triangles in the overlapping area of the current model are corrected by averaging them with the corresponding data of the global model. The boundary of the current mesh is adjusted in order to close any gap which might have appeared after removing the overlapping triangles. Finally, the current model is added to the global model to obtain a single representation of the entire scene.

5. TEXTURE MAPPING

Texture mapping in VRML is achieved by assigning a 2D texture map coordinate to each 3D vertex. The texture-coordinate is then interpolated between vertices. The simplest approach is to treat each camera image as a texture map and store the correspondences between 3D model points and image pixels in the VRML file. To do this requires *calibration* of the camera with respect to the model. The mapping between 3D points and a given 2D image is encapsulated in the *camera model*, denoted by the function M. In homogeneous coordinates, M projects a model point $\mathbf{X_L} = (X,Y,Z,1)$, in 3D laser coordinates, onto a 2D image point $\mathbf{x} = \lambda(u,v,1)$ as follows:

$$\lambda \begin{pmatrix} u \\ v \\ 1 \end{pmatrix} = M \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \qquad (2)$$

Camera models vary in complexity, depending on how accurately they attempt to characterise the image formation process. This, in turn, determines the number and types of parameters in M. We use Tsai's camera model (Tsai, 1987) which is based on the pin-hole model of perspective projection, and has eleven parameters. Six *extrinsic* parameters define the position and orientation of the camera with respect to the world coordinate system. Five *intrinsic* parameters describe the optical properties of the camera, including a 1st order approximation of a coefficient to correct for radial distortion (Weng et al., 1992).

Given at least seven non-coplanar 3D points and their corresponding 2D image positions, then the camera model parameters can be determined (eleven correspondences are

required for a fully optimised solution). Correspondence information is supplied by the system operator via a graphical user interface. When more than the minimum number of correspondences are known, we employ the RANSAC parameter estimation technique (Fischler and Bolles, 1981). This attempts to identify and discard erroneous data, and thus use only the most accurate data to obtain a solution.

Since the field of view of the video camera is relatively small, compared to that of the laser range finder, a series of video images must be captured in order to cover a laser scan view, typically in a 3 × 5 grid. In addition to the range and video information, the laser generates a *reflectance image* at each scan location. This is computed by analysing the amount of light returned to the laser at each 3D point and quantising these values into grey-levels. There is a known correspondence between a reflectance image pixel and a 3D point. Thus, the user can generate the 2D/3D correspondences required for calibration by matching points in the camera and reflectance images. This is more accurate than picking 3D VRML model points directly, as, the triangulation and surface-fitting processes lead to a loss of resolution and the original raw data is featureless and difficult to interpret.

Once the parameters of the model have been computed, the camera/image pair is said to be *calibrated*. The function M can then be used to project *any* 3D point into the camera image plane to generate 2D texture coordinates. These are stored in the VRML file and the texture-mapping process is completed.

The *hand-calibration* process could be performed for each of the camera images which are to be used for texture-mapping the model. This is very flexible in that it allows the model to be texture-mapped using an image taken with any camera, at any time, from any location, however, it does require user interaction. It is possible to automate this process in cases where the model is to be texture-mapped using only images obtained with the pan/tilt mounted camera. If the camera intrinsic parameters remain fixed, the change in the projection transformation between each of the images is entirely due to the change in extrinsic parameters, which can be computed from the (known) pan/tilt values. Thus, only a single hand calibration step is required (i.e. only an one-off calibration procedure is needed for a sensor head design) from which the camera parameters for the remaining images can be derived. See Sequeira et al. (1998) for a fuller description.

5.1 Perspective Correction

In order for a VRML object to look realistic from any viewpoint, it is necessary for the texture-mapping process to take account of perspective effects in the image formation process. To see why this is important we must consider the mechanics of VRML texture mapping.

The choice of which texture map element (texel) to apply to a given point on a 3D triangular facet is made by interpolating between the coordinates of the texel at each vertex. This technique can lead to distortion, because it approximates the perspective projection of the imaging process with a linear transformation (Weinhaus and Devarajan, 1997). This is very noticeable if the 3D model is composed of relatively large triangles, but improves as the triangles get smaller. In order to avoid such distortion altogether, each triangle should be textured using data from a camera image taken orthogonal to it. Since such data is not immediately available for the majority of triangles, we must devise a method to recover the orthogonal view from an oblique view.

Instead of simply using the original camera image, we create a new texture-map image. This is the same size as the original camera image, and the vertices of its triangles are in exactly the same places, but it starts off devoid of texture and is filled in as perspective correction is applied to each triangle.

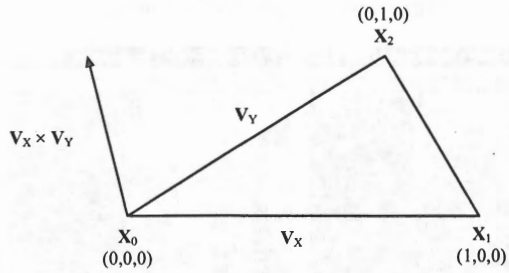


Figure 8: Constructing a facet-centred coordinate system.

The perspective correction process involves several steps. The first of these is to construct a 3D coordinate system for each triangle (see Figure 8). Suppose the non-homogeneous 3D coordinate vectors of a triangular facet are $\mathbf{X}_0, \mathbf{X}_1, \mathbf{X}_2$. We set:

- The origin (0,0,0) of our facet coordinate system as $\mathbf{X}_0$.
- The x-axis vector (1,0,0) as $\mathbf{V}_X = \mathbf{X}_1 - \mathbf{X}_0$.
- The y-axis vector (0,1,0) as $\mathbf{V}_Y = \mathbf{X}_2 - \mathbf{X}_0$.
- The z-axis vector as the normal $\mathbf{V}_X \times \mathbf{V}_Y$.

The transformation from facet coordinates to 3D world coordinates is given by the 4×4 matrix, $\mathbf{F}$:

$$\mathbf{F} = \begin{pmatrix} \mathbf{V}_X & \mathbf{V}_Y & \mathbf{V}_X \times \mathbf{V}_Y & \mathbf{X}_0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{3}$$

Two views of a planar object, taken with the same camera, are related by an affine transformation. The aim is to find the transformation which maps from the camera image view of a triangle to the orthogonal view. This is determined by the affine correspondence between texture triangle vertices $\mathbf{t}_i = (s_i, t_i, 1)$ and facet vertices, in the $z = 0$ plane. It can be calculated by solving for the six unknowns (a_{ij}) in the following system of equations:

$$\begin{pmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = \begin{pmatrix} a_{00} & a_{01} & a_{02} \\ a_{10} & a_{11} & a_{12} \end{pmatrix} \begin{pmatrix} s_1 & s_2 & s_3 \\ t_1 & t_2 & t_3 \\ 1 & 1 & 1 \end{pmatrix} \tag{4}$$

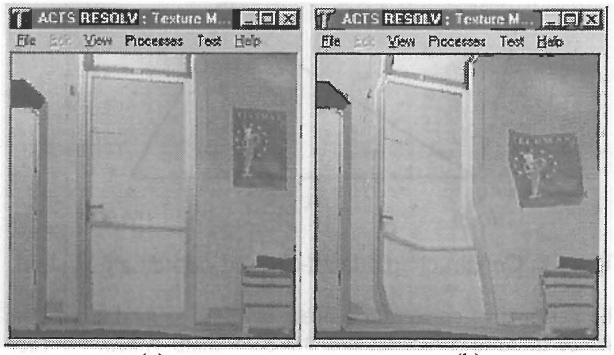
We can now form the pipeline of transformations between the texture map and original image coordinates:

$$\underbrace{\lambda \begin{pmatrix} u \\ v \\ 1 \end{pmatrix}}_{I} = M \underbrace{\begin{pmatrix} \mathbf{V}_X & \mathbf{V}_Y & \mathbf{V}_X \times \mathbf{V}_Y & \mathbf{X}_0 \\ 0 & 0 & 0 & 1 \end{pmatrix}}_{\mathbf{F}} \underbrace{\begin{pmatrix} a_{00} & a_{01} & a_{02} \\ a_{10} & a_{11} & a_{12} \\ 0 & 0 & 0 \\ 0 & 0 & 1 \end{pmatrix}}_{\mathbf{A}} \underbrace{\begin{pmatrix} s \\ t \\ 1 \end{pmatrix}}_{\mathbf{t}} \tag{5}$$

Working backwards, the transform $\mathbf{A}$ converts from texture coordinates to facet coordinates; $\mathbf{F}$ transforms from facet coordinates to model coordinates and, finally, M projects 3D model coordinates into 2D camera image coordinates. Thus, texture-mapping proceeds as follows: for a given model

triangle, use its vertices to generate **F** and **A**. For each empty pixel **t** within the bounds of the triangle in the texture-map obtain the coordinates of a pixel in the original camera image, as $\mathbf{x} = M(\mathbf{FAt})$. This pixel is stored in the texture-map at the current location.

When a texture-map generated in this fashion is applied to a model, it will look realistic regardless of the viewing angle. Figure 9 shows a view of part of a model which has been texture-mapped with and without perspective correction.



(a) (b)

Figure 9: Results of texture-mapping with (a) and without (b) perspective correction.

5.2 Hidden Texture Removal

In cases where a model has been reconstructed from several different data capture points, it is probable that some 3D triangles are not visible in one or more camera images. If this is due simply to the fact that a triangle is outside the camera's field of view, there is no cause for concern. However, if a triangle is *occluded* by other scene structure, care must be taken not to texture-map it with texture belonging to the occluding surface.

A little bit of preprocessing solves the problem. Looking again at equation 2, each model triangle is projected into an image, using its camera model function M. Although two different 3D points can project onto the same 2D image location $(u,v,1)$, the projective scale factor, λ, which is a function of the depth of the 3D point relative to the camera, will be different. Therefore, λ can be used to decide whether to texture-map a 3D triangle, or part thereof.

To begin with, each 3D triangle is assigned an id-number, and depth and label maps of the same dimension as the camera images are initialised. Each triangle, in turn, is projected into each camera image and the λ values at its 2D vertices are noted. Values of λ are interpolated for every 2D pixel within the boundary of the 2D triangle and compared against those stored in the depth map. If the λ value indicates that the 3D point is closer to the camera then it replaces that value in the depth map and the current triangle-id is stored in the corresponding label map location.

The label maps are used later, when loading the texture-map. As before, camera image coordinates are computed from texture coordinates. Now though, the camera image pixel is only stored in the texture-map if the current triangle-id matches that stored in the corresponding label map location. This ensures that the texture available in a particular camera image is only applied to parts of the model which are visible in that image.

5.3 Multiple Image Texture Mapping

It is common for a particular segment of 3D scene structure to appear in several different camera images. This may be caused, for example, by overlapping in the image mosaic generated by the pan/tilt camera, or images taken from radically different views. There is also the case where the camera has zoomed to get very high resolution data for a particularly important scene feature.

Whatever the circumstances, the availability of multiple texture sources for a single triangle, necessitates additional consideration. The first problem is to decide *which* texture source to use. In fact, our solution is very simple as the hard work has already been done by the hidden texture removal preprocessing. Effectively, each label map tells us how many pixels in the corresponding image would be used to texture map any given model triangle. All that is required is to sum the number of instances of the triangle-id in the label map. This is done for each triangle in every image and the image contributing the largest number of pixels is used as the texture source. This method requires some restructuring of the VRML file, for the sake of efficiency, such that all those triangles sharing a common texture source are grouped together.

Another problem when dealing with multiple texture sources is that of redundant information. If the original camera images are used as texture maps, some of the texture will never be used as the model triangle it belongs to may have already been textured from another camera image, having better coverage. A more efficient storage mechanism is required. The approach we have taken is to extract the data from the camera images and load it into one or more packed texture-maps, storing only that data which is actually used during texture mapping. The problem then is to determine the sizes and arrangement of texture triangles which minimise the amount of wasted space. There is a trade-off between the time it takes to perform the texture-packing process and the amount of texture-map space it saves. Therefore, packing is not performed automatically, but only when the data redundancy is at a high enough level to make it worthwhile. The texture-packing process is described in detail in Sequeira et al. (1998).

6. RESULTS AND CONCLUSIONS

In Figure 10 several screen snapshots from a reconstructed model of the lobby area at the Royal Institute of Chartered Surveyors, London are presented.

Work towards improving the accuracy and realism of our reconstructions is ongoing. We are currently investigating ways to augment the models to enhance the sense of "being there". For example, an animated movie texture could be superimposed on a virtual TV screen (see Figure 11), or a video camera could track people in the real world scene and then representations of those people would mimic their movements in the reconstructed model.

With its integrated autonomous 3D acquisition system, RESOLV provides a powerful technology that can be expected to play an important role in supporting a new kind of telepresence experience.

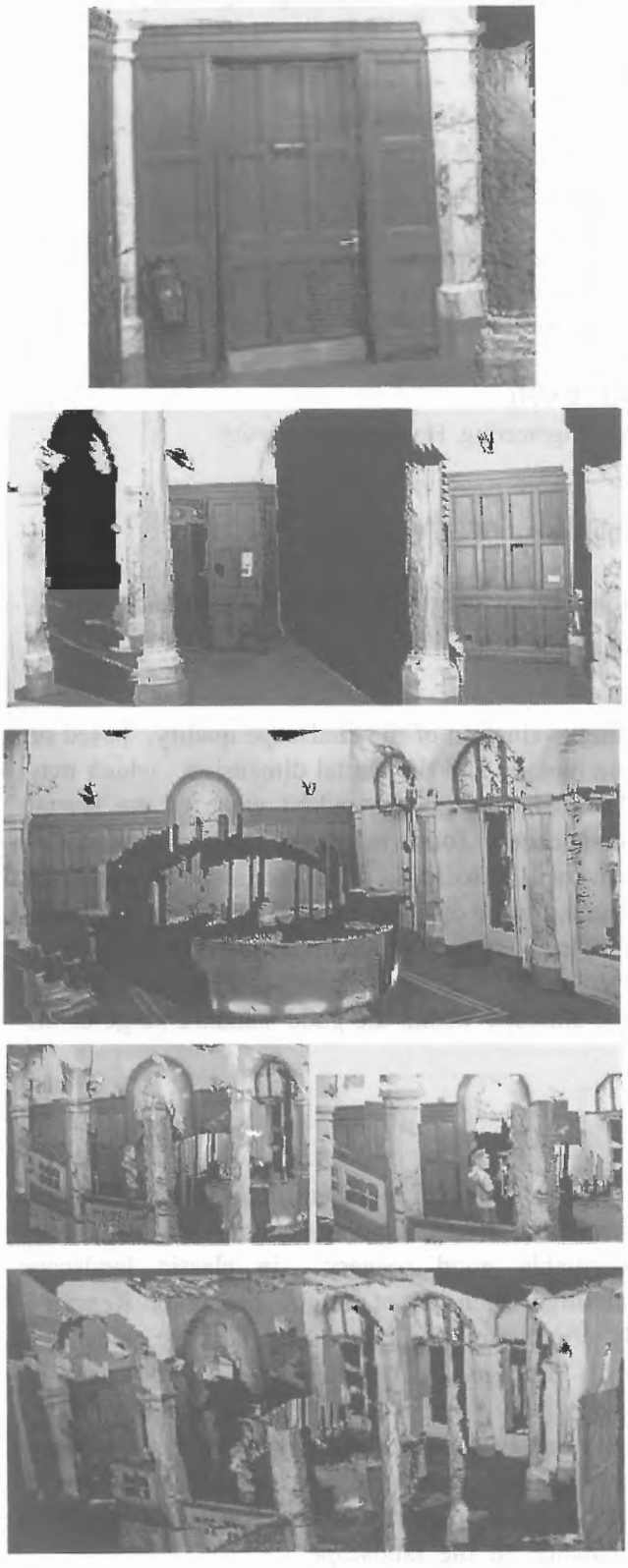


Figure 10: Several screen-snapshots from a reconstructed model of the lobby area at the Royal Institute of Chartered Surveyors, London. (http://mortimer.jrc.it/3DRoyal.html)

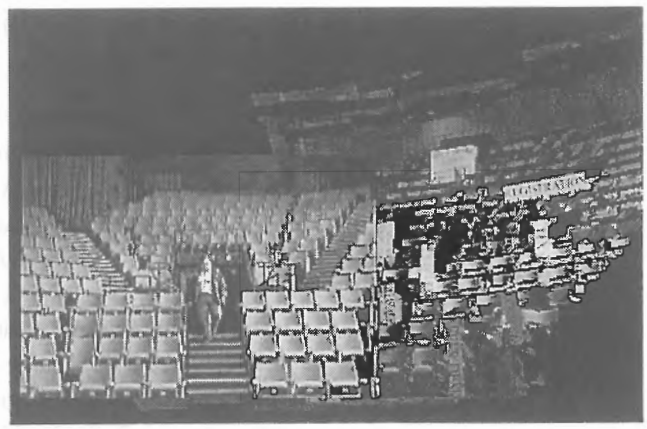


Figure 11: Reconstructed model of the University of Leeds Conference Auditorium with superimposed animated movie texture. (http://www.scs.leeds.ac.uk/vislib/resolv/conf_aud.htm)

7. REFERENCES

Fischler, M.A., and Bolles, R.C., 1981. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6), pp. 381 — 395.

Sequeira, V., Gonçalves, J.G.M., and Ribeiro, M.I., 1995. High-Level Surface Descriptions from Composite Range Images. In: *Proceedings IEEE Int. Symposium on Computer Vision*, pp. 163-168.

Sequeira, V., Gonçalves, J.G.M., and Ribeiro, M.I., 1996. Active View Selection for Efficient 3D Scene Reconstruction, In: *Proceedings ICPR'96 – 13th Int. Conf. on Pattern Recognition,* Vol. I - Track A-Computer Vision, pp. 815-819.

Sequeira, V., 1996b. Active Range Sensing for Three-Dimensional Environment Reconstruction, PhD Thesis, Dept. of Electrical and Computer Engineering, IST-Technical University of Lisbon, Portugal.

Sequeira, V., Ng, K.C., Butterfield, S., Gonçalves, J.G.M., and Hogg, D. C., 1998. Three-dimensional textured models of indoor scenes from composite range and video images. In: *Proceedings of SPIE, Three-Dimensional Image Capture and Applications*, edited by Ellson, R.N. and Nurre, J.H., vol. 3313.

Tsai, R.Y., 1987. A versatile camera calibration technique for high-accuracy 3D machine vision metrology using off-the-shelf TV cameras and lenses. *IEEE Journal of Robotics and Automation*, RA-3(4), pp. 323 — 344.

Weinhaus, F.M. and Devarajan, V., 1997. Texture mapping 3D models of real-world scenes. *ACM Computing Surveys*, 29(4), pp. 325 — 365.

Weng, J., Cohen, P. and Herniou, M., 1992. Camera calibration with distortion models and accuracy evaluation. In: *IEEE Trans. PAMI*, vol. 14, pp 965 — 980.