3-D Object Recognition using Passively Sensed Range Data. *

Kenneth M. Dawson and David Vernon

University of Dublin, Trinity College, Dept. of Computer Science, Dublin 2, Ireland

Abstract. Model-based object recognition is typically addressed by first deriving structure from images, and then matching that structure with stored objects. While recognition should be facilitated through the derivation of as much structure as possible, most researchers have found that a compromise is necessary, as the processes for deriving that structure are not sufficiently robust. We present a technique for the extraction, and subsequent recognition, of 3-D object models from passively sensed images. Model extraction is performed using a depth from camera motion technique, followed by simple interpolation between the determined depth values. The resultant models are recognised using a new technique, *implicit model matching*, which was originally developed for use with models derived from actively sensed range data [1]. The technique performs object recognition using secondary representations of the 3-D models, hence overcoming the problems frequently associated with deriving stable model primitives. This paper, then, describes a technique for deriving 3-D structure from passively sensed images, introduces a new approach to object recognition, tests the approach robustness of the approach, and hence demonstrates the potential for object recognition using 3-D structure derived from passively sensed data.

1 3-D Model Extraction from passively sensed images.

The extraction of a 3-D model can be performed in a series of steps: (1) computing depth using camera motion, for the significant intensity discontinuities, (2) interpolating range data between the significant intensity discontinuities, (3) smoothing of the resultant depth map and (4) deriving a 3-D model from the depth map.

1.1 Depth from Camera Motion.

In order to compute the 3-D location of a point we must obtain two vectors to that point. For passive approaches to vision those two vectors must be obtained from two separate observations of the point. However the accuracy of the resulting 3-D data is limited by the sensor resolution and the disparity between the viewpoints. Also, it is important to note that the identification of points which correspond in the two images is difficult, and that the complexity of the correspondence problem increases with the disparity between the viewpoints. Hence, although an increase in disparity increases the potential accuracy of the 3-D data, it also increases the complexity of (and hence the likelihood of error within) the correspondence problem.

* The research described in this paper has been supported by ESPRIT P419, EOLAS APT-VISION and ESPRIT P5363.

One solution to this dilemma is provided by the computation of depth using camera motion. The technique employed in this paper (see [2,3,4] for details) uses nine images taken from different positions on an arc around a fixation point. The instantaneous optic flow, representing the apparent motion of zero-crossings in each successive image, is computed from the time derivative of the Laplacian of Gaussian of each image. The global optic flow is computed by exploiting the instantaneous optic flow to track the motion of each zero-crossing point thoughout the complete sequence. This provides a vector field representing the correspondence between zero- crossing points in the initial and final image in the sequence, i.e. over an extended base-line of camera displacement. The depth of each zero-crossing is then computed by triangulation, using the start point and the end point of each global optic flow vector.

An example is shown in Figure 1 in which the third image in a sequence of nine images of a book is shown, along with its significant intensity discontinuities, the optical flow determined between the third and eighth images in the sequence, and finally the depth map which results after interpolation and smoothing. The nine images were taken with an angular disparity of approximately 2^o between successive camera positions and a fixation distance of 600mm.

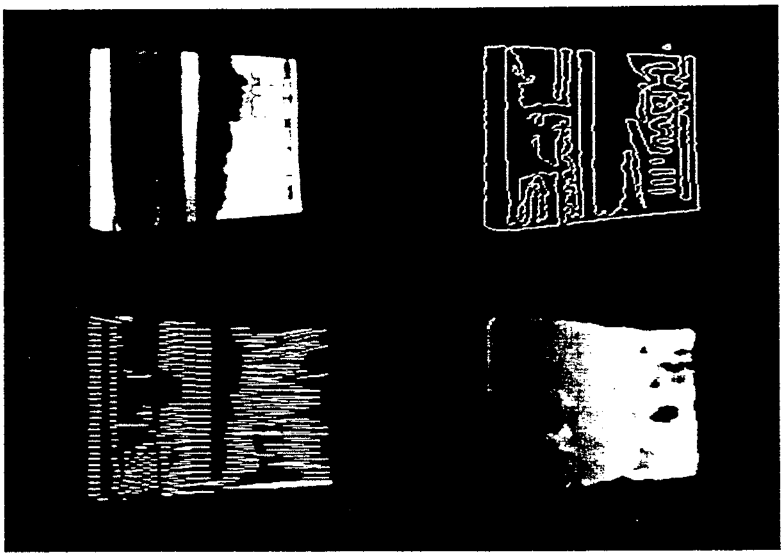

Fig. 1. Depth from camera motion. See text for details.

1.2 Interpolation of sparse range data.

The result of the previous algorithm is a sparse depth map, where depth values are known only at locations corresponding to the significant zero-crossings which were successfully tracked throughout the sequence of images. However, the purpose of this research is to investigate the potential for recognising 3-D structure dervied from passively sensed data, and hence we must interpolate between the available depth information.

The majority of interpolation techniques attempt to fit a continuous surface to the available depth information (e.g. [6]). This requires that range data be segmented into likely surfaces prior to the application of the interpolation technique, or alternatively that only a single surface be presented. We employ a simpler technique invloving planar interpolation to ensure that the surfaces are correct for polyhedral objects.

The interpolation method defines a depth value for each undefined point in the depth map by probing in five directions (to both the East and West, where East and West are parallel to the direction of motion) from that point in order to find defined depth values. A measure of those defined depth values (based also on the orientations of the features with which the depth values are associated, and the distances from the undefined point) is then employed to define the unknown depth value; e.g. for point (x, y):

$$Depth(x,y) = \frac{Range_{east} * Distance_{west} + Range_{west} * Distance_{east}}{Distance_{west} + Distance_{east}} \quad (1)$$

where $Range_{east}$ and $Range_{west}$ are the weighted average of the range values located to the east and to the west of point (x, y) respectively, and $Distance_{east}$ and $Distance_{west}$ are the average distances to those range values.

1.3 Filtering range data.

The depth map which results from the interpolation can be quite noisy. That is to say, that there can be local regions of depth values which vary significantly to those in a larger area around them. In order both to overcome this and to define values for isolated undefined points, a smoothing filter was applied to the data. A smoothing filter which simply averages all depth values within a mask was not appropriate, as resultant values would be affected by local regions of noise. This restricted the potential choice of filter considerably, and only two types of filter, median and modal, were considered. It was found, experimentally, that a reasonably large (e.g. 11x11) modal filter produced the best results, in terms of the resultant depth map, and hence 3-D structure.

1.4 Building 3-D models.

Finally, having obtained a reasonably smooth depth map, it is still necessary to convert from a viewer-centered description to an object-centered surface model. This can be done by first employing the relevant camera model, to convert from image coordinates $(i, j, depth)$ to Cartesian coordinates (x, y, z), and then deriving '3-point seed' surfaces [7] (i.e. surfaces are instantiated between any three points which are within the sampling distance of each other).

2 Object Recognition - Implicit Model Matching.

The basic problem of rigid object recognition is to establish the correspondence between an object model, which is *viewed*, and a particular *known* object model, and the computation of the associated pose. The majority of object recognition techniques match *known* object models with viewed instances of objects through the comparison of model primitives (e.g. edges). However, it is extremely unlikely that the model primitives extracted will be identical to those of a model which is *known a priori*. In order to overcome that problem it is possible to employ secondary representations, such as the Extended Gaussian Image (or EGI) [8], although using the EGI has proved difficult [9].

The technique of implicit model matching introduces a new, more powerful, but similar idea. It employs *several* secondary representations which allow the problem to be considered in terms of sub-problems. Initially, orientations of *known* objects which may potentially match the *viewed* object are identified through the comparison of visible surface normals, for each possible orientation of each *known* object (where 'each possible orientation' is defined by the surface normals of a tesselated sphere). Potential orientations are then fine tuned by correlating 1-D histograms of specific components of surface orientations (known as directional histograms). The object position is estimated using approximate knowledge of the physical configuration of the camera system, and fine tuned using a template matching technique between needle diagrams derived from the ***known*** and *viewed* models. Finally, using normalised correlation, each hypothesis is evaluated through the comparision of the needle diagrams.

At each stage in the generation, tuning and verification of hypotheses only comparisons of the various secondary representations are employed. The central concept behind implicit model matching is, then, that 3-D object models may be reliably compared through the use of secondary representations, rather than (or, more properly, as well as) by comparison of their component primitives. Additionally it is important to note that object pose may be determined to an arbitrarily high degree of accuracy (through the fine-tuning stages), although initially only a limited number of views are considered.

2.1 Approximating Object Orientation.

The first stage in this technique is the computation of approximate orientations of a known object model which may, potentially, correspond to the viewed object model. This is achieved by considering the known object from every possible viewpoint, as defined by a tesselated sphere, and comparing directional histograms of tilt (which will be explained presently) for every viewpoint with a directional histogram of tilt derived from the viewed model. The orientations which generate locally maximum correlations between these histograms (i.e. as compared to the correlations associated with neighbouring tesselations on the sphere) may be regarded as the potentially matching orientations.

Directional Histograms. The concept of the Directional Histogram was developed as part of the technique of implicit model matching and embodies the notion of mapping a single component of the 3-D orientations of a model visible from a given viewpoint to a 1-D histogram, where the component of orientation is defined about the axes of the viewing device. Four different components are employed: roll, pitch, yaw and tilt; where roll, pitch and yaw are defined as rotations about the Z, X and Y axes of the viewing device respectively, and tilt is defined as π radians less the angle between the orientation vector and the focal axis (i.e. the Z axis of the viewing device). See Figures 2 and 3.

Resulting Orientations. The result of these comparisons of tilt directional histograms is the identification of potentially matching orientations. However, only two degrees of freedom have been constrained, as only the tilt of the object is approximated. In order to complete the approximation of orientation, we must also compute potentially matching values of roll around the focal axis of the viewing device (tilt and roll are independent).

This identification may again be performed using directional histograms, but using roll rather than tilt. Hence, for every determined value of tilt, a directional histogram of roll is derived and compared (using normalised cross correlation) with that from the viewed model in every possible value of roll. Each 'possible value' of roll is defined by the resolution of the directional histogram, and the directional histogram is simply shifted in a circular fashion in order to consider the various possible values of roll.

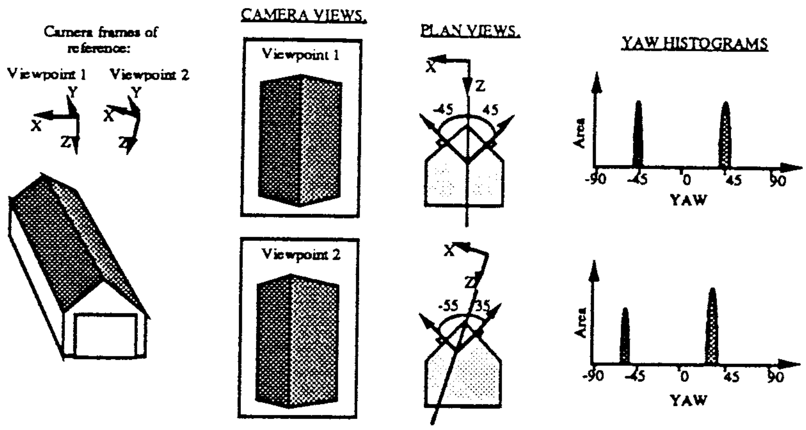

Fig. 2. Example yaw directional histograms. These two yaw histograms of two views of a garage-like object are a simple example of how directional histograms work. The visible surface areas of the views of the object are mapped to the histograms at their respective yaw angles (defined with respect to the focal axes of the camera). Notice the shift in the histograms, which is due to the slightly different values of yaw of the two viewpoints.

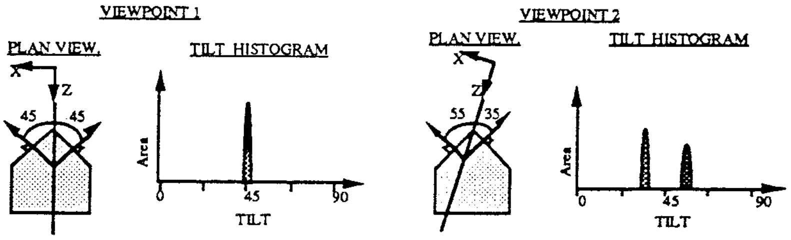

Fig. 3. Example tilt directional histograms. These two tilt histograms are derived from the two views of the garage-like object shown in Figure 2. Notice how, for the first view, the two orientations result in the same value of tilt, and in the second view how the values change.

2.2 Fine-tuning Object Orientation.

The potentially matching orientations computed can only be guaranteed to be as accurate as the quantisation of the sampled sphere. Increasing the resolution of the sphere to an arbitrarily high level, however, would obviously cause a signifcant increase in the computational overhead required in determining potential orientations. Alternatively, it is possible to fine-tune the orientations using directional histograms (of roll, pitch and yaw) in a similar fashion to the method used for the approximate determination of object roll. Pitch, yaw and roll directional histograms are derived from the view of the known object and compared with histograms derived from the viewed model. The differences between the directional histograms indicate the amount by which the orientation may best be tuned (e.g. see Figure 2). The various directional histograms are derived and compared sequentially and iteratively until the tuning required falls below the required accuracy of orientation or until the total tuning on any component of orientation exceeds the range allowed (which is defined by the quantisation of the tesselated sphere).

This stage allows the accuracy of potentially matching orientations to be determined to an arbitrarily high level (limited only by the resolution of the directional histograms). Hence, although only a limited number of possible viewpoints of any known object are considered, the orientation of the object may be determined to a high level of accuracy.

2.3 Approximating Object Position.

Turning now to the *approximate* determination of object position, it is relatively straightforward to employ the position of the viewed model with respect to its viewing camera. The imaged centroid of the viewed model, and an approximate measure of the distance of the viewed model from the viewing camera are both easily computed. The position of the camera which views the known model may then be approximated by placing the camera in a position relative to the known model's 3-D centroid, such that the centroid is at the correct approximate distance from the camera and is viewed by the camera in the same position as the viewed model's imaged centroid.

2.4 Fine-tuning Object Position.

Fine tuning object position may be considered in terms of two operations; tuning position in a directional orthogonal to the focal axis of the viewing device, and tuning of the distance of the object from the same viewing device (i.e. the depth). This separates the 3 degrees of freedom inherent in the determination of object position.

Tuning Viewed Object position (i.e. orthogonal to viewing device). This operation is performed using a template matching technique in which a needle diagram (i.e. an iconic representation of the visible local surface orientations) of the known model is compared using a normalised correlation mechanism, with a needle diagram of the viewed model. The position of the template which returns the highest correlation is taken to be the optimal position for the known model.

The standard method of comparing iconic representations is normalised cross correlation, but this form of correlation is defined only for scalars. For the comparison of needle diagrams 3-D vectors must be compared and that correlation (NV) for each possible position of the template (m, n) is defined as follows:

$$NV(m,n) = \frac{\sum_i \sum_j f(viewed(i,j)) * (\pi - angle(viewed(i,j), known(i-m)(j-n)))}{\sum_i \sum_j f(viewed(i,j)) * \pi} \quad (2)$$

where $viewed(i,j)$ and $known(i,j)$ are the 3-D orientation vectors from the ***viewed*** and ***known*** needle diagrams respectively, $f(vector)$ is 1 if the vector is defined and 0 otherwise, and $angle(vector1, vector2)$ is the angle between the two vectors (or 0 if they are undefined).

In order to make this template matching operation more efficient, the needle diagrams are first compared at lower resolutions, using a somewhat simpler measure-of-fit.

Tuning Object Depth. Fine tuning the distance between the ***known*** model and its viewing camera is done through direct comparison of the depth maps generated from both the viewed model and the known model (in its determined pose). The Depth Change Required (or DCR) is defined as follows:

$$DCR = \frac{\sum_i \sum_j f(viewed(i,j)) * f(known(i,j)) * (viewed(i,j) - known(i,j))}{\sum_i \sum_j f(viewed(i,j)) * f(known(i,j))} \quad (3)$$

where $viewed(i,j)$ and $known(i,j)$ are the depths from the ***viewed*** and ***known*** depth maps respectively, where $f(depth) = 1$ if the depth is defined and 0 otherwise. The DCR is directly applied as a translation to the pose of the camera which views the known model, in a direction defined by the focal axis of the camera. Due to perspective effects this operation will have effects on the depth map rendered, and so is applied iteratively until the DCR falls below an acceptable level.

Final tuning of Object Position. Tuning of object position is limited in accuracy primarily by the resolution used in the needle diagrams (for tuning position orthogonal to the focal axis of the viewing camera). As a final stage in tuning we attempt to overcome this by determining the position to sub-pixel accuracy. This is accomplished using a combination of template matching, normalised correlation and quadratic modelling techniques. The best position may be determined to pixel accuracy using the technique described in section 2.4. In order to determine the position to sub-pixel accuracy the normalised correlations, from the comparison of needle diagrams, around the best position (as determined to pixel accuracy) are used and are modelled as quadratics in two orthogonal directions (i.e. parallel to the two image axes).

2.5 Verifying Hypotheses.

Having hypothesised and fine tuned poses of known objects it is necessary to determine some measure of fit for the hypothesis so that it may be accepted (subject to no better hypothesis being determined), or rejected. The normalised correlation of local surface orientations (i.e. needle diagrams) between the viewed model and the known model in a determined pose, as used when fine tuning object position (see section 2.4) gives a degree-of-fit which represents all aspects of object position and orientation. This degree-of-fit, then, provides a powerful hypothesis verifcation measure.

The known model, in the computed position and orientation, which gives the best degree of fit with the viewed model (i.e. maximal correlation between derived needle diagrams), and which exceeds some predefined threshold, is taken to be the best match; the viewed model is assumed to be the corresponding object, in the pose of the known model.

2.6 An Exception.

There is, however, one situation in which this technique, implicit model matching, will fail and that is when only a single surface/orientation is visible. Computation of object roll in this instance is impossible using directional histograms (as there is an inherent ambiguity with respect to roll around the orientation vector of the surface).

This situation can be detected by considering the standard deviation of the visible orientations as mapped to an EGI (as if the standard deviation is less than a small angle then it may be taken that only one surface is visible). The problem must then be regarded as one of shape recognition, although it should be noted that it is possible to adapt the technique of implicit model matching to cope with this situation (see [10]).

3 Experimental Results and Conclusions.

The intention of the testing detailed herein is to investigate the robustness of the recognition technique, and to demonstrate the potential for recognising 3-D models derived from passively sensed data. The objects employed were all of simple rigid geometric structure, and scenes contained only one object. The rationale for these choices is that the segmentation and identification of occlusion problems, etc., still require much further research.

As an example of recognition, consider the book shown Figure 1. The model determined is quite accurate, with the main exception being the title. Regardless of these errors, however, sufficient of the model is computed correctly to allow reliable identification of the book (See Figure 5) from the database of objects (See Figure 4).

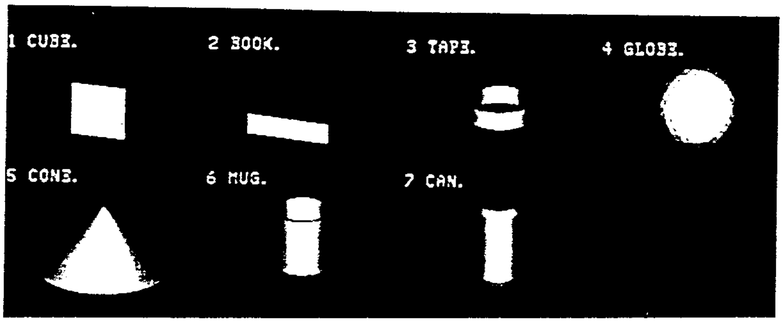

Fig. 4. The database of *known* object models.

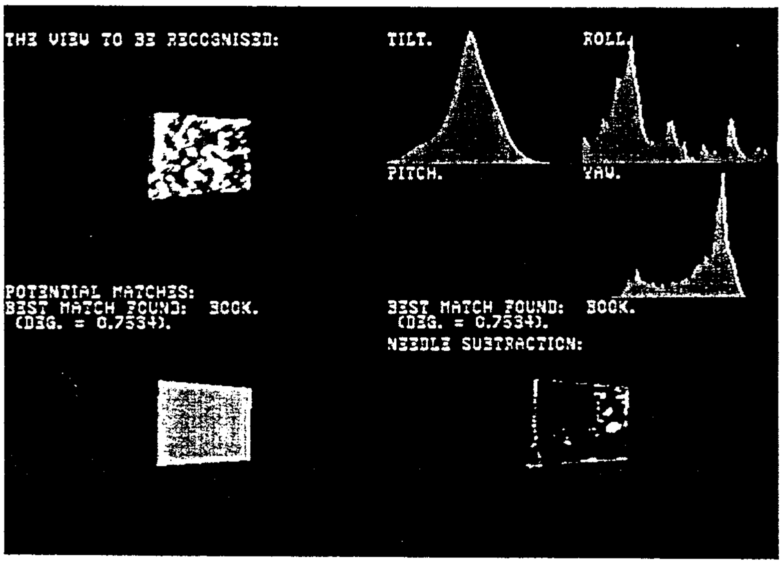

Fig. 5. Recognition of the model derived from the depth map shown in Figure 1.

The discrimination between the various recognition hypotheses is not that significant however, and as more complex objects are considered (See Table 1) the discriminatory ability gets progressively worse resulting, eventually, in mistaken recognition. The testing allows a number of conclusions to be drawn, which follow, and satisfies both of the stated intentions of this research.

1. It is demonstrated that there is potential for the recognition of objects in passively sensed images on the basis of derived 3-D structural information.

2. Implicit model matching was found to degrade reasonably gracefully in the presence of noisy and incorrect data. However, its performance with models derived from actively sensed range data [1] was significantly more reliable.
3. Finally, the limitations on the techniques presented for the development of three dimensional models are emphasised. The visual processing performed in this research quite obviously deals in a trivial way with many important issues, and these issues remain for future research.

Further details of all aspects of the system described in this paper are given in [4].

		Known models and best d.o.f.s							
Scene	**Figure**	Cube	Cone	Book	Mug	Can	Globe	Tape	**Result**
Cube		0.8229	0.6504	0.7220	0.6810	0.5270	0.6344	0.3656	Cube
Cone		0.2508	0.7311	0.5130	0.2324	0.1962	0.3777	0.1332	Cone
Book	5	0.2763	0.5432	0.7534	0.2958	0.2189	0.3412	0.1883	Book
Mug		0.6872	0.6061	0.5770	0.6975	0.5892	0.6580	0.3276	Mug
Pepsi Can		0.6643	0.5829	0.7099	0.6705	0.6852	0.5150	0.3658	Book *
Globe		0.4492	0.5620	0.5789	0.4036	0.3305	0.5725	0.2897	Book *
Sellotape		0.5706	0.6270	0.6689	0.4979	0.5039	0.4295	0.3735	Book *

Table 1. The complete table of the degrees-of-fit determined between *viewed* instances of objects and the *known* models.

References

1. Dawson, K. Vernon, D.: Model-Based 3-D Object Recognition Using Scalar Transform Descriptors. Proceedings of the conference on Model-Based Vision Development and Tools, Vol. 1609, SPIE - The International Society for Optical Engineering (November 1991)
2. Sandini, G., Tistarelli, M.: Active Tracking Strategy for Monocular Depth Inference Over Multiple Frames. IEEE PAMI, Vol.12, No.1 (January 1980) 13-27
3. Vernon, D., Tistarelli, M.: Using Camera Motion to Estimate Range for Robotic Part Manipulation. IEEE Robotics and Automation, Vol.6, No.5 (October 1990) 509-521
4. Vernon, D., Sandini, G. (editors): Parallel computer vision - The VIS a VIS System. Ellis Horwood (to appear)
5. Horn, B., Schunck, B.: Determining Optical Flow. Artificial Intelligence, Vol.17, No.1 (1981) 185-204
6. Grimson, W.: From Image to Surfaces: A Computational Study of the Human Early Visual System. MIT Press, Cambridge, Massachusetts (1981)
7. Faugeras, O., Herbert, M.: The representation, recognition and locating of 3-D objects. International Journal of Robotics Research, Vol. 5, No. 3 (Fall 1986) 27-52
8. Horn, B.: Extended Gaussian Images. Proceedings of the IEEE, Vol.72, No.12 (December 1984) 1671-1686
9. Brou, P.: Using the Gaussian Image to Find Orientation of Objects. The International Journal of Robotics Research, Vol.3, No.4 (Winter 1984) 89-125
10. Dawson, K.: Three-Dimensional Object Recognition through Implicit Model Matching. Ph.D. thesis, Dept. of Computer Science, Trinity College, Dublin 2, Ireland (1991)

This article was processed using the LaTeX macro package with ECCV92 style