ELSEVIER

Image and Vision Computing 14 (1996) 465–472

image AND vision COMPUTING

Realistic range rendering for object hypothesis verification

Patrick J. Flynn*

School of Electrical Engineering and Computer Science, Washington State University, Pullman, WA 99164-2752, USA

Received 13 March 1995; revised 27 September 1995; accepted 27 September 1995

Abstract

In many model-based object recognition systems, a synthesize-and-verify technique is used to evaluate the quality of hypotheses. This technique synthesizes images of hypothesized objects in hypothesized poses, and compares them against the input imagery, producing a matching score. In this paper, we examine the image synthesis process in the context of triangulation-based range finding. We motivate the use of synthetically shadowed range data for verification, present a simple algorithm for generation of shadowed range imagery, and demonstrate its usefulness in a set of experiments on real imagery.

Keywords: CAD-based vision; Experimental computer vision; Object recognition; Sensor modeling; Range image

1. Introduction

A critical component of many three-dimensional object recognition systems is a mechanism to quantitatively evaluate the goodness of a hypothesis, which often is composed of a number of bindings between geometric primitives from one of the models in the object database and compatible primitives in the scene whose contents are being identified. This verification task can involve synthesis of an image of the object in the hypothesis with a hypothesized pose; the numerical 'goodness' score is then computed by comparison between the real image under consideration and the synthetic image. Among the systems which employ this synthesize/compare approach to verification are 3DPO [1], BONSAI and recent variants [2,3], and the system developed by Hansen and Henderson [4]. These scores are used to identify the 'best' of the set of hypotheses under examination; this best element is typically reported as the system's decision on the identity of a component of the scene. In BONSAI and its descendants these scores are normalized; hence, in addition to the relative character of the score (higher scores reflect more confidence), an absolute connotation exists and can be used to (for example) establish a minimum score level below which no hypothesis will be deemed acceptable.

Obviously, the sensing model used to produce the synthetic imagery in these systems must correspond closely to the real sensor in use for verification to perform reliably. Explicit attention to sensor models in object recognition systems has recently become a topic of interest, as evidenced by work on vision algorithm compilers incorporating sensor-specific visibility in precompiled recognition strategies [5,6]. The type of sensor, its configuration, the objects being viewed (in particular, their material properties), and ambient environmental conditions jointly define what can and cannot be seen in a typical view of the objects. For images obtained from a photometric stereo sensor, surface points must be visible to the intensity sensor as well as each illumination source. Similarly, only surface points which receive active illumination and are visible to the intensity camera yield estimates of range in a light-stripe-based range finder.

In this paper, we describe an algorithm for producing range images of objects described as polyhedral meshes of arbitrary density; these images will reflect the self-shadowing artifacts arising in light-stripe range finders. After describing the algorithm, we demonstrate the utility of such 'realistic' synthetic imaging on the verification stage of a CAD-model-based 3D object recognition system.

2. Motivation: rendering-based verification

Consider a correspondence-based 3D object recognition system which employs solid models of the objects to be recognized and range imagery as the sensing

* Email: flynn@eecs.wsu.edu

SSDI 0262-8856(95)01063-7

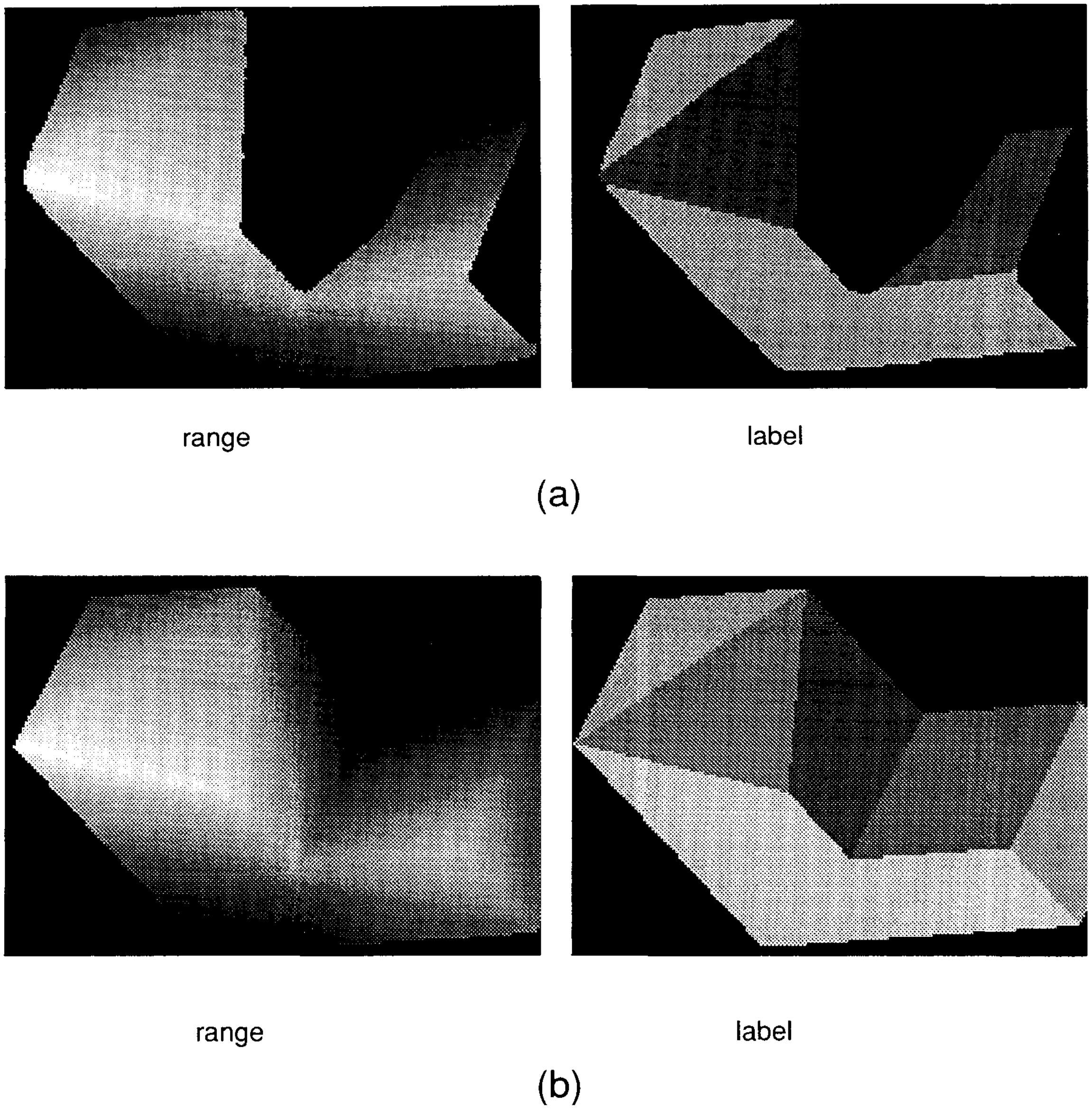

Fig. 1. Real (a) and synthetic (b) range images of a polyhedral object (left), and segmentations (right).

modality. Such systems build a set of correspondences (or bindings) between reliably-detectable entities (e.g. points, curves, or surfaces) in the scene and compatible entities in one of the object models in order to recognize that object. Many such systems employ a hypothesize-and-test approach to recognition, where a pose transformation (which aligns the entities involved in the current set of bindings) is estimated and used to validate the bindings. It has often been observed (particularly in situations where there are a large number of entities in the scene) [2,7,8] that only a few bindings are necessary for estimation of pose, although the quality of the pose estimate is of course affected by the number of bindings used in its computation.

Once a pose estimate is available, many recognition techniques invoke a verification procedure to evaluate the goodness of existing bindings in the hypothesis, and perhaps acquire additional bindings [2]. Most systems apply the estimated pose to the geometric primitives composing the hypothesized model. In some systems, the scene is then searched for these transformed entities. In others (including the system used in this paper), the transformed model entities are then used to generate a synthetic image of the object using the same view specification and resolution as that of the input imagery, producing a synthetic image which is ‘registered’ in 3D with the input image. The quality of the hypothesis is then calculated by comparisons on a pixel-by-pixel basis between the input and synthesized images.

For the remainder of this paper, we will assume that range imagery is used and our attention will focus on the generation of imagery that exhibits artifacts similar to those inherent in a triangulation-based range finder. The two most common techniques for rendering synthetic range imagery from 3D model descriptions are polygon rendering (in essence retaining the Z-buffer of a polygon scan-converter) and ray-casting. Since we have high-quality polyhedral approximations to all of our object models, we will employ scan-conversion here, since this rendering method is generally faster than even first-hit ray tracing.

3. Scan-conversion range rendering

Algorithms for scan-converting polygons (2D or 3D) into an image buffer are already well-known in the computer graphics community [9,10], and do not need detailed explanation here. What is atypical about the use of polygon scan-conversion for range image synthesis is the absence of a 'visible' intensity buffer. The depth buffer (used to cache the frontmost surface's depth at each pixel location) contains a range image after the algorithm is complete. We also need to identify the visible polygon at each pixel. Hence, the output of the scan converter is a range image and a corresponding segmentation image.

3.1. Rendering triangulation shadowing

Polygon scan-conversion produces excellent range data for polyhedra and objects with curved surfaces (the density of the polyhedral approximation of curved surfaces determines the faithfulness of the synthetic range data); indeed, the data is unrealistically good when compared with images taken from triangulation-based range finders. Fig. 1 shows real (part (a)) and synthetic (part (b)) range and segmentation images of a polyhderal object. The real range image (Fig. 1a, left) contains a substantial amount of shadowing. Indeed, one face of the object has been obscured entirely. The images in Fig. 1(b) were synthesized by the BONSAI/IFI object recognition system [3] from a 'correct' hypothesis of identity for three of the surfaces extracted from the input data.

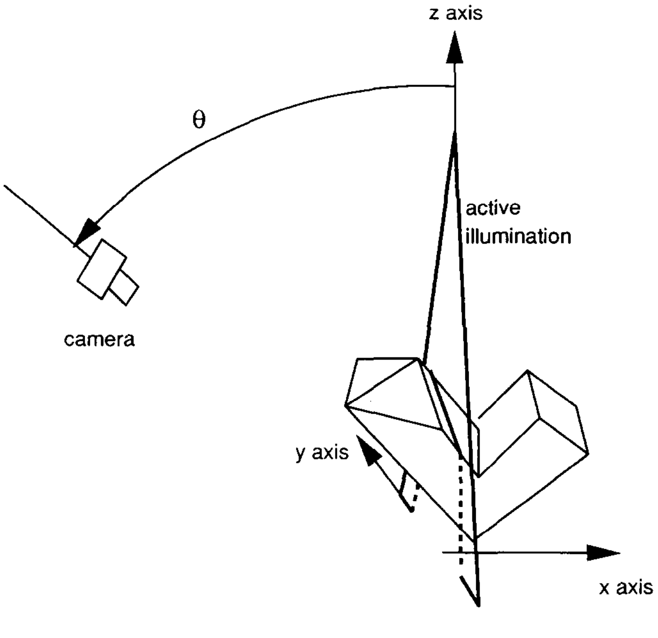

Fig. 2. Triangulation range sensing geometry.

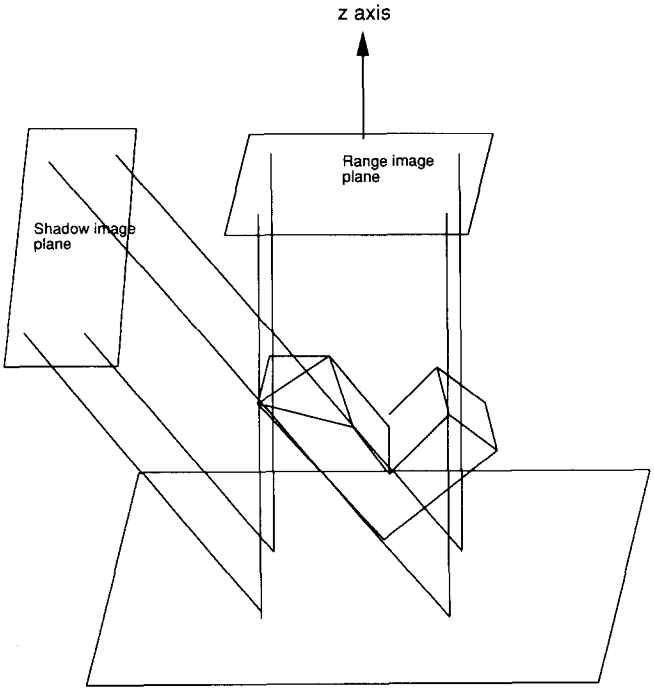

Fig. 3. Configuration of image planes for synthetic triangulation shadowing.

The discrepancy between these two images is due almost entirely to the triangulation shadowing effect: at some locations, object surfaces were either actively illuminated but not visible to the intensity sensor, or visible to the sensor but not illuminated. In our experience, the former effect is most prevalent.[1] Fig. 2 shows the placement of the light source and intensity sensor in the range finding system used to produce our images. The angle θ ($\pi/4$ in our system), along with the geometry of the scene, determines the degree of shadowing present in images of the scene.

To simulate the effects of shadowing in a polygon-based renderer, a second rendering step is required; this additional step places the view direction parallel to the camera's optical axis. A subsequent scan-conversion of the object using this view direction yields an additional range and label image, which we call the shadow buffer range and label images. This second rendering is

[1] The triangulation shadowing effect can be exploited for background pixel removal; if the object being imaged is placed on a light-absorbing surface (e.g. black felt), the laser stripe on that surface will not be seen by the camera and that pixels will be considered 'shadowed'.

```
Input: polyhedral object (a set of polygons) $\mathscr{P} = \{\mathbf{P}_i\}$
Input: angle $\theta$ between camera axis and $z$ axis of range image (see Fig. 2)
Output: range image $\mathscr{R} = \{(x_{ij}, y_{ij}, z_{ij}), (i, j) \in [0 \ldots N_r - 1] \times 0 \ldots N_c - 1]\}$
    ($N_r$ and $N_c$ are the number of rows and columns in the image, respectively)
Output: label image $\mathscr{L} = \{l_{ij}, (i,j) \in [0 \ldots N_r - 1] \times [0 \ldots N_c - 1]\}$

Allocate $\mathscr{R}$, $\mathscr{L}$, and temporary images $\mathscr{R}_s$ and $\mathscr{L}_s$ to hold the range and label images rendered from the camera's viewpoint.
Scan-convert $\mathscr{P}$ into $\mathscr{R}$ and $\mathscr{L}$.
Form $\mathscr{P}' = \mathbf{R}_y(\theta)\mathscr{P}$ using the same image plane specifications as used for $\mathscr{R}$.
    ($\mathbf{R}_y(\theta)$ denotes a transformation that rotates the object by an angle of $\theta$ about the $y$-axis).
Scan-convert $\mathscr{P}'$ into $\mathscr{R}_s$ and $\mathscr{L}_s$.
FOR EACH pixel $(i, j)$
    Compute $[x'y'z'] = \mathbf{R}_y(-\theta)[x_{ij}\, y_{ij}\, z_{ij}]$.
    Compute $(i', j')$, the image row and column number closest to $(x', y')$ in $\mathscr{R}_s$.
    IF $l_{s,i'j'} \neq l_{ij}$, $l_{ij} \leftarrow 0$, i.e. mark pixel $(i, j)$ as shadowed.
END FOR

END
```

Fig. 4. Triangulation shadowing algorithm.

performed with the same pixel spacing and resolution as the first (only the view direction is different). Fig. 3 illustrates the configuration of the two image buffers. To determine whether a particular range pixel in the output image is shadowed, its 3D coordinates are transformed into shadow buffer coordinates, and the corresponding labels (surface indices) are examined. If these surface labels do not agree, then the face in the shadow buffer occludes the face in the range buffer, and the pixel is marked as a shadow pixel. If the labels agree, the pixel is visible, and its label is not modified. Fig. 4 contains a pseudocode version of this algorithm.

Fig. 5 shows the effect of shadowing on the synthetic range and segmentation images of the hypothesis in Fig. 1(b). Note the close correspondence between the synthetic image and the original input image (Fig. 1(a)). The missing pixels at segment edges are an artifact of the process which checks for identical label values at corresponding pixels in the two range images; near edges, rounding can place a pixel into either of the segments defining the edge. A postfiltering operation could be applied to assign labels to these 'missing' pixels but their presence has a minimal effect on the scores calculated below and the filtering operation would consume processing time. For that reason, the missing pixels along edges are ignored in subsequent processing.

4. Verification with 'realistic' range data

Our primary motivation for incorporating shadowing into synthetic range image generation is twofold.

1. During model-building (the 'compile-time' phase of a typical model-based recognition system), synthetic range images (and accompanying segmentations) of object models are generated to accumulate lists of visible surface areas for different views of the objects.

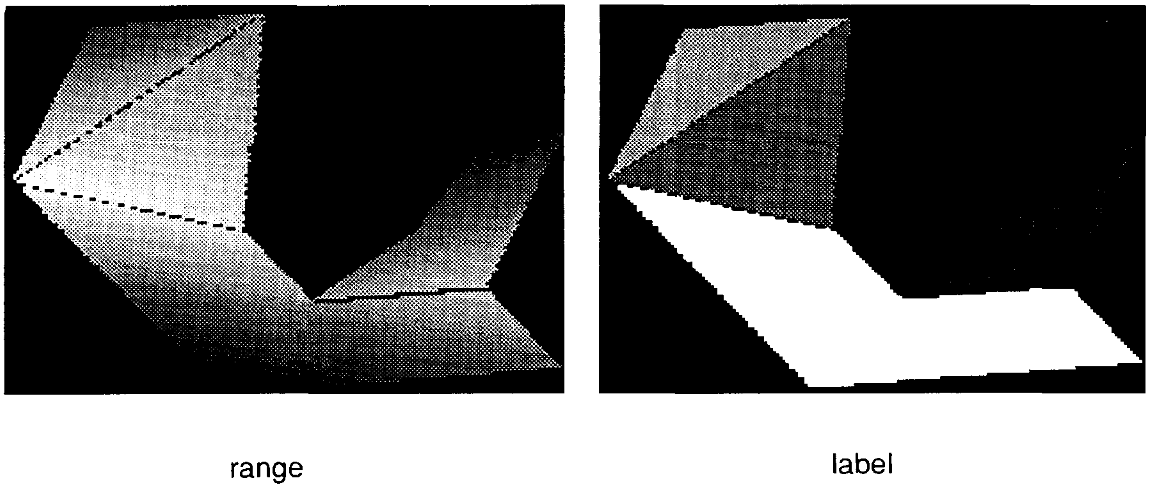

Fig. 5. Synthetic image of a polyhedron with simulated triangulation shadowing.

It is important for the images used in this estimation step to be as representative as possible of the true sensor.

2. When performing object recognition (the 'run-time' phase), hypothesis verification employs synthetic imagery of the hypothesized object and pixel-by-pixel comparison between the input and synthetic images. Synthetic imagery should reflect the sensor's properties in order for the matching 'scores' to be accurate.

We now present a verification example which demonstrates the utility of synthetic range shadowing in separating correct hypotheses from incorrect. The verification system in use was originally developed for the BONSAI CAD-based vision system [2], adapted for use with a recognition system based on invariant feature indexing [3], and is summarized as follows. Synthetic range and segmentation images of the hypothesized model in the hypothesized pose are generated and used to produce the six matching scores $s_1, \ldots, s_6$, each between zero and one. The overall matching score for a hypothesis is

$$s = \prod_{i=1}^{6} s_i.$$

The subscores were determined empirically based on a large set of experiments with real range data, and attempt to capture the several different ways that hypotheses and the resultant images can be considered 'close' to an input image:

1. s_1 measures discrepancies in the depth at each pixel. Define N_1^+ as the number of pixels in the input range image within 0.1 inch of the predicted value,[2] and N_1^- as the number of pixels where the predicted value is larger (i.e., closer to the sensor) than the sensed value. Intuitively, we would like N_1^+ to be large and N_1^- to be small. s_1 is defined as $N_1^+/(N_1^+ + N_1^-)$. If the predicted depth is further from the sensor than the measured depth, s_1 is not affected as this could be caused by occlusion.
2. s_2 is a product of subscores, each calculated from observed overlaps between segments in the input and synthetic segmentation images. For each pair of segment labels (i, j), let t_{ij} be the number of locations in the label images where surface i in the real image coincided with label j in the synthetic image. If the hypothesis under consideration does indeed contain a correspondence between model surface j and scene surface i, the subscore for (i, j) is $t_{ij}^2/(N_{M_j} N_{S_i})$, where N_{M_j} and N_{S_i} are the number of pixels with the appropriate label in the synthetic and real segmentation images, respectively. If (S_i, M_j) is not present in the current hypothesis, the contribution to s_2 is $1 - t_{ij}^2/(N_{M_j} N_{S_i})$. Hence, these subscores reward large correct overlaps and small incorrect overlaps, and penalize missing or large incorrect overlaps between segments in the two segmentation images.
3. s_3 is a global overlap score, which is simply the sum of the 'correct' overlap populations divided by the sum of all overlap populations. Unlike s_2, this overlap score is not penalized by small 'incorrectly-overlapping' regions.
4. s_4 measures the proximity of the estimated areas of segments in the synthetic range image to the corresponding areas in the input range image. Let (S_i, M_j) be a binding in the current hypothesis, and A_{S_i} and A_{M_j} be the estimated areas of S_i and M_j, respectively.[3] If the ratio

$$r = \frac{A_{S_i}}{A_{M_j}}$$

is less than 1.0, the contribution of that binding to s_4 is r. If $1 < r < 2$, the contribution to s_4 is $1 - r$. If $r > 2$, the hypothesis is rejected. The final value of s_4 is the product of the individual contributions from each binding.
5. s_5 summarizes the closeness between the number of valid (nonzero) pixels in the input and synthetic segmentations. If N_i is the number of valid pixels in the input segmentation and N_s is the number of valid pixels in the synthetic segmentation, then

$$s_5 = \begin{cases} \frac{N_i}{N_s} & \text{if } N_i < N_s \\ \frac{N_s}{N_i} & \text{if } 1 < \frac{N_i}{N_s} < 2 \\ 0 & \text{if } \frac{N_i}{N_s} > 2 \end{cases}$$

The last clause in the conditional form for s_5 discards hypotheses where the predicted image of the hypothesized model occupies more than twice as many pixels in the synthetic segmentation as the object in the input image.
6. s_6 measures the number of times in the segmentation images where an invalid pixel in one corresponds to a valid pixel in the other. Let N_0 be the number of pixel locations where the synthetic segmentation has a valid pixel but the input segmentation does not, and N_1 be the number of locations where the input segmentation is valid but not the synthetic segmentation. Then

$$s_6 = \frac{(N_i - N_0)(N_s - N_1)}{N_i N_s}$$

[2] 0.1 was an experimentally determined value and depends most critically on sensor noise.

[3] Segment areas are measured by accumulating the areas of small triangles formed from 2 × 2 pixel blocks in the range image which share the segment label.

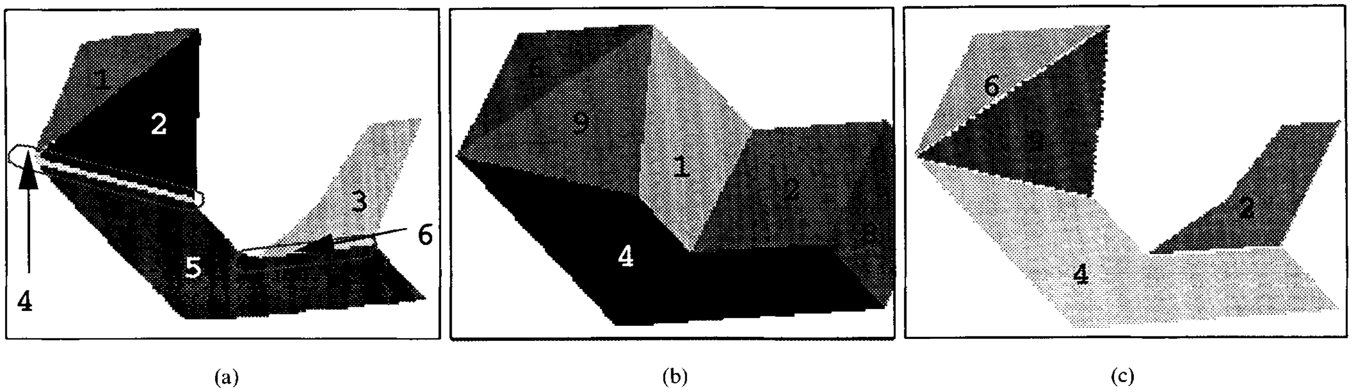

Fig. 6. Input segmentation, (a) a synthetic segment image with no shadowing, (b) and a synthetic label image with shadowing, (c) for a correct hypothesis of identity and pose. Segments 4 and 6 in part (a) lie along object edges.

We have not yet conducted a thorough analysis of the sensitivity of these scores to small perturbations in the pose estimate. However, the sensitivity can be characterized informally as follows. The depth discrepancy matching score (s_1) is most sensitive to errors in the rotation component of pose. The threshold value (0.1 inch in our experiments) can be increased to relax the restrictiveness of the classification as 'close' or 'not close' in depth at each pixel, at the cost of inflating the score for synthetic images from incorrect hypotheses, which often contain bands of pixels which meet the 'close' criterion essentially by accident. In addition, errors in the z component of the translation portion of the pose estimate can also affect s_1. The remaining scores are not as sensitive to small perturbations in translation or rotation, since they are based on either the overlap and correspondence between labels in the real and synthetic segmentations, or the presence or absence of valid pixels in the two images. The effect of a perturbation is to depress the scores through mismatches of edges between segments or between a segment and the (invalid) background pixels neighboring it. An experimental or theoretical study of the extent of this effect remains to be conducted.

In Fig. 6, we show segmentation (segment label) images generated from a data-driven segmentation procedure applied to the image in Fig. 1, and two label images generated from correct hypotheses of object identity and pose (the hypotheses were obtained from the invariant feature indexing system described in [5]). Obviously, one of the hypothesis images was generated without shadowing, and one incorporates shadowing closely approximating that of the original range sensor.

What effect does shadowing have on the hypothesis scores s_i described above? In the context of this example,

- s_1 rose from 0.868 (unshadowed image) to 0.885 (shadowed image). Both N_1^+ and N_1^- dropped slightly due to the shadowing, but the fractional decrease in N_1^- was larger.
- s_2 rose from 0.359 (unshadowed) to 0.497 (shadowed). Since shadowing completely removed the model segment labeled 1 in Fig. 6(b), and the segment-wise overlap fractions remained in the neighborhood of 0.8 in both images, the lack of a pair of matching segments inflated the product of matches.
- s_3 rose from 0.848 to 0.857. This small increase reflects only some small edge effects (perhaps caused by slightly different pose estimates).
- s_4 rose from 0.357 to 0.622. This dramatic jump occurred because predicted and observed segment areas were much closer when the shadow image was used. This reinforces our visual impression that our shadowing algorithm reproduces the shadowing artifacts in triangulation sensing.
- s_5 rose from 0.324 to 0.862; since the synthetic shadowing procedure produces segments much closer in size and shape to those observed in the scene, the N_i and N_s values are closer, making the ratio closer to 1.0.
- s_6 rose from 0.338 to 0.804. Since shadowing produces fewer valid pixels, and the pixels rendered invalid by shadowing do correspond to truly hidden pixels in the input range data, this increase is expectable.

To summarize this experiment, the introduction of shadowing into the synthetic images generated for verification has a measurable effect on the matching scores. The final score for a correct hypothesis with unshadowed synthetic imagery was 0.0103. When shadowing was added to these images and the scores recomputed, the final score was 0.162.

5. Experiments

We conducted a set of verification experiments with the proposed method for generating shadowed range data. Five real range images of thirteen different 3D objects (a total of 65 images) were segmented and given to the invariant feature indexing system for object

Table 1
Statistics for improvement of matching scores in experiments with real range data

Model name	# views used	correct/ incorrect	# hypotheses	average score (no shadowing)	standard deviation	average score (shadowing)	standard deviation	increase (percent)
adapter	5	correct	5	10.32E-3	0	31.76E-3	0	207
		incorrect	48	784.0E-6	1.335E-3	1.233E-3	2.949E-3	57
agpart2	2	correct	1	571.0E-6	0	2.269E-3	0	297
		incorrect	10	101.0E-6	88.00E-6	1.131E-5	1.939E-3	1017
bigwye	1	correct	3	1.609E-3	1.094E-3	3.928E-3	1.513E-3	144
		incorrect	15	193.0E-6	447.0E-6	624.0E-6	1.445E-3	223
block1	5	correct	16	61.56E-3	35.59E-3	69.81E-3	40.39E-3	13
		incorrect	441	1.546E-3	4.059E-3	1.626E-3	4.176E-3	5
block2	3	correct	4	10.14E-3	3.315E-3	52.66E-3	14.19E-3	420
		incorrect	69	2.889E-3	4.668E-3	3.236E-3	5.178E-3	12
block4	5	correct	18	40.79E-3	15.67E-3	41.97E-3	12.53E-3	3
		incorrect	13	1.627E-3	1.259E-3	1.760E-3	1.294E-3	8
box2inch	5	correct	114	87.18E-3	26.83E-3	89.60E-5	27.57E-3	3
		incorrect	865	2.217E-3	3.014E-3	2.509E-3	3.268E-3	13
column1	5	correct	68	15.40E-3	6.899E-3	27.46E-3	15.45E-3	78
		incorrect	1216	1.439E-3	2.907E-3	1.895E-3	3.486E-3	32
column2	5	correct	3	32.21E-3	0	59.68E-3	0	85
		incorrect	60	3.695E-3	11.13E-3	3.988E-5	11.66E-3	8
curvblock	5	correct	57	14.21E-3	16.28E-3	19.14E-3	20.28E-3	35
		incorrect	4618	170.0E-6	1.353E-3	171.0E-6	1.399E-3	1
grnblk3	5	correct	21	59.44E-3	31.92E-3	75.55E-3	37.32E-3	27
		incorrect	523	825.0E-6	3.339E-3	800.0E-3	3.060E-3	−3
hump	5	correct	17	4.801E-3	1.205E-3	7.102E-3	4.458E-3	48
		incorrect	170	2.851E-3	6.039E-5	4.626E-3	11.63E-3	62
taperoll	3	correct	6	24.40E-3	108.0E-6	45.16E-3	99.00E-6	85
		incorrect	0	N/A	N/A	N/A	N/A	N/A
Average	–	correct	333 (total)	45.60E-3	17.89E-3	52.62E-3	21.67E-3	15
	–	incorrect	8048 (total)	811.9E-6	2.240E-3	963.2E-6	2.515E-3	19

recognition described in [3]. Of these images, 12 generated either no hypotheses (because of an insufficient number of features or mis-estimation of feature attributes), or a list containing no correct hypotheses; these images were ignored in further testing. Of the 54 remaining images, we measured the net increase in matching score obtained from synthetic shadowing. Table 1 shows the average net increase in matching scores for both correct and incorrect hypotheses on a model-by-model basis and for the entire set of images. The average scores in each line of the table are weighted by the number of correct or incorrect hypotheses returned by the system; the summary line is likewise a weighted sum of averages for each model. For five of these 65 experiments, the top-ranked hypothesis was incorrect when verification did not employ the shadowing procedure, but correct when shadowing was incorporated. These results demonstrate that shadowing does increase the matching scores for both correct and incorrect hypotheses in most cases. These increases are quite large in some situations. In effect, the technique helps to differentiate correct hypotheses from incorrect ones, and as such is a valuable addition to a model-based object recognition system.

6. Conclusions and future work

We have motivated the use of synthetically 'shadowed' range data in the verification step of model-based object recognition and presented a simple algorithm for generating such imagery. Experiments on real data indicate that shadowing raises the numerical scores calculated from verification of correct hypotheses without a major effect on the scores of incorrect hypotheses. There are certainly additional sources of error in synthetically generated range data that reduces their ability to be compared with the real images they are expected to be verified against; misregistration (arising from poor pose estimates) and sensor noise (typically contaminating the z or depth coordinate of the range image) are both notable effects that this system does not take into account. It can be argued that these two effects either cannot be modeled well for the purposes of synthesis and verification, or that the verification step is not the correct place to remedy them.

In future research, we will conduct a more thorough set of experiments with both real and synthetic input imagery, and integrate the synthetic shadowing technique into our existing object recognition software environment.

Acknowledgements

This research was supported by NSF grants CDA-9121675 and IRI-9209212 and by the Washington State University Research Grant-In-Aid program. The author thanks Dr John C. Hart and two anonymous reviewers for helpful suggestions and comments. A preliminary version of this paper appears in *Proc. 1994 IEEE Computer Society Conf. on Computer Vision and Pattern Recognition (CVPR '94)*, Seattle, June 1994, pp. 848–851.

References

[1] R.C. Bolles and P. Horaud, 3DPO: A three-dimensional part orientation system, Int. J. Robotics Research, 5(3) (Fall 1986) 3–26.

[2] P.J. Flynn and A.K. Jain, BONSAI: 3D object recognition using constrained search, IEEE Trans. Pattern Analysis and Machine Intelligence (October 1991) 1066–1075.

[3] P.J. Flynn and A.K. Jain, 3D object recognition using invariant feature indexing of interpretation tables, Computer Vision, Graphics, and Image Processing: Image Understanding, 55(2) (March 1992) 119–129.

[4] C. Hansen and T. Henderson, CAGD-based computer vision, IEEE Trans. Pattern Analysis and Machine Intelligence, 11(11) (November 1989) 1181–1193.

[5] K. Ikeuchi and T. Kanade, Applying sensor models to automatic generation of object recognition programs, Proc. 2nd Int. Conf. on Computer Vision, pp. 228–237, December 1988.

[6] K. Gremban and K. Ikeuchi, Appearance-based vision and the automatic generation of object recognition programs, in A.K. Jain and P.J. Flynn (eds.), Three-Dimensional Object Recognition Systems, Elsevier, 1993, pp. 229–258.

[7] W.E.L. Grimson, Object Recognition by Computer, MIT Press, 1990.

[8] C. Chen and A. Kak, A robot vision system for recognizing 3-D objects in low-order polynomial time, IEEE Trans. Systems, Man, and Cybernetics, 19(6) (November/December 1989) 1535–1563.

[9] D.F. Rogers, Procedural Elements for Computer Graphics, McGraw-Hill, 1985.

[10] J. Foley, A. van Dam, S. Feiner and J. Hughes, Computer Graphics: Principles and Practice, Addison-Wesley, 2nd ed., 1990.