

Adaptive Matching Using Object Modes Generated from Photometric Stereo Images

G. Bellaire, K. Schlüns, A. Mitritz, K. Gwinner

Institut für Tech. Informatik, TU Berlin, Sekr. FR 3-11
Franklinstr. 28-29, 10587 Berlin, GERMANY
{bellaire,karsten}@cs.tu-berlin.de

Abstract. We present a complete object recognition system for 3-D objects using a viewer-centered object description, so-called surface normal images (SNIs), recently introduced by Park et al. [3]. Based on this representation we utilize a weak active technique (the Photometric Stereo Method (PSM)) to extract 3-D features from the objects. We combine surface orientations with an approximated line drawing to build 2.5-D models.

Furthermore we develop an accumulator based matching method, which is adaptive and tolerant regarding the measurement errors. This includes a module to analyze the composition of the actual object library, that supports the construction of the index hierarchy. An effective technique is proposed that combines the results of the sequential feature matching of the rotated 2.5-D scene model set. Both the reconstruction level and the matching level of the object recognition system were tested successfully with synthetic and real object data bases.

1 Introduction

Viewer-centered approaches [2] develop models that represent the information visible from a certain view point. The projective relation between the 3-D object and the image is not considered during the matching process.

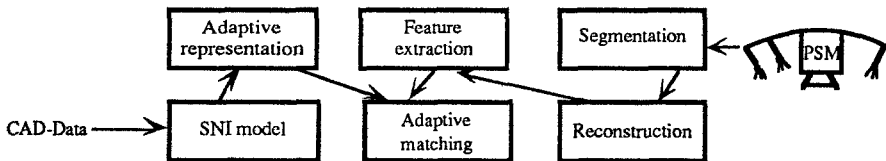


Fig. 1. Instances of our object recognition system.

The surface normal image (SNI) model [3] is a new and promising representative of this approach. It groups a set of views defined as follows: A view is represented as a 2-D line drawing. In the SNI set each view is a normal view. A normal view is defined by the alignment of the surface normal of a base face parallel to the line of sight. A SNI model of a 3-D object contains each object face as base face exactly once. In the scene domain a SNI is matched with so-called rotated input images (RII). The RIIs are generated from a 2.5-D scene model in the same manner as the SNIs. Following these descriptions the demand of an especially adapted matching strategy is obvious.

2 2.5-D model reconstruction

For generating the RIIs the first partial derivatives generated from PSM [5] have to be transformed to a height map or a geometrical 2.5-D model. This can be done for a large class of diffuse and hybrid reflecting surfaces [4]. We have developed a method, that combines

dense gradient information and a line drawing of the object. The method consists of two steps. The projected object is taken apart in planar and curved patches. Then these parts are fitted together to a 2.5-D surface. First the gradient images are segmented with region growing. Using this kind of segmentation technique planar as well as curved patches can be extracted without additional effort. Curved patches are approximated by planar patches. These planar patches are attributed as belonging to a curved region. This is necessary for the treatment of occlusions and the elimination of approximation edges in the recognition part. Since region boundaries in the interior of curved patches are determined by the growing process, these patches are post-processed with a balancing algorithm. Pixels on region boundaries are reclassified, if the average orientation in an adjacent region has a smaller angular deviation than the original region. The reclassification process is done iteratively until an equilibrium is reached. Subsequently to this process the boundaries are approximated polygonally. Consequently, for the next steps consistent region boundary information is available.

2.1 Part Assembler

Now, from the region and boundary data a winged edge model is generated. The model includes the vertices, edges, faces and the face orientations from the 2-D structure. This structure must be modified, if concave objects with partially occluded boundaries occur. In this case, we have to assign more than one depth value to some vertices. To prepare the depth calculation, such occluding edges have to be detected. Occluding boundaries are detected by using the face orientations. Since an orthogonal projection (denoted by op) is assumed, the expected edge orientation depends on the adjacent face orientations \bar{n}_1 and \bar{n}_2 as follows:

$$(x \ y)^T = (q_2 - q_1 \ p_1 - p_2)^T = op(\bar{n}_1 \times \bar{n}_2), \text{ with } \bar{n}_1 = (p_1 \ q_1 \ -1)^T, \bar{n}_2 = (p_2 \ q_2 \ -1)^T.$$

If this orientation is inconsistent with the line drawing, the edge becomes an occluding edge. Vertex splitting is done, if 1. both adjacent edges are occluding edges, or 2. one edge is occluding and the other is a 3-D boundary edge.

2.2 Determination of Depth Values

The depth is calculated locally for each vertex. Since a fixated depth of a vertex constrains the depth values of each adjacent face vertex, five cases have to be distinguished:

I. No adjacent vertex depth values are available: This vertex is neither constrained with respect to its 2-D coordinates nor with respect to its depth value. The depth value can be chosen arbitrarily.

II. For exactly one of the adjacent faces a fixation is made: The 2-D coordinates of the vertices can be substituted in the plane equation. This determines the depth values of these vertices.

III. For exactly two of the adjacent faces a fixation is made: 2-D inconsistencies of vertices can arise. Therefore an orthogonal projection onto the line of intersection of the two planes is determined. Thereafter the point of intersection is substituted into the plane equation.

IV. For exactly three of the adjacent faces a fixation is made: Consistency and the depth can be attained simultaneously by calculating the intersection of the planes.

V. More than three faces are fixed: If there is more than one point of intersection, than this inconsistency cannot be repaired. Therefore the order of depth calculation is made to be dependent on the number of adjacent faces. Such accidental events are less likely, especially for objects in general orientations and poses.

This procedure ensures that face orientations determined by the shape recovery method leave the assembling process unchanged. The edge structure in this process is variable. Surface

slopes on and nearby edges are less reliable than in the interior of regions due to physical, non-ideal object edges. Accordingly it is appropriate to let the edge structure alterable. Surface orientations are measured over the whole region for each face, as a result the surface orientation associated with a face is sufficiently stable.

The above scheme is applied to each vertex in the described order. When the process is finished, the 2-D winged edge model derived from the line drawing is transformed to a 2.5-D model.

3 Accumulator based adaptive matching

Park et al compare sequentially all RII sets with the model data base. The sole recognizable optimization of this trivial matching strategy is the ordering of the RIIs' by base face attributes as face size and face type. They classify RII faces in not occluded and occluded to manage inexact data. This data is then especially treated during the matching procedure.

We introduce an adaptive inexact matching strategy. It gains an optimal matching scheme, that is adapted to the actual object data base, the a priori knowledge about the current sensor configuration, see [1], and the measured sensor errors. This article regards the processing of unavoidable sensor tolerances. The CAD-data processing modeling component computes a set of SNIs for each object. Weak perspective projection or scaled orthographic projection is assumed. The SNI sets of all model objects build the object data bases. Feature extracting functions generate feature vectors that describe the particular SNIs. For each SNI one feature vector exists. During the matching procedure the RII set is processed by comparing RIIs' feature vectors one by another with the complete SNI set feature vectors. The matching algorithm consists of an adaptive hierarchical indexing scheme, that contains an inexact matching module and that is enlarged by a learning module.

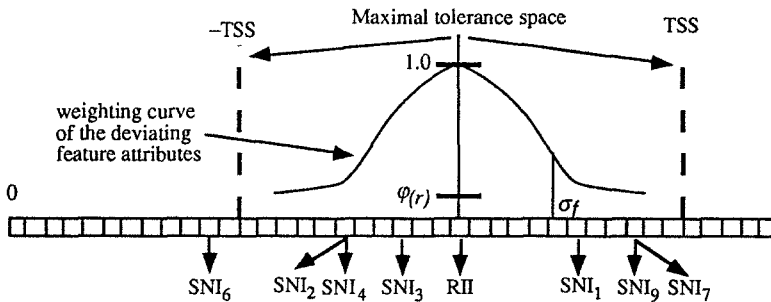


Fig. 2. Index layer showing the tolerance space dimension and a Gaussian distributed voting function.

The indexing process consists of a relation from the area of the scene features into the area of the index space. If a set of n scene features exists, then a vector of g ($g \leq n$) feature attributes determine an index I_g . The values of the index depend on the content of the feature attributes. An index vector specifies exactly a point in the index space. The hierarchical system organizes g features of interest in g index layers. Each feature is represented by a certain layer in the layer model. The layer model is adapted for the data base and the used sensors. Layers delineate the index space of a feature. The layer indexes point to accumulators of those SNIs, that contain similar feature attributes.

In the case of inexact data a method has to be found, that relates the data to comparative models. The proposed inexact matching strategy increases the accumulators of several SNIs by the value $e^{-k^2/2}$, with $k = 0, \dots, k_{\max}$ denoting the deviation of a model feature to the measured data. That means the accumulator of an object with a feature value close to the mea-

sured data is increased more than an accumulator of an object view with a less similar feature value. This is called Gaussian distribution method. The exact method is denoted as spike distribution method. The accumulator values determine the ranking of the SNI best list.

The following conditions constrain the indexing procedure, see Fig. 2:

- The width of the Gaussian curve used to calculate the vote value is adapted for the standard deviation of the regarded feature σ_f multiplied by a scale factor s . For $s = 2.0$ it is well known, that ca. 95% of the deviations caused by measuring errors are covered by the curve. Pursuing this, the absolute amount of deviations remains relatively small.
- A tolerance space, that is larger than $s \cdot \sigma_f$, causes an assignment of the smallest possible vote $v(k) = \phi(r)$ to the nodes outside the Gaussian curve. SNI candidates with larger errors in particular features are allowed to concur the matching race.
- A SNI is omitted from searching if a feature value drops out of the layer's tolerance space. Initially the tolerance space size (TSS) is 50%.

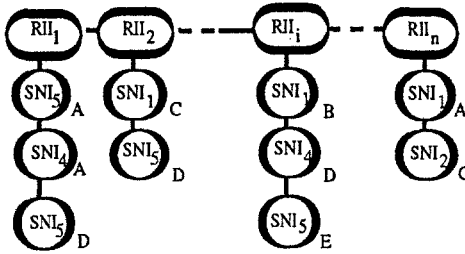


Fig. 3. The RII list matching algorithm concludes the resulting SNI lists: A SNI is marked by an object label (A, B, ...) and a number for the concerning SNI set. For each RII a list of SNI's is generated. This list is sorted with respect to the accumulated votes of the observed features.

In the proposed system a list of RII's is compared with the object data base. This process calculates a list of the most voted SNI for every RII. Some regulation has to be done to conclude the particular lists into a single object ranking list. The following constraints constrict SNI lists as shown in Fig. 3:

- The objects best list integrates only the best matching SNI of a particular object for each RII: This constraint prevents, that objects with many faces are privileged for objects with fewer faces.
- A SNI, that votes for a RII is removed from the remaining best list. This constraint dismisses the possibility, that a particular object's SNI votes for several RIIs.
- If no object SNI matches an especial RII, then this object is excluded from the matching process. This constraint solely controls the elimination of objects from the matching process.

3.1 Learning by integration of measured sensor accuracies

A technique to improve the performance of the matching is to adapt the tolerance space size (TSS) on the feature attribute distribution of the measured data. The system performs an On-line adaptation by integrating the sensor accuracy: The matching strategy generates a list of the best matching SNI sets. A localization function verifies or discards the hypothesis of the matching procedure. Correctly selected view sets are compared with the RII sets.

The accuracy is measured by comparing of the correct result feature vector x_{SNI} and the vector x_{RII} of the corresponding RII. This process repeats for all feature vectors of the two view sets. For each feature attribute (element of the feature vectors) the maximal difference between the real and the measured feature value is stored in a learning map. This learning map supervises the mean standard deviation and maximal deviation value max_{dev} of the particular features. After t successfully processed recognition tasks the TSS and the σ_f of each index layers adapt. The learning map calculates the standard deviation for a feature g after the $n+1$

recognition task. The TSS is calculated by the max_{dev} of the features. Since the TSS excludes object views from the matching procedure, the learning algorithm allows the expansion of the TSS by incrementing max_{dev} by $max_{dev}/3$. The algorithm decreases the TSS by deleting the current max_{dev} after t recognition tasks and choosing the next smaller max_{dev} as new max_{dev} .



Fig. 4. Two data base objects and their reconstructed 2.5-D models.

3.2 The accuracy probability of results

In the following we estimate the accuracy of object propositions depending on the calculated votes of the SNI features. The sizes of the Gaussian curves of the several features rely on the standard deviations of the features' errors. Pursuing this, it is possible to regard votes as function values of the proportion z . This z represents the proportion between the actual feature difference of the RII and SNI and the standard deviation of the regarded feature. To get an estimation of the matching probability of a proposed model to an object, we calculate the inverse function $z = v^{-1}$ of the used Gaussian weight voting function: $z(vote) = \sqrt{-2 \ln(vote)}$. We use the results of z to calculate the probability of the accuracy of a model proposition, on the assumption of normal distributed feature errors. The probability, that a measured feature deviates at Δx from the correct feature value follows the Gaussian error integral:

$$p(z(vote)) = \sqrt{2/\pi} \cdot \int_0^z e^{-t^2/2} dt; \quad t = \Delta x / \sigma_n \text{ and } z = a / \sigma_n$$

For a deviation Δx with $\Delta x / \sigma_n \geq z(vote)$ ($a \leq \Delta x$) the value $pi = 1 - p(z(vote))$ delineates the probability for this Δx . For $z(vote) = 0$ it follows $pi = 1.0$.

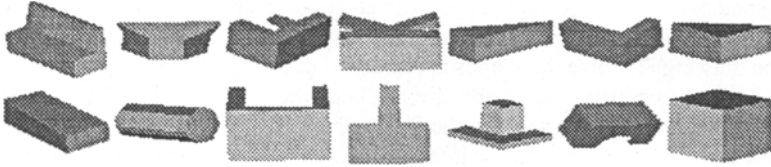


Fig. 5. Shaded image of reconstructed real objects.

4 Results

We have examined the shape recovery scheme (PSM) and the proposed face assembling method with several synthetic and real objects. We have analyzed several PSM realizations, including analytic, LUT based and discrete solution finding approaches. Our 2-D-LUT technique has proved to give the best results with respect to calibration expenditure and accuracy. Fig. 4 shows single PSM input images of two objects in the data base. For each object two aspects are shown. Since only a 2.5-D model can be derived from the input images two faces are missing in the second aspect of the L-shaped object. Also, the F-shaped object has been reconstructed in a reliable manner, although there arise shadows and mutual illuminations. Fig. 5 shows the Lambertian shaded versions of the derived 2.5-D models of the real objects in the data base. It can be seen, that the 2.5-D models of some objects are not complete due to cast and self shadowing effects. To verify the recovered surface orientations

the angle between right-angled faces has been measured. For most face pairs the angular difference was not higher than three degrees.

To test the recognition system we used a model data base with 14 objects. The objects are categorized in the classes convex, concave and smooth. We demonstrate six test series with varying parameters. The following features are calculated for the experiments:

- junction types, edge types, relations of the length of edge pairs, classes of edges with the same length, face number, face relations, angle types.

experiment #	adaptation steps	convex	curved	concave	sum
1	---	0 % ---	0 % ---	0 % ---	0 % ---
2	4	100 % 0.92	50 % 0.82	57 % 0.80	66 % 0.85
3	21	100 % 0.92	75 % 0.83	71 % 0.77	80 % 0.83
4	39	100 % 0.92	75 % 0.82	86 % 0.74	87 % 0.81
5	56	100 % 0.91	75 % 0.80	86 % 0.72	87 % 0.80
6	---	100 % 0.93	50 % 0.87	86 % 0.82	80 % 0.86

Tab. 1. Recognized real objects in percent and their recognition probability.

The experiment results shown in Tab. 1 depend on the variation of the tolerance parameters:

- Experiment 1: The TSS and the standard deviation for all features are zero. These settings correspond to the exact matching method.
- Experiment 2-5: The parameters are adapted stepwise. Distinct objects cause the generation of a distinct number of RIIs. This explains the non continuous growing of the adaptation steps. The recognition rate of the system grows continuously.
- Experiment 6: The system runs with extremely large TSS and standard deviation.

5 Conclusion

It has been shown, that a shading based shape recovery method (Photometric Stereo) leads to sufficient 2.5-D models for application in an elaborate object recognition system. Particularly this is appropriate, if the derived 3-D shape information is combined with an approximated line drawing. The line drawing is necessary, if occlusions have to take into consideration. We have shown, that good results are reached when combining PSM, the SNI model and the adaptive matching strategy. In the future, the shape recovery part of the project will be focused on making use of shadow information and mutual illuminations. The matching will be extended by an automatic verification module.

6 References

1. G. Bellaire, "Feature-Based Computation of Hierarchical Aspect-Graphs", Machine GRAPHICS & VISION, Vol. 2, No. 2, pp. 105-122, 1993
2. K. Ikeuchi, "Determining Linear Shape Change: Toward Automatic Generation of Object Recognition Programs", CVGIP, Vol. 53, pp. 154-170, 1991
3. J. H. Park, T. G. Chang, J. S. Choi, "Three-Dimensional Object Representation and Recognition Based on Surface Normal Images", PR, 26, No. 6, pp. 913-921, 1993
4. K. Schlüns: "Photometric Stereo for Non-Lambertian Surfaces Using Color Information", Proc. 5th Int. Conf. on Computer Analysis of Images and Patterns, Budapest, Hungary, Sept. 13-15, pp. 444-451, 1993
5. R.J. Woodham R.J.: "Photometric Method for Determining Surface Orientations from Multiple Images", Optical Engineering, Vol. 19, No. 1, pp. 139-144, 1980