Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 387-392, 1993.

Fast Alignment Using Probabilistic Indexing

Clark F. Olson
Computer Science Division
University of California at Berkeley
Berkeley, CA, 94720
clarko@robotics.eecs.berkeley.edu

Abstract

The alignment method [4] is a model-based object recognition technique that determines possible object transformations from three hypothesized matches of model and image points. For images and/or models with many features, the running time of the alignment method can be large. This paper presents methods of reducing the number of matches that must be examined. The techniques we describe are: Using the probabilistic peaking effect [1] to eliminate unlikely matches (implemented in a probabilistic indexing system [6]) and eliminating groups of model points that produce large errors in the transformation determined by the alignment method. Results are presented that show we can achieve a speedup of over two orders of magnitude while still finding a correct alignment.

1 Introduction

The alignment method [4] is a model-based object recognition technique for recognizing three-dimensional objects from a single view in two-dimensional images. For each model in the database, triples of image points are matched with triples of model points. For each possible match, the transformation that brings them into alignment is determined, and each of these transformations must be tested to determine if it is correct. For complex object models or images the running time of the alignment method can be large.

This paper addresses techniques for reducing the number of matches between triples of model and image features that must be examined. This is accomplished by eliminating model triples and matches between image and model triples that are unlikely to produce correct transformations. The techniques that we present are based on the following two principles:

1. The probability density functions of angles and distance ratios in images peak strongly at the pre-projection (model) value [1, 2, 3]. This principle has been used to build an indexing system that determines groups of model points likely to match groups of image points of size three [6].

2. Matches that produce a transformation with a large uncertainty are unlikely to result in a good alignment of the model.

These techniques have been implemented for the alignment method and a speedup of over two orders of magnitude has been achieved while still finding a correct alignment. Additional details of the techniques presented in this paper can be found in [5] and [6].

2 The Alignment Method

The premise of the alignment method is that a unique (up to a reflection) affine transformation between the model and image of the model can be found by matching three model points with three image points. Huttenlocher and Ullman describe how to determine this transformation [4].

Let us call the triple of model points being matched the *model group* and the image points hypothetically matched to them the *image group.* If each of the points in the image group is the result of the projection of its corresponding model group point then we will say the two groups are in *actual correspondence.* For the rest of this paper we will consider a single object model. In practice, each model must be examined separately in the alignment method.

It is not desirable to examine each combination of three image points and three model points, since, if there are n image points and m model points, the entire algorithm requires $O(m^4 n^3 \log n)$ operations, due to a $O(m \log n)$ verification step. If a model object is present in the image, it is likely that a substantial number of triples of model points can be detected. In the best case, only one of these triples needs to be found and matched to recognize the object. If all combinations are examined to find the best scoring match, then much is being done that is not necessary. Even if we stop once a adequate match has been found, we can use information about the likelihood of each match being correct to determine which matches to examine. This can reduce the running time considerably.

It is important to note that the affine transformation is used in this algorithm as an approximation to the full perspective projection and is valid only when the distance to the object is large compared to the size of the object in the z-direction in camera-centered coordinates. Experiments determining when this approximation is valid are described in Section 5.

Huttenlocher and Ullman have proposed techniques to lower the complexity of the algorithm. For example, virtual points found by using the orientations at two model and image points can be used in some situations to reduce the complexity to $O(m^3 n^2 \log n)$ operations. The methods presented here can be easily modified to accommodate such techniques.

3 Probabilistic Indexing

It has been observed that there is a large peak in the probability density of image angles and ratios of lengths at the values taken by the features in the model [1, 2, 3]. This information can be used to discard matches between image points and model points that have a small likelihood of being in actual correspondence. We have used this effect to create a probabilistic indexing system [6].

We use the same features as Ben-Arie [1] to determine which groups are unlikely to match. These features are easy to determine for sets of three ordered points. For model group (p_1, p_2, p_3) and corresponding image group (q_1, q_2, q_3), let α be the angle $\angle p_1 p_2 p_3$ and β be the angle $\angle q_1 q_2 q_3$. Define the segment lengths as follows: $a_1 = |\overline{p_1 p_2}|, a_2 = |\overline{p_2 p_3}|, b_1 = |\overline{q_1 q_2}|, b_2 = |\overline{q_2 q_3}|$. Figure 1 gives an illustration. The features used are:

1. The angles formed by the points in the model (α) and in the image (β).

2. The ratios of the lengths of the segments ($\frac{a_1}{a_2}$ and $\frac{b_1}{b_2}$).

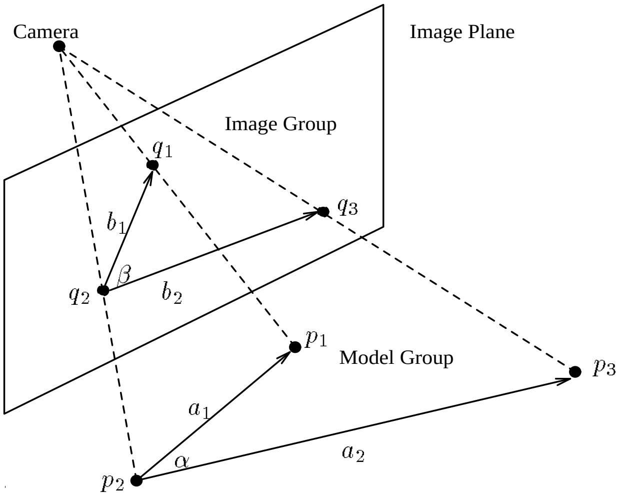

Figure 1: Projected Model Group

When integrated over the viewing sphere, the joint probability density of β and $\frac{b_1}{b_2}$ peaks strongly at α and $\frac{a_1}{a_2}$. We have generated probability histograms describing this effect by sampling the viewing sphere and numerically integrating in a manner similar to Ben-Arie. The joint prior probability histogram of β and $\frac{b_1}{b_2}$ has also been generated using similar techniques.

The probabilistic indexing system creates an index table by discretizing the α-$\frac{a_1}{a_2}$ space and placing each model group into the cell of the table corresponding to its parameters. Let b_i denote the bin that corresponds to the image group features and let h be the hypothesis that the model group and the image group are in actual correspondence. We can determine the probability of h being correct by applying Bayes' rule:

$$P(h \mid b = b_i) = \frac{P(h)P(b = b_i \mid h)}{P(b = b_i)}$$

$P(b = b_i \mid h)$ is given by the peaking effect joint probability histograms and $P(b = b_i)$ is given by the prior probability histogram. We assume that the prior probability of each possible match (and thus each possible hypothesis h) is the same, so we can drop the $P(h)$ term without changing the ranking of the hypotheses.

When presented with an image group, the system determines β and $\frac{b_1}{b_2}$ and examines the probability histograms to determine the index table cells that contain model groups that have a large enough probability of generating the image features. The model groups at those locations are then considered as possible matches for the image group. So, for a specific image group k, a model group is eliminated if:

$$\frac{P(b = b_i^{(k)} \mid h)}{P(b = b_i^{(k)})} < \gamma_p$$

where $b_i^{(k)}$ is the bin for the kth image group and γ_p is an empirically determined constant.

4 Eliminating Groups Using Error Statistics

Here we will show how model groups that produce large uncertainty in the computed transformation can be determined and thus eliminated from consideration. Let the model group be (p_1, p_2, p_3) and the image group be (q_1, q_2, q_3).

4.1 Condition Number

Huttenlocher and Ullman transform the model group such that it lies in the x-y plane. Let p_i^* be the coordinates of the ith point of the model group after this transformation and:

$$p'_2 = p_2^* - p_1^* \qquad q'_2 = q_2 - q_1$$

$$p'_3 = p_3^* - p_1^* \qquad q'_3 = q_3 - q_1$$

They then solve for the affine transformation L that aligns the points in two-dimensions. L is determined by the following equations:

$$Lp'_2 = q'_2 \qquad Lp'_3 = q'_3$$

These equations can be transformed into:

$$ML_1 = I_1 \qquad ML_2 = I_2$$

where

$$M = \begin{bmatrix} p'_{2_x} & p'_{2_y} \\ p'_{3_x} & p'_{3_y} \end{bmatrix}$$

$$L_1 = \begin{bmatrix} l_{11} \\ l_{12} \end{bmatrix} \qquad L_2 = \begin{bmatrix} l_{21} \\ l_{22} \end{bmatrix}$$

$$I_1 = \begin{bmatrix} q'_{2_x} \\ q'_{3_x} \end{bmatrix} \qquad I_2 = \begin{bmatrix} q'_{2_y} \\ q'_{3_y} \end{bmatrix}$$

Our localization of the image points will have some error, of course. Therefore, our solutions for L_1 and L_2 will also have some error. Let $\mathcal{I}_1$, $\mathcal{I}_2$, $\mathcal{L}_1$, and $\mathcal{L}_2$ denote the true values of I_1, I_2, L_1, and L_2, and let δI_1, δI_2, δL_1, and δL_2 denote their errors such that:

$$\mathcal{I}_1 = I_1 + \delta I_1 \qquad \mathcal{I}_2 = I_2 + \delta I_2$$

$$\mathcal{L}_1 = L_1 + \delta L_1 \qquad \mathcal{L}_2 = L_2 + \delta L_2$$

From matrix computations we can bound the error on δL_1 and δL_2 as follows [7]:

$$\tfrac{||\delta L_1||}{||L_1||} \leq \kappa(M)\tfrac{||\delta I_1||}{||I_1||} \qquad \tfrac{||\delta L_2||}{||L_2||} \leq \kappa(M)\tfrac{||\delta I_2||}{||I_2||}$$

where $||\cdot||$ is any vector norm (and its induced matrix norm) and $\kappa(M) = ||M|| \cdot ||M^{-1}||$ is the condition number of M. So, if M has a large condition number we may have large errors δL_1 and δL_2. We eliminate a model group k if the inverse of its' condition number is less than some percentage of the average:

$$\frac{1}{\kappa(M^{(k)})} < \frac{\gamma_k}{N} \sum_i \frac{1}{\kappa(M^{(i)})}$$

where $M^{(i)}$ refers to the model matrix for the ith model group and γ_k is an empirically determined constant.

4.2 Model Group Area

Even when the model group produces a condition number of one (the best case), it is possible that the group leads to a large error in the computation of the transformation matrix, as can be seen from the following analysis. Assuming non-singularity we have:

$$\begin{bmatrix} l_{11} \\ l_{12} \end{bmatrix} = \begin{bmatrix} p'_{2_x} & p'_{2_y} \\ p'_{3_x} & p'_{3_y} \end{bmatrix}^{-1} \begin{bmatrix} q'_{2_x} \\ q'_{3_x} \end{bmatrix}$$

Computing this inverse we get:

$$l_{11} = \frac{p'_{3_y} q'_{2_x} - p'_{2_y} q'_{3_x}}{p'_{3_y} p'_{2_x} - p'_{2_y} p'_{3_x}}$$

Substituting the original model and image values, this becomes:

$$l_{11} = \frac{(p^*_{3_y} - p^*_{1_y})(q_{2_x} - q_{1_x}) - (p^*_{2_y} - p^*_{1_y})(q_{3_x} - q_{1_x})}{(p^*_{3_y} - p^*_{1_y})(p^*_{2_x} - p^*_{1_x}) - (p^*_{2_y} - p^*_{1_y})(p^*_{3_x} - p^*_{1_x})}$$

Similar equations (with the same denominator) can be found for l_{12}, l_{21}, and l_{22}. The denominator is twice the area of the triangle formed by p_1, p_2, and p_3. If the denominator is small, this will have the effect of magnifying the errors present in the numerator and the denominator. So, we eliminate a model group k if it satisfies:

$$\text{area}(\triangle p_1^{(k)} p_2^{(k)} p_3^{(k)}) < \gamma_a \max_i \left\{ \text{area}(\triangle p_1^{(i)} p_2^{(i)} p_3^{(i)}) \right\}$$

where $p_1^{(i)}$ refers to the first point of the ith model group and γ_a is an empirically determined constant.

5 Experimental Results

The techniques presented above have been tested on both randomly generated and real data. This section presents those experiments and results.

5.1 Complete System

We have implemented the alignment method and used these techniques to eliminate groups and matches unlikely to produce correct transformations. We used the following procedure in our experiments: The condition number and area for each model group was determined. If the group was not eliminated then it was placed in the index table. Each image group was used to index the table to find possibly matching model groups. For each indexed match, the transformation aligning the points was found and it's correctness was determined.

A transformation was considered correct if the closest transformed model point to each correct image point was the correct match. Thus, a transformation was considered successful only if each corresponding model and image point were brought close together.

5.2 Random Data

To determine the efficacy of the affine transformation as an approximation to the perspective transformation and to determine the best parameters for eliminating unlikely matches, tests were carried out on random point sets. Each experiment consisted of tests on 100 random objects with 10 randomly generated points each. The models underwent random transformations and were projected onto the image using the full perspective transformation. Gaussian noise was added to each of the feature coordinates.

Figure 2 shows the percentage of actual correspondences which produced successful transformations using objects at varying ratios of object distance to object depth. The three plots are for images with gaussian noise of standard deviation 0.0, 2.0, and 5.0 added to the image feature coordinates. The x-axis is the ratio of object distance to approximate depth of the object. We see that for each of the three plots serious degradation of the percentage successful begins at approximately a ratio of 8, which we conclude is the minimum ratio for which the affine approximation to the perspective projection is accurate for use with the alignment method.

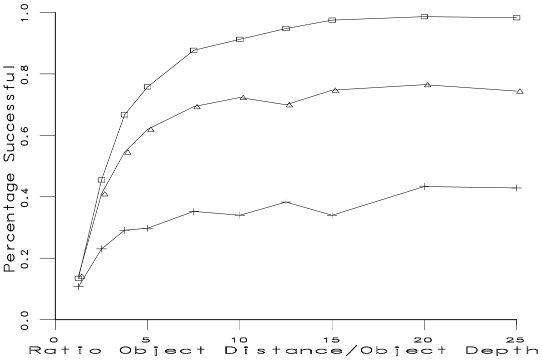

Figure 2: Percentage of successful transformation with various object distances. □: no noise, △: noise ($\sigma = 2.0$), +: noise ($\sigma = 5.0$)

For the remainder of the experiments, we uses gaussian noise with standard deviation 2.0 and objects at a distance such that the ratio of object distance to object depth is 10. Figure 3 shows the percentage of total matches examined (not eliminated) and the percentage of correct matches examined for various values of each of the elimination parameters when used alone. The most powerful parameter is the peaking probability, as seen by the large difference between the percentage of correct and total matches examined. The condition number and model group area parameters also appear to be useful for elimination.

5.3 Real Images

The techniques were also tested on real images. In these experiments, model features were selected and measured by hand. Image features were determined by hand with the help of an edge detector. Random extraneous points were added to these images.

Figure 4 shows one of the images that was used to test the elimination techniques. Using each of the parameters at reasonable values, we examined 4.2% of the total matches and 24.9% of the correct matches in seven real images (compared with 2.5% and 18.2% for the random points.)

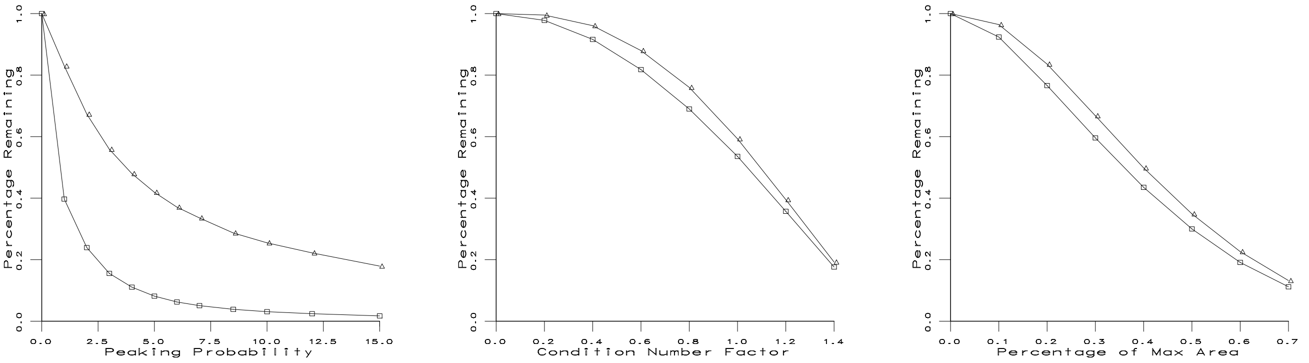

Figure 3: Percentage of matches examined for each elimination parameter. □: total, △: correct.

Figure 4: An image used for testing.

6 Discussion

We will examine the speedup and probability of a false negative produced by these techniques under two recognition models:

1. Each possible match between an image group and a model group that is not eliminated receives a score from some verification process. The best scoring match is accepted as correct if the score meets some criteria.

2. Matches are examined in some order. Matches that are not eliminated receive a score from the verification process. As soon as the score for one of the matches meets some criteria, it is accepted as correct and the remainder of the matches are not examined.

The speedup will be defined as the expected number of matches that must be verified by the algorithm without using the elimination techniques, divided by the expected number when using the techniques. Let n be the total number of matches examined, p be the percentage of total matches not eliminated, ν be the total number of matches examined that produce a correct transformation, and ρ be the percentage of these matches not eliminated.

In the first model, we examine np matches when using these techniques and n matches when not using these techniques, so the speedup is simply $\frac{1}{p}$.

In the second model, if we assume that matches are examined in random order, we have a hypergeometric distribution. For large values of n, this can be approximated by the binomial distribution. The expected number of matches that must be verified when not using the elimination techniques is then approximately $\frac{n}{\nu}$. When using the elimination techniques the expected number of matches that must be verified is approximately $\frac{pn}{\rho\nu}$. So, we see that the speedup is approximately $\frac{\rho}{p}$. This analysis assumes that $\nu > 0$, that is, a correct match exists. If $\nu = 0$ either because the model is not present in the image or because none of the matches in actual correspondence produce a good enough transformation, then the speedup is the same as for model 1 ($\frac{1}{p}$).

Table 1 shows ρ, p and the expected speedups for some values of the elimination parameters. Impressive speedups are attained for recognition model 1 and for model 2 when $\nu = 0$. The speedups for recognition model 2 when $\nu > 0$ are more modest. We note that the case $\nu = 0$ is the common case, since each model must be examined and there are usually few of the models present in the image.

If we assume that the probability of a correct match being eliminated is independent of whether other correct matches have been eliminated (this assumption will be discussed below), the probability of a false negative as a result of eliminating correct matches is

γ_p	γ_k	γ_a	ρ	p	$\frac{1}{p}$	$\frac{\rho}{p}$
11.0	1.0	.37	.062	.0069	144.9	8.9
10.0	0.9	.34	.085	.0104	96.2	8.2
8.5	0.8	.30	.124	.0174	57.5	7.1
6.0	0.7	.20	.229	.0418	23.9	5.5

Table 1: Speedups for various elimination parameters.

ρ	ν =50	ν =100	ν =200
.062	$4.08\text{x}10^{-2}$	$1.66\text{x}10^{-3}$	$2.76\text{x}10^{-6}$
.085	$1.18\text{x}10^{-2}$	$1.39\text{x}10^{-4}$	$1.92\text{x}10^{-8}$
.124	$1.33\text{x}10^{-3}$	$1.78\text{x}10^{-6}$	$3.17\text{x}10^{-12}$
.229	$2.25\text{x}10^{-6}$	$5.08\text{x}10^{-12}$	$2.58\text{x}10^{-23}$

Table 2: Probability of a false negative for various values of ρ and ν.

$(1-\rho)^\nu$ for both models.

If even 10 points from a model are present in an image, there are 120 correct model groups (for 20 model points there are 540 model groups.) For moderate values of noise ($\sigma \approx 2.0$) the majority of these will result in correct transformations, so we expect ν to be at least 50. Table 2 shows the probability of a false negative resulting from eliminated matches for the values of ρ from Table 1 and ν =50, 100, and 200.

Even for relatively small values of ν and a high percentage of elimination of matches, there is a small chance of a false negative. As ν increases or the percentage of elimination decreases the likelihood of a false negative becomes negligible.

We now address the question of the independence of the probability of correct matches being eliminated. More specifically, we want to know if it is possible for some object to be in an orientation for which all model groups appear in unlikely configurations in an image. Unlikely configurations occur when the viewing directions is not within some proximity to perpendicular to the plane of the points in the group. This means that groups of coplanar points will have occur in unlikely configurations from the same viewing directions. Objects that are not nearly flat should not be a problem since they have groups in all orientations. A flat or nearly flat object rotated such that it is foreshortened in the image may produce angles and/or distance ratios far from the probability peaks. Such images would not benefit much from the elimination of matches based on these techniques, since too many correct matches would be eliminated. Problems with such images are common to most object recognition systems including the human visual system. It is possible that special-purpose techniques for recognizing flat or nearly-flat objects may be of use in these cases.

7 Conclusions

We have presented techniques that greatly reduce the number of matches that must be examined in the alignment method through use of the probabilistic peaking effect and error criteria, thus greatly increasing the speed at which objects can be recognized. Experimental results were given that showed that these techniques work in practice and still result in a correct transformation being found.

Acknowledgements

The author thanks Jitendra Malik for his advice on aspects of this research. This research has been supported by a National Science Foundation Graduate Fellowship to the author, NSF Presidential Young Investigator Grant IRI-8957274 to Jitendra Malik, and NSF Grant IRI-9114446.

References

[1] J. Ben-Arie. The probabilistic peaking effect of viewed angles and distances with application to 3-d object recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(8):760–774, August 1990.

[2] T. O. Binford, T. S. Levitt, and W. B. Mann. Bayesian inference in model-based machine vision. In L. N. Kanal, T. S. Levitt, and J. F. Lemmer, editors, *Uncertainty in Artificial Intelligence 3*, pages 73–95. Elsevier Science Publishers B.V. (North-Holland), 1989.

[3] J. B. Burns, R. Weiss, and E. M. Riseman. View variation of point set and line segment features. In *Proceedings of the DARPA Image Understanding Workshop*, pages 650–659, 1990.

[4] D. P. Huttenlocher and S. Ullman. Recognizing solid objects by alignment with an image. *International Journal of Computer Vision*, 5(2):195–212, 1990.

[5] C. F. Olson. Fast alignment by eliminating unlikely matches. Technical Report UCB//CSD-92-704, University of California at Berkeley, November 1992.

[6] C. F. Olson. Probabilistic indexing: Recognizing 3d objects from 2d images using the probabilistic peaking effect. Technical Report UCB//CSD-93-733, University of California at Berkeley, May 1993.

[7] D. S. Watkins. *Fundamentals of Matrix Computations*. John Wiley and Sons, 1991.