Matching Perspective Views of Coplanar Structures using Projective Unwarping and Similarity Matching

Robert T. Collins
J. Ross Beveridge

CMPSCI TR94-06

February 1994

This paper was presented at the 1993 IEEE Conference on Computer Vision and Pattern Recognition, New York City, June 1993. This work was funded in part by DARPA/TACOM contract DAAE07-91-C-R035 and by the RADIUS project under DARPA/Army contract TEC DACA76-92-R-0028.

Matching Perspective Views of Coplanar Structures using Projective Unwarping and Similarity Matching *

Robert T. Collins
Department of Computer Science
University of Massachusetts
Amherst, MA. 01003
rcollins@cs.umass.edu

J. Ross Beveridge
Department of Computer Science
Colorado State University
Fort Collins, CO. 80523
ross@cs.colostate.edu

Abstract

We consider the problem of matching perspective views of coplanar structures composed of line segments. Both model-to-image and image-to-image correspondence matching are given a consistent treatment. These matching scenarios generally require discovery of an eight parameter projective mapping. However, when the horizon line of the object plane can be found in the image, done here using vanishing point analysis, these problems reduce to a simpler six parameter affine matching problem. When the intrinsic lens parameters of the camera are known, the problem further reduces to four parameter affine similarity matching.

1 Introduction

Matching is a ubiquitous problem in computer vision. Correspondence matching can be broken into two general areas: *model-to-image* matching where correspondences are determined between known 3D model features and their 2D counterparts in an image, and *image-to-image* matching where corresponding features in two images of the same scene must be identified. Fast and reliable matching techniques exist when good initial guesses of pose or camera motion are available [6, 7] or when the distance between views is small [1]. What is lacking are good methods for finding matches in monocular images, formed by perspective projection, and taken from arbitrary viewpoints.

This paper examines the problem of matching coplanar structures composed of line segments. A simple method is presented that, when applicable, allows fast and accurate matching of coplanar structures across multiple images, and of locating structures that correspond to a model consisting of significant planar patches. The main point to this paper is that the full perspective matching problem for coplanar structures can often be reduced to a simpler four parameter affine matching problem when the horizon line of the planar structure can be determined in the image. Given this horizon line, the image can be transformed to show how the structure would appear if the camera's line of sight was perpendicular to the object plane. This process is called *rectification* in aerial photogrammetry.

2 Planar Transformations

Essentially all matching problems involve solving for both a discrete correspondence between two sets of features (model-image or image-image) and an associated transformation that maps one set of features into registration with the other. These two solutions together constitute matching: a match being a correspondence plus a transformation. For planar structures under a perspective camera model, the relevant set of transformations is the eight parameter projective transformation group [12].

More restrictive transformations are worth special attention. Often these transformations are more easily computed, thus making matching easier. One such special case occurs for *frontal planes,* planar structures viewed "head-on" with the viewing direction of the camera held perpendicular to the object plane. When the intrinsic camera parameters are known, perspective mapping of a frontal plane to its appearance in the image can be described with just four affine pa-

*This paper was presented at the 1993 IEEE Conference on Computer Vision and Pattern Recognition, New York City, June 1993. This work was funded in part by DARPA/TACOM contract DAAE07-91-C-R035 and by the RADIUS project under DARPA/Army contract TEC DACA76-92-R-0028.

rameters: an image rotation angle, a 2D translation vector, and an image scale [22].

2.1 Frontal Planes

Under the standard pinhole camera model, the image projection of world point (X, Y, Z) is the image point $(X/Z, Y/Z)$. In this case, the appearance of *any* 3D object is governed only by the relative position and orientation of the camera with respect to the object, i.e. the camera *pose.* There are 6 degrees of freedom for camera pose: three for rotation and three for translation. Constraining the camera to point directly perpendicular to an object plane, to yield a *frontal view* of the plane, fixes two degrees of its rotational freedom. The four remaining degrees of freedom, one free camera rotation about the normal of the object plane and three translation parameters, can be characterized by how they affect the appearance of the object plane in the image. For example, translation directly towards or away from the object plane manifests itself as a uniform change of scale in the projected image. Translation parallel to the planar surface shows up as a proportional 2D translation in the image. Finally, a rotation of the camera about its principle axis (which is normal to the object plane) causes the projected image to rotate by the same angle about a point in the image plane. The pinhole camera projection of a frontal plane is therefore described by four affine parameters that are directly related to the physical pose of the camera with respect to the plane. Said in another way, the function that maps object coordinates to image coordinates for a planar structure viewed frontally by a pinhole camera is a four parameter affine mapping.

A more realistic camera model must take into account the camera lens parameters. To a first approximation, lens effects can be modeled by a set of *linear* parameters that include focal length, lens aspect ratio, optical center, and optical axis skew. The combined effects of these parameters can be described by a general six parameter affine mapping of the ideal pinhole image onto the observed raster image [16]. A more realistic model of the projection of a frontal plane is thus a four parameter affine mapping of object features onto an idealized pinhole image, followed by a six parameter affine mapping onto the observed raster image.

In summary, the perspective projection of a frontal plane is described in general by a six parameter affine transformation. When a calibrated camera is used its intrinsic lens effects are known, and can be inverted to recover the ideal pinhole projection image. After correction for intrinsic lens effects, the frontal view of an object plane can be described by a four parameter affine similarity mapping.

2.2 Arbitrary Orientations

For planes viewed at an angle, the function mapping object coordinates to image coordinates is no longer affine, but is instead a more general projective transformation [12]. Lines that are parallel on a tilted object plane appear to converge in the image plane, intersecting at a *vanishing point.* Two or more vanishing points from different sets of coplanar parallel lines form a line in the image called the *vanishing line* or *horizon line* of the plane.

For frontal planes, all parallel lines on the object remain parallel in the image. This is because the image projection of a frontal plane is described by an affine transformation, which preserves parallelism. To avoid continually treating frontal views as a special case, by convention a set of parallel lines in the image is said to intersect in a point "at infinity." For frontal planes, all vanishing points of parallel lines appear at infinity, and the vanishing line passing through them is also said to be at infinity.

This convention allows the relation between views of frontal object planes and tilted object planes to be precisely stated. The mapping of object coordinates to image coordinates for a plane viewed at *any* orientation, either frontal or tilted, is described by a projective transformation. Parallel lines on the object map to lines in the image that converge to a vanishing point, all of which lie on the vanishing line associated with the object plane. Frontal views are distinguished from more general views in that the vanishing line is located at infinity. In this case the projective transformation mapping object coordinates to image coordinates also happens to be an affine transformation.

These considerations lead to a simple yet powerful observation. By applying a projective mapping to the image that takes the vanishing line of a coplanar structure to the line at infinity, the vanishing points of all lines in the object plane will also appear at infinity. Once this is done, all parallel lines in the planar structure will appear parallel in the image. This implies that the new image is a frontal view of the object plane, and thus the mapping from object to image can be represented as an affine transformation.

2.3 Rectification

We have seen that the vanishing line of a frontal plane appears at infinity in the image plane, and furthermore, that it is possible to recover a frontal view from the image of a tilted object plane by applying a projective transformation that maps the object's vanishing line to infinity. There is, however, a six-dimensonal space of projective transformations that all map a given line in the image off to infinity. How to choose a "best" one is described in this section.

For a pinhole camera image, the location and orientation of the vanishing line of an object plane determines the true 3D orientation of the plane with respect to the camera's line of sight. When the equation of the vanishing line is $ax + by + c = 0$, the normal to the object plane, in camera coordinates, is

$$n = (a, b, c)/||(a, b, c)||. \tag{1}$$

For a frontal plane, the normal of the plane must be parallel to the Z-axis of the camera. If the camera could move, the image of a frontal plane could be recovered from the image of a tilted plane by merely rotating the camera to point directly towards the plane. The camera can no longer be moved physically, of course, but the image *can* be transformed artifically to achieve the desired 3D rotation.

Assume the unit orientation of the object plane has been determined to be n, as in equation 1, oriented into the image ($c \geq 0$). To bring this vector into coincidence with the positive Z axis requires a rotation of angle $\mathrm{Cos}^{-1}(n \cdot (0,0,1))$ about the axis $n \times (0,0,1)$. The effects of this camera rotation on the image can be simulated by an invertible projective transformation in the image plane [19]. In homogeneous coordinates,

$$k_i \begin{bmatrix} x'_i \\ y'_i \\ 1 \end{bmatrix} = \begin{bmatrix} E & F & a \\ F & G & b \\ -a & -b & c \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix}$$

where

$$E = \frac{a^2c + b^2}{a^2 + b^2}, F = \frac{ab(c-1)}{a^2 + b^2}, G = \frac{a^2 + b^2c}{a^2 + b^2}.$$

The image is transformed to appear as if the camera had been pointing directly towards the object plane. The result therefore is a frontal view of the object plane, as seen by a pinhole camera, i.e. a rectified four parameter affine view.

This transformation can be used to map a vanishing line to infinity even when the camera lens parameters are not known. In this case the pure pinhole image can not be recovered and the position of the vanishing line in the image can no longer be interpreted geometrically in terms of 3D plane orientation. Nevertheless, the image will be rectified to present some six parameter affine mapping of the frontal object plane.

3 Correspondence Matching

A two step approach is used to match coplanar line segments seen from two arbitrary 3D viewpoints. The first step is to rectify both sets of line segments using the techniques described above. This reduces the perspective matching problem to a simpler affine matching problem. The second step is to use a local search matching algorithm [6, 8] to find the optimal affine map and correspondence between the two sets of line segments. If both sets of line segments are extracted from images, then an image-to-image matching problem results. If one set of segments is derived from a geometric object model, then a model-to-image matching problem results.

The local search matching algorithm is used in place of more commonly known types of affine matching for the following two reasons. First, the goal of local search matching is to find optimal, rather than acceptable, matches. When started from random positions in the search space, the local search algorithm finds the optimal match with high probability. This means that the best possible match will be returned even when the image data is of poor quality, and that the algorithm will continue to perform well even when repetitive image structure generates a profusion of partial matches. Second, the local search matching algorithm we use operates over a space of *many-to-many* line segment correspondence mappings. This is important, particularly for image-to-image matching, because bottom-up line extraction processes frequently fragment line segments. Finding the true match may require piecing together fragments from both images.

Most previous approaches to affine matching do not seek optimal matches, and do not allow many-to-many mappings between features. Common approaches to affine matching roughly fall into one of the following four categories: 1) key-feature algorithms [9, 21], 2) generalized Hough transforms [4, 17, 24] and pose clustering algorithms [23], 3) geometric hashing [18, 20], and 4) constraint-based tree search [3, 14, 15]. Key-feature approaches to matching seek some easily identified and distinctive feature which predicts the presence of the model as a whole. Generalized Hough transforms, and more generally pose clustering approaches, seek support for specific model-to-image transformations based upon partial matches between small sets of model and image features. Geometric hashing extends the key-features approach by considering larger sets of local subfeatures and multiple models. Constraint-based tree search seeks locally consistent matches by searching, usually depth first, a tree of potential model-to-image feature bindings. All of these techniques hypothesize the presence of an acceptable match. Typically they do not enforce global geometric consistency, leaving this to an auxiliary post-processing algorithm. None deals with the problem of searching through a profusion of possible partial matches for the one which is best.

Unlike the approaches to affine matching just cited, local search matching uses a combination of iterative improvement and random sampling to search the discrete space of many-to-many correspondence mappings between model and image line segments for one that minimizes a match error function. The match error depends upon the relative placement implied by the correspondence. More particularly, to compute the match error the model is placed in the scene so that the appearance of model features is most similar to the appearance of corresponding image features. For affine matching, a least-squares procedure determines the best-fit similarity transformation registering the model to the image. The first term in the match error is a function of the residual squared-error. A second term penalizes matches which omit portions of the model.

Local search matching iteratively improves a current match by repeatedly testing a local neighborhood of matches defined with respect to the current match. Each neighbor is a distinct correspondence mapping between model and image features. Tractable neighborhood sizes, for instance n neighbors in a space of 2^n possible matches, tend to yield tractable algorithms. However, there is an art to designing small neighborhoods that do not induce a profusion of local optima. New neighborhoods definitions have been developed that are particularly well suited to matching geometric features [6, 8].

Despite clever neighborhood definitions, local search can become stuck on local optima. Random sampling offers a probabilistic solution to the local optima problem. The probability of finding the globally optimal match starting from a randomly chosen initial match is analogous to the probability of getting heads when flipping an unfair coin. Even with an unfair coin, it is a good bet that heads will appear at least once in a large number of throws. For instance, using a coin that only comes up heads in 1 out of 10 throws, the odds of getting heads 1 or more times in 50 throws are 99 out of 100. Similarly for local search matching, even if the probability of seeing the optimal match on a single trial is low, the probability of seeing the optimal match in a large number of trials is high.

The combination of iterative refinement and random sampling has proven to be very effective. Under difficult circumstances, this basic form of algorithm reliably finds excellent, and usually globally optimal, matches. The algorithm performs well even when scenes are highly cluttered and significant portions of a model instance are fragmented or missing entirely.

4 Examples

Although other methods are available (see discussion in Section 5), the results in this paper rely exclusively on vanishing point analysis for finding vanishing lines in the image. This simple approach works surprisingly well for many man-made scenes, both indoor, outdoor, and aerial. Vanishing points are found using a standard Hough transform approach [5]. Each line in the image is entered into a two dimensional Hough array representing the surface of a unit hemisphere. Each image line "votes" in a great (semi)circle of accumulators, and potential vanishing points are detected as peaks in the array where several great circles intersect. For most man-made scenes, either two or three clusters will dominate the Hough array. These clusters correspond to the vanishing points of the two or three dominant line directions in the scene. Each pair of vanishing points defines a vanishing line for planes containing lines of those orientations.

The present version of the local search matching system supports four parameter, but not six parameter, affine transformations. [1] We therefore needed to know the lens parameters of the camera for each experiment. It should be stressed that only rough knowledge of the calibration parameters is generally needed to find acceptable matches. The most important parameters to determine are focal length and aspect ratio. We assumed for all our experiments that the image center was at the numeric center of the image, and that the optical X and Y axes were perpendicular and aligned with the row and column axes of the raster image. Aspect ratio was determined from the camera manufacturer's specifications, when available, otherwise it was assumed to be one-to-one. The focal length for each experiment was determined from vanishing point information and *apriori* knowledge that the dominant line directions were perpendicular in the scene [11]. This computation amounts to varying the distance of the focal point from the image until two vectors pointing from the (variable) focal point towards two (fixed) vanishing points in the image are perpendicular.

4.1 Model-to-Image Matching

Figures 1a) and b) show a set of straight line segments extracted from an image of a wall poster using the Burns algorithm [10], and a set of model lines to be matched to the image. The first stage in matching is to detect two clusters of lines converging to the two main vanishing points in the image, and from the resulting vanishing line rectify the image to present a frontal view of the poster (Figure 1c).

The four parameter affine match found by the local search matching algorithm yielded a set of correspondences between model lines and image lines. These correspondences were used to estimate an eight parameter planar projective transformation to bring the model lines into registration with the image data lines, using the least-squares estimation procedure of [12]. Figure 1d shows the transformed model overlaid on the input image lines.

[1]There is a full 3D perspective version [7], but it is inappropriate for these matching problems because exact camera parameters and an initial object pose estimate are required.

4.2 Image-to-Image Matching

Because it does not rely on computing 3D object pose, this approach extends easily to image-to-image correspondence matching. In this case, both images are rectified using the techniques of the last section, and one is treated as the model while the other becomes the data to be matched. The goal is to discover the affine transformation that maps one set of rectified image lines into another.

When both cameras are calibrated, both images can be rectified into four parameter affine mappings of object coordinates to image coordinates. Since the mapping from one image to another can be written by inverting one object-to-image transformation and composing it with the other, and since the four parameter affine group is closed under inversion and composition, the resulting image-to-image transformation can also be described by a four parameter affine mapping. Similarly, when either camera is uncalibrated the resulting transformation between rectified views is a general six parameter affine mapping.

Figure 2 shows an example of image-to-image matching in the context of aerial image registration. Figures 2a) and b) show sets of extracted straight line segments from two aerial photographs. The first image presents a frontal view of the ground plane, a fact verified by vanishing point analysis, which finds two orthogonal sets of nearly parallel lines. At this point we should mention that the term "frontal" was coined with terrestrial robotics in mind, and that within the aerial domain the correct term to use is "nadir". The second image is clearly not a nadir view, a fact again verified via vanishing point analysis. Figure 2c shows these image lines after rectification.

To apply local search matching, image 1 was assumed to be the model and rectified lines from image 2 the data. Both line sets were filtered to include only lines greater than 100 pixels long, reducing the matching problem to 55 long lines in one image and 68 lines in the other. Additionally, the search space was partitioned based upon the dominant orthogonal line directions. The best match found is displayed in Figure 2d.

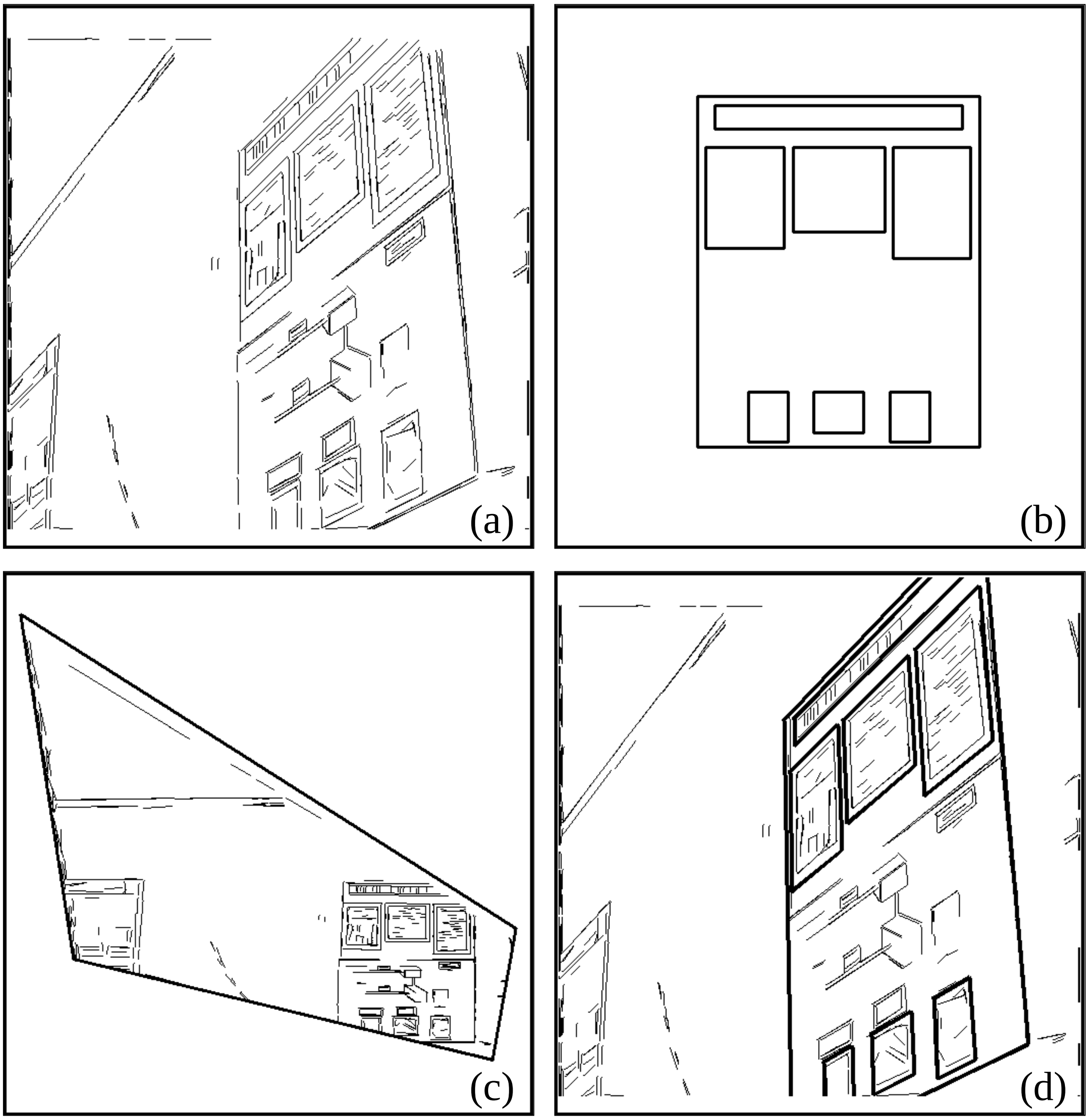

Figure 1: Model-to-image matching example on a poster image: (a) data lines from poster image, (b) poster model, (c) rectified poster data lines, (d) poster model registered with the image data.

Figure 2: Image-to-image matching example on an aerial image: (a) image lines from nadir view, (b) image lines from oblique view, (c) rectified oblique view, (d) registration of nadir view with rectified oblique view.

5 Issues and Extensions

The domains we anticipate are scenes depicting either indoor or urban outdoor environments with much planar and parallel linear structure. Such scenes often contain lines and planes in two or three dominant directions. The approach to matching taken here requires each plane to be matched separately, so a method is needed to partition lines in the image into sets belonging to planes in the world. This would be nearly impossible in monocular images, were it not for the rich structure of man-made environments, suggesting domain-specific heuristics based on corners and perpendicularity. In particular, L-junctions composed of two lines from different vanishing point clusters are good candidates for coplanar corners. We are currently exploring heuristic geometric methods, as well as more formal approaches based on projective invariance, for partitioning image lines into coplanar groups.

We are also exploring other methods besides vanishing point analysis for detecting the horizon line of an object plane in the image. Possibilities include analyzing texture gradients [13], and exploiting properties of the perspective projection of convex planar curves [2].

The techniques presented here may not be adequate to determine feature correspondences when structures are present in the scene that deviate significantly from coplanarity with respect to the viewing distance. However, to the extent that *some* scene features are found to be coplanar and can be successfully matched, this initial set of planar correspondences provide strong constraints on the positions of remaining features. For calibrated cameras, the relative rotation and direction of translation between two camera positions can be computed from the perspective transformation describing how the appearance of a planar structure differs in the two images [12]. This reduces the search for other 3D feature correspondences to that of induced stereo, where corresponding feature points lie along known *epipolar* lines. Even for uncalibrated camera systems, knowledge of the perspective transformation relating the image features of a planar structure constrains the positions of arbitrary point features in one image to lie along epipolar lines in a second image.

In its current form, the local search affine matching algorithm described in this paper is used for image-to-image feature matching simply by declaring the features in one image to be a model. This is not ideal, since the the current treatment of model and image lines is not symmetric. Future work on the affine matcher may include developing a more symmetric error metric for image-to-image matching, and extending the range of the match transformation space to handle six parameter affine matching so that images from uncalibrated camera systems can be used.

References

[1] P. Anandan, "Measuring Visual Motion from Image Sequences," Ph.D. Thesis and COINS Tech Report 87-21, University of Massachusetts, Amherst, MA, 1987.

[2] J. Arnspang, "Moving Towards the Horizon of a Planar Curve,"*IEEE Workshop on Visual Motion,* 1989, pp. 54–59.

[3] H.S. Baird, *Model-Based Image Matching Using Location,* MIT Press, Cambridge, MA, 1985.

[4] D.H. Ballard, "Generalizing the Hough Transform to Detect Arbitrary Shapes," *Pattern Recognition,* Vol. 13(2), 1981, pp. 111–122.

[5] S.T. Barnard, "Interpreting Perspective Images," *AI Journal,* Vol. 21(4), November 1983, pp. 435-462.

[6] J.R. Beveridge, R. Weiss and E.M. Riseman, "Combinatorial Optimization Applied to Variable Scale 2D Model Matching," *Proceedings IEEE International Conference on Pattern Recognition*, Atlantic City, June 1990, pp.18-23.

[7] J.R. Beveridge and E.M. Riseman, "Hybrid Weak-Perspective and Full-Perspective Matching," *Proceedings IEEE Computer Vision and Pattern Recognition,* Champaign, IL, June 1992, pp.432-438.

[8] J.R. Beveridge, "Local Search Algorithms for Geometric Object Recognition: Optimal Correspondence and Pose," Ph.D. Thesis, Department of Computer Science, University of Massachusetts, Amherst, MA 01003, February 1993.

[9] R.C. Bolles and R.A. Cain, "Recognizing and Locating Partially Visible Objects: the Local-feature-focus Method," *International Journal of Robotics Research,* Vol.1, No.3, 1982, pp.57-82.

[10] J.B. Burns, A.R. Hanson and E.M. Riseman, "Extracting Straight Lines," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* Vol. 8, No. 4, July 1986, pp.425-456.

[11] B. Caprile and V. Torre, "Using Vanishing Points for Camera Calibration," *International Journal of Computer Vision,* Vol. 4, 1990, pp. 127–140.

[12] O.D. Faugeras and F. Lustman, "Motion and Structure from Motion in a Piecewise Planar Environ-

ment," *International Journal of Pattern Recognition and Artificial Intelligence,* Vol. 2, 1988, pp. 485–508.

[13] J. Garding, "Shape from Surface Markings," Ph.D. dissertation, Royal Institute of Technology, S-100 44 Stockholm, Sweden, May 1991.

[14] W.E.L. Grimson and T. Lozano-Pérez, "Localizing Overlapping Parts by Searching the Interpretation Tree," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* Vol. 9(3), 1987, pp. 469–482.

[15] W.E.L. Grimson, *Object Recognition by Computer: The Role of Geometric Constraints,* MIT Press, Cambridge, MA, 1990.

[16] B.K.P. Horn, *Robot Vision,* MIT Press, Cambridge, MA, 1986.

[17] J. Illingworth and J. Kittler, "A Survey of the Hough Transform," *Computer Vision, Graphics, and Image Processing,* Vol.44, 1988, pp. 87-116.

[18] A. Kalvin, E. Schonberg, J.T. Schwartz and M. Sharir, "Two-dimensional, Model-based, Boundary Matching using Footprints," *International Journal of Robotics Research,* Vol.5(4), 1986, pp. 38-55.

[19] K. Kanatani, "Constraints on Length and Angle," *Computer Vision, Graphics, and Image Processing,* Vol. 41, 1988, pp. 28-42.

[20] Y. Lamdan and H.J. Wolfson, "Geometric Hashing: A General and Efficient Model-based Recognition Scheme," *Proceedings IEEE Second International Conference on Computer Vision,* Tampa, December 1988, pp.238-249.

[21] D.G. Lowe, *Perceptual Organization and Visual Recognition,* Kluwer Academic Publishers, 1985.

[22] H.S. Sawhney, Ph.D. Thesis, Computer Science Department, University of Massachusetts, Amherst, MA, 1992.

[23] G. Stockman, "Object Recognition and Localization via Pose Clustering," *Computer Vision, Graphics, and Image Processing,* Vol. 40, 1987, pp.361-387.

[24] D.W. Thompson and J.L. Mundy, "Three-Dimensional Model Matching from an Unconstrained Viewpoint," *IEEE Conference on Robotics and Automation,* Raleigh, NC, 1987, pp. 208–220.