A Multiscale Region-Based Approach to Image Matching

Mark Tabb and Narendra Ahuja

Beckman Institute and Department of Electrical and Computer Engineering
University of Illinois, 405 N. Mathews Ave., Urbana, IL USA, 61801
mtabb, ahuja@vision.csl.uiuc.edu

Abstract

This paper presents a new technique for the estimation of 2–D motion fields from image sequences. Homogeneous regions are identified and matched using a multiscale segmentation algorithm. Many matching difficulties result from structural changes in an image which occur only at certain scales, and, hence, a multiscale approach results in a denser set of matched regions. Pixel correspondences are then obtained by estimating an affine transformation between matched regions. A motion field is then calculated from these correspondences. Areas where occlusion is present are identified, and the effects of the occlusion on the affine parameter estimation process are compensated, resulting in a more accurate estimated motion field.

1. Introduction

This paper is concerned with the problem of estimating the 2–D motion, or optical flow, field from a time sequence of images. Previous approaches to this problem can be classified as either pixel-based (intensity-based) or feature-based. Some reviews can be found in [1, 5, 13]. The pixel-based approaches [12, 19] utilize constraints based upon local spatial and temporal derivatives. These methods work well as long as the motion is relatively small between frames and sufficient intensity variation is present. Feature-based methods extract features from images such as points [4], edges [9, 11, 20], corners [15, 20], or regions [7, 10, 14, 16, 17] and match them across frames, thereby obtaining a displacement field. The advantages of this approach are that coarse (large) motions can be modelled and features are more robust to noise and lighting changes than pixel-based methods. The difficulties with this approach include mismatched features and obtaining a dense displacement field from sparse feature matches. This paper presents a new feature-based technique which segments and matches regions of homogeneous intensity or color across frames of an image sequence, and then interpolates a motion field down to the pixel level. Some advantages of a region-based approach over other feature-based methods are that regions are more stable to noise and lighting changes than other kinds of features, and the probability of an incorrect match is much lower because regions have a larger variety and complexity of attributes than the other types of features.

It is important that the segmentation method used to extract regions from the images be multiscale. A multiscale segmentation provides a much richer description of regions available for matching. Both structural changes and noise within a certain area of an image may cause there not to be any matches for regions within that area at a particular scale. However, it is often the case that matches can be found within that area at other scales. As a result, a multiscale method will be able to find region correspondences over a larger fraction of the image than a method which extracts regions at only a single scale. Of the previous region-based methods, only [10] uses a multiscale segmentation method. After the extracted regions have been matched across frames, the methods of [7, 8, 10, 16] assign to the matched region centroids motion vectors corresponding to the displacement of the centroids of the matched regions resulting in a very sparse motion field. Both our method and that of [14] assume that the motion between a pair of matched regions in adjacent frames can be modelled by an affine transformation. This allows very general types of motion to be modelled, including many kinds of nonrigid motion. A descent-type iterative approach is used to calculate the best affine transformation between each pair of matched regions. These transformations provide a dense set of motion vectors between the matched regions. The method of [14] computes the affine transformation which best matches the boundaries of the matched regions, however, this often gives an incorrect result. Instead, our method finds the affine transformation which best accounts for the observed intensities or colors within the matched regions. This results in much more accurate estimated motion.

All types of motion estimation algorithms make errors because of occlusion, but region-based methods are particularly sensitive to this problem. In pixel-based methods, occlusion can cause motion vectors to be estimated incorrectly in the neighborhood of the occlusion. For feature-based methods, occlusion may cause matches not to exist for features near the occlusion, and, as a result,

1051-4651/94 $04.00 © 1994 IEEE

the motion cannot be estimated in these areas. However, if the features are regions, it is usually the case that occlusion will only partially obscure a region. Hence, a region unoccluded in one frame and partially occluded in the following frame will still be matched, but the partial occlusion can change the shape of the region enough to cause the estimated motion parameters to be incorrect. If this region is large, motion vectors may be estimated incorrectly far away from the occlusion. Thus, occlusion can cause global errors in a region-based method. The method presented in this paper reduces this problem by estimating and compensating for the effects of occlusion.

The rest of this paper is organized as follows: Section 2 gives an overview of the multiscale region segmentation and matching process. Section 3 describes the method for obtaining a dense motion field within matched regions by computing an affine transformation between each pair of matched regions. Section 4 presents the method by which the effects of occlusion are compensated when the affine transformations are calculated. Section 5 shows experimental results, and Section 6 makes some concluding remarks.

2. Region segmentation and matching

The processes of region segmentation and matching are integrated together in this approach. We treat an image sequence as a 3–D volume with time as the third axis. A 3–D, multiscale segmentation is then performed on this volume. The assumption is that a given region will overlap spatially with itself at least partially between adjacent frames. Each segmented 3–D region is a "tube" consisting of a 2–D region and its correspondences throughout the sequence.

The region segmentation algorithm presented in [18] transforms an image into a hierarchy of force fields using the transform of [2, 3]. This transform encodes the structural information of the image into the field hierarchy. This encoding allows regions to be extracted from the field hierarchy using only local operations. This method is robust to noise, yet the identified regions retain detailed boundary features. The resulting multiscale region hierarchy has the property that a region at any given scale is composed of a set union of regions existing at any finer scale. Also, the scale parameter is directly related to the similarity of a region with its neighboring regions. For grayscale images, this similarity is measured in terms of intensity differences, and for color images the similarity is measured by distance in some color space. Originally, the 3–D segmentation was done by extending both the transform and subsequent region identification from 2–D to 3–D. Evaluating this transform in 3–D incurs significant computational expense, and, as a result, a method was developed which performs the 3–D segmentation using a 2–D transform.

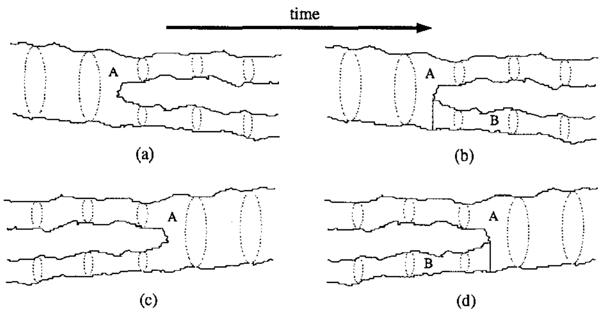

Figure 1. Tube A undergoes a bifurcation in (a), which causes a region at some time to be matched to two regions. In (c), Tube A undergoes a merging, which causes two regions to map to the same region at some time. These problems are resolved by the creation of Tube B as shown in (b) and (d).

The segmented tubes require some post-processing because structural changes occurring in an image sequence may result in tubes which contain bifurcations and mergings. As a result, a region in one frame may be matched to multiple regions in the following frame and vice versa. When a multiple match situation exists, one of the regions is selected arbitrarily as being the correct match, and a new tube is created for each of the other regions. If the selected match is incorrect, it will be identified when the affine transformation parameters are computed. Figure 1 illustrates this process. Tube A contains a bifurcation in 1(a). The multiple match at the bifurcation was resolved with the creation of Tube B as shown in 1(b). Tube A contains a merging in 1(c), which resulted in the creation of Tube B shown in 1(d).

3. Estimating affine transformation parameters and the resulting motion field

For each pair of regions matched in adjacent frames, the best affine transformation between them is estimated iteratively. Denote by R_i^t the region corresponding to the cross-section of the ith tube at frame t. Also, denote the coordinates of the pixels within R_i^t by $\left(x_{ij}^t, y_{ij}^t\right)$ with $j = 1..|R_i^t|$, where $|R_i^t|$ is the cardinality of R_i^t, and denote the pixel nearest the centroid of R_i^t by $(\bar{x}_i^t, \bar{y}_i^t)$. Each $\left(x_{ij}^t, y_{ij}^t\right)$ is mapped by an affine transformation to the point $\left(\hat{x}_{ij}^t, \hat{y}_{ij}^t\right)$ according to

$$\begin{aligned}\left(x_{ij}^t, y_{ij}^t\right)^T \rightarrow & \left[\mathbf{A_k}\begin{pmatrix} x_{ij}^t - \bar{x}_i^t \\ y_{ij}^t - \bar{y}_i^t \end{pmatrix} + \vec{T}_k + \begin{pmatrix} \bar{x}_i^{t+1} \\ \bar{y}_i^{t+1} \end{pmatrix}\right] \\ & = \left(\hat{x}_{ij}^t, \hat{y}_{ij}^t\right)_k^T \end{aligned} \qquad (1)$$

A 2x2 deformation matrix, $\mathbf{A_k}$, and a translation vector, $\vec{T}_k$, comprise the affine transformation. The subscript k

indicates the iteration number, and $[\cdot]$ indicates a vector operator which rounds each vector component to the nearest integer. Define the indicator functions

$$\lambda_i^t(x,y) = \begin{cases} 1, (x,y) \in R_i^t \\ 0, \ \ else \end{cases} \tag{2}$$

The amount of mismatch is measured as

$$\left(M_i^t\right)_k = \sum_{x,y} \|\vec{I}_t(x,y)-\vec{I}_{t+1}(\hat{x},\hat{y})\| \cdot \left[\lambda_i^t(x,y) + \lambda_i^{t+1}(\hat{x},\hat{y}) \right. \\ \left. -\lambda_i^t(x,y) \cdot \lambda_i^{t+1}(\hat{x},\hat{y})\right] \tag{3}$$

where $\vec{I}_t(x,y)$ represents the value of the pixel at coordinate *(x,y)* in frame *t*. For grayscale images, $\vec{I}$ is a scalar representing pixel intensities. For color images, we define $\vec{I}$ as a 3–D vector representing pixel values in the CIE 1976 (L*u*v*) space [6]. The Euclidean norm is used in both cases. The rounding performed in (1) causes M_i^t to be larger than would be the case if $(\hat{x},\hat{y})$ was a rational coordinate and interpolation kernels were used to estimate $\vec{I}_{t+1}(\hat{x},\hat{y})$ from the four pixels nearest to $(\hat{x},\hat{y})$. However, the estimated affine transformations are very similar with and without rounding.

The affine transformation parameters which minimize M_i^t are estimated iteratively using a straightforward local descent criterion. As initial guesses, we set $\vec{T}_0 = \left(\bar{x}_i^{t+1} - \bar{x}_i^t, \bar{y}_i^{t+1} - \bar{y}_i^t\right)^T$ and $\mathbf{A}_0$ to the final value of $\mathbf{A}$ computed between R_i^{t-1} and R_i^t if R_i^{t-1} exists, and to the identity matrix otherwise. The initial guess is usually very close to the optimal transformation, hence, it is assumed that the first local minimum reached in the descent process corresponds to the global minimum. Typically, about 10 iterations are required for convergence.

Due to structural changes in the scene and very fast motion, some of the identified region matches will be incorrect. Bad matches typically result in a computed affine transformation with a larger amount of deformation than will typically occur between adjacent frames. Thus, letting a_{ij} denote the elements of $\mathbf{A}$, if

$$(|1-a_{00}| > \epsilon) \cup (|a_{01}| > \epsilon) \cup (|a_{10}| > \epsilon) \cup (|1-a_{11}| > \epsilon) \tag{4}$$

is true, a match is considered incorrect. The value of the constant ϵ used is 0.3.

A motion field is computed for each frame from the pixel correspondences of the well-matched regions. Because the segmentation is multiscale, a given pixel may belong to more than one region which is well-matched. Hence, multiple motion vectors will exist for this pixel, although they will typically be very similar. This ambiguity is resolved by selecting the motion vector resulting from the affine transformation which gives the smallest mismatch error when applied to the smallest (finest scale) well-matched region.

4. Compensating for occlusion

If a region is unoccluded in one frame and its match is partially occluded in the following frame, then the motion vectors estimated in section 3 for this region may have some error caused by the changed shape of the region due to the partial occlusion. In this section, we show how to compensate for this changed region shape.

We first compute a predicted motion field for the present frame before the affine transformation parameters for this frame are computed. This is accomplished by taking the affine transformation parameters for the regions in the previous frame which were used in computing the previous motion field, and then applying these parameters to the corresponding regions in the *present* frame. If the motion field is reasonably smooth from frame to frame, then this predicted motion field will be fairly close to the actual motion field in the present frame. Motion vectors from two or more regions which map onto the same pixel indicate that the pixels are visible in the present frame but all but one of them are occluded in the next frame (VPON). Similarly, a pixel to which no motion vectors map is occluded in the present frame and visible in the next frame (OPVN). Before the affine transformation parameters are computed between a given region in the present frame and its match in the next frame, the shape of the region in the next frame is modified. First, connected sets of VPON pixels are identified. Any of these sets which are adjacent to the region are appended to it. Next, any OPVN contained within the region are subtracted from it. The shape of the resulting region is now no longer distorted by the occlusion. Figure 2 demonstrate why this is the case.

5. Experimental results

Figure 3 gives an example of the performance of our motion estimation algorithm. Two consecutive frames of a color image sequence of a football game are shown in 3(a-b). The motion in this sequence is very fast and non-rigid, and there is also a significant amount of occlusion. Computed motion fields are shown sub-sampled in 3(c-d). The effects of occlusion were not compensated in 3(c), whereas in 3(d) they were, resulting in a significant improvement in the accuracy of the estimated motion field. The best set of matched region pairs for these two frames is shown in 3(e-f). Each matched pair is displayed with one of six different intensity values, and all neighboring regions have been assigned different intensities. Regions for which no motion could be computed are shown in black. The main reason motion could not be computed for these regions is that no matches were found for them.

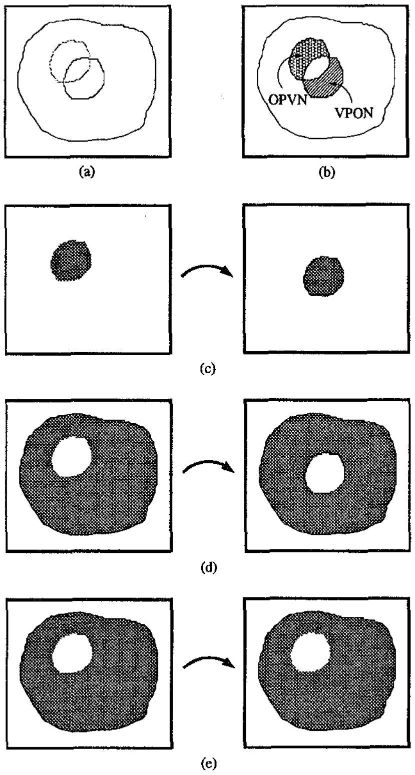

Figure 2. (a) A stationary region is occluded by a smaller region whose position in the present frame is shown with a dotted boundary and in the next frame with a solid boundary. (b) The corresponding VPON and OPVN pixels are shaded. (c) The shape of the smaller region is shown for both frames. The shape of the smaller region in the next frame is unchanged by the occlusion compensation. (d-e) The shape of the larger region in both frames is shown before (d) and after (e) the occlusion compensation. In (e), the larger region in the next frame has had the VPON appended to it and the OVPN subtracted from it, which compensates for the shape distortion caused by the occluding region.

This is due to the matches being fully occluded in the other frame or structural changes which cause the match to belong to a different tube. A total of 43 best matched region pairs were identified, resulting in motion vectors computed over 96% of the first frame. Through visual inspection, the motion estimated for each of these region pairs was determined to be consistent with the actual motion.

6. Conclusions

We have presented a new region-based technique for 2–D motion estimation. A multiscale segmentation algorithm is used which provides a rich set of matched regions. Pixel correspondences are obtained for each pair of matched regions by estimating the parameters of the best affine transformation relating the regions. The affine transformation allows very complex motion fields to be modelled. Before computing the affine transformation parameters, the distortion in region shapes caused by occlusion is compensated. A motion field is then estimated from the affine parameters calculated for the set of regions identified as being the best matched.

References

[1] J. Aggarwal and N. Nandhakumar. On the computation of motion from sequences of images- a review. *Proc. IEEE*, 76:917–935, August 1988.

[2] N. Ahuja. A transform for detection of multiscale image structure. In *Comp. Vision Patt. Recog. '93*, pages 780–781, New York, June 1993.

[3] N. Ahuja. A transform for detection of multiscale image structure. In *Image Understanding Workshop*, pages 893–903, Washington, D.C., April 1993.

[4] S. Barnard and W. Thompson. Disparity analysis of images. *IEEE Trans. on Patt. Anal. and Mach. Intell.*, PAMI-12(4):333–340, July 1980.

[5] J. Barron, D. Fleet, S. Beauchemin, and T. Burkitt. Performance of optical flow techniques. In *Comp. Vision Patt. Recog. '92*, pages 236–242, Champaign, 1992.

[6] C.I.E. Colorimetry proposal for study of color spaces. *J. Optical Soc. Am.*, 64:896–897, June 1974.

[7] C. Fuh and P. Maragos. Region-based optical flow estimation. In *Comp. Vision Patt. Recog. '89*, pages 130–135, San Diego, 1989.

[8] C. Fuh, P. Maragos, and L. Vincent. Visual motion correspondence by region-based approaches. In *Proc. ACCV '93*, pages 784–789, Osaka, Japan, 1993.

[9] S. Haynes and R. Jain. Detection of moving edges. *Comput. Vision, Graphics, & Image Process.*, 21(3):345–367, 1982.

[10] G. Healey. Hierarchical segmentation-based approach to motion analysis. *Image and Vision Computing*, 11(9):570–576, November 1993.

[11] E. Hildreth. Computations underlying the measurement of visual motion. *Artif. Intell.*, 23(3):309–354, 1984.

[12] B. Horn and B. Schunck. Determining optical flow. *Artif. Intell.*, 17(1):185–203, August 1981.

[13] T. Huang and R. Tsai. Image sequence analysis: Motion estimation. In T. Huang, editor, *Image Sequence Analysis*. Springer-Verlag, 1981.

[14] D. Kalivas and A. Sawchuk. A region matching motion estimation algorithm. *CVGIP: Image Understanding*, 54(2):275–288, 1991.

[15] H. Nagel. Displacement vectors derived from second-order intensity variations in image sequences. *Comput. Vision, Graphics, & Image Process.*, 21(1):85–117, 1983.

[16] K. Price and R. Reddy. Matching segments of images. *IEEE Trans. on Patt. Anal. and Mach. Intell.*, PAMI-1(1):110–116, January 1979.

[17] S. Sull and N. Ahuja. Integrated 3–d analysis and analysis-guided synthesis of flight image sequences. *IEEE Trans. on Patt. Anal. and Mach. Intell.*, 16(4):357–372, April 1994.

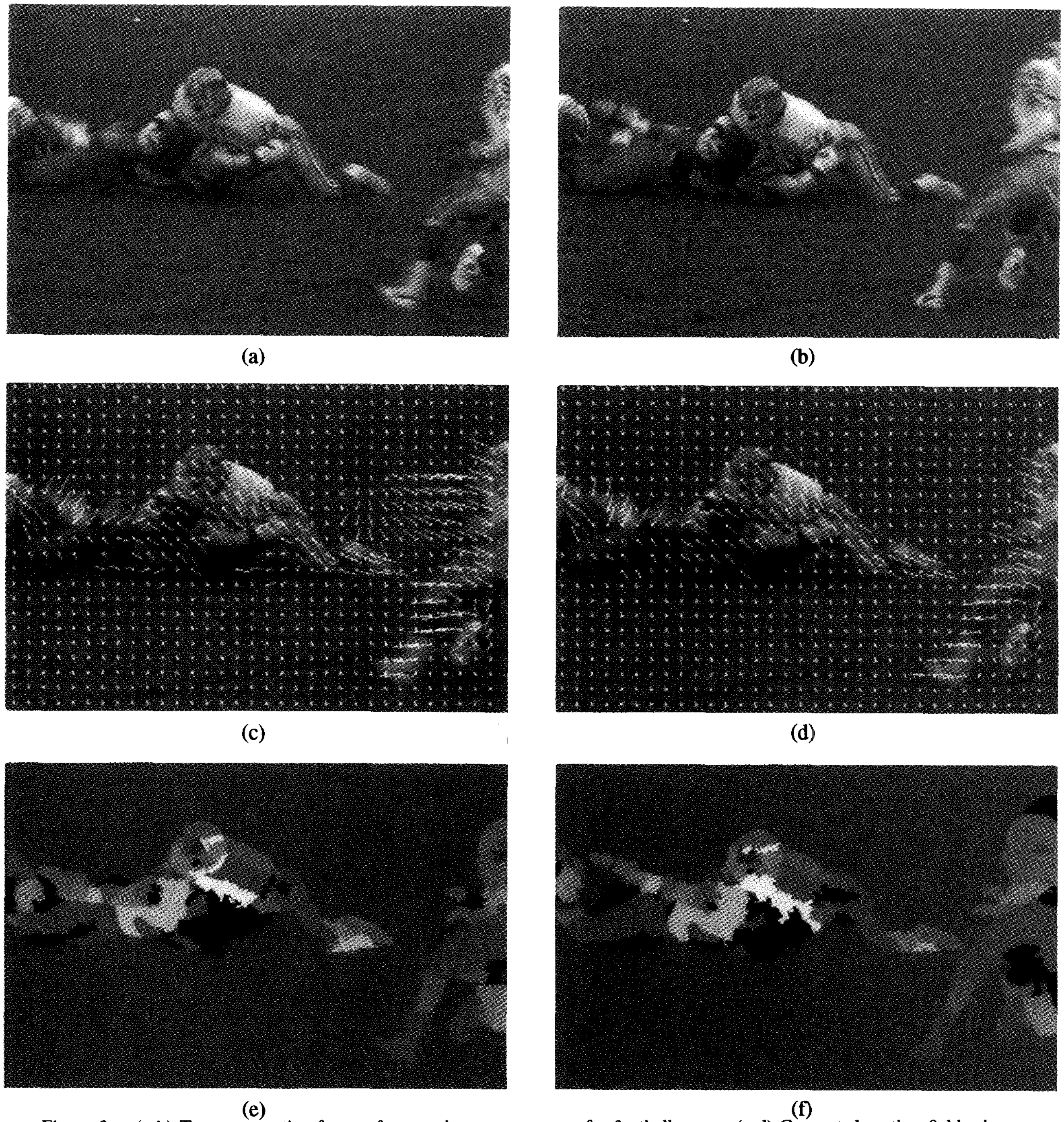
(a) (b) (c) (d) (e) (f)

Figure 3. (a-b) Two consecutive frames from an image sequence of a football game. (c-d) Computed motion fields shown sub-sampled and overlaid onto the image in (a). Motion fields are shown without compensating for occlusion (c) and with the occlusion compensation (d). (e-f) The best set of matched region pairs. Each region pair is displayed with one of six different intensities with all neighboring regions being assigned a different intensity. Regions for which no motion could be computed are shown in black.

[18] M. Tabb and N. Ahuja. Detection and representation of multiscale low-level image structure using a new transform. In *Proc. ACCV '93*, pages 155–158, Osaka, Japan, 1993.

[19] A. Verri, F. Girosi, and V. Torre. Differential techniques for optical flow. *J. Optical Soc. Am.*, 7(5):912–922, 1990.

[20] J. Weng, N. Ahuja, and T. Huang. Matching two perspective views. *IEEE Trans. on Patt. Anal. and Mach. Intell.*, 14(8):806–825, August 1992.