Proceedings of the 2007 IEEE Symposium on Computational Intelligence in Image and Signal Processing (CIISP 2007)

Fuzzy Motion Interpolation for Mesh–Based Motion Estimation

Abouzar Eslami
Robotics and Machine Vision Laboratory
Department of Electrical Engineering
Sharif University of Technology
P. O. Box 11365-9363, Tehran, Iran
Email: eslami@ee.sharif.edu

Nasser Sadati, *Member IEEE*
Intelligent Systems Laboratory
Department of Electrical Engineering
Sharif University of Technology
P. O. Box 11365-9363, Tehran, Iran
Email: sadati@sina.sharif.edu

Mehran Jahed
Robotics and Machine Vision Laboratory
Department of Electrical Engineering
Sharif University of Technology
P. O. Box 11365-9363, Tehran, Iran
Email: jahed@sharif.edu

Abstract— **Mesh–based motion estimation is an important tool for video coding especially with low bit rate. In this paper, a new method for interpolating pixel motion from adjacent mesh nodes with the ability of omitting independent nodes is proposed. By exploiting fuzzy rules to determine the association of pixel and neighboring nodes, the proposed interpolation can detach pixels from some nodes. Consequently, it can deal with those critical patches on objects boundary which their nodes do not belong to one object. Updating the membership functions of each rule with specified strategy makes the interpolation adaptive with non–stationary conditions of image sequences and decreases sensitivity to initial selection of parameters. Experimental results show that the proposed method, in comparison with the conventional methods, increases the motion interpolation accuracy while does not accompanies with massive time cost and complexity.**

I. Introduction

Motion estimation is a basic part of video coding regarding its efficiency in reducing temporal redundancy [1]. Two frequently addressed motion estimation methods are block–based and mesh–based motion estimation. In conventional block–based method, the motion model is restricted to translational applied not to each object but to the blocks that can contain more than one object [2], [3]. This results in low pick signal to noise ratio (PSNR) of block–based motion estimation especially when moving objects are small. On the other hand, mesh–based motion estimation can cover more sophisticated motion models as affine, perspective and etc., and reach to larger PSNR [4], [5]. In this method, a mesh is established for the video frame and the video motion is represented by the deformation of that mesh. Indeed, the frame motion is represented by warping of deformable patches resulted from mesh nodes motion. While the block-based method can lead to severe block distortions, the mesh-based method may cause warping artifacts [6].

The mesh structure can be re–initialized by few frames but it is more usual to re–initialize it per frame regarding to non–stationary inherent of image sequences. In nomenclature, the frames associated with the initial and warped meshes are called reference and target frames, respectively. The reference frame is previous to target one in forward computing and conversely in backward. There are two alternatives for initial mesh geometry: regular, and adaptive or content–based including regular non-uniform, irregular conformal connected and irregular non-conformal connected [7]. Regular mesh is commonly used due to its simplicity. Moreover, with regular mesh there is no need to send the mesh topology to the decoder because it is predefined. Rectangular or triangular patches are two usual patch shapes used in the mesh structure.

Principally, mesh–based motion estimation (synthesis problem) consists of two steps: motion estimation for mesh nodes and motion interpolation for others. A simple approach to estimate the node's motion is exhaustive search for the best match of node and its neighborhood in target frame. Such an approach does not consider global distortion and is critically local. Resulted motion vectors can cause nodes to cross each other and so destroying patch connectivity, especially for small patches [8]. In particular, hexagonal matching procedure is proposed where the motion vector at a node point is estimated by iterative local minimization of the prediction error [9]. This method is extended by employing a hierarchy of regular meshes such that the motion estimation with a coarse mesh provides the initialization for the next (finer) level of the mesh [10]. However, the primary exhaustive search method is not obsolete and can be used with some constraints regarding its simplicity and speed.

The final step is determination of the motion of pixels inside each patch by interpolating from neighboring nodes motion vector. Motion modelling is a powerful tool for representing spatial transformation of patches. Whether the mesh structure is rectangular or triangular, the bilinear or affine model parameters will be approximated from mesh nodes motion vector. Finally, the motion of internal pixels will be determined by a function of their coordination and motion model with acquired parameters [8].

A simpler but faster scheme is spline method and averaging. In this approach, the motion of inner pixels will be determined by a weighted averaging of surrounding nodes motion. A decreasing function of Euclidian distance generates the weight of each node. This strategy, applied to patches in here, is close to the RBF registration method in which the whole image spatial transformation will be approximated from land mark translation and their distance from pixels [11]. Despite the simplicity, this method is completely flexible and can cover

1-4244-0707-9/07/$25.00 ©2007 IEEE

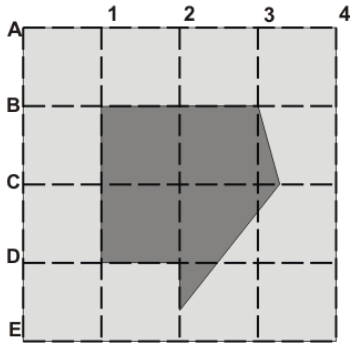

Fig. 1. A synthetic sample of patches containing more than one object.

many forms of motion both locally and globally.

Since the inner patch pixels occupy a large surface of image, the more comprehensive interpolation causes the less motion estimation error. Whether the motion model (affine or bilinear) or averaging is employed, the patches are assumed fully connected. Except the smooth deformations representable by addressed interpolation methods, occlusion and detachment are challenging problems in literature [12].

Referring to Fig. 1, where a simple frame including one object and regular mesh is shown. Patch $[D2, D3, E3, E2]$ is an instance in which the smooth previous interpolations are inappropriate and result in high distortion. Explicitly, it is not meaningful to estimate the motion of background pixels from foreground ones and vice versa. A primary solution is preventing from such conditions using content–based meshes and locating initial nodes so that each patch contains just one object [13]. Although this scheme successfully increases the total accuracy, but also introduces extra process to locate the nodes on boundaries and moving regions (with optical flow aspect). Despite nodes freedom in their initial position, some considerations are also required to inhibit mesh shrinking which makes the resulted mesh disable of representing the whole image deformation. Because of adaptive inherent of content–based meshes, all processes should be performed per frame. This amount of process is not desirable and not affordable in fast applications.

The exhaustive splitting scheme proposed by Hsu and et. al. is based on comparing all possible cases in splitting each patch to reach the minimum distortion [14]. The mean square or absolute error between target and motion compensated reference frames, after comparing with a predefined threshold, determines the need to split the undergoing patch. Despite the objective of having the maximum accuracy in choosing the split pattern, this adaptive interpolation is still suboptimal because of the predefined threshold and limited number of split patterns. A series of processes including motion interpolation, compensation and distortion computation should be performed for each splitting pattern. Regarding to the large number of patterns considered for each patch (32 patterns) and the whole number of patches to split, it can be inferred that the practical applications of this adaptive interpolation is restricted because of its computational complexity.

In this paper, in order to remove the inherent disability of regular meshes, against patch discontinuity, an adaptive interpolation is proposed based on fuzzy splitting of patches. For this purpose, three fuzzy rules determine the dependency of undergoing pixel to each adjacent node and their influence on pixel interpolated motion. Because of considering each pixel individually, various splitting patterns can be generated. Furthermore, there is no backward computation (motion compensation for distortion calculation) required and the motion interpolation performs through fuzzy rules. The fuzzy inference can be now based on the difference of intensity, difference of temporal intensity variation and distance.

As a result of adaptation in membership functions, the proposed interpolation is less sensitive to non–stationary condition of image sequence and the initial selection of parameters. The adaptation removes the need for exhaustive checking of all possible states as in exhaustive splitting method ([14]). Indeed, the proposed method attempts to reach to less motion estimation error by training and adaptive selection of membership parameters during frames. Experimental results show that the proposed method provides high accuracy results, close to the content–based meshes while it is faster than them and much more faster than the exhaustive method. So the regular meshes with fuzzy motion interpolation can be recommended for many fast and accurate applications with low computational cost.

In the following section, the theory of fuzzy interpolation and its practical considerations are proposed. The experimental results of applying the proposed scheme in comparison with other schemes are available in section III. Finally, in section IV some concluding remarks are briefly discussed.

II. Fuzzy Motion Interpolation

The accuracy of mesh–based motion estimation strictly depends on interpolation strategy for inner patch pixels. Assumption of smoothness in motions over patch is not fair on object boundaries, where patches contain more than one moving objects. Among all mesh–based methods, which their properties are theoretically discussed in section I, the exhaustive splitting method exploits regular mesh and attempts to make splitting patches. This is exactly the principal of the proposed method in this study, since the regular mesh decreases the process time cost and also the required bit rate, while the patch splitting increases the total accuracy. By slicing patches with straight line in different positions and into two groups of 1–3 and 2–2 nodes, 32 different patterns are generated in the exhaustive splitting method. Pixel–based processing and examining the independency of each pixel–node pair, can provide much more arbitrary patterns.

Determining the coherency for each pixel–node pair is a semi–segmentation problem over each patch that uses general information but performs locally. There are different approaches for segmentation both for still images and image sequences. Various features as pixel motion and intensity are proposed, although common problem of pixel–based segmentation is uncertainty. In processing of individual pixels, there is

no confidential information about objects shape, texture and etc. In the case of comprehensive database, it is practically impossible to approximate histogram and the position information of all objects so that exploiting these information will be inefficient. Fuzzy logic is appropriate tool for dealing with such problems by considering possibility for all conditions [15]. In special case of this study, instead of crisp decision about each pixel–node coherency, a possibility is assigned. The possibility value determines the ratio of each node influence on pixel motion.

The following three fuzzy rules, based on three pixel properties, can determine the coherency ratio between each pixel and node:

1) If their difference in intensity is low, then they are dependent.
2) If their difference in temporal intensity variation is low, then they are dependent.
3) If they are close, then they are dependent.

Optimal thresholding of intensity and fuzzy clustering (FCM) segmentation of intensity both are shown to be useful in medical and natural images. In most images, especially natural ones, adjacent pixels from the same object, have similar intensities. The reason is the limited spatial frequency content of natural objects. Generally, it is not true to assume all pixels of one object will have the same intensity and it is not true to assume two pixels with the same intensity belong to one object. The first rule is not held all over the image but since the support region of employed rules is just one patch, which inherently applies a spatial restriction, it is desirably applicable.

An appropriate membership function for *having identical intensity* is the joint probability of two intensities estimated from objects histogram. This value represents the probability of two intensities both belonging to one object. Having priory knowledge about objects number and their histogram, the membership value is the average of such joint probabilities over all objects when objects have the same occurrence probability. In lack of priory knowledge, the gaussian probability density function (which is also used for threshold selection in segmentation problems [16]) can be assumed for dominant objects. Assuming gaussian pdf for each object intensity with mean of μ and variance of σ^2, the joint probability of two independent intensities (i_1 and i_2) can be written as

$$p = \frac{1}{\sqrt{2\pi}\sigma} \exp(-\frac{(i_1-\mu)^2 + (i_2-\mu)^2}{2\sigma^2}) \qquad (1)$$

Now suppose i_2 is a sample of object intensities, the maximum likelihood approximation of μ is i_2 which is a rough estimate but best available one. Consequently, the membership function of *having identical intensity* can be given as:

$$u1 = \exp(-\frac{(i-i_0)^2}{\sigma^2}) \qquad (2)$$

where i_0 is the node intensity and i the intensity of undergoing pixel. The parameter of σ^2 determines the influence of intensity in comparison with other two features. The larger σ^2, the more dominant will be the first rule so that more pixels satisfy the first condition. This parameter can be fixed experimentally or trained adaptively (strategy of this study) to reach to less distortion.

Optical flow equation contains spatio–temporal property, used both for motion estimation and segmentation. The variation rate of intensity per time unit for each pixel is a function of its spatial frequency ($\mathbf{f} = \nabla I(x,y)$) and motion vector ($\mathbf{m} = [dx/dt, dy/dt]^T$), expressed by

$$\frac{dI}{dt} = -\mathbf{f}^T\mathbf{m} \qquad (3)$$

where $I(x,y)$ represents the intensity of pixel $[x,y]^T$. Equation 3 shows the relation between temporal intensity variation rate (dI/dt) with spatial frequency and motion vectors. If two pixels belong to one object (hopefully have the same spatial frequencies and have identical motion vectors), then their intensity variation from reference to target frame will be identical. Generally, it is not true to conclude from two pixels with identical dI/dt that they belong to the same object. It is not even true to assume two pixels of one object have identical spatial frequency or identical motion vector. So, the second rule is not justifiable unless for close pixels and ambiguity is associated with segmentation based on optical flow equation. It is even worse since the temporal intensity variation itself is the product of two other features. Between these features, the spatial frequency is not reliable for segmentation because it is critically localized and incapable of representing neighborhood texture. In other hand, the motion vectors are much more useful. It is expected for two close pixels of one object to have almost similar motion vectors. It is hard to explicitly conclude from dI/dt whether its variation is because of $\mathbf{f}$ or $\mathbf{m}$. The exploited membership function for the premise part of rule number two (*having identical temporal intensity variation*) is as follows

$$u_2 = (1+\alpha x)^{-1/2} \qquad (4)$$

where x is the absolute value of difference between the dI/dt of undergoing pixel and each node. Equation 4 is useful when intensities are unsigned integers between $[0, 255]$. The only adaptive parameter is α which determines the pass width of membership function.

Totally, for two close pixels, it is more probable to have the same motion vector. So, the influence of each landmark motion in interpolating the motion of one pixel, can be decreased by their distance. This is much similar to the principals of RBF interpolation. it should be mentioned that this hypothesis can be false even for two neighboring pixels. The other rules are mentioned to remove this disadvantage of conventional RBF method. A well known radial basis function is $r^2 \log r$, namely thin plate spline (TPS). The TPS minimizes the bending energy (quadratic function of variation over image), but is not descending. The alternative function used as the membership function of *being close* can be given by

$$u_3 = \exp(-\beta r^2) \qquad (5)$$

where β is the adaptive parameter and r the Euclidian distance between pixel and node.

It is possible to include more rules based on other features as texture (considering their neighborhood). As another instance, the type of intensity variation over the straight line from undergoing pixel to node, can be employed as a dependency factor. A rapid step or impulse variation probably indicates the separability. The reliability, accuracy and computational cost are the important factors in selecting features and rules. Those features as texture that consider neighborhood, cause blurring that is not desirable regarding to the small surface of patches. All above three pixel features (intensity, temporal intensity variation and distance) can be computed quickly. Complex and time consuming features are not desirable in motion estimation, where process speed is important. The experimental results also approve these properties both in accuracy and speed.

The membership function of rules' consequence part (*being dependent*) is *tansig* function, having two adaptive parameters; shift and slope as stated in equation 6. Dynamic range of all proposed membership function is restricted to $[0, 1]$, So there is no further normalization required. Figure 2 depicts a typical output membership function as described by

$$y = 0.5 - 0.5tansig(\frac{x-a}{b}) \tag{6}$$

In order to decrease sensitivity to initial selection of parameters and increase the ability of proposed method in following the image sequences' dynamic, training of parameters accomplishes in some iterations during a few sequential frames.

Using the $\max - \min$ rule of inference and the center of gravity's defuzzification, the weight of each node in interpolating the pixel motion is determined. The total motion of pixel is the sum of neighboring nodes motion weighted by their specific dependency inferred from the above procedure. Since just one rule will be fired for each pixel, then only one rule will be trained (suboptimal but fast and simple solution). Training strategy is the gradient descent (GD) method (same as the back propagation training in neural networks). For training by GD method, the ideal motion vectors (targets) are required in addition to the estimated ones. So in each frame, in addition to the mesh nodes, the real motion of some other randomly selected pixels should be estimated by exhaustive search. The exactly determined motion of these pixels are used to modify the weights (membership function parameters).

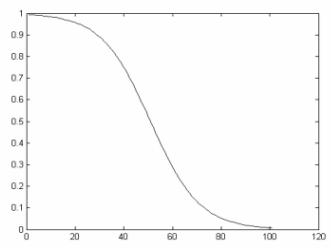

Fig. 2. A typical output membership function.

The limited support range of rules is already discussed and justified. But local processing associates with blocking artifacts which can cause severe discontinuities from one patch to the next one. Global training of rules membership functions by choosing the training control points is necessary to prevent from such artifacts and to generalize the adapted parameters. It is hard to certify the convergence but experimental results show that the training accomplishes in few iterations.

It is possible to have three separate rules for each patch individually. But this will increases the required memory for process which is not desirable especially with respect to no considerable increase in motion estimation accuracy. On the other hand, increasing the number of membership functions introduces more adaptive parameters to be trained which results in two problems; 1) More required control points with exactly determined motion vector for training, which costs more time. 2) Overtraining is more probable, in which the error is minimized in training but the estimation result is not appropriate and general. The other choice is contributing farther nodes in interpolating pixels motion. But all three rules are justified by assumption of restricted support range. By including farther nodes, the assumptions will be less reliable. It should be mentioned that, the membership functions could be trained in just one or two first frames by providing more training samples from these frames. But this scheme removes the ability of the proposed method in following the dynamics of image sequence and results in more average error. Whether the motion estimation is applied forward or backward and wether the mesh initializes per frame or not, the fuzzy motion interpolation is applicable since it exploits the segmentation basics with adaptive scheme.

III. Simulation Results

The proposed method with some other conventional motion estimation strategies are applied to three typical image sequences (*garden*, *house*, *tennis*). Applied methods include four step search (FSS) [2] and normalized partial ridgelet distortion search (NPRDS) [3] from block–based, and adaptive interpolation from mesh–based motion estimation. First, the proposed training procedure should be tested. As discussed before, exploiting few rules and few adaptive parameters results in fast training and appropriate decrease in interpolation error after few number of frames. Figure 3 depicts the motion estimation error for two sequences (*garden* and *house*). It can be inferred that after few number of frames of *garden*, the training is accomplished with acceptable accuracy while the rate of learning is much less in sequence *house*. The rate of learning depends on undergoing image sequence and the type of motions.

Table I shows the experimental results where the PSNR criterion represents the motion estimation error and is calculated using the following equation

$$PSNR = 10 \log \frac{255^2}{|I(x,y) - \hat{I}(x,y)|} \tag{7}$$

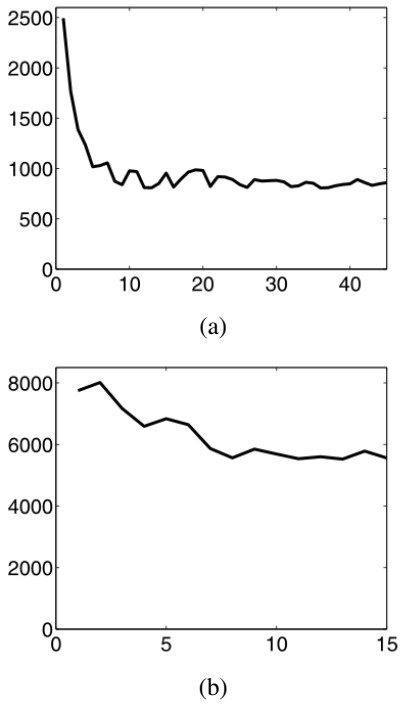

Fig. 3. Motion compensation error of typical image sequences: (a) *garden*, (b) *house*.

TABLE I
THE PSNR OF MOTION ESTIMATION

method	*garden*	*house*	*tennis*
FSS	52.23	30.71	65.27
NPRDS	51.05	27.35	58.61
Hexagonal	61.23	36.85	75.84
Exhaustive splitting	61.78	37.12	76.43
Fuzzy interp.	66.34	40.88	80.72

where $\hat{I}(x, y)$ is the motion compensated frame. Table I implies that the fuzzy rules result in a 5-db increase in motion estimation accuracy. It should be mentioned that although fuzzy rules have explicit influence, since better and more precise representation of pixel dependency, but their important ability is introducing detachment to mesh–based motion estimation. In contrast to motion interpolation with motion modelling, the proposed method inspects motion smoothness and determines dependency of pixel to each node independently. The proposed method can split the patch and omit one or some nodes from interpolation.

Among previous methods of motion estimation, the exhaustive splitting interpolation is the only approach with similarity to the proposed interpolation. Table I shows that the result using the exhaustive splitting are somehow similar to the proposed approach. But it must be considered that the exhaustive splitting is based on exhaustive try and error scheme. So that all possible patterns will be considered, and interpolation accuracy is calculated in all patterns and the best is chosen. Consequently, the time and computational cost is too much. Furthermore, the exhaustive splitting does not apply a precise interpolation when just one node is separated. In this situation the motion of two nodes are used for interpolation which decreases the accuracy in comparison with fuzzy interpolation. Actually, the fuzzy interpolation of this study outperforms the exhaustive splitting strategy in both fields.

IV. CONCLUSION

In this paper, a new motion interpolation method is proposed for mesh–based motion estimation. In this method, the fuzzy rules determine the dependency of undergoing pixel to each node. Consequently, it has the ability of splitting the patches and removing the conventional methods limiting assumption of motion smoothness. Fuzzy logic can appropriately deal with uncertainty in segmentation and determining dependency of pixels. Dependency of each pixel to one node is determined by the rules on their difference in intensity, difference in temporal intensity variation and also distance. Adaptive training reduces sensitivity to initial selection of membership functions, parameters and makes the fuzzy rules to be compatible with variations of image sequences and frame contents. Experimental results show that the fuzzy interpolation enhances the motion estimation accuracy with rate depending on temporal and spatial frequencies and also the object size. Although in here the proposed fuzzy interpolation is just applied to mesh–based motion estimation, but it can also be used in registration application using radial basis functions.

REFERENCES

[1] Y. Wang, J. Ostermann, Y. Q. Zhang, *Video Processing and Communications,* Prentice Hall, 2002.

[2] L.M. Po and W.C. Ma, "A novel four-step search algorithm for fast block motion estimation," IEEE Trans. Circuits Syst. Video Technol., vol. 6, pp. 313-317, June 1996.

[3] M. Eslami, E. Fatemizadeh, "Edge sensitive block motion estimation employing partial ridgelet distortion search," Proceedings of IEEE symp. on Signal Processing and Information Technology, pp. 902–907, 2006.

[4] Y. Wang and O. Lee, "Use of two-dimensional deformable mesh structures for video coding, part I - the synthesis problem: mesh-based function approximation and mapping," IEEE Trans. on Circuits and Systems for Video Technology, vol. 6, no. 6, pp. 636-646, Dec. 1996.

[5] Y. Wang and O. Lee, "Use of two-dimensional deformable mesh structures for video coding, part II-the analysis problem and a region-based coder employing an active mesh representation," IEEE Trans. on Circuits and Systems for Video Technology, vol. 6, no. 6, pp. 636-646, Dec. 1996.

[6] Y. Wang, J. Ostermann, "Comparison of block-based and mesh-based motion estimation algorithms," Proceedings of IEEE International Symposium on Circuits and Systems, vol. 2, pp. 1157-1160, June 1997.

[7] C. M. Kuo, M. S. Hung, "An effective mesh-based motion compensation technique for video coding," Elsevier Journal of Visual Communication and Image Representation, vol. 14, no. 4, pp. 405-427, Dec. 2003.

[8] Y. Altunbasak, A. M. Tekalp, "Closed-form connectivity-preserving solutions for motion compensation using 2-D meshes," IEEE Trans. on Image Processing, vol. 6, issue 9, pp. 1255-1269, Sept. 1997.

[9] Y. Nakaya and H. Harashima, "Motion compensation based on spatial transformations," IEEE Trans. Circuits Syst. Video Technol., vol. 4, pp. 339356, June 1994.

[10] C. Toklu, A. T. Erdem, M. I. Sezan, and A. M. Tekalp, "Tracking motion and intensity variations using hierarchical 2D mesh modeling for synthetic object transfiguration," Graph. Models Image Process., vol. 58, no. 6, pp. 553573, Nov. 1996.

[11] K. Rohr, "Image registration based on thin plate splines and local estimates of anisotropic landmark localization uncertainities," Proc. of International conf. on Medical Image Computing and Computer Assisted Intervention, pp. 1174-1183, Oct. 1998.

[12] C. Toklu, A. M. Tekalp, A. T. Erdem, M. I. Sezan, "2-D mesh-based tracking of deformable objects with occlusion," IEEE Proc. of International Conference on Image Processing, 1996, vol. 1, pp. 933-936, Sept. 1996.

[13] G. Al-Regib, Y. Altunbasak, R. M. Mersereau, "Hierarchical motion estimation with content-based meshes," IEEE Trans. on Circuits and Systems for Video Technology, vol. 13, issue 10, pp. 1000-1005, Oct. 2003.

[14] P. Hsu, K. J. Ray Liu and T. Chen "An adaptive interpolation scheme for 2-D mesh motion compensation," Proceedings of International Conference on Image Processing, vol. 3, pp. 646-649, Oct. 1997.

[15] H. J. Zimmermann *Fuzzy Set Theory,* Kluwer Academic Publishers, third edition, 1996.

[16] A. K. Jain, "Fundamentals of digital image processing," Prentice Hall Information and System Sciences Series, 1989.