PERGAMON

Pattern Recognition 35 (2002) 1303–1313

PATTERN RECOGNITION
THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY

www.elsevier.com/locate/patcog

Generalised correlation for multi-feature correspondence

C.V. Jawahar *, [1], P.J. Narayanan[1]

Centre for Artificial Intelligence and Robotics, Raj Bhavan Circle, High Grounds, Bangalore 560 001, India

Received 11 August 2000; accepted 3 May 2001

Abstract

Computing correspondences between pairs of images is fundamental to all structures from motion algorithms. Correlation is a popular method to estimate similarity between patches of images. In the standard formulation, the correlation function uses only one feature such as the gray level values of a small neighbourhood. Research has shown that different features—such as colour, edge strength, corners, texture measures—work better under different conditions. We propose a framework of *generalized correlation* that can compute a real valued similarity measure using a feature vector whose components can be dissimilar. The framework can combine the effects of different image features, such as multi-spectral features, edges, corners, texture measures, etc., into a single similarity measure in a flexible manner. Additionally, it can combine results of different window sizes used for correlation with proper weighting for each. Relative importances of the features can be estimated from the image itself for accurate correspondence. In this paper, we present the framework of generalised correlation, provide a few examples demonstrating its power, as well as discuss the implementation issues. © 2002 Published by Elsevier Science Ltd on behalf of Pattern Recognition Society.

Keywords: Stereo vision; Correspondence computation; Correlation; Feature integration

1. Introduction

A central problem in computer vision is the analysis and interpretation of 3D scenes. The high degree of success of the human vision system on this task and the 3D conception of space that developed as a consequence can perhaps be held responsible for the prominence of 3D reconstruction in computer vision research. There have also been attempts to reason about the 3D world using only the underlying geometry, without generating a 3D reconstruction of the scene. Projective geometry has played a major role in such efforts. Some researchers exploit the algebraic relationships between multiple images to solve the problems which were hitherto answered after 3D reconstruction [1].

There are a number of methods to reconstruct the 3D information from images. Shading, focus, interreflections, motion have all been exploited to extract the 3D shapes of objects in the images. Depth from multiple views is an important method, partly motivated by the human binocular vision system. Two views with known viewing geometries are sufficient to compute the 3D information of the visible surfaces. The depth information can be computed using triangulation once corresponding points are identified in the "left" and "right" images. Replicating most of the human vision capabilities with computational models is a difficult task. Humans seem to subjectively visualize the spectral variations and integrate the outputs from correlated as well as independent sources to visualize a scene. Though there are a few excellent articles in modeling human vision [2], most of the computer vision researchers attempt to tackle the

* Corresponding author.

E-mail addresses: jawahar@iiit.net (C.V. Jawahar), pjn@iiit.net (P.J. Narayanan).

[1] Authors are presently with the Indian Institute of Information Technology, Gachibowli, Hyderabad-500 019, India.

PII: S0031-3203(01)00111-X

problem independently. The geometry based methods that gained popularity recently attempt to build a less demanding reconstruction of the scene. Projective and affine reconstructions are frequently used. There are also strategies to progressively build projective, affine and finally Euclidean reconstructions of a scene from multiple—structured or unstructured—views of it [3,4]. The algebraic methods are closely related to these. They avoid explicitly reconstructing the scene. Instead, they try to answer the questions posed by the specific navigation or interpretation task using the algebraic relationships between multiple images of the scene.

A central requirement of all these methods is pixel correspondence or the identification of the corresponding points in multiple views. The results of the algebraic methods of reconstruction are sound if correspondences are accurate. However, correspondence computation tends to involve a lot of engineering optimizations. The lighting conditions, camera settings, discretization and quantization effects, signal noise are some of the factors that complicate precise correspondence computation. The precision of the computer vision task is bound by the precision of the correspondences, however. The feature points used to compute correspondences could be the individual pixels or derived features like edges and corners. Sparse and sharp correspondence maps result from using good feature points such as edges and corners. Dense maps can be obtained by computing correspondences for every pixel. Small patches of the image, instead of single pixels, usually result in better disambiguation, in either case.

The conventional stereo correspondence schemes employ intensity values to compute dense correspondences and edges or corners for sparse correspondences. The absolute and relative geometry of the gray level distribution is often more important in a matching process; hence intensity values or their derivatives alone may not suffice. Thus, secondary features derived from the image play an important role in improving correspondences.

The gray level or colour intensity value, texture measure, edge strength, etc., are some of the features that can be used at each pixel for matching. Each works well under different conditions. However, no attempt has been made to combine this heterogenous collection of features to produce better results than each individually can. We provide a framework, called the *generalised correlation*, for combining different types of features in a flexible way. The framework generalizes the traditional correlation function to compute correspondences based on a multidimensional feature vector. The components of this vector could be made of different features, including outputs of different spectra of multi-spectral images. The integration of the results of multiple methods in a flexible way provides superior accuracy in the correspondences computed. A preliminary version of this paper appeared in ACCV [5].

In the next section, we discuss the basics of a correlation based correspondence scheme. Notations are introduced there. Section 3 describes the generalised correlation functions and the facility to emphasise or de-emphasise a particular set of features. A number of examples are provided in Section 4 to demonstrate the power of the method. They include the correspondence computation of colour images and integration of correlation functions computed using variable size windows. Relative importance of features and their advantages in integrating multiple evidences are described in Section 5. In Section 6, various computational aspects of the proposed functions are described. Section 7 provides a few concluding remarks.

2. Stereo correspondence with similarity measures

Correspondence computation between a pair of images involves the identification of the same physical point in both the images. It is usually posed as the problem of finding the matching points in the second image of the selected points in the first image. An exhaustive search for such a point can be very expensive computationally. The epipolar constraint [6] restricts the search to a line for calibrated cameras. The task is further simplified by restricting the maximum possible value for the disparity for which an estimate could exist. We restrict our attention to a parallel rectified pair of cameras in this paper.

Major approaches for stereo correspondence are based on dynamic programming [7,8], relaxation [2,9], and correlation [10–12]. Often stereo matching methods extract edges, lines or corners in the left image as the feature points and try to identify corresponding points in the right image by minimizing a suitable objective function. Since dynamic programming is an efficient method of optimisation for functions with many discrete variables, stereo correspondence problem is well suited for dynamic programming. Relaxation based algorithms start with an initial match and identify the optimal one by iteratively modifying the matches based on some geometrical constraints. These algorithms are highly parallel in nature and are well suited to simulate the human stereo perception mechanism.

The process of matching involves identification of a similar pixel in the second image. Similarity is often measured in terms of photometric properties such as the gray level or colour values. Correlation is the most popular similarity measure. In this paper, the terms correlation and similarity are used interchangeably. We either minimize the distance between the template vector and the observation vector or maximize their normalized dot product to compute the match. Numerous examples are available in the literature that use correlation for correspondence computation and image matching [10–12]. For a small window $W^1_{pq}(i,j)$ of size $p \times q$ around a pixel (i,j) in left image, correspondence in right image is computed by moving a similar window $W_{pq}^{\mathrm{r}}(i+d,j)$ in the right image along the epipolar line. The location that optimises a similarity function or a distance function

$$\mathscr{S}(W_{pq}^{\mathrm{l}}(i,j),\ W_{pq}^{\mathrm{r}}(i+d,j)) \tag{1}$$

for all d, is the matching point of pixel (i,j). The function $\mathscr{S}$ is proportional to either $W^{\mathrm{l}} \cdot W^{\mathrm{r}}$ or to $-|W^{\mathrm{l}} - W^{\mathrm{r}}|$. A correlation function using the dot product is given in Eq. (2) and one using the differences is given in Eq. (3). We define a real valued correlation function $\mathscr{C}(F_{\mathrm{r}}, F_{\mathrm{l}}, p, q, d)$ (in short $\mathscr{C}(d)$) at a pixel (i,j) to depict the similarity of pixels in left and right images displaced by d pixels.

$$\mathscr{C}_1(d) = \frac{[F_{\mathrm{l}}^{pq}(i,j)]^{\mathrm{T}}[F_{\mathrm{r}}^{pq}(i+d,j)]}{||F_{\mathrm{l}}^{pq}(i,j)|| \times ||F_{\mathrm{r}}^{pq}(i+d,j)||}, \tag{2}$$

$$\mathscr{C}_2(d) = \frac{||F_{\mathrm{l}}^{pq}(i,j) - F_{\mathrm{r}}^{pq}(i+d,j)||}{||F_{\mathrm{l}}^{pq}(i,j)|| \times ||F_{\mathrm{r}}^{pq}(i+d,j)||}. \tag{3}$$

Here, $F_{\mathrm{l}}^{pq}(i,j)$ and $F_{\mathrm{r}}^{pq}(i,j)$ are vectors of dimension pq consisting of the gray level values of a $p \times q$ window around the pixel (i,j) of the left and right images, respectively and d is the disparity being searched for, and $||X||$ denotes the L_2 norm $\sqrt{[X]^{\mathrm{T}}[X]}$ of the vector X. The correlation function is computed for various values of $d \in [-d_{\mathrm{left}}, d_{\mathrm{right}}]$, and the disparity d corresponding to maximum of $\mathscr{C}_1$ is chosen as the matching point in the case of Eq. (2). The minimum of $\mathscr{C}_2$ defines the matching point in the case of Eq. (3).

These correlation functions can be normalized by subtracting the mean of each of the windows from each pixel value before matching, reducing the effects of differences in illumination or camera gains. The normalized version of Eq. (2) is given by Eq. (4).

$$\mathscr{C}_1^N(d) = \frac{[\tilde{F}_{\mathrm{l}}^{pq}(i,j)]^{\mathrm{T}}[\tilde{F}_{\mathrm{r}}^{pq}(i+d,j)]}{||\tilde{F}_{\mathrm{l}}^{pq}(i,j)|| \times ||\tilde{F}_{\mathrm{r}}^{pq}(i+d,j)||}, \tag{4}$$

where $\tilde{F}_{\mathrm{l}}^{pq}$ and $\tilde{F}_{\mathrm{r}}^{pq}$ are the normalized vectors with the means over the respective $p \times q$ windows, $\bar{f}_{\mathrm{l}}^{pq}$ and $\bar{f}_{\mathrm{r}}^{pq}$ respectively, subtracted from each pixel.

An important drawback of this formulation is its incapability to employ multiple features to compute the disparity map. The gray values may yield the best disparity map in some situations whereas the edge strengths may be best in other situations. A proper combination of the effects of gray level features, such as the gray value and texture measures, and the geometric features, such as edges, corners and lines, can provide better results in general [13–15]. In our opinion, a good matching algorithm should be able to select the best combination of features to generate an accurate correspondence map. The conventional correlation formulation is also not able to take advantage of multi-spectral images and requires modifications to do so.

Selecting an appropriate window size is crucial to a window-based similarity measure. Small window sizes increase the probability of mismatches, but yield better localization. Large window sizes reduce the mismatches, but at the cost of localization. It would be advantageous to integrate the strengths of both by combining large and small windows. The framework of *generalized correlation* we present next can not only combine different types of evidences, such as gray levels, edges and texture measures, but also integrate multi-spectral information as well as different sized windows to compute more accurate correspondences. We consider the features as a multidimensional vector, each dimension containing similar or dissimilar information. A generalized correlation function evaluates the similarity between two vectors, combining the effects of their components appropriately. Our formulation, in addition, has the flexibility to weigh the features according to their relative importance.

3. Stereo correspondence with multidimensional features

We can generalize the notion of the feature vector given above to include not only the gray level values, but other derived features like edge strength, texture measures, etc. Let $\bar{F}_{\mathrm{l}}(i,j) \in \mathbb{R}^p$ and $\bar{F}_{\mathrm{r}}(i,j) \in \mathbb{R}^p$ be two p-dimensional feature images, with $\bar{F}_{\mathrm{l}}^k(i,j)$ as the kth component of the feature vector of the pixel (i,j) in the left image. The dimension p of the vector is not directly linked to the window size used for matching. The term feature is used quite liberally. Different feature types can have different windows of support in the image. Some of the components may have values derived from photometric features and others from geometric features. The p-dimensional feature vector is formed for each pixel by stacking the values from the windows of support for each feature. It is possible to use correlation defined in the previous section over a 3 × 3 window as one feature and the same over a 5 × 5 window as another, for instance, providing the ability to combine multiple window sizes. We can modify the correlation function given in Eq. (2) as follows:

$$\mathscr{C}_1'(d) = \frac{[\bar{F}_{\mathrm{l}}(i,j)]^{\mathrm{T}}[\bar{F}_{\mathrm{r}}(i+d,j)]}{||\bar{F}_{\mathrm{l}}(i,j)|| \times ||\bar{F}_{\mathrm{r}}(i+d,j)||}. \tag{5}$$

A similar function based on generalized distance between feature vectors similar to the one given in Eq. (3) can be written as

$$\mathscr{C}_2'(d) = \frac{[\bar{F}_{\mathrm{l}}(i,j) - \bar{F}_{\mathrm{r}}(i+d,j)]^{\mathrm{T}}[\bar{F}_{\mathrm{l}}(i,j) - \bar{F}_{\mathrm{r}}(i+d,j)]}{||\bar{F}_{\mathrm{l}}(i,j)|| \times ||\bar{F}_{\mathrm{r}}(i+d,j)||}. \tag{6}$$

Each component of the feature vector can have different magnitudes and will need to be scaled appropriately. Flexibility in giving different weights to different feature components would be welcome to adapt the framework to different situations. To achieve this, we replace the popular L_2 norm with the following matrix norm

$$||X||_M = \sqrt{[X]^{\mathrm{T}}\boldsymbol{M}[X]}, \tag{7}$$

where, $\boldsymbol{M}$ is a $p \times p$ positive semidefinite matrix that encodes the relative importances as well as the cross relationships among the components of the feature vector. The Eqs. (5) and (6) can be rewritten as follows to express the *generalized correlation function* $\mathscr{GC}$.

$$\mathscr{GC}_1(d) = \frac{[\bar{F}_1(i,j)]^{\mathrm{T}}\boldsymbol{M}[\bar{F}_{\mathrm{r}}(i+d,j)]}{||\bar{F}_1(i,j)||_{\mathrm{M}} \times ||\bar{F}_{\mathrm{r}}(i+d,j)||_{\mathrm{M}}}, \tag{8}$$

$$\mathscr{GC}_2(d) = \frac{[\bar{F}_1(i,j)-\bar{F}_{\mathrm{r}}(i+d,j)]^{\mathrm{T}}\boldsymbol{M}[\bar{F}_1(i,j)-\bar{F}_{\mathrm{r}}(i+d,j)]}{||\bar{F}_1(i,j)||_{\mathrm{M}} \times ||\bar{F}_{\mathrm{r}}(i+d,j)||_{\mathrm{M}}}. \tag{9}$$

Here the *feature relation matrix* $\boldsymbol{M}$ encodes the relationships among the features as well as their relative importances. Though the feature vectors are multidimensional, the correlation function is real valued and the computation of its extrema can be carried out similar to the conventional correlation techniques by moving the correlation window with integer increments along the epipolar line. Normalized versions can also be devised similarly, but the normalization should be done by subtracting a mean value appropriate for each component of the feature vector.

It will be interesting to probe the importance of the feature relation matrix $\boldsymbol{M}$ for evaluating the correlation function. If $\boldsymbol{M} = [m_{ij}]$ is a diagonal matrix with diagonal elements m_{ii} encoding the relative weights of the ith feature, $\mathscr{GC}$ provides a weighted similarity measure. Clearly, if m_{ii} is zero, the influence of the ith feature component will be zero. The components of the feature vector can be combined in a flexible way by varying the elements in the feature relation matrix.

The elements m_{ij} need not be zero or unity. The total emphasis becomes a weighted combination of the features with values other than 0 and 1. If the matrix $\boldsymbol{M}$ is the inverse of the covariance matrix of the feature components, the generalized correlation function becomes conceptually similar to the Mahalanobis distance [16]. This would be very useful in the presence of features with widely different statistical properties. $\boldsymbol{M}$ need not be a diagonal matrix if correlation between feature components needs to be taken into account. In such cases, it will be desirable to know whether features are correlated or uncorrelated.

The feature relation matrix $\boldsymbol{M}$ need not be a precomputed constant matrix; it can depend on the images being matched. It may also be different on different parts of the image. Ideally, its elements should be estimated based on the local statistics of the image regions in terms of the relative strengths of the components of the feature vector. It should, thus, be possible to emphasize the edge based evidence when it is significant, but suppress it when not so. Estimating the relative importance of each component is similar to feature selection in pattern recognition. In most situations, one can learn or fine tune $\boldsymbol{M}$ offline and use it to compute accurate correspondences in similar situations. We outline a strategy for estimating $\boldsymbol{M}$ in Section 5.

The primary advantage of the generalized correlation framework is its ability to combine multiple types of features in a flexible way. Area-based features produce better matches when the patch being compared is large due to the presence of more information. However, larger patches produce fattening problem at occlusion boundaries due to the inability to localize them. Edge and corner based features localize the occlusion boundaries better, but do not produce dense enough correspondences unless the edges are present uniformly. The generalized correlation framework has the potential to combine the advantages of the multiple methods as is demonstrated by the examples given below.

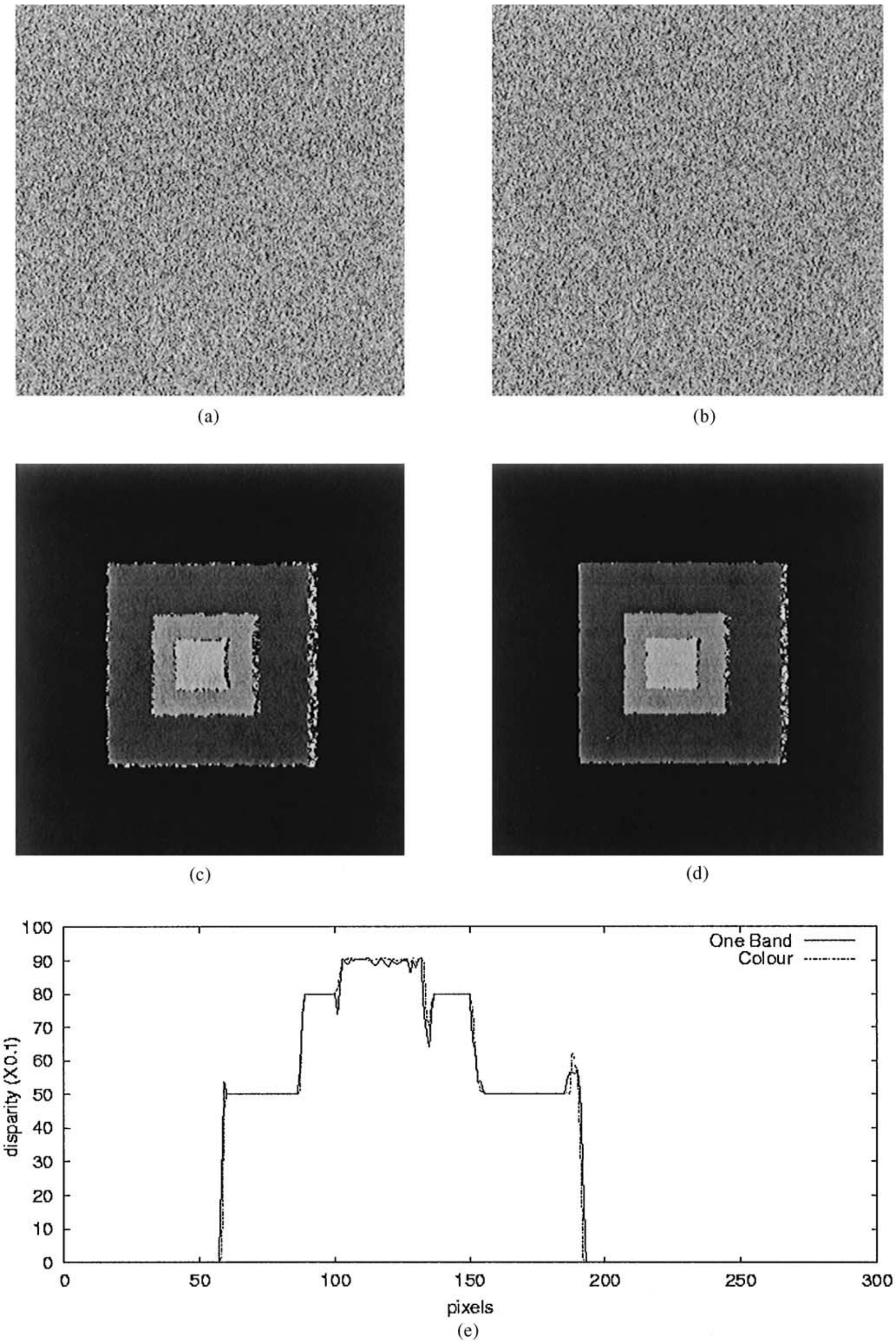
4. Examples

We present a set of examples in this section to demonstrate the effectiveness of generalized correlation to solve the correspondence problem.

4.1. Example 1: colour stereogram

Stereo algorithms are often verified on monochrome images. Though monochrome images are advisable for a number of real-time applications, there are enough situations where the colour image can yield better results. Here we consider a “random colour stereogram” for the validation of the methodology for stereo correspondence. The image pair has a three-layered wedding cake structure embedded in it. It was generated in a manner similar to the classical random dot stereogram but gray-values replaced by different colours in a consistent manner. The images are shown in Figs. 1a and b.

Each band of the colour stereograms is equivalent to a conventional random stereogram. We consider only one band initially. The computed disparity map with 3 × 3 window is shown in Fig. 1c. This result is same as the conventional correlation. Disparity levels are shown in different shades, with brighter colours used for higher disparities. Though the images are synthetic and free of noise, there are mismatches, in particular at the bound-


(a)

(b)

(c)

(d)

(e)

Fig. 1. Left and right images of a random colour stereogram along with the computed disparity maps and comparisons.

aries. Mismatches at the right end of the uniform disparity regions are due to occlusion. Occlusion occurs when a pixel is visible only in one image. But there are many more mismatches in the result. This is due to the multiple extrema of the correlation function. Additional geometrical constraints can possibly reduce the mismatches.

Next, we compute correspondences between the same image pair using generalized correlation with the feature vector comprising of the red, green and blue components of the 3 × 3 neighbourhood of each pixel. The computed disparity map is shown in Fig. 1d. Clearly, our strategy results in more crisp and sharp disparity map. The generalized correlation using multiple features works well by removing the ambiguity due to multiple extrema. For a quantitative comparison between the two disparity maps, we look at the mean disparity of all the scanlines containing the three disparity levels. They are plotted in Fig. 1e. A smooth and continuous disparity map is obtained using generalized correlation; discontinuities (edges) are also more sharp and precise.

Generalized correlation can take advantage of multi-spectral images, like colour and satellite images. A more challenging problem in correlation-based correspondence scheme is achieving localization and preserving the sharp discontinuities. If one employs a larger window, localization will be lost at the expense of reduction in number of mismatches. These errors will be prominent at the depth discontinuities. Therefore a good stereo correspondence algorithm based on correlation should be able to employ variable window sizes and incorporate the advantages of each of them [17].

4.2. *Example 2: variable size windows*

The generalized correlation formulation acts as an integrating mechanism for various stereo correspondence strategies. In this example, we demonstrate the performance of an integrating mechanism, which will merge the positive aspects of small and large window sizes (i.e., localization and reduction in mismatches).

In Figs. 2a and b, we show a pair of images with a wedding cake structure embedded on it. Unlike a synthetic random dot stereogram, the structure and the natural gray-distribution in the scene will lead to a large number of mismatches if one employs a small window. Matching results with a 1 × 1 window (pixel to pixel match) is shown in Fig. 2c. As the window size increases, the number of mismatches decreases, but with loss in sharpness of edges. Disparity maps for 3 × 3 and 5 × 5 windows using ordinary correlation are shown in Figs. 2d and e. The results of generalized correlation is shown in Fig. 2f. It clearly integrates the advantages of both small and large correlation windows. To investigate the performance further, we added a small additive noise to the right image to study the performance of the methodology in presence of distortions. Respective results for windows of size 1 × 1, 3 × 3, 5 × 5 and the proposed method are shown in Figs. 2g–j.

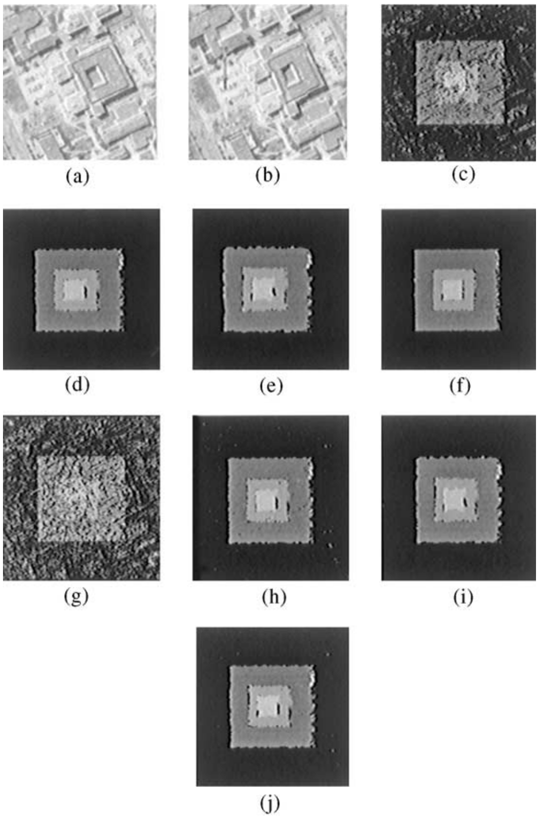


Fig. 2. Performance of the generalized correlation scheme with variable window sizes (for more details, refer e.g. 2).

In both the cases, we construct a feature vector comprising of elements from all the three (1 × 1, 3 × 3 and 5×5) windows. In this case, the dimensionality of the feature vector considered is 35 (i.e, 25+9+1). The feature vector is an ordered array of gray-values from these three windows. We have used the same weight for all pixels from a single window. This results in a diagonal feature relation matrix with exponentially decreasing weights as the window size increases to weigh them appropriately.

Gray level values are used widely for computing correspondences. Other features can be useful in other situations. It is advantageous to integrate multiple types of features. In the next example, we demonstrate how the integration of dissimilar features help correspondence computation.

4.3. *Example 3: multidimensional features*

We use the image of a natural scene (Fig. 3a) as the left image and generate a right image by first

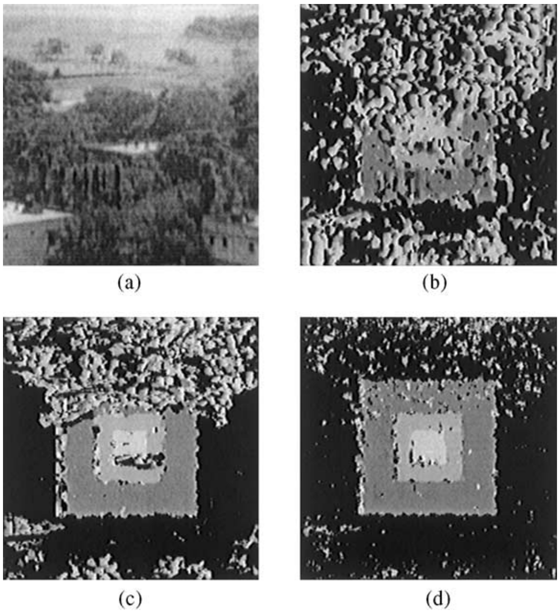


Fig. 3. (a) Natural texture used for the experimentation. Disparity maps computed with Gray value (b), Edge strength (b) and Texture measure (c).

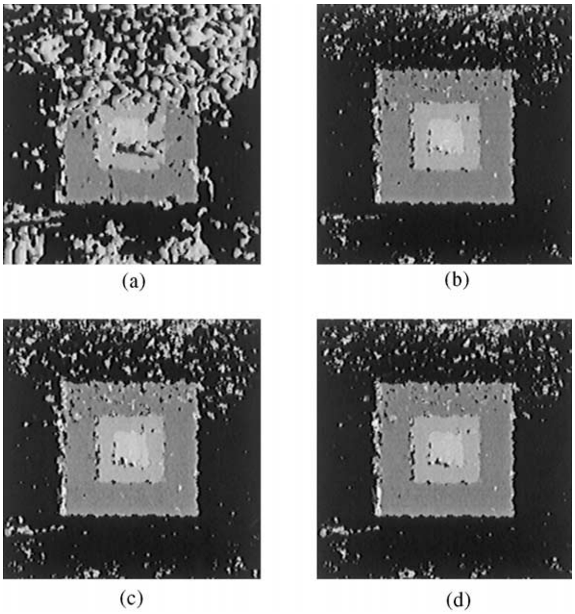


Fig. 4. Disparity maps computed with combinations of features. (a) Gray value and edge strength. (b) Gray value and texture measure (c) Edge strength and texture. (d) All the three together.

photometrically distorting it by equalizing its histogram and then imposing a wedding cake structure on it. Also an additive Gaussian noise (mean $= 0$, $\sigma = 3$) is added to the right image. The image regions which were occluded in the left image are filled randomly in the right image. Here we demonstrate that the features vary in their performance and their combination is the possible option in a generic environment.

Initially, we employ a conventional correlation using gray-value alone. Disparity map computed based on correlation of gray-values in a 5×5 neighbourhood is shown in Fig. 3b. It identifies proper matching pixels only at pixels wherever there is a reasonable variation in gray-values within a neighbourhood. This performs poorly at bottom and top half of the image.

We additionally tried edge information for correspondence computation. Edge strength given by

$$\sqrt{\left(\frac{\partial I(x,y)}{\partial x}\right)^2 + \left(\frac{\partial I(x,y)}{\partial y}\right)^2}$$

is considered as the feature. This resulted in a disparity map as shown in Fig. 3c. One may notice that the performance is poor wherever the edge information is less.

As a third feature, we tried a texture measure, which is invariant to the photometric distortions. If $a_1, a_2, \ldots, a_8$ are the neighbours of pixel a, then the texture unit number [18] corresponding to a is defined as

$$T_a = 3^i E(a_i, a),$$

where $E(a_i, a)$ is 0,1 or 2 according to the gray-value of a_i is less, equal or more than that of a. The disparity map computed using texture number is shown in Fig. 3d. Since the texture measure employed can take care of the photometric distortion, this performs better than the rest.

In the next phase of the experimentation, we try all combination of these features. Results are shown in Fig. 4. While integrating, the features were normalized to a common range of $[0, 255]$. The combinations are found to perform better than the individual features. A combination of gray value and edge strength integrated the advantages of both these features as can be seen in Fig. 4a. Combinations containing texture number performs better as can be seen in Fig. 4 (b) and (c). A combination of all the three (Fig. 4 (d)) combines the advantages of all the features.

Indeed, if one can decide on the importance of these individual features a priori, a better performance can be expected. In this experimentation, all the features were equally emphasized with a diagonal feature relation matrix. The issues related to the automatic emphasis/deemphasis is discussed in detail in the next section.

5. Estimating the feature relation matrix

Generalizing the feature vector to one of heterogeneous components and incorporating a feature relation matrix to encode the relative importances and mutual relationships between the different features has significant benefits. The estimation of the feature relation matrix $\boldsymbol{M}$ to be used in real situations, however, becomes crucial when this is done. The matrix need not be the same for the entire image either; it can change with the local properties of the image.

Any prior knowledge about the relative importances of the different features used and the cross relationships between pairs of features, if known, can be used to estimate $\boldsymbol{M}$. Otherwise, $\boldsymbol{M}$ can be computed from the local statistical properties of the images. The behaviour of the correlation function near the matching point can also be used to compute $\boldsymbol{M}$. It is often the case that $\boldsymbol{M}$ can be estimated off-line from selected example images and used later with the real images.

Estimation of the feature relation matrix $\boldsymbol{M}$ satisfying some optimality constraint and validation of the efficiency of the methodology is quite similar to the feature selection problem in pattern recognition. A detailed discussion on this topic is beyond the scope of this paper. The feature relevance estimation may be attempted in a supervised or completely unsupervised manner. If the ground truth is available, one can estimate the importance of individual features based on the relative performance in the computation of the precise matches. However, the availability of groud truth is not a valid assumption in many real-life situations. In such situations, one could formulate an algorithm which optimally estimates the importances of features while identifying the appropriate matches. The feature-relevance adaptation can be an iterative process in the case of dynamic stereo. An appropriate learning paradigm like neural network can be employed for this.

Here, we outline a simple strategy to heuristically evaluate the relative importances of features. For the sake of this discussion, two simplifying assumptions are made. (1) The different components of the feature vector are uncorrelated. Thus, $\boldsymbol{M}$ is a diagonal matrix. (2) The same feature relation matrix is used for the whole image. We include a few comments at the end of the section about the impact of relaxing these assumptions.

We can look at how the partial correlation function for each component of the feature vector behaves around the matching point. A strong peak in the partial correlation function indicates a good match; it also validates the use of the particular feature for matching. Thus, we should emphasize the entries of $\boldsymbol{M}$ corresponding to the features with strong peaks and de-emphasize the features with weak peaks. An implementation of the scheme is as follows:

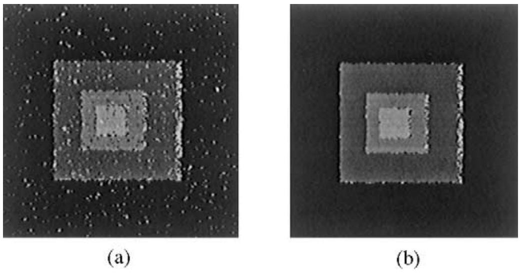


Fig. 5. Estimating $\boldsymbol{M}$ from the images. (a) The disparity map computed with equal weights. (b) Same using the estimated feature relation matrix.

1. Start with m_{f} set to 1 for all features.
2. Compute the generalized correlation $\mathcal{GC}$ function and the partial correlation functions $\mathcal{GC}_{\mathrm{f}}$ for each component f of the feature vector for a few good matches or for all points.
3. Estimate the sharpness of the partial correlation functions $\mathcal{GC}_{\mathrm{f}}$ for each f at the matching point. A simple measure of sharpness could be obtained by convolving the partial correlation function with the mask $[-1,-2,6,-2,-1]$. Let s_{f} be the average sharpness, over all points considered, of the partial correlation function for feature f.
4. Set m_{f} proportional to a sharpness measure of the partial correlation function at the matching point. Let $s_{\max}$ be the maximum average sharpness among all features. Set m_{f} to $s_{\mathrm{f}}/s_{\max}$ for each feature f. This de-emphasizes features with low peaks and emphasizes those with high peaks.

The last three steps can be repeated till the diagonal elements m_{f} are reasonably stable. The estimate of $\boldsymbol{M}$ given above will be tuned for the pair of images used in the computation.

5.1. Example 4: relative importance of features

We applied the above estimation procedure on the colour image pair used in the previous section. To simulate the noisy and unrelated features, we set random, uncorrelated values to one of the bands of an image. The disparity map computed with equal emphasis for all the features worsened, as expected. When we estimated $\boldsymbol{M}$ using the procedure given above, the weight corresponding to the noisy band was estimated to be 0.054 in the first step itself, while the other two bands retained high weights of 1.0 and 0.94. We used all the points in the scene for estimating $\boldsymbol{M}$. Note that the methodology does not assume any a priori knowledge of correct matches. Thus, this is a fully unsupervised process. Fig. 5 gives the results of the experiment. The disparity map computed using estimated weights is much better compared to the one computed with equal weightage to all three bands.

Another strategy to estimate $\boldsymbol{M}$ can be used if it is possible to guide the estimation process with good and bad match points. An iterative procedure can adjust the entries of $\boldsymbol{M}$ to optimize the number of mismatches, if the true disparity map is given. A gradient descent scheme may be used in such situations to estimate the entries of $\boldsymbol{M}$. Such a scheme can find applications in dynamic stereo where the characteristics of the scene remains consistent across frames. The feature relation matrix estimated from the first few frames can be used for the entire video sequence.

Though a constant $\boldsymbol{M}$ will suffice for most applications, there can be situations where different $\boldsymbol{M}$ matrices are used for different parts of the image. In this case, the entries of $\boldsymbol{M}$ can be estimated after a segmentation of the image. An initial segmentation of the image based on tone or texture properties can split the image into multiple, mutually exclusive regions. The points in each region contribute only to the $\boldsymbol{M}$ for that region. There could also be situations when non-diagonal $\boldsymbol{M}$ matrices are necessary. In such situations, estimating the elements of feature relation matrix is more involved, and may possibly be based on the analysis of inter-feature correlations. We intend to study this aspect in the future.

6. Computation of the generalized correlation function

Our formulation of the corresponding point has the following general form,

$$\max_{d} \frac{T_1}{T_2 \times T_3}. \tag{10}$$

Minimum function is used when differences are involved. The term T_2 is fixed for each source window and is independent of d. Being a positive constant it has no influence on the maximum and hence the match. The matching criterion using generalized correlation can then be written using the left image as the source as

$$\max_{d} \frac{T_1}{T_3} = \max_{d} \frac{[\bar{F}_{\mathrm{l}}(i,j)]^{\mathrm{T}}\boldsymbol{M}[\bar{F}_{\mathrm{r}}(i+d,j)]}{||\bar{F}_{\mathrm{r}}(i+d,j)||_M}. \tag{11}$$

The computation is similar to that of ordinary correlation. The feature vectors $\bar{F}_{\mathrm{l}}$ and $\bar{F}_{\mathrm{r}}$ can be assembled for each i,j and d and then the multiplication can proceed. In practice, however, the feature relation matrix will be a sparse matrix with zeroes making up for most entries. Full-fledged matrix multiplication need not be performed as a result. If $\boldsymbol{M}$ is a diagonal matrix with elements m_k,

$$T_1(i,j) = \sum_k m_k \bar{F}_{\mathrm{l}}^k(i,j)\bar{F}_{\mathrm{r}}^k(i+d,j). \tag{12}$$

This computation can be performed in a recursive manner. The components of the vector $\bar{F}_{\mathrm{l}}$ and $\bar{F}_{\mathrm{r}}$ are values from the pixels in the neighbourhood such as intensity, edge strength and texture measure. Hence a shift and accumulate strategy can also reduce the computation involved, as in the case of the computation of correlation functions. The techniques given in [10] to speed up the computations of correlation functions like $\mathscr{C}_1$ and $\mathscr{C}_2$ can be adapted easily to the computation of the generalized correlation function also. These methods on average require only a constant number—2 for normal correlation computation—of multiply and add operations per pixel to compute correlations, independent of the image size and the size of the feature vector. This is achieved by keeping partial sums for each row and column of the moving window and computing only the additional elements when the window is moved. A full correlation needs to be computed for the starting window, and a constant number of operations is necessary at the start of each row. The total algorithmic complexity for normal correlations using this method, therefore, is $O(p) + O(rc)$, where $p = w^2$ is the size of the feature vector, r is the number of rows of the image, and c is the number of columns of the image. Since p is a much smaller number than rc, the computation of correlation takes a constant amount of time per pixel.

We modified this strategy to compute the generalized correlation function for the case where $\boldsymbol{M}$ is a diagonal matrix. The final complexity again is $O(rc)$, with preprocessing that takes time $O(p)$, where p is the size of the feature vector. A pseudo code of the implementation for a diagonal feature relation matrix is described in Appendix A. The generalized correlation function is also amenable to parallel implementation in case speed of computation is the critical factor.

7. Discussions and conclusions

We presented the framework of generalized correlation that can combine multiple types of image features in a flexible manner to obtain more accurate matches between two images. Generalized correlation can thus integrate multiple techniques for correspondence computation, compensating for the weaknesses of each using the strengths of the others. We demonstrated the effectiveness of generalized correlation using a number of examples in this paper. Integration of different sorts of evidences is critical to compute the best possible correspondences in any situation. The framework we laid out provides an effective means to do that.

While our examples demonstrate the power, two areas remain to be explored, namely, the composition of the feature vector and the estimation of the feature relation matrix. Which features among gray level, colour, edge strength, texture number, etc., should be used in a particular situation? That should depend on the properties of the images themselves. We intend to develop a few guidelines for automatic selection of the components of the feature vector. The estimation of the feature relation matrix M used in the framework is also critical. We have given a preliminary procedure to estimate $\boldsymbol{M}$, but much more work needs to be done on this front, especially when the features used for correlation are correlated and $\boldsymbol{M}$ is not a diagonal matrix.

The matching point provides the best match under the circumstances. A confidence measure in the match, however, could be useful later. A fuzzy measure of confidence in the final correspondence output can be of help in decisions taken using the correspondence map. The relative weights of the components of the feature vector encodes the confidences in each implicitly. The value of the $\mathscr{GC}$ function can also serve as a confidence measure and can be propagated to the next level of processing. The fuzzy measures can be propagated till a crisp decision becomes necessary, but no information will be lost till such a decision is forced. The framework of generalized correlation can provide a much more reliable measure of confidence by combining multiple features in a flexible way.

8. Summary

Computing correspondences between pairs of images is fundamental to many computer vision and pattern recognition algorithms. Area based matching algorithms often employ correlation scores as the matching measure between pixels or regions. Most of these matching schemes employ only the gray-value for computing the correspondence. Performance can be improved by additionally employing secondary or derived features in addition to the intensity values. They could be based on colour, edge strength, corners, texture measures, etc. In this paper, we presented the framework of generalized correlation that can compute a real valued similarity measure using a multi-dimensional feature vector whose components can be heterogeneous.

Generalized correlation framework can combine the effects of different image features, such as multi-spectral features, edges, corners, texture measures, etc., into a single similarity measure in a flexible manner. Each of these features can be emphasized or deemphasized for specific image pairs. The same formulation can also combine results of different window sizes used for correlation with proper weighting of each. Relative importances of the features can be estimated from the image itself for better correspondence computation. Examples are provided to demonstrate the applicability of the generalized correlation under various situations. Features used in these case studies include, gray-value, edge strength, colours and texture number. We have also provided a heuristic method for computing the feature relation matrix from image to image.

Appendix A. Pseudo code

```
// Note: $F_r^f(i,j)$ is the $f$th feature of $\bar{F}_{\mathrm{r}}(i,j)$,
// correlation windows of feature $f$ are of
// size $(2h_{\mathrm{f}}+1)\times(2w_{\mathrm{f}}+1), m_{\mathrm{f}}$ is the weight of
// $f$th feature and is repeated
// $(2h_{\mathrm{f}}+1)(2w_{\mathrm{f}}+1)$ times on the principal diagonal
// of the feature relation matrix.
For $r \leftarrow$ startRow to EndRow
    // Because of parallel camera situation,
    // computations can be done row by row
    // First, compute all the partial products
    // ($pNR$ and $pDR$) required for this row.
    If $r$ equals startRow Then
        // For the startRow, there is no short cut.
        $\forall f, \forall c, \quad pDR_{\mathrm{f}c} \leftarrow \sum_{j=-h_{\mathrm{f}}}^{h_{\mathrm{f}}} m_{\mathrm{f}} F_{\mathrm{r}}^f(j+\mathrm{startRow}, c)$
                  $F_{\mathrm{r}}^f(j+\mathrm{startRow}, c)$
        $\forall f, \forall c, \forall d, \quad pNR_{\mathrm{f}cd} \leftarrow \sum_{j=-h_{\mathrm{f}}}^{h_{\mathrm{f}}} m_{\mathrm{f}}$
                  $F_{\mathrm{l}}^f(j+\mathrm{startRow}, c)$
                  $F_{\mathrm{r}}^f(j+\mathrm{startRow}, c+d)$
    Else
        // For other rows, new terms can be
        // computed from the previous
        // row's corresponding terms by adding a new
        // term and subtracting another term
        $\forall f, \forall c, \quad pDR_{\mathrm{f}c} \leftarrow pDR_{\mathrm{f}c} - m_{\mathrm{f}}(F_{\mathrm{r}}^f(r-h_{\mathrm{f}}-1, c)$
                  $F_{\mathrm{r}}^f(r-h_{\mathrm{f}}-1, c) + F_{\mathrm{r}}^f(r+h_{\mathrm{f}}, c) F_{\mathrm{r}}^f$
                    $(r+h_{\mathrm{f}}, c))$
        $\forall f, \forall c, \forall d, \quad pNR_{\mathrm{f}cd} \leftarrow pNR_{\mathrm{f}cd}$
                  $-m_{\mathrm{f}}(F_{\mathrm{l}}^f(r-h_{\mathrm{f}}-1, c)$
                  $F_{\mathrm{r}}^f(r-h_{\mathrm{f}}-1, c+d) + F_{\mathrm{r}}^f(r+h_{\mathrm{f}}, c)$
                  $F_{\mathrm{r}}^f(r+h_{\mathrm{f}}, c+d))$
    EndIf
    // Partial products required for computation of
    // Numerator and Denominator are ready
    // Compute the NR and DR so that
    // $\mathscr{GC}(d) = NR(d)/DR$ can be computed
    For $c \leftarrow$ startCol to EndCol
        // Numerator and Denominator can be
        // computed by
        // adding the partial terms $pNR$s and $pDR$s.
        If $c$ equals startCol Then
            // For the startCol, compute $NR$ and
              // $DR$ explicitly
            $DR(c) \leftarrow \sum_f \sum_{i=-w_{\mathrm{f}}}^{w_{\mathrm{f}}} pDR_{\mathrm{f}c}$
            $NR(c,d) \leftarrow \sum_f \sum_{i=-w_{\mathrm{f}}}^{w_{\mathrm{f}}} pNR_{\mathrm{f}cd}$
        Else
            // For the rest, compute them incrementally
            $DR(c) \leftarrow DR(c-1) - \sum_{\mathrm{f}} pDR_{\mathrm{f}(c-w_{\mathrm{f}}-1)}$
              $+ \sum_f pDR_{\mathrm{f}(c+w_{\mathrm{f}})}$
            $\forall d, \quad NR(c,d) \leftarrow NR(c-1,d)$
              $- \sum_{\mathrm{f}} pNR_{\mathrm{f}(c-w_{\mathrm{f}}-1)d}$
              $+ \sum_{\mathrm{f}} pNR_{\mathrm{f}(c+w_{\mathrm{f}})d}$
        EndIf
    EndFor
    // Now, find the (pixel) position of maximum
    // correlation for each pixel
    For $c \leftarrow$ startCol to EndCol
        maxm $\leftarrow -1.0, \text{disparity}(r,c) \leftarrow d_{\text{left}}$
        For $d \rightarrow d_{\text{left}}$ to $d_{\text{right}}$
            corrFn $\leftarrow \frac{NR(c+d,d)}{\sqrt{DR(c+d)}}$
            If maxm < corrFn then
                maxm $\leftarrow$ corrFn
                disparity$(r,c) \leftarrow$ d
            EndIf
        EndFor
    EndFor
EndFor
```

References

[1] A. Shashua, Algebraic functions for recognition, IEEE Trans. Pattern Anal. Mach. Intell. 17 (1995) 779–789.

[2] D. Marr, VISION: A computational investigation into the human representation and processing of visual information. W.H. Freeman and Company, San Francisco, USA, 1982.

[3] O. Faugeras, S. Laveau, L. Robert, G. Csurka, C. Zeller, 3-D reconstruction of urban scenes from sequence of images, Technical Report INRIA, 1995.

[4] M. Pollefeys, R. Koch, L.V. Gool, Self-calibration and metric reconstruction in spite of varying and unknown internal camera parameters, Proc. ICCV (1998) 90–95.

[5] C.V. Jawahar, P.J. Narayanan, Generalised correlation for stereo correspondence, Proc. ACCV (2000) 631–636.

[6] O. Faugeras, Three dimensional computer vision, MIT Press, USA, 1996.

[7] Y. Ohta, T. Kanade, Stereo by intra- and inter-scanline search, IEEE Trans. Pattern Anal. Mach. Intell. 7 (1985) 139–154.

[8] S.H. Lee, J.J. Leou, A dynamic programming approach to line segment matching, Pattern Recog. 27 (1994) 961–986.

[9] M.N. Nasrabadi, A stereo vision technique using curve segments and relaxation matching, IEEE Trans. Pattern Anal. Mach. Intell. 14 (1992) 566–572.

[10] O. Faugeras, B. Hotz, H. Mathieu, T. Vieville, Z. Zhang, Real time correlation-based stereo: algorithm, implementation and applications, Technical Report, INRIA, 1991.

[11] P. Fua, Combining stereo and monocular information to compute dense depth maps that preserve septh discontinuities, International Joint Conference on Artificial Intelligence, Sydney, Australia, 1991.

[12] T. Kanade, O. Okutomi, A stereo matching algorithm with an adaptive window: Theory and experiment, IEEE Trans. Pattern Anal. Mach. Intell. 16 (1994) 902–932.

[13] C.V. Jawahar, Stereo correspondence based on correlation of fuzzy texture measures, Proc. ICVGIP (1998) 273–278.

[14] D. Scharstein, Matching images by comparing their gradiant fields, Proceedings of the 12th IAPR (1994) 572–575.

[15] G.-Q. Wei, W. Brauer, G. Hirzinger, Intensity- and gradiant-based stereo matching using hierarchical gaussian basis functions, IEEE Trans. Pattern Anal. Mach. Intell. 20 (1998) 1143–1160.

[16] K. Fukunaga, Introduction to Statistical Pattern Recognition, Academic Press, New York, 1972.

[17] Y. Boykov, O. Veksler, R. Zabin, A variable window approach to early vision, IEEE Trans. Pattern Anal. Mach. Intell. 20 (1998) 1283–1294.

[18] L. Wang, D.C. He, Unsupervised textural classification of images using the texture spectrum, Pattern Recog. 25 (3) (1992) 247–255.

About the Author—C.V. JAWAHAR received Ph.D. from Indian Institute of Technology, Kharagpur in 1997. He was with Centre for Artificial Intelligence and Robotics, Bangalore, India, as a scientist till December 2000. Presently he is an Assistant Professor at the Indian Institute of Information Technology, Hyderabad, India. His areas of interest include computer vision, soft computing, image processing, multimedia systems and pattern recognition.

About the Author—P.J. NARAYANAN got his Ph.D. in Computer Science from the University of Maryland, College Park in 1992. From 1992 to 1996, he was a research faculty member at the Carnegie Mellon University and worked on the Virtualized Reality project. He was with the Centre for Artificial Intelligence and Robotics (CAIR) from 1996 to 2000. He is presently an Associate Professor at the Indian Institute of Information Technology. His research interests include Computer Vision, Computer Graphics, and Virtual Reality.