

COMPUTING THE VISIBLE INVARIANCE IN GREY SCALE IMAGERY ON THE TRANSPUTER

N A Chalabi and T S Durrani

Signal Processing Division
Department of Electronic and Electrical Engineering
University of Strathclyde
Glasgow, Scotland (UK)

ABSTRACT

In this paper we present a robust method for computing the visible invariance characteristics of 2D digitised grey-scale images. In essence visible invariance characterises features in the image plane, which are invariant to a specified set of transformations. The problem is formulated in variational terms and a least-squares technique is used to extract an optimum set of local features from the raw raster array of pixels. Appropriate mask operators for different window sizes to compute the image derivatives are given. These derivatives are then used for the computation of a new set of parameters, provided by classical differential geometry, the mean and Gaussian curvatures, which possess many invariant properties, and may be used to quantify the visible invariance. The method is parallel in nature and has successfully been implemented on the INMOS IMS T414 transputer running under an occam environment. An analysis of the parallel implementation based on convolution type operator is also presented.

1. INTRODUCTION

A crucial step that has become traditionally important in low-level machine vision consists of mapping the digitised gray-value image into an intermediate representation that could be used more efficiently by other higher level modules of the vision system hierarchy. In general, raw images contain much redundant information and can have a poor signal to noise ratio which may render them unsuitable as direct inputs to stereo-matching or motion estimation algorithms. Preprocessing is essentially a filtering process that aims to preserve certain intrinsic features of the original image and yields a new description which in general consists of significant intensity changes, discontinuities, local extrema, curvatures and corners of the image brightness function. In many applications as in object recognition, correspondence and image coding it is highly desirable that such low level image description is invariant to many transformations such as translation, rotation, scaling, skewing and parameterisation...

Several techniques have also been proposed in recent years for the automatic analysis of range imagery. Brady and Horn [1] developed a theoretical framework for surface representation. Besl and Jain[2] have also carried extensive investigations on local surface description for object recognition. Their study derives its roots from differential geometry. One interesting feature of their findings is that the mean and Gaussian curvatures are the fundamental second order surface characteristics that possess many desirable invariance properties and represent extrinsic and intrinsic surface geometry. They

also show that for a visible surface these 2 curvatures are invariant to surface parameterisation, rotation and translation of the object surfaces. Yang and Kak[3] use this concept successfully to determine the identity of the topmost object in a pile.

The question that naturally arises is whether one could apply such methods to gray-scale images. Range maps as obtained from active sensors (laser based, structured-light based...) are available as regular arrays of pixel values which quantify the depths from the viewer of the visible surface points. They constitute an implicit 3D surface description. This does not unfortunately apply to grey level imagery since the image irradiance results from the interaction of several factors, many of which (lighting, shadows, viewing angle...) are not properties of the surface in view. A common dilemma is that we cannot distinguish between a physical edge and a luminance edge if no additional cue is available. Thus the meaning attached to the slopes and curvatures in general of the intensity values is not related in a simple manner to the actual 3D structure of the visible parts of the object. However the invariance properties of the Gaussian and mean curvature may be exploited for feature labeling and object identification. The signs of these two curvatures can be combined to classify a surface patch into one of eight primitive types [2]. The image intensity function can be also segmented on this basis to discriminate between flat, hollow, peak and saddle portions...

2. THE CONCEPT OF VISIBLE INVARIANCE

Besl and Jain[1] define in broad terms an invariant feature as one which does not change under a specified group of transformations that do not affect its visibility. Thus visible invariance is more precise and takes into account the important fact that the imaging system may fail to record certain features of the objects when the viewing direction is modified or when objects occlude one another. This problem is well recognised by the stereo-vision and optical flow research community. To be more specific we shall express a formal mathematical statement to characterise the invariant visibility concept. Consider e_k a member of the set R_e of the original features in an image and these e_k can be the numerical values for the monochrome intensities or their contrast or even vectors with the 3 components for the R,G,B spectral intensities in the case of a colour camera. Let f_k be a member of the set R_f and T a specified composition of transformation T_l ($1 \leq l \leq m$) that maps each elements e_k into a unique element f_k

$$f_k = T_m \circ T_{m-1} \dots \circ T_1 (e_k)$$

The feature e_k is said to be invariant if and only if :

$$f_k = e_k$$

Thus ideally an invariant feature is unaffected by the idiosyncrasies of the individual transformations. In a practical situation one desires to compute features which have the same value regardless of their position, orientation or the coordinate system used in the image plane.

3. MEAN AND GAUSSIAN CURVATURES

Consider a generalised smooth surface (S) of class $C^r(R)$ which is continuous and r times differentiable over a region R in the 3D space. For our purpose we require $r \geq 2$. A Taylor series expansion about the target point (x_0, y_0) provides

a local description in terms of six independent parameters which quantify the surface shape in the neighbourhood of (x_0, y_0) , i.e. :

$$S(x_0+h, y_0+l) = S_0 + \nabla S \cdot \underline{X} + \frac{1}{2} \underline{X}^t H \underline{X} + O^3(h, l)$$

Where :

$$- S_0 = S(x_0, y_0)$$

is the surface value at the target point,

$$- \nabla S = (S_x, S_y)^t$$

is the local slope vector (gradient),

$$- H = \begin{vmatrix} S_{xx} & S_{xy} \\ S_{xy} & S_{yy} \end{vmatrix}$$

is the local curvature matrix (Hessian),

$$- \underline{X} = (h, l)^t$$

represents the coordinates of the point of interest relative to (x_0, y_0) ,

$$- O^3(h, l)$$

represents the higher order terms in the expansion and may be neglected in a small neighbourhood.

Note that the 2 principal curvatures of (S) at the target point are given by the eigenvalues of the Hessian matrix H . The Gaussian and the mean curvature on the other hand are given by the determinant and half the trace of the Hessian respectively.

i.e. :

$$\gamma_c = \det(H) = S_{xx} S_{yy} - S_{xy}^2$$

$$\mu_c = \frac{1}{2} \text{tr}(H) = \frac{S_{xx} + S_{yy}}{2}$$

The mean curvature averages the information of two measurements and hence appears more robust to noise. The computation of both the Gaussian and mean curvature relies upon the second order partial derivatives of the surface. Therefore such partial derivatives must be accurately determined. In the next section we propose a least-squares technique for extracting the derivatives of the image brightness function.

4. LOW-LEVEL FEATURES EXTRACTION

We focus our attention on discrete raster images whose intensity function $E(x, y)$ is defined over a finite $M \times N$ rectangular lattice of points :

$$L_{MN} = \{ (i, j) : 0 \leq i < M \text{ and } 0 \leq j < N \}$$

The numbers M and N designate the number of column and rows respectively. A square window of size $(2n+1)^2$ centered around the point (x, y) can be defined as :

$$W_n(x, y) = \{ E(x+i, y+j) : -n \leq i \leq n \text{ and } -n \leq j \leq n \}$$

If the function $E(x, y)$ varies smoothly then it can be locally approximated by a quadratic surface patch $F(x+i, y+j)$. The problem then amounts to fitting a second order polynomial surface over a small neighbourhood $\Omega_{x, y}$ defined over the

window region $W_n(x, y)$. The analytical expression for the surface patch is :

$$F(x+i, y+j) = F_0 + i F_x + j F_y + \frac{1}{2} F_{xx} +$$

$$\frac{1}{2} F_{yy} + i j F_{xy}$$

for $-n \leq i \leq n$ and $-n \leq j \leq n$.

The parameters $F_0, F_x, F_y, F_{xx}, F_{yy}, F_{xy}$ describe the actual value, slopes and curvatures of the sur-

face at the point (x, y) . In order to compute these coefficients one needs to minimise a norm which expresses the total discrepancy between the irradiance function and surface function fitted. The problem can be formulated as an optimisation procedure using a least-squares technique, as :

$$\text{Minimize : } \epsilon = \sum_{j=-n}^{+n} \sum_{i=-n}^{+n} [F(i, j) - E(i, j)]^2$$

which amounts to solving the system of equations provided by :

$$\frac{\partial \epsilon}{\partial F_0} = 0, \quad \frac{\partial \epsilon}{\partial F_x} = 0, \quad \frac{\partial \epsilon}{\partial F_y} = 0,$$

$$\frac{\partial \epsilon}{\partial F_{xy}} = 0, \quad \frac{\partial \epsilon}{\partial F_{xx}} = 0, \quad \frac{\partial \epsilon}{\partial F_{yy}} = 0$$

$$F_0 = \frac{5n(n+1)}{(2n-1)(2n+3)} (2A-B+C) + A$$

$$F_x = \frac{3}{n(n+1)(2n+1)^2} \sum_{i,j} i E(i,j)$$

$$F_y = \frac{3}{n(n+1)(2n+1)^2} \sum_{i,j} j E(i,j)$$

$$F_{xx} = \frac{30(B-A)}{(2n-1)(2n+3)}$$

$$F_{yy} = \frac{30(C-A)}{(2n-1)(2n+3)}$$

$$F_{xy} = \frac{9}{n^2(n+1)^2(2n+1)^2} \sum_{i,j} i j E(i,j)$$

$$A = \frac{1}{(2n+1)^2} \sum_{i,j} E(i,j)$$

$$B = \frac{3}{n(n+1)(2n+1)^2} \sum_{i,j} i^2 E(i,j)$$

$$C = \frac{3}{n(n+1)(2n+1)^2} \sum_{i,j} j^2 E(i,j)$$

TABLE-1 : Solution for general window.

The analytical solutions of the system for any window size are listed in table 1. The results for n=1 and n=2 are shown in tables 2 and 3 respectively. Such expressions can be developed to derive convolution mask operators. Fig-1 and Fig-2 illustrate the kernels obtained for n=1 and n=2.

Since the mean curvature is a linear sum of two convolution operators, it can also be computed by a convolution operator. In Fig-3 the 3x3 and 5x5 kernels for the mean curvature are given. The Gaussian curvature on the other hand is nonlinear and requires three separate convolutions for its evaluation.

$$F_0 = 5A-2B-2C$$

$$F_x = \frac{1}{6} \sum_{i,j} i E(i,j)$$

$$F_y = \frac{1}{6} \sum_{i,j} j E(i,j)$$

$$F_{xx} = 6(B-A)$$

$$F_{yy} = 6(C-A)$$

$$F_{xy} = \frac{1}{4} \sum_{i,j} i j E(i,j)$$

$$A = \frac{1}{9} \sum_{i,j} E(i,j)$$

$$B = \frac{1}{6} \sum_{i,j} i^2 E(i,j)$$

$$C = \frac{1}{6} \sum_{i,j} j^2 E(i,j)$$

TABLE-2 : Solution for 3x3 window (n=1).

$$\begin{array}{ccc|ccc|ccc} -1 & +2 & -1 & -1 & 0 & +1 & -1 & -1 & -1 \\ +2 & +5 & +2 & -1 & 0 & +1 & 0 & 0 & 0 \\ -1 & +2 & -1 & -1 & 0 & +1 & +1 & +1 & +1 \\ \hline & & & 9F_0 & & 6F_x & & & 6F_y \\ \hline +1 & 0 & -1 & +1 & -2 & +1 & +1 & +1 & +1 \\ 0 & 0 & 0 & +1 & -2 & +1 & -2 & -2 & -2 \\ -1 & 0 & +1 & +1 & -2 & +1 & +1 & +1 & +1 \\ \hline & & & 4F_{xy} & & 3F_{xx} & & & 3F_{yy} \end{array}$$

fig-1 : Mask operators for 3x3 window.

$$F_0 = \frac{1}{7} (27A-10B-10C)$$

$$F_x = \frac{1}{50} \sum_{i,j} i E(i,j)$$

$$F_y = \frac{1}{50} \sum_{i,j} j E(i,j)$$

$$F_{xx} = \frac{10}{7} (B-A)$$

$$F_{yy} = \frac{10}{7} (C-A)$$

$$F_{xy} = \frac{1}{100} \sum_{i,j} i j E(i,j)$$

$$A = \frac{1}{25} \sum_{i,j} E(i,j)$$

$$B = \frac{1}{50} \sum_{i,j} i^2 E(i,j)$$

$$C = \frac{1}{50} \sum_{i,j} j^2 E(i,j)$$

TABLE-3: Solution for 5x5 window (n=2).

$$\begin{array}{ccc|ccc|ccc|ccc} -13 & 2 & 7 & 2 & -13 & 4 & 2 & 0 & -2 & -4 \\ 2 & 17 & 22 & 17 & 2 & 2 & 1 & 0 & -1 & -2 \\ 7 & 22 & 27 & 22 & 7 & 0 & 0 & 0 & 0 & 0 \\ 2 & 17 & 22 & 17 & 2 & -2 & -1 & 0 & 1 & 2 \\ -13 & 2 & 7 & 2 & -13 & -4 & -2 & 0 & 2 & 4 \\ \hline & & & & & 25F_0 & & & & 100F_{xy} \\ \hline -2 & -1 & 0 & 1 & 2 & 2 & 2 & 2 & 2 & 2 \\ -2 & -1 & 0 & 1 & 2 & -1 & -1 & -1 & -1 & -1 \\ -2 & -1 & 0 & 1 & 2 & 0 & 0 & 0 & 0 & 0 \\ -2 & -1 & 0 & 1 & 2 & 1 & 1 & 1 & 1 & 1 \\ -2 & -1 & 0 & 1 & 2 & 2 & 2 & 2 & 2 & 2 \\ \hline & & & & & 50F_x & & & & 50F_y \\ \hline 2 & -1 & -2 & -1 & 2 & 2 & 2 & 2 & 2 & 2 \\ 2 & -1 & -2 & -1 & 2 & -1 & -1 & -1 & -1 & -1 \\ 2 & -1 & -2 & -1 & 2 & -2 & -2 & -2 & -2 & -2 \\ 2 & -1 & -2 & -1 & 2 & -1 & -1 & -1 & -1 & -1 \\ 2 & -1 & -2 & -1 & 2 & 2 & 2 & 2 & 2 & 2 \\ \hline & & & & & 50F_{xx} & & & & 50F_{yy} \end{array}$$

fig-2 : Mask operators for 5x5 window.



Fig-10

convolution kernel. For the computations of the invariant curvatures many multiplications can be avoided by noting some coefficient of the mask operators (Fig-1,2 and 3) are zero or one. Hence further code optimisation can be achieved.

6. RESULTS

In the preliminary stages the method described in this paper has been implemented on a VAX-8750 computer system linked via an ethernet remote server to a CRS 4000 frame store and a vidicon camera which captures real images with 8-bit resolution. The algorithms for calculating the mean and Gaussian curvatures as well as for segmenting the object surface on the basis of the sign of these two curvatures were implemented in the C language. It was observed that the time taken for the computation of the Gaussian curvature can become excessively long. For a 256x256 image and a 9x9 window, approximately 10 mns were required to produce the results. The mean curvature computation requires in general less processing time which is expected. More recently much of our efforts have been concentrated on the parallel simulation of the algorithms and their fast implementation on a target array of transputers hosted on an IBM PC-AT running DOS. The transputer development system version 2 (TDS2) supports occam 2 programming environment. We have therefore developed parallel versions of the C programs for calculating the invariant curvatures.

The algorithms were tested on many images with different window sizes. To illustrate our results two examples have been selected in this paper. The first is a coin image Fig-4, and is characterised by a relatively large amount of specular reflection inherent to the polished metallic surface. The coin has been rotated by 15° (Fig-5). To demonstrate the principle of the visible invariance the mean and Gaussian curvatures of the coin in both images were computed and these are shown in Figs-6,7,8 and 9 respectively. The rotated curvature images were then rotated back by 15° and subtracted from their

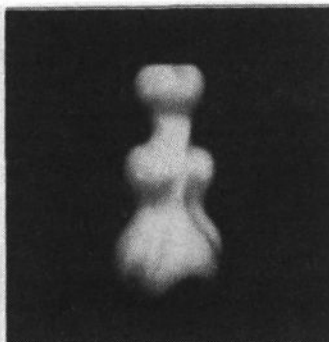


Fig-11



Fig-12



Fig-13



Fig-14

corresponding non-rotated versions. The two difference images obtained were then thresholded at 9.8% (for the Gaussian) and 7.1% (for the mean curvature) to produce a zero (black) image.

Fig-10 shows the result of the segmentation algorithm based on the sign of the invariant curvatures. Black regions are flat surface patches with $\mu_c = 0$ and $\gamma_c = 0$, white regions correspond to peaks of the intensity function where $\mu_c < 0$ and $\gamma_c > 0$ etc... See [2] for more details on surface types.

The second example is shown in Fig-11 which illustrates the image of a vase with smooth non-linear surface structure. The mean curvature was successively computed with different mask sizes (3x3, 5x5 and 7x7). The results are shown in Figs 12,13 and 14 respectively. A most noteworthy feature of these results is the clear improvement of the signal to noise ratio with increasing window size. The background fluctuations which are perceived in Fig-12 appear to have died out in Fig-14 while the object surface structure has become much in evidence. The Gaussian curvature of the vase has also been calculated and is shown in Fig-15 and then thresholded below 15%. The resulting image Fig-16 shows the outlines of the significant edges. Finally the results of the curvatures sign segmentation algorithm was performed on the vase image and gave the results shown in fig-17, which illustrates the object surface formed by many patches of differing surface types. A thresholding operation above level 10 thus removing planar segments produced Fig-18 which discriminates strongly the object against the background. Hence this method could be used for shape recognition.



Fig-15



Fig-16

7. CONCLUSION

We have presented a method for computing invariant features in grey scale imagery. It was shown that such feature detection schemes are based on convolution type operations. General formulae for the determination of the appropriate kernel in terms of the order of the window were given. Parallel implementations have been investigated to overcome the computational bottleneck limiting the capabilities of current sequential machines. The results carried on real images

have shown that the method lend itself to general purpose computer vision tasks such as edge detection, shape recognition, surface segmentation and classification...The invariance property can be exploited for more specific tasks such as the correspondence problem which is central to many stereo-matching and visual motion estimation algorithms. It was also shown how the SNR of the output images can be enhanced by using larger window operators.

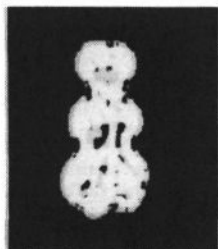


Fig-17



Fig-18

REFERENCES

- (1) M Brady and B K P Horn : Describing surfaces. Computer Vision Graphics and Image Processing 32,pl-28,1985.
- (2) P Besl and R Jain : Invariant surface characteristics for 3D object recognition in range images. Computer Vision Graphics and Image Processing 33,p33-80,1986.
- (3) H Yang and A Kak : Determination of the identity, position, and orientation of the topmost object in a pile. Computer Vision Graphics and Image Processing 36,p229-255,1986.
- (4) INMOS Limited : occam 2 Reference Manual, Prentice Hall International 1988.
- (5) J Kerridge : occam programming : a practical approach, Blackwell Scientific Publications.
- (6) INMOS : transputer IMS T414 data sheet, June 1987.