Shape Features

A Model Based Method for Characterization and Location of Curved Image Features

Thierry Blaszka and Rachid Deriche

INRIA Sophia-Antipolis, BP 93, 06902 Sophia-Antipolis Cedex, France

Abstract. This paper deals with the development of a parametric model based method to locate and characterize accurately important curved features such as ellipses and B-splines based curves. The method uses all the grey level information of the pixels contained within a window around the feature of interest and produces a complete parametric model that best approximates in a mean-square sense the observed grey level image intensities within the working area. Promising experimental results have been obtained on real data.

1 Introduction

This paper presents an approach which is a natural extension and generalization of the work presented in [3]. It deals with the localization and characterization of curved image features.

After this introduction, a first section is devoted to the modelization of the image features, then the next one will present the evaluation of the parameters of our models, the third section will be devoted to the experimental results and the perspectives and the applications of this work will conclude.

2 Characterization of Image Features

The linear models, defined in [3], are very useful for indoor scenes because of their polygonal environment. But for more general processes primitives not limited to lines are required, and to this end features delimited by curves will be considered.

The motivation is to have a complete characterization of curved features and to propose an approach that allows us to detect them with a sub-pixellic accuracy. Following the ideas used in the case of the linear features, the considered curve features are of the same global type: n regions with constant intensity defined by lines or by curve boundary in the working area. This type of attributes can be defined by the use of the function of Heaviside U. This function allows us to define features with sharp edges, but in the real images such attributes don't appear because of the blur introduced by the acquisition system. Then, a convolution operation with some smoothing kernel S is used to characterize this blur; these functions are defined as:

$$U(x) = \begin{cases} 1 \text{ if } x \geq 0 \\ 0 \text{ if else} \end{cases} \quad I \otimes S(x,y) = \int_{\mathbb{R}} \int_{\mathbb{R}} S(\alpha,\beta) I(x-\alpha, y-\beta) d\beta d\alpha \ . \quad (1)$$

The considered smoothing kernels are the Gaussian and the exponential filters introduced in [1]:

$$g(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{x^2}{2\sigma^2}} \; G(x,y) = g(x)g(y) \; e(x) = \frac{\alpha}{4}\left(\alpha|x|+1\right)e^{-\alpha|x|} \; E(x,y) = e(x)e(y) \; .$$

These filters lead us to define the models which will be denoted, in the rest of the paper, as Gaussian model and exponential model depending of the smoothing kernel used.

2.1 Ellipse Models

In our context, the simplest way to define ellipses, is to consider their analytical formulation: $N_l(x,y) = \frac{x^2}{a^2} + \frac{y^2}{b^2} - 1 = 0$, where a and b denotes the lengths of the ellipse axis. The combined use of this equation and the Heaviside function yields to the model of a sharp ellipse: $U(N_l(x,y))$.In order to consider more general ellipses, a frame change is done to take into account the orientation θ of its axis, and the position (x_0, y_0) of its center. Considering the new coordinates (x', y') and adding the grey-level intensities inside (A) and outside (B) the ellipse, the expression of the model becomes: $N_l'(x, y, x_0, y_0, \theta, a, b, A, B) = (A-B)\,U\,(N_l(x', y')) + B$. Convolving this model with one of the smoothing kernels (Gaussian or exponential), leads to the general model of ellipse, defined by eight parameters, $M_l(x, y, x_0, y_0, \theta, a, b, \sigma, A, B)$.

At this stage, one can note the difference between the approach of Lipson et al in [4] which first computes the ellipse parameters, and then evaluates the mean grey-level inside it, while our model intrinsically includes the radiometric (grey-level and blur) and geometric informations.

2.2 Closed B-Spline Models

In order to deal with a larger class of curved features, another type of curve model based on B-Spline closed curves is defined.

Within the large set of possible B-spline curves, only the subset of the smooth closed curves defined by their degree d and their control vertices of multiplicity one are used. The points $\mathbf{M}$ of such a B-Spline curve are defined as:

$$\mathbf{M}(t) = \sum_{i=1}^{n} \mathbf{V}_i \mathbf{B}_i^d(t) + \sum_{i=1}^{d} \mathbf{V}_i \mathbf{B}_{n+i}^d(t) \tag{2}$$

where t is the parameter varying along the curve, n the number of control vertices $\mathbf{V}_i$, and $\mathbf{B}_k^d$ the basis functions of degree d.

For our application, the selected curve has a fixed degree and a fixed number of control vertices, then its parameters are only the position of the control vertices. This leads to a model with $2n+3$ parameters: the coordinates of the control vertices, the grey-level intensities inside and outside the curve (A and B), and the blur coefficient (σ).

This representation of curve prohibits the use of the Heaviside function in its analytical form, and consequently a close form for the smoothed model can't be derived. To solve this problem, an algorithmical solution is used. First, a synthetic image of the curve is created and it is filled by the use of one classic algorithm. At this stage, the model of a B-Spline curve $N_b(V_1,\ldots,V_n,A,B)$ including the grey-level intensities, but without smoothing is defined. The next step is to smooth this image, using a discrete convolution. As for the previous features, two models of B-Spline curve are considered depending on the smoothing filter (f) used, Gaussian or exponential:

$$M_b^f(V_1,\ldots,V_n,\sigma,A,B) = N_b(V_1,\ldots,V_n,A,B) \otimes F(\sigma) \tag{3}$$

Due to the CPU time needed by a direct convolution operation, the recursive implementation of the Gaussian and the exponential smoothing described in [2] and [1] are used. These approaches lead us to reduce the computation time twice, at least, without any lost of precision.

3 Approximation of the Data

To characterize the features from the images, using the previous models, an iterative method called the model based approach is used. This method supposes to have a region of interest around the feature to characterize and a feature type selected. But an iterative method needs a first vector to initiate the process, and even if the method has been proven to be robust (see the experimental part), starting with a parameter vector far from the solution leads to a great CPU time. To tackle this problem, a fast method called *variance descent approach* has been developed. This method is designed to fastly produce a close initial parameter vector which is a rough solution to the minimization process of the model based approach.

3.1 Variance Descent Approach

This method is based on the remark that the considered curves define two iso-intensity regions in the working area. If the parameters of this feature are known and if there is no blur then the sum of the standard deviations within each region will be null. Therefore this method consist to define an energy criterion Σ which is the sum of the standard deviations of each region defined by the curve in the working area.

Case of Ellipses. Considering a first ellipse given inter-actively by the user or by a previous process, the energy Σ corresponding to this initial ellipse is evaluated. Then, each geometric parameter of the ellipse is moved from its initial position, keeping the others unchanged, and the energy Σ is computed. The parameter set corresponding to the minimal energy term computed is retained and the process iterates until the energy stops to decrease. The way the parameters are moved depends on the considered parameter: the center of the ellipse is moved in the eight directions corresponding to its eight neighbors, the axis lengths and orientation are increased and decreased separately. The initialization vector is composed of the founded geometric parameters, the grey-level intensities are set to the means of each region and the blur coefficient is set to 1.

Case of B-Spline Curves. the method used in this case is derived directly from the ellipse algorithm: the initial energy term is calculated; each control point is moved in the eight directions of its eight neighbors, while keeping the other control points invariants; the energy term corresponding to this new set of control points is calculated; the set of control points corresponding to the lower energy term is retained and the process iterates until the energy term stop to decrease.

As expected due to the number of control points of the curve, this direct method, denoted direct *vda* approach in the rest of the paper, is computationally very expensive. Then, a more efficient method in term of CPU have been developed: the following steepest gradient method, denoted gradient *vda* method. This approach corresponds to evaluate the initial energy, to compute the gradient of the energy function using finite difference, to find the best step in order to minimize the energy function in the gradient direction and to iterate until the process stop to decrease.

The initialization parameter vector is composed of the set of control points founded with one of the previous *vda* approaches and the grey-level intensities are set to the mean grey-levels of each region and the blur coefficient is set to 1.

3.2 Model Based Approach

The final step is now to start from the close initial conditions provided by the previous approaches and use the following method to evaluate the solution parameters with sub-pixellic precision while taking into account the blur introduced by the acquisition system. The refinement of the parameters is done by a numerical method which is intended to minimize the error function:

$$F(\mathbf{P}) = \frac{1}{m} \sum_{i=1}^{m} \left(M_a\left(i, \mathbf{P}\right) - I(i)\right)^2 \tag{4}$$

where M_a denotes the model of the considered feature, $I(i)$ the intensity of the pixel i, m the number of pixels of the working area, and $\mathbf{P}$ the vector of parameters of the considered feature model. The minimization of this function which is a sum of squares of non-linear functions is done by the routine *lmdif* of the *Minpack* library which implements the Levenberg-Marquardt algorithm.

4 Experimental Results

To test our models and the robustness of our method a lot of experimentations on noisy synthetic data and on real images have been done. But only results on real images are presented here.

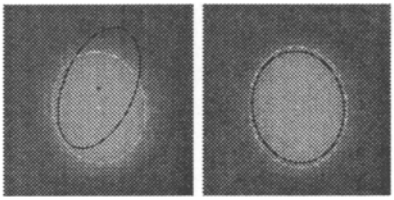

Fig. 1. Application of ellipse models.

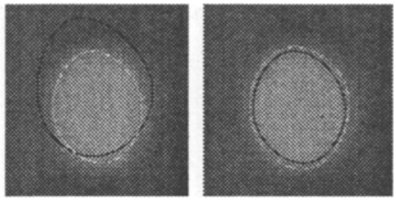

Fig. 2. Application of B-Spline models.

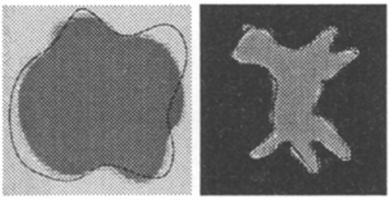

Fig. 3. The direct *vda* method.

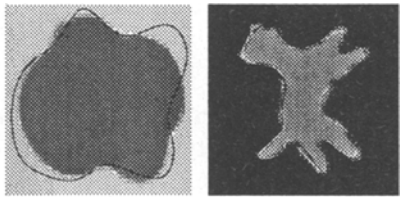

Fig. 4. The gradient *vda* method.

Figure 1 presents the application of the ellipse model. On the left image the manual initialization is drawn in black, and the convergence of the *vda* approach in white. In the right image lies the result of the model based approach in white initialized by the *vda* in black. Following the same scheme, Fig. 2 presents the application of the B-Spline curve model on the same image. One can note that the final results are the same in the two types of model, but the B-spline based approach is two-times faster.

In the case of B-Spline curves Fig. 3 and 4 show the results of the application of the direct *vda* method and of the gradient *vda* method respectively. The CPU time required for the first approach is about 200 seconds for the cloud image (left) and about 650 seconds for the dog image (right). But the gradient *vda* method takes roughly just 30 seconds. However, it is worthwhile to note that the initial conditions provided by these two approaches both leads to the same result when applying the model based approach.

The robustness of the method is illustrated by Fig. 5 where the initialization (in black in the left image) was given far from the solution. The result of the gradient *vda* method is shown in white of the left image and the result of the Gaussian model of B-spline initialized by the previous process corresponds in a satisfactory way to the solution (in white in the right image). This illustrates the fact that the model of B-Spline has a good convergence on images of smooth curves and the use of the *vda* allows to reduce drastically the CPU time of the convergence.

The application of our B-Spline models on real images is presented in Fig. 6. The black curves show the results of the *vda* approaches (Left direct *vda* and right gradient *vda*) and the white curves show the results of the application of the exponential model initialized by the black curves. In term of CPU time the direct *vda* approach is very long, up to ten times longer than the gradient *vda* approach. The CPU time required by the model based approach initialized by

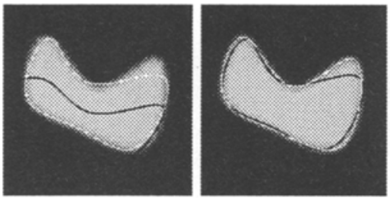

Fig. 5. Combination of the gradient *vda* and the Gaussian model of B-Spline.

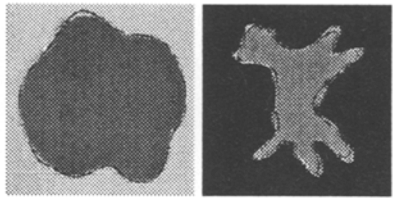

Fig. 6. The *vda* approaches and the Exponential model of B-Spline (see text).

the direct *vda* output was 250 sec on the image representing the dog, and 300 sec using the output of the gradient *vda* approach (right Fig. 6). On the image representing the cloud, the model based approach has converged in 140 sec with the output of the direct *vda* approach as initial condition (left Fig. 6) and in 150 sec using the output of the gradient *vda* approach.

The direct *vda* method allows to be slightly faster but globally the sum of the CPU times of the two steps is smaller in the case of the use of the gradient *vda* method. Due to the fact that the accuracy of the model based approach is the same in the two cases of initialization, one can consider the *vda* gradient approach as the good way to produce a close initialization.

5 Conclusion

An efficient model based method has been developed to locate and characterize precisely curved image features. Two different models have been developed to describe efficiently these features and a minimization process has been proposed to find the parameters that best approximate locally the observed grey level image intensities. Among the directions in which the approach presented in this paper can be extended, one can consider the generalization of the models to take into account non planar intensity regions, and the application of the estimation of the blurring parameter to the problem of recovering depth from focus.

References

1. R. Deriche. Fast Algorithms For Low-Level Vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(1):78–88, January 1990.
2. R. Deriche. Recursively Implementing the Gaussian and Its Derivatives. In *Proc. Second International Conference On Image Processing*, pages 263–267, Singapore, September 7-11 1992.
3. R. Deriche and T. Blaszka. Recovering and Characterizing Image Features Using An Efficient Model Based Approach. In *Computer Vision And Pattern Recognition*, pages 530–535, New-York, June 14-17 1993.
4. P. Lipson, A.L. Yuille, D. O'Keeffe, J. Cavanaugh, J. Taaffe, and D. Rosenthal. Deformable Templates for Feature Extraction from Medical Images. In O.D. Faugeras, editor, *First European Conference on Computer Vision*, pages 413–417, Antibes France, April 1990.