Unsupervised Detection and Semi-Automatic Extraction of Contours Using a Statistical Model and Dynamic Programming

François Destrempes and Max Mignotte
DIRO, Département d'Informatique et de Recherche Opérationnelle
Université de Montréal
C.P. 6128, Succ. Centre-Ville
Montréal, Canada (Quebec), H3C 3J7
email : destremp@iro.umontreal.ca

ABSTRACT

In this paper, we describe a statistical model for the gradient vector field of the gray level in images based on empirical results. Moreover, we present a global Markov model for contours in images that uses this statistical model for the likelihood. Our model is suitable for an ICE procedure for the estimation of the parameters and dynamic programming (or a simulated annealing) for the segmentation. This yields an unsupervised method for detection of contours. Also, we present a Markov model for paths in images based on the statistical distribution of the gradient. Our model can be used to define cost functions suitable for dynamic programming as in the intelligent scissors algorithm for semi-automatic extraction of contours.

1 Introduction

Unsupervised detection and semi-automatic extraction of contours are important problems in Image Processing.

One popular tool for solving the first problem is the Canny edge-detector or its variants (Deriche, Shen edge-detectors) based on the gradient of the gray levels at the pixels of the image. In [1], a Markov model with constraints for segmentation of an image into regions or contours has been presented. One tool for the second problem is the intelligent scissors algorithm [2] and [3]. The main inconvenient with these algorithms is the supervision in the specification of thresholds. Moreover, these algorithms are not directly based on the statistics of the image.

In [4], an advance has been made in proposing a statistical model for the norm of the gradient of the gray levels outside contours and for a related random variable on contours. However, we depart from the model presented in [4].

We present in this paper the statistical model [5] for the norm of the gradient of the gray levels for points off and on contours. We also introduce a new model for the angle between level curves and contour curves. Those models are based on empirical results.

We use an ICE procedure for the estimation of the parameters. The estimation procedure is based on a Markov model with constraint that takes into account the statistical distribution of the gradient vector field of the gray levels in the image. Comparison between histograms and estimated distributions suggests that our model is valid.

A preliminary step consists in pre-segmentation of the image into potential contours using directly an idea in the Canny edge-detector method. We obtain a subset T of the image that contains the contours of interest but also false contours which appear as noise. Based on the same model as for the estimation step, we can remove the noise from the set T by finding the global minimum of a corresponding energy function using the Viterbi algorithm. This yields a new unsupervised method for detection of contours in images.

For the semi-automatic extraction of contours, we introduce a Markov model for paths in images similar to [4] also based on the distribution laws of the gradient vector field of the gray level. We can then use dynamic programming exactly as in [2] and [3] though with different cost functions now motivated by the statistics of the image.

This paper is organized as follows. In Section 2, we explain the pre-segmentation step. In Section 3, we present the statistical model for contours. Section 4 presents in details the estimation procedure of the statistical distributions. In Section 5, we explain an unsupervised method for detection of contours, whereas in Section 6, we present a method for semi-automatic extraction of contours. In Section 7, we discuss briefly empirical results.

2 Pre-segmentation of the image

Given an image of size N, $G = (S, U)$ will denote the non-oriented graph consisting of the N pixels of the image together with the edges given by the usual 8-neighbors. If s and t are adjacent sites of G, we denote the edge joining s and t by (s,t) (so, $(s,t) \in U$). S' will denote the set of all sites of G together with the set of its edges ($S' = S \cup U$). We form a graph $G' = (S', U')$ by making two adjacent sites in G also adjacent in G', and by making (s,t) adjacent to its end points s and t.

We will need a subset T of the set of sites S obtained from a pre-segmentation step as in the Canny edge-detector. Namely, we first compute the gradient of the gray levels in the image. At each pixel s of the image, we approximate the direction of the gradient to the nearest direction $\hat{d}$ among the directions $d_1, ..., d_4$ corresponding to the angles $-\pi/4, 0, \pi/4, \pi/2$. If the norm of the gradient at s is smaller than one of its two neighbors along $\hat{d}$, then $s \notin T^+$; otherwise, $s \in T^+$. Next, we remove the isolated points of T^+ thus obtaining a set T. This simple procedure gives us a set of potential contour points from which we will extract contour points using our statistical model for the gradient distribution in the image.

Finally, we decompose T into a family $\mathcal{P} = \{p_\alpha\}$ of disjoint non-empty paths (in G) with the property that each path is maximal (no two distinct paths have adjacent end points). Such, a decomposition is not unique, but we fix one in all that follows.

We then define T' to be the set of all edges (s,t) appearing in the paths of $\mathcal{P}$.

3 Statistical Model for contours

For each site $s \in S$, the random variable Y_s represents the norm of the gradient of the gray level at the corresponding pixel.

Based on empirical results, we model $P_{Y_s}(y_s)$ for the sites off contours by a Weibull law [6],

$$\mathcal{W}(y_s; \min, C, \alpha) \;=\; \frac{C}{\alpha^C}(y_s - \min)^{C-1}\, e^{-\left(\frac{y_s - \min}{\alpha}\right)^C}$$

with $y_s > \min$.

For points on contours, we model $P_{Y_s}(y_s)$ by a mixture of two Gaussian laws,

$$\mathcal{M}(y_s; w_j, \mu_j, \sigma_j) \;=\; \sum_{j=1}^{2} w_j \mathcal{N}(y_s; \mu_j, \sigma_j).$$

If $(s,t) \in U$, we consider the distribution of the angle Y_{st} between the mean of the gradient at s and t with the normal to the vector from s to t. The angle is normalized between $-\pi/2$ and $\pi/2$. We have observed that $|Y_{st}|$ follows a Weibull law up to a factor

$$k_0 \mathcal{W}(y_{st}; \min_0, C_0, \alpha_0)$$

whenever the edge (s,t) belongs to a contour. We impose the condition that $C_0 \leq 1$ and the factor is taken so as to obtain a distribution in the interval $[0, \frac{\pi}{2}]$.

In the case where (s,t) is off contours, we model $|Y_{st}|$ by a uniform distribution on $[0, \frac{\pi}{2}]$,

$$\mathcal{U}(y_{st}; 0, \frac{\pi}{2}) \;=\; \frac{2}{\pi}.$$

4 Estimation of parameters

We consider a couple of random fields $Z = (X, Y)$ on the graph G', where $Y = \{Y_s, Y_{st}, s \in S, (s,t) \in U\}$ represents the field of observations, and $X = \{X_s, X_{st}, s \in S, (s,t) \in U\}$ the label field.

Here, the observable data is as in Section 3 and each X_s, X_{st} is in $\{e_1 = \text{"off"}, e_2 = \text{"on"}\}$. The distribution of (X, Y) is defined by a prior distribution $P_X(x)$ and by site-wise likelihoods $P_{Y_s/X_s}(y_s/x_s)$, $P_{Y_{st}/X_{st}}(y_{st}/x_{st})$ whose parameter Φ depends on the class label x_s, x_{st}. We assume independence between each random variable Y_s given X_s and Y_{st} given X_{st}. The observable Y is called the "incomplete data", and Z the "complete data".

In order to obtain a reliable estimation of the parameters, we resort to the ICE algorithm. This procedure, described in detail in [7], is outlined as follows :

Parameter initialization : we use the parameters estimated on an initial segmentation obtained by the K-means algorithm described in [8] for $\Phi^{[0]}$. Then $\Phi^{[p+1]}$ is computed from $\Phi^{[p]}$ in the following way :

1. Stochastic step : using the Gibbs sampler, one realization x is simulated according to the posterior distribution $P_{X/Y}(x/y)$, with parameter vector $\Phi^{[p]}$.

2. Estimation step : the parameter vector $\Phi^{[p+1]}$ is estimated with the ML estimator of the "complete data" corresponding to each class.

3. If $\Phi^{[p+1]} \not\approx \Phi^{[p]}$, for some $k = 1, 2$, return to step 1.

We now explain the estimation procedure in details.

Posterior distribution

The posterior distribution is modeled according to a global Bayesian formulation using the independence hypothesis

$$P_{X/Y}(x/y) \;\propto\; \prod_{s \in S} P_{Y_s/X_s}(y_s/x_s) \prod_{(s,t) \in U} P_{Y_{st}/X_{st}}(y_{st}/x_{st})\; P_X(x).$$

For the likelihood, we use the distributions presented in Section 3

$$\begin{aligned}
P_{Y_s/X_s}(y_s/e_1) &= \mathcal{W}(y_s; \min, C, \alpha) \\
P_{Y_s/X_s}(y_s/e_2) &= \mathcal{M}(y_s; w_j, \mu_j, \sigma_j) \\
P_{Y_{st}/X_{st}}(y_{st}/e_1) &= \mathcal{U}(y_{st}; 0, \frac{\pi}{2}) \\
P_{Y_{st}/X_{st}}(y_{st}/e_2) &= k_0 \mathcal{W}(y_{st}; \min_0, C_0, \alpha_0).
\end{aligned}$$

For the local *a priori* model, we adopt the simple model corresponding to the function $\sum_{<s,t>}(1 - \delta(x_s, x_t))$ in order to favor a homogeneous segmentation, where (s,t) ranges over T'. We also impose the following constraints : $x_s = e_1$ for all $s \notin T$; $x_{st} = e_1$ whenever $(s,t) \notin T'$; $x_s = e_2$ and $x_t = e_2$ if and only if $x_{st} = e_2$. Given this *a priori* model with constraint, the prior distribution $P_X(x)$

can be written as,

$$P_X(x) = \frac{1}{Z}\exp\left\{-\sum_{<s,t>:(s,t)\in T'}\left(1-\delta(x_s,x_t)\right)\right\} \prod_{s\notin T}\delta(x_s,e_1)\prod_{(s,t)\notin T'}\delta(x_{st},e_1) \prod_{(s,t)\in T'}\eta(x_s,x_t,x_{st})$$

where summation is taken over all edges of T', $\delta(.)$ is the Kronecker delta function, Z is a normalizing constant, and $\eta(x_s,x_t,x_{st})=1$ if $x_s=x_t=x_{st}=e_2$, or $x_{st}=e_1$ and $x_s=e_1$ or $x_t=e_1$, and $\eta(x_s,x_t,x_{st})=0$ otherwise.

Parameter initialization

We initialize the ICE procedure by the parameters estimated on the complete data obtained by a K-means clustering segmentation of T into two classes based on the norm of the gradient of the gray levels. The class with smallest mean is labeled e_1.

Stochastic step

The energy corresponding to the posterior distribution is the Gibbs field

$$\begin{aligned} U(x,y) = & \sum_{s\in S} -\ln P_{Y_s/X_s}(y_s/x_s) \\ & +\sum_{(s,t)\in U} -\ln P_{Y_{st}/X_{st}}(y_{st}/x_{st}) \\ & +\sum_{<s,t>:(s,t)\in T'}\Big(1-\delta(x_s,x_t)\Big) \end{aligned}$$

with the constraint that $V(x)$ is minimal, where

$$\begin{aligned} V(x) = & \sum_{s\notin T}\Big(1-\delta(x_s,e_1)\Big) \\ & +\sum_{(s,t)\notin T'}\Big(1-\delta(x_{st},e_1)\Big) \\ & +\sum_{(s,t)\in T'}\Big(1-\eta(x_s,x_t,x_{st})\Big). \end{aligned}$$

The constrained stochastic relaxation has been developed in this context [9] and could be used in the simulation of X. However, for computational reasons, we impose the constraints directly by setting $x_s=e_1$ whenever $s\notin T$, $x_{st}=e_1$ whenever $(s,t)\notin T'$, and $x_{st}=e_2$ if and only if $x_s=e_2$ and $x_t=e_2$. Thus, we simply use the Gibbs sampler algorithm [10] on $T\cup T'$. In doing so, we visit the sites of $T\cup T'$ by groups of three : a site $s\in S$ and its two adjacent edges (s,t), (r,s). There are two possible labels for s and each one determines the labels of (s,t), (r,s) (since the labels of the other sites of S are fixed). For each possibility, we consider $p(x_s)$ given by

$$P_{Y_{rs}/X_{rs}}(y_{rs}/x_{rs})P_{Y_s/X_s}(y_s/x_s)P_{Y_{st}/X_{st}}(y_{st}/x_{st}) \exp(-2+\delta(x_r,x_s)+\delta(x_s,x_t))$$

with x_{rs} et x_{st} adjusted so that $\eta(x_r,x_s,x_{rs})=1$ and $\eta(x_s,x_t,x_{st})=1$. We then choose x_s to be e_1 with probability $p(e_1)/(p(e_1)+p(e_2))$.

Estimation step

Assuming that the "complete data" is known, the parameters of the statistical distribution associated to each class can then be computed with the Maximum Likelihood (ML) estimator on each class $k=1,2$, thanks to good asymptotic properties.

In principle, one should take $\Phi^{[p+1]}$ as the conditional expectation of $\hat{\Phi}$ given $Y=y$, computed according to the current value $\Phi^{[p]}$. It is the best approximation of Φ in terms of the mean squares error [7]. The computation of this expectation is impossible in practice, but we can approach it thanks to the law of large numbers by,

$$\Phi^{[p+1]} = \frac{1}{n}\,[\hat{\Phi}(x_{(1)},y)+\cdots+\hat{\Phi}(x_{(n)},y)],$$

where $x_{(i)},\ i=1,\ldots,n$ are realizations of X drawn according to the posterior distribution $P_{X/Y}(x/y,\Phi^{[p]})$. In order to decrease the computational load, we can take $n=1$ without altering the quality of the estimation [11].

We use the following estimators for the distribution laws introduced in Section 3.

• If $Y=(Y_1,\ldots,Y_M)$ are M random variables i.i.d. according to a "single" Weibull law $\mathcal{W}(y;\min,C,\alpha)$, and $y=(y_1,\ldots,y_M)$ is a realization of Y, the ML estimator of $C_{\text{ML}},\alpha_{\text{ML}}$ is given by [12],

$$\begin{aligned} \hat{C}_{\text{ML}} &= F(\hat{C}_{\text{ML}}), \\ \hat{\alpha}_{\text{ML}} &= \left(\frac{1}{M}\sum_{i=1}^{M}\tilde{y}_i^{\hat{C}_{\text{ML}}}\right)^{\frac{1}{\hat{C}_{\text{ML}}}}, \end{aligned}$$

where $\tilde{y}=(y-\min)$ and

$$F(x) = \frac{M\sum_{i=1}^{M}\tilde{y}_i^x}{M\sum_{i=1}^{M}(\tilde{y}_i^x\ln\tilde{y}_i)-\sum_{i=1}^{M}\ln\tilde{y}_i^x\sum_{i=1}^{M}\tilde{y}_i^x}.$$

Here, we simply take $\min=-10^{-5}\nu$, where ν is the maximum of the norm of the gradient in the given image. We use an iterative method [12] to find $\hat{C}_{ML}$.

• The ML estimator of the parameters of a mixture of Gaussian distributions cannot be computed directly. We use the SEM algorithm [13]. This algorithm is a stochastic version of the EM algorithm [14] and is identical to the ICE algorithm except for the prior distribution which is replaced by the proportion of each class. We use the ML estimator on the result given by a K-means clustering segmentation to obtain an initial solution for the parameters of this mixture.

• The ML estimator for a truncated Weibull law cannot be found directly. So, we estimate the Weibull law for the interval $(\min_0,\infty)$ and then we adjust it by its integral from 0 to $\frac{\pi}{2}$ computed with the numerical method Simpson $1/3$. We take $\min_0=-10^{-3}$. If $C_0>1$, we set $C_0=1$ and re-estimate α_0 and k_0. This last case happens with small images when there is over-fitting.

5 Unsupervised detection of contours

Based on the estimates given by the ICE procedure, we can compute an unsupervised Markov segmentation of the image into contours. In this framework, the Markov segmentation can be viewed as a statistical labeling problem according to a global Bayesian formulation in which the posterior distribution $P_{X/Y}(x/y) \propto \exp -U(x,y)$ has to be maximized [15]. The corresponding posterior energy is given in Section 4 and is subject to the constraint that $V(x)$ is minimal.

We could use the simulated annealing algorithm with constraint [9] to minimize this global energy function with the above constraint.

However, we can find directly an optimal solution using the Viterbi algorithm on each path p of the set T obtained by the pre-segmentation step. If p is listed as a sequence $s_0, s_1, ..., s_n$ of adjacent pixels, and $e_{i_0}, e_{i_1}, ..., e_{i_n}$ is the corresponding segmentation, we consider the energy function

$$l_p(i_0, ..., i_n) = -\ln P_{Y_{s_0}/X_{s_0}}(y_{s_0}/e_{i_0}) + \sum_{t=1}^{n} \lambda_t(i_{t-1}, i_t)$$

where $\lambda_t(i,j)$ is defined by

$$\begin{array}{l} 1 - \delta(e_i, e_j) - \ln P_{Y_{s_t}/X_{s_t}}(y_{s_t}/e_j) \\ - \ln P_{Y_{s_{t-1},s_t}/X_{s_{t-1},s_t}}(y_{s_{t-1},s_t}/e_{i,j}) \end{array}$$

and $e_{i,j}$ is adjusted so that $\eta(e_i, e_j, e_{i,j}) = 1$. Then, the above constrained minimizing problem is equivalent to the minimizing of the functions l_p independently on each path. By applying the Viterbi algorithm [16], on each path with the above cost functions, we obtain an optimal segmentation of T.

6 Semi-automatic extraction of contours

Recall that e_1 = "off" and e_2 = "on". Given a path $c = (s_0, ..., s_n)$ in the graph G of Section 3, we consider similarly to [4] and [17] the likelihood $P_{Y/C}(y/c)$ given by

$$\begin{array}{ll} & \prod_{s \notin c} P_{Y_s/X_s}(y_s/e_1) \prod_{s \in c} P_{Y_s/X_s}(y_s/e_2) \\ & \prod_{(s,t) \notin c} P_{Y_{st}/X_{st}}(y_{st}/e_1) \prod_{(s,t) \in c} P_{Y_{st}/X_{st}}(y_{st}/e_2) \\ = & k(y) \prod_{s \in c} \frac{P_{Y_s/X_s}(y_s/e_2)}{P_{Y_s/X_s}(y_s/e_1)} \prod_{(s,t) \in c} \frac{P_{Y_{st}/X_{st}}(y_{st}/e_2)}{P_{Y_{st}/X_{st}}(y_{st}/e_1)} \\ \propto & \prod_{i=0}^{n} \frac{\mathcal{M}(y_{s_i}; w_j, \mu_j, \sigma_j)}{\mathcal{W}(y_{s_i}; \min, C, \alpha)} \prod_{i=1}^{n} \frac{k_0 \mathcal{W}(y_{s_{i-1},s_i}; \min_0, C_0, \alpha_0)}{\mathcal{U}(y_{s_{i-1},s_i}; 0, \frac{\pi}{2})} \end{array}$$

where

$$k(y) = \prod_{s \in S} P_{Y_s/X_s}(y_s/e_1) \prod_{(s,t) \in U} P_{Y_{st}/X_{st}}(y_{st}/e_1).$$

Moreover, we consider an *a priori* model that imposes a restriction on the length of the path from the starting point to the end point. Namely, we consider

$$P_C(c) = \frac{1}{Z} \exp(-\rho\,|c|)$$

where $|c|$ is the number of edges in the path c and Z is a normalizing factor.

This yields the posterior probability

$$\begin{array}{rcl} P_{C/Y}(c/y) & \propto & P_{Y/C}(y/c) P_C(c) \\ & \propto & \exp(-l(c,y)) \end{array}$$

where the corresponding posterior energy function (up to a constant) is given by

$$l(c,y) = l(s_1, ..., s_n) = \sum_{i=1}^{n} \lambda(s_{i-1}, s_i),$$

with $\lambda(s,t)$ defined by

$$\begin{array}{l} \rho + \log(\mathcal{W}(y_s; \min, C, \alpha)) - \log(\mathcal{M}(y_s; w_j, \mu_j, \sigma_j)) \\ + \log(\mathcal{U}(y_{st}; 0, \frac{\pi}{2})) - \log(k_0 \mathcal{W}(y_{st}; \min_0, C_0, \alpha_0)). \end{array}$$

We choose ρ minimal so that $\lambda(s,t) \geq 0$. Therefore, we can use dynamic programming [18] in order to find the path with lowest cost $l(c,y)$ from a specified pixel s_0 to any given pixel t. Such a procedure has been presented in [2] and [3] though with *ad hoc* cost functions. One important property of our model is the in-variance of the likelihood under affine transformations of the gray levels.

7 Experimental Results

In our implementation of the estimation procedure, we first re-calibrate the gray levels of the image between 0 and 255, and then apply a 3×3 Gaussian mask. Afterward, we multiply the gradient by $100/\nu$, where ν is the maximum of the norm of the gradient in the image. Thus, $\min = -10^{-3}$. We fix the number of iterations to 10 for the ICE procedure, to 100 for each application of the SEM algorithm and to at most 30 for each estimation of the Weibull law. The entire procedure takes about 55 seconds on a PC workstation 400MHz for the image presented here. The comparison between the empirical density functions and the estimated distributions suggests that the statistical model presented here is valid. See Fig. 2 and 3 for the estimated values of the parameters and the distributions.

We present the images obtained by the detection method in one instance (cf. Fig. 4). One can observe that the noise and false contours due the pre-segmentation step are completely removed by the segmentation into contours.

In our implementation of the semi-automatic extraction of contours, we have sometimes to freeze a portion of the contour manually, but this is also the case with the intelligent scissors algorithm. Our method seems to work as well as the original algorithm, if not better. Thus, it helps validate our statistical model for contours. See Fig. 1.

8 Conclusion

In this paper, we have described a statistical model for the gradient vector field of the gray level in images based on empirical results. Moreover, we have presented a global Markov model for contours in images that uses this statistical model for the likelihood. Our model is suitable for an ICE procedure for the estimation of the parameters and dynamic programming (or a simulated annealing) for the segmentation. This yields an unsupervised method for the detection of contours. Finally, we have presented a Markov model for paths in images based on the statistical distribution of the gradient. Our model can be used to define cost functions suitable for dynamic programming as in the intelligent scissors algorithm for the semi-automatic extraction of contours.

Références

[1] D. Geman, S. Geman, C. Graffigne, and P. Dong. Boundary detection by constrained optimixation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 12(7) :609–628, July 1990.

[2] E. N. Mortensen, B. S. Morse, W. A. Barrett, and J. K. Udupa. Adaptative boundary detection using live-wire two-dimensional dynamic programming. In *IEEE Proceedings of Computers in Cardiology*, pages 635–638, October 1992.

[3] E. N. Mortensen and W. A. Barrett. Interactive segmentation with intelligent scissors. *GMIP*, 60(5) :349–384, September 1998.

[4] M. Pérez, A. Blake, and M. Gangnet. Jetstream : Probabilistic contour extraction with particles. In *Int. Conf. on Computer Vision, ICCV' 2001*, Vancouver, Canada, July 2001.

[5] F. Destrempes and M. Mignotte. Unsupervised detection of contours using a statistical model. In *ICIP' 2002*, accepted.

[6] N. S. Matloff. *Probability Modeling and Computer Simulation*. PWS-KENT Publishing Company, 1988.

[7] F. Salzenstein and W. Pieczynski. Parameter estimation in hidden fuzzy markov random fields and image segmentation. *Graphical Models and Image Processing*, 59(4) :205–220, 1997.

[8] S. Banks. *Signal processing, image processing and pattern recognition*. Prentice Hall, 1990.

[9] D. Geman and S. Geman. Relaxation and annealing with constraints. *Division Appl. Math., Brown Univ., Complex Systems Tech. Rep. 35*, 1987.

[10] S. Geman and D. Geman. Stochastic relaxation, Gibbs distributions and the Bayesian restoration of images. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 6(6) :721–741, 1984.

[11] B. Braathen, P. Masson, and W. Pieczynski. Global and local methods of unsupervised Bayesian segmentation of images. *GRAPHICS and VISION*, 2(1) :39–52, 1993.

[12] M. Mignotte, C. Collet, P. Pérez, and P. Bouthemy. Three-class markovian segmentation of high resolution sonar images. *Computer Vision and Image Understanding*, 76(3) :191–204, December 1999.

[13] G. Celeux and J. Diebolt. L'algorithme SEM : un algorithme d'apprentissage probabiliste pour la reconnaissance de mélange de densités. *Revue de statistiques appliquées*, 34(2) :35–52, 1986.

[14] A.P. Dempster, N.M. Laird, and D.B. Rubin. Maximum likelihood from incomplete data via the EM algorithm. *Royal Statistical Society*, pages 1–38, 1976.

[15] J. Besag. On the statistical analysis of dirty pictures. *Journal of the Royal Statistical Society*, B-48 :259–302, 1986.

[16] G. D. Forney Jr. The Viterbi algorithm. *Proc. IEEE*, 61(3) :263–278, March 1973.

[17] D. Geman and B. Jedynak. An active testing model for tracking roads in satellite images. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 18(1) :1–14, 1996.

[18] E. W. Dijkstra. A note on two problems in connexion with graphs. *Numerische Mathematik*, 1 :269–270, 1959.

FIG. 1. *Semi-automatic extraction of contours using dynamic programming and based on the parameters estimated by the ICE procedure.* $\rho = 6.1195$.

min	C	α
-0.001	0.637668	3.59364
w_1	μ_1	σ_1
0.804729	52.4051	14.867
w_2	μ_2	σ_2
0.195271	15.867	6.745
$\min_0$	C_0	α_0
-0.001	1	0.388712
k_0	ν	
1.02051	107.366	

FIG. 2. *Values of the parameters estimated by the ICE procedure.*

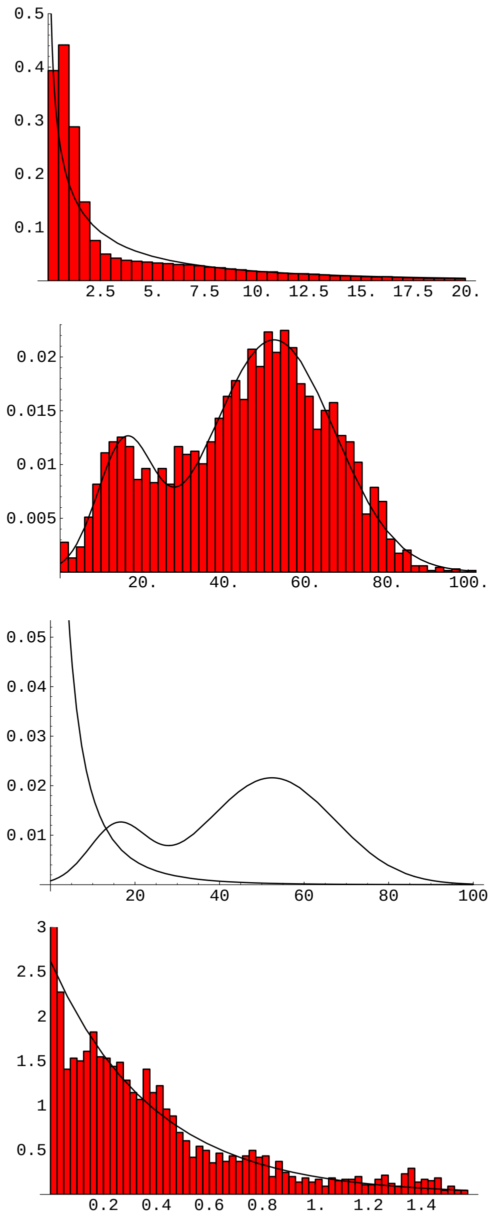

FIG. 3. *Example of the empirical density functions of the gradient of the gray levels and the distributions estimated by the ICE procedure. From top to bottom : norm of the gradient off contours ; norm of the gradient on contours ; comparison of the two previous distributions ; angle between the gradient and the normal to the curve on contours.*

FIG. 4. *Example of an unsupervised detection of contours of an image using dynamic programming and based on the parameters estimated by the ICE procedure. Top : original image. Center : pre-segmentation. Bottom : contours detected in the image.*