ROBUST OBJECT BOUNDARY DETERMINATION USING A LOCALLY ADAPTIVE LEVEL SET ALGORITHM

Eftychios Sifakis[*], *Student Member IEEE*

Department of Computer Science
Stanford University
Stanford, CA 94305-9025, USA
e-mail: *sifakis@cs.stanford.edu*

Georgios Tziritas, Senior Member IEEE

Computer Science Department
University of Crete
P.O.Box 2208, Heraklion, GREECE
e-mail: *tziritas@csd.uoc.gr*

ABSTRACT

This paper introduces a level set methodology for the precise boundary localization of image objects within an indicated region, designed to be particularly robust against weak or spurious edges, triple points or inhomogeneity of object features in the proximity of the actual interface. The proposed technique requires a reliable classification for a subset of the object interiors, which is propagated towards the unclassified space using a competitive, statistically motivated fast marching region growing algorithm. Color and texture features are used on a locally adaptive, dynamically updated fashion to allow for the robust discrimination of inhomogeneous objects and an efficient implementation. Applications are illustrated in the context of moving object localization and semi-automatic object extraction.

1. INTRODUCTION

Object boundary determination or, equivalently, segmentation into regions is a fundamental image processing task. Although it is one of the oldest problems in image processing, it still remains an open issue, because of its difficulty and complexity. Furthermore, in the emerging new standards of multimedia content description (MPEG-4 [1] and MPEG-7 [2]) image/video object extraction is an imperative step. The object localization may be implemented interactively [3] or automatically [4, 5].

A rough object boundary is often extracted in some early stage, for example in video segmentation, where an object might be at first localized using change detection or motion segmentation. In other cases the process of accurate object localization could be interactively initialized and the boundary approximately determined.

For the final boundary determination methods based on energy minimization could be used [6]. Since these methods are gradient-based, they cannot accurately handle textured images, junctions and other complex situations. In [7] we have also introduced a level set algorithm searching for local maxima of image gradient features. This approach also suffers from inherent weaknesses of gradient descriptors.

Region merging or region growing methods appear to be more adapted to such complex situations. Seeded region growing algorithms [8] could be used, as for example in interactive object extraction [3], where color similarity is used for advancing two contours in opposite directions until they meet each other on the region boundary.

[*] Supported under a SAP AG Stanford Graduate Fellowship

In this paper we introduce a new algorithm based on Bayesian level set methods and multi-region fast marching algorithms [7]. Each class is densely described using combined color and texture features. The region description is locally adaptive and sufficiently detailed in order to handle many complex situations.

2. LEVEL SET ALGORITHMS

2.1. Fast marching algorithms for multiple interfaces

The well known fast marching algorithm [9, 10, 11] provides a constructive finite differences solution to the Eikonal-like stationary level set equation

$$F|\nabla T| = 1 \tag{1}$$

governing the propagation of a monotonically advancing contour in R^N under a normal velocity field F, given a specific level set of the arrival time function T (commonly the zero level set) as initial input. The algorithmic complexity is $O(n \log n)$ over the number of pixels swept by the algorithm.

The multi-region fast marching algorithm [7] expands upon the basic formulation by allowing for the simultaneous propagation of several evolving contours, possibly with independently defined velocity fields, competitively expanding against each other and halting their evolution at points of contact.

An arbitrary number of propagating contours can be handled while the execution time of $O(n \log n)$ is preserved regardless of the number of propagation classes present, through parallel simulation of the evolution processes and dynamic limiting of the effective range for each contour class. Simultaneous evolution also guarantees topological stability and prevents penetration of one expanding region into another, which would be possible should the propagation processes had been handled independently.

An inherent limitation of the fast marching algorithm also applicable to the multi-region extension is the difficulty of incorporating local geometric curve properties, such as normal direction and curvature in the velocity definition. This is usually offset by the additional benefits of an increased stability resulting from the competitive evolution and a straightforward selection of the final segmentation result without the need for an explicit propagation instance selection.

2.2. Bayesian formulation and competitive region growing

Several definitions of the velocity function F of equation (1) are to be found in the literature for the purposes of different segmentation applications, most notably the edge-detecting velocity field $F(\boldsymbol{x}) = 1/(1+|\nabla I(\boldsymbol{x})|^N)$ designed to slow down the evolution of the deforming contour in the proximity of object boundaries. Several problems are to be expected should the original fast marching algorithm be used with such a velocity field in the presence of edges with considerably different strengths since the propagation is likely to penetrate the object boundaries at different time instances in several parts of its outline, an issue particularly problematic in the case of coexistence of strong, sharp edges with smooth transitions or stealth edges coincidentally covered up by lighting conditions or similar coloring of the background.

The multi-region extension has been illustrated [7] to exhibit significantly increased robustness in such cases, due to the simultaneous expansion of competing contours, typically corresponding to other objects to be segmented, counterbalancing the protrusion of the evolving interface through weak edges with the convergence onto a competing contour, in the fashion of other region-growing algorithms. Nevertheless, the shortcomings of a simple gradient-based velocity definition are often beyond any remedy in the presence of texture or noise in the interior of the objects to be discriminated. A velocity definition designed specifically to address such issues is based on the statistical modeling of the regions to be segmented and motivates the evolution of the deforming contours on the *a posteriori* probability of the intended classification. Using Bayes' rule this can be rewritten as a function of the individual region distributions

$$F_l(\boldsymbol{x}) = Pr\{\boldsymbol{x} \in U_l | o(\boldsymbol{x})\} = \frac{Pr\{U_l\} p(o(\boldsymbol{x})|U_l)}{\sum_k Pr\{U_k\} p(o(\boldsymbol{x})|U_k)} \quad (2)$$

where F_l is the expansion velocity for the boundary of region U_l and $o(\boldsymbol{x})$ is the actual feature-dependent observation at the image site $\boldsymbol{x}$. Evidently, this formulation is particularly suited to the multi-region level set framework described in subsection 2.1 with an independent velocity definition for each class.

The generality of this formulation illustrated by equation (2) allows for the utilization of the described framework in every application where a reliable modeling of the image features is feasible and an initialization for the region-growing process, specifically a confidently classified subset of the image regions, is obtainable through statistical analysis or user interaction. Documented applications include unsupervised static segmentation on color and texture features [7], change detection and moving object localization/tracking [4] and motion field segmentation.

3. ALGORITHM DESCRIPTION

We introduce a new algorithmic framework aiming to broaden the range of applications of the multi-region fast marching algorithms to a class of more delicate problems. As mentioned in subsection 2.2 the applicability of the Bayesian framework is highly dependent on the feasibility of a single statistical model consistently describing each of the regions to be segmented. This is not always the case when the object to be extracted exhibits substantial inhomogeneity due to its consisting of several regions of distinct content. Furthermore, in several cases the sub-object components of each region might exhibit some particular common feature, such as temporal variability or independent motion in video segmentation, which greatly facilitates the initialization process of selecting a reliably classified image subset.

The key observation is that on a local scale a consistent model is almost always feasible, provided that enough data are present to guarantee a robust model estimation, a limitation having obvious repercussions on the minimum size and shape regularity of the objects to be extracted. In particular this method is aimed towards the extraction of sufficiently large objects (commonly at least 10-15 pixels wide) without extremely thin and narrow components. Several descriptors are used for a single region with the applicable at each occasion being selected through criteria of geometric proximity. The rest of this section provides a description of the utilized features, the statistical models adopted and the core level set algorithm for the region-growing process.

3.1. Feature definition

Edge features often have limited usability when an object to be segmented is highly textured or when there is no consistent edge strength throughout the extent of the image boundary. Moreover, in the case of an object consisting of distinct components their boundaries commonly form triple points near the outline of the whole object which are often poorly handled by edge-based segmentation techniques.

The proposed approach uses a combination of color and texture features in order to provide a robust feature descriptor for the proper manipulation of such situations. Color features are derived from the *CIE Lab* color space, selected for its near-linear perceptual behavior. Color intensity of an image region is modeled through the empirical distribution of intensity values. Chromaticity features are independently modeled as well using the histogram of (a, b) pairs in the considered region. For the purposes of efficient discretization, both intensity and chromaticity values are quantized prior to further processing. The likelihood of the observed color at a given pixel is used directly in equation (2).

Texture features are derived from the Discrete Wavelet Frames analysis [12] implementing an iterative frequency band bisection without subsampling the original image. In the case of optimal filters for the frequency band decomposition the wavelet components can be shown to be zero mean and uncorrelated. For the current implementation the fourth order binominal lowpass and its conjugate highpass filter are used for the iterative creation of the wavelet frame range. The lowest frequency component is discarded while the higher frequency components are assumed to follow a generalized zero-mean Gaussian distribution for the individual frames, combined under the hypothesis of independence. It should be noted that for modeling an entire, only piecewise-homogeneous region a standard Gaussian distribution is often tractable by virtue of the central limit theorem. Nevertheless, experimental data illustrated the suitability of a Laplacian distribution for features derived on a very local basis, which is the model used in the current implementation. The *a priori* probabilities are assumed all equal in lack of other evidence. Likelihood values due to color and texture are combined additively under the hypothesis of independence.

3.2. Dynamic local features

In order to properly handle objects consisting of unlike components, feature description for a given region is performed in a locally adaptive fashion. Each region boundary pixel on the initial user-supplied or statistically derived classification constitutes a local node utilized in the above process. The node used to provide the distribution parameters required in equation (2) is selected for each pair of image site and region as the node belonging to the same region having the minimal geometric distance from the image site in question. The feature descriptor for each node is obtained from statistics derived in a window centered at the node location masked by the pixels already classified into the node's host region (figure 1). Additionally, the local node features are dynamically updated in the process of the propagation as new pixels are classified against the region owning the specific node, in effect causing a change in the mask used to derive the node statistics.

3.3. Implementation

The algorithm consists of a preprocessing stage and a propagation/update stage

```
Preprocess() {
  BuildFeaturePrimitives();
  BuildNodes(); {
  CalcClosestPoint();
  BuildNodeFeatures();
  BuildUpdateLists(); }

PropagateAndUpdate()
  while (UnclassifiedExists) {
  pxl = PropagateOnePoint();
  Update(ClosestNode(pxl)); }
```

The preprocessing stage includes the preparation of the raw features, such as colorspace quantization and the Discrete Wavelet Frames analysis. Subsequently, the feature nodes are selected on the boundary of the initially classified regions of each propagation class. The closest node of each region to each of the initially unclassified points is calculated afterwards, an operation performed offline to allow for optimized computation. Finally, the initial node features are calculated using their local square neighborhoods masked by the initial classifications. All points in the range of each node which were initially masked out are put in separate update lists arranged per image pixel, in order to facilitate the update process during the propagation. Each pixel is associated with a list of pointers to nodes whose values should be updated when the specific region's contour sweeps through the particular point.

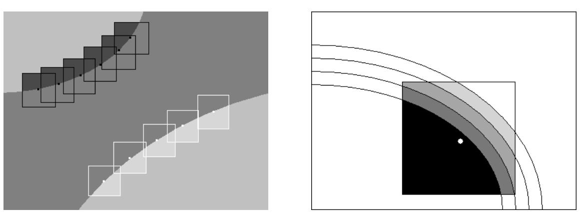

Fig. 1. Local feature descriptors densely placed on the initial boundaries and dynamic update of the effective feature data

The propagation stage is the regular fast marching loop, augmented by an update operation each time a new classification is carried out or, in fast marching terminology, when a narrow band pixel is turned into an alive pixel. It should be noted that this update for the feature descriptor used is no more than a single bin increment in the node intensity/chromaticity histograms, a simple addition of the square of the wavelet frame coefficients into the respective variances and a renormalization.

3.4. Performance

Standard execution tests indicated a processing time of less than one second for 300x400 pixel with 20% overall unclassified space images on a 1.2 GHz Pentium III Unix machine, with a 60% of the execution time being attributed to the preprocessing stage and only 40% to the propagation/update algorithm itself. It should be noted that complex operations such as the closest point estimation could be performed through sub-optimal but very fast heuristics (the closest point according to the l_1 metric is within a factor of $\sqrt{2}$ of the distance from the l_2 optimal closest point, a well acceptable compromise, yet admits a fast $O(n)$ implementation).

4. APPLICATIONS AND RESULTS

4.1. Object extraction

The proposed framework can address applications of static segmentation and object extraction in the context of a user-assisted interactive editing environment. The zone containing the region boundary could be input by the user through a brush tool or a freehand outline of the object's interior and exterior. Figure 2 illustrates the application of the introduced framework for the semi-automatic segmentation of natural scenes exhibiting hard to determine region boundaries due to excessive texture, massively inhomogeneous content or several unlike sub-object components. Initialization is user supplied in the form of wide zones surrounding the actual boundaries. The process is applicable to any number of image regions as demonstrated in the final example.

4.2. Moving object localization

Partial classification maps for a given video frame can arise from a change detection algorithm operating onto an image sequence. Reliable decisions for the mobility of certain image sites may result from statistical analysis of video stream features, such as inter-frame difference or optical flow, where hard to decide image sites appear on the neighborhood of the moving objects as outliers of the motion model estimation. Different texture and distinct motion often force the boundaries of the unclassified region dangerously close to the object boundary, possibly compromising the effectiveness of region-growing algorithms. In figure 3 a multi-region fast marching algorithm [4] is used to implement a change detector with limited accuracy, followed by a relaxation stage aiming to frame the actual moving object boundary between two extremal curves. The illustrated examples are among the most problematic in that one of the boundary outlines detected often lie too close to the actual boundary, inducing serious problems with the operation of traditional region-growing algorithms such as SRG [8].

5. CONCLUSION

This paper presents a new fast level set algorithm for the precise localization of an object boundary given a highly confident partial classification of the object interiors. Statistical formulations are utilized to motivate the level set evolution. Color histograms and wavelet analysis are used to provide the image features used in a dynamic, locally adaptive region-growing scheme in order to robustly describe inhomogeneous and noisy objects, while preserving a favorable algorithmic complexity. Favorable performance is demonstrated in applications of user–assisted object extraction and fully autonomous moving object detection and localization.

6. REFERENCES

[1] T. Sikora, "The MPEG-4 video standard verification model," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 7, no. 1, pp. 19–31, Feb 1997.

Fig. 2. Region boundary determination for a user provided (grey outlines) classification into two (left, middle) or three (right) segments

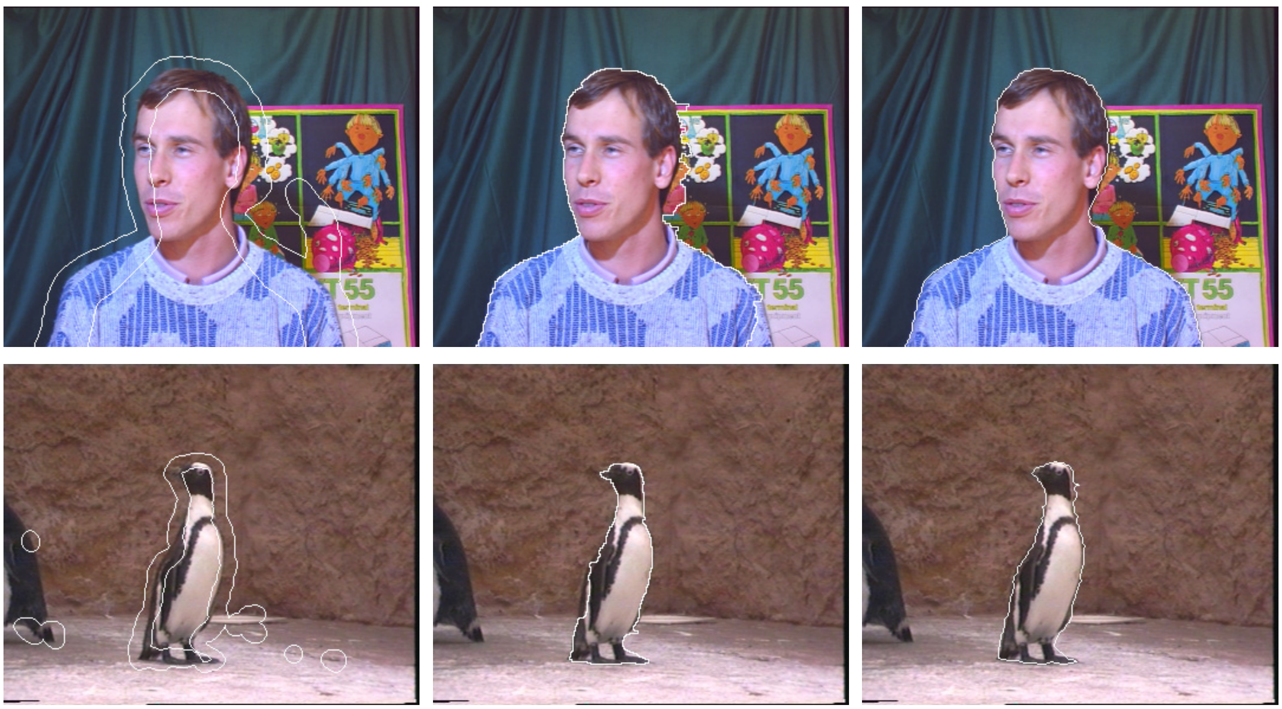

Fig. 3. Comparison of SRG (middle) and Locally adaptive Multi-region Fast Marching (right) for moving object localization.

[2] P. Salembier, "Overview of the MPEG-7 standard and of future challenges for visual information analysis," *EURASIP Journal on Applied Signal Processing*, vol. 4, pp. 343–353, Apr 2002.

[3] P. Daras et. al., "MPEG-4 authoring tool using moving object segmentation and tracking in video shots," *EURASIP Journal on Applied Signal Processing*, 2003 (to appear).

[4] E. Sifakis, I. Grinias, and G. Tziritas, "Video segmentation using fast marching and region growing algorithms," *EURASIP Journal on Applied Signal Processing*, vol. 4, pp. 379–388, Apr 2002.

[5] Y. Tsaig and A. Averbuch, "Automatic segmentation of moving objects in video sequences: a region labeling approach," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 12, pp. 597–612, Jul 2002.

[6] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *International Journal of Computer Vision*, vol. 1, no. 4, pp. 321–331, 1987.

[7] E. Sifakis, C. Garcia, and G. Tziritas, "Bayesian level sets for image segmentation," *Journal of Visual Communication and Image Representation*, vol. 13, no. 1, pp. 44–64, Mar 2002.

[8] R. Adams and L. Bischof, "Seeded region growing," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, no. 6, pp. 641–647, Jun 1994.

[9] J. N. Tsitsiklis, "Efficient algorithms for globally optimal trajectories," *IEEE Transactions on Automatic Control*, vol. 40, no. 9, pp. 1528–1538, Sep 1995.

[10] J. Sethian, "A fast marching level set method for monotonically advancing fronts," *Proceedings of the National Academy of Sciences*, vol. 93, no. 4, pp. 1951–1955, 1996.

[11] S. Osher and R. Fedkiw, *Level Set Methods and Dynamic Implicit Surfaces*, Springer-Verlag, Nov 2002.

[12] M. Unser, "Texture classification and segmentation using wavelet frames," *IEEE Transactions on Image Processing*, vol. 4, no. 11, pp. 1549–1560, Nov 1995.