

An Algorithm for Binary Image Segmentation Using Polygonal Markov Fields

Rafał Kluszczyński¹, Marie-Colette van Lieshout², and Tomasz Schreiber¹

¹ Nicolaus Copernicus University, ul. Chopina 12-18, 87-100 Toruń, Poland
klusi@mat.uni.torun.pl, tomeks@mat.uni.torun.pl

² CWI, Kruislaan 413, 1098 SJ Amsterdam, The Netherlands
Marie-Colette.van.Lieshout@cwi.nl

Abstract. We present a novel algorithm for binary image segmentation based on polygonal Markov fields. We recall and adapt the dynamic representation of these fields, and formulate image segmentation as a statistical estimation problem for a Gibbsian modification of an underlying polygonal Markov field. We discuss briefly the choice of Hamiltonian, and develop Monte Carlo techniques for finding the optimal partition of the image. The approach is illustrated by a range of examples.

1 Introduction

One of the fundamental image analysis tasks is that of *segmentation*, i.e. to partition the image in relatively homogeneous regions [10]. Indeed, segmenting the data is often the first step in image interpretation problems. The partition may be achieved at several conceptual levels. At the lowest level, that of individual pixels, segmentation amounts to classification of pixel values. At the other extreme, the focus of attention are the objects that constitute a given image and the goal is to extract them from the image.

A myriad of segmentation methods has been proposed, from elementary thresholding through level set approaches and contour extraction methods to scene modelling. In this paper, we propose to use polygonal field models. Thus, we place ourselves at the intermediate conceptual level that regards a segmentation as a coloured tessellation [8]. The advantage of such an approach is that - in contrast to pixel based ones - global aspects of the image are captured. At the same time there is no need to model all objects in the image, which is feasible in restricted application domains only. Furthermore, a coloured polygonal tessellation is a reasonable and widely applicable mathematical formalisation of the intuitive concept of a segmentation, especially when compared to the rather artificial level set model or to the notion of a collection of pixel labels that do not necessarily have any spatial coherence. The idea can be traced back to Clifford and Middleton [6]; from a computational point of view, a Metropolis-Hastings style sampler was developed by Clifford and Nicholls [7] and applied to an image reconstruction problem within a Bayesian framework. We shall use a modification of the algorithm in Schreiber [12] which is conceptually and computationally easier.

2 Preliminaries and Notation

We aim to describe the contents of an image domain D , assumed to be an open, convex and bounded subset of the plane (typically a rectangle), by means of a family of non-intersecting polygonal contours in D , possibly nested or chopped off by the boundary. We restrict ourselves to foreground/background segmentation, so that a contour may be interpreted as a polygonal boundary between black (foreground) and white (background) regions. We shall use the notation \bar{y} for a discretised image, $S \subset D$ for the pixel set. The value y_s at pixel s belongs to some finite set L . A collection of contours is denoted by γ . For each collection, there are exactly two admissible black and white colourings. We use a hat notation, $\hat{\gamma}$, to describe a family of non-intersecting polygonal contours with its associated colouring.

Our approach, as developed in [12] and [9], involves the following building blocks.

1. The first ingredient is, intuitively speaking, the ability to generate a 'completely random' polygonal field, both as a benchmark or reference field, and as a tool for exploring the space of admissible polygonal configurations without favouring any particular one. A reasonable choice is the so-called Arak process [1,2,3,4], denoted by \mathcal{A}_D .
2. Secondly, we need a *goodness-of-fit measure* $\mathcal{H}(\bar{y} | \hat{\gamma})$ to quantify how well a coloured polygonal configuration $\hat{\gamma}$ matches the data image \bar{y} . Moreover, we would like to be able to influence the geometry of $\hat{\gamma}$ by assigning a higher probability to large polygons with smooth boundaries that do not have spurious edges. This is captured by a so-called *regularisation term* $\mathcal{H}(\bar{y})$, cf. Section 3 in [9].
3. The third and last ingredient is an updating mechanism which keeps the distribution of the reference Arak field invariant while exhibiting good exploratory properties in the configuration space. To this end, we develop the so-called disagreement-loop birth and death algorithm, originally introduced in [12] and further extended in [9]. This mechanism can then be combined with standard Metropolis and simulated annealing techniques to find an optimal segmentation $\hat{\gamma}$.

The next three sections will elaborate a little further on the above points. A full account can be found in [12,9].

3 The Arak Process: Dynamic Representation

The crucial idea underlying Arak's construction [1] is to interpret the polygonal boundaries of the field as the trace left by a particle travelling in two-dimensional time-space. Thus, the two-dimensional image domain D is seen as a set of *time-space* points $(t, y) \in D$, with t referred to as the *time* coordinate and with y standing for the (1D) *spatial* coordinate. In this language, the basic Arak process is constructed as follows.

Birth events. The new particles are born in pairs at *birth sites* chosen in the interior of the domain D according to a homogeneous Poisson point process of intensity π . There are also *boundary birth sites* emitting single particles, generated by a Poisson point process with an appropriate intensity measure $\kappa(\cdot)$ concentrated on ∂D , whose analytic details we omit here for simplicity of presentation, referring the reader to [2] for an exhaustive description.

- * Each interior birth site emits two particles, moving with initial velocities v' and v'' chosen according to the joint distribution $\theta(dv', dv'') := \pi^{-1}|v' - v''|(1+v'^2)^{-3/2}(1+v''^2)^{-3/2}dv'dv''$ on $v' < v''$. This is equivalent to choosing the angle $\phi \in (0, \pi)$ between the straight lines representing the space-time trajectories of the emitted particles according to the density $\sin(\phi)/2$.
- * Each boundary birth site $x \in \partial D$ yields one particle with initial speed v determined according to an appropriate distribution $\theta_x(dv)$, see [2] for its explicit form.

The colour in the interior of the newly created angle is chosen so as not to clash with the one to the left of the trajectory (the past in time-space terms), with minor modification for left-extreme points.

Evolution rules. All the particles evolve independently in time according to the following rules.

- * Between the critical moments listed below each particle moves freely with constant velocity, hence $dy = vdt$.
- * When a particle touches the boundary ∂D , it dies.
- * In case of a collision of two particles (equal spatial coordinates y at some moment t with $(t, y) \in D$), both of them die.
- * The time evolution of the velocity v_t of an individual particle is given by a pure-jump Markov process so that $\mathbf{P}(v_{t+dt} \in du \mid v_t = v) = q(v, du)dt$ for the transition kernel $q(v, du) := |u - v|(1 + u^2)^{-3/2}du$.

The random polygonal configuration obtained in the above procedure is precisely the basic Arak process in D . As shown in [2], \mathcal{A}_D enjoys a number of striking properties. One of them is the two-dimensional germ *Markov property*, stating that the conditional distribution of the field inside an open bounded region with piecewise smooth boundary given the outside configuration depends only on the trace of this configuration on the boundary (colouring, intersection points and intersection directions). The next important property is *consistency*: for bounded open and convex D_1 and D_2 with $D_1 \subseteq D_2$ the restriction of \mathcal{A}_{D_2} to D_1 coincides in distribution with \mathcal{A}_{D_1} , see Theorem 4.1 of [2]. A crucial property is the isometry invariance of the Arak process - while the translational invariance can be easily deduced from the construction above, the rotational invariance is a deep and non-trivial result that follows from the particular choice of the kernels $\theta(\cdot)$ and $q(\cdot, \cdot)$ above. Moreover, one dimensional sections of \mathcal{A}_D happen to be homogeneous Poisson point processes. Finally, a number of explicit formulae are available for various numeric characteristics of \mathcal{A}_D , such as the expected total edge length, mean number of vertices, edges etc, see [2,4,7]. These properties suggest that the Arak process is a suitable reference field.

4 Model-Based Image Segmentation

The model to be used for inference will be a Gibbsian modification of the polygonal random field \mathcal{A}_D by means of a Hamiltonian (sometimes referred to as energy function)

$$\mathcal{H}(\bar{y}; \hat{\gamma}) = \mathcal{H}(\bar{y} | \hat{\gamma}) + \mathcal{H}(\hat{\gamma}) \tag{1}$$

that is the sum of two terms, cf. Section 2. In other words, upon observation of \bar{y} , the likelihood of $\hat{\gamma}$ with respect to the reference field is weighted by a factor $\exp[-\mathcal{H}(\bar{y}; \hat{\gamma})]$, then normalised to have total probability mass 1. We take $\mathcal{H}(\bar{y} | \hat{\gamma}) = L_1(\bar{y}, \hat{\gamma})$, the sum of absolute differences between the actual pixel values and those assigned by $\hat{\gamma}$. For binary images, the L_1 distance reduces to $|\bar{y}\Delta\hat{\gamma}|$, the cardinality of the set of sites at which the observed colour does not match that of the polygon, which can be interpreted probabilistically as a random noise model in which each pixel value is flipped to the wrong colour independently of other pixels with some probability $p < 1/2$ (see [5,9] for details). Thus, finding an optimal $\hat{\gamma}^*$ in the absence of further regularisation would amount to minimising the misclassification rate $|\bar{y}\Delta\hat{\gamma}|/|S|$. In general, $\hat{\gamma}^*$ is not unique. Moreover, it tends to result in an over-segmentation. To overcome these problems, we added a regularisation term $\mathcal{H}(\hat{\gamma}) = \beta\ell(\gamma)$ proportional to the total edge length.

We are now ready to rephrase image segmentation as the task of finding a coloured polygonal configuration $\hat{\gamma}$ and a real β so that the Hamiltonian value in (1) is a sufficiently good approximation of the global minimum.

5 Disagreement Loop Birth and Death Dynamics

To minimise (1), we use simulated annealing. Briefly, given a sequence of temperatures T_n decreasing to zero as $n \rightarrow \infty$, we sample from a probability density proportional to $\exp[-\mathcal{H}(\bar{y}; \hat{\gamma})/T_n]$ with respect to the reference field. Clearly it suffices to describe how to sample for $T_n = 1$. A crucial concept in our algorithm is that of a *disagreement loop* [11, Section 2.2]. Consider adding a new birth site x_0 to configuration γ , and denote the resulting polygonal configuration by $\gamma \oplus x_0$. Then, for $x_0 \in \text{Int } D$, the symmetric difference $\Delta^\oplus[x_0; \gamma] := \gamma \Delta [\gamma \oplus x_0]$ is just a single loop (a closed polygonal curve), possibly self-intersecting and possibly chopped off by the boundary. Likewise, a disagreement loop $\Delta^\ominus[x_0; \gamma]$ arises by removing a birth site x_0 in polygonal configuration γ ; we write $\gamma \ominus x_0$ for the resulting configuration. This is discussed in details in [11,12], here we only give an illustration in Fig. 5. Recall that $\kappa(\cdot)$ is the boundary birth intensity measure as in Section 3. Then,

- (DL:birth) with intensity $[\pi dx + \kappa(dx)]ds$ set $\delta := \gamma_s \Delta \Delta^\oplus[x; \gamma_s] = \gamma_s \oplus x$. Choose either of the two admissible colourings with probability 1/2 to obtain $\hat{\delta}$. Then, with probability $\min\left(1, \exp\left[\mathcal{H}(\bar{y}; \hat{\gamma}_s) - \mathcal{H}(\bar{y}; \hat{\delta})\right]\right)$ put $\hat{\gamma}_{s+ds} := \hat{\delta}$, otherwise keep $\hat{\gamma}_{s+ds} := \hat{\gamma}_s$;

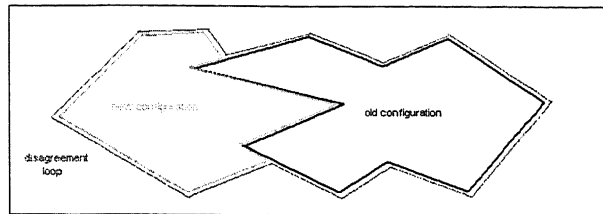


Fig. 1

(DL:death) for each birth site x in γ_s , with intensity $1 \cdot ds$ set $\delta := \gamma_s \Delta \Delta^\ominus[x; \gamma_s] = \gamma_s \ominus x$. Choose either of the two admissible colourings with probability $1/2$ to obtain $\hat{\delta}$. Then, with probability $\min(1, \exp[\mathcal{H}(\bar{y}; \hat{\gamma}_s) - \mathcal{H}(\bar{y}; \delta)])$ put $\hat{\gamma}_{s+ds} := \hat{\delta}$, otherwise keep $\hat{\gamma}_{s+ds} := \hat{\gamma}_s$.

By Theorem 1 in [12] and Theorems 1–2 in [9], under these dynamics the current polygonal configuration converges in distribution to the Gibbsian modification of \mathcal{A}_D with the Hamiltonian $\mathcal{H}(\bar{y}; \hat{\gamma})$. This algorithm exploits the fact that disagreement loops $\Delta^{\oplus, \ominus}[x; \gamma]$ are very well suited for simulation. Since both the dynamic representation of the Arak process in Section 3 and the simulation algorithm presented here involve a notion of time, in the sequel we refer to the former as to the *representation time* (r-time) and to the latter as to the *simulation time* (s-time).

Extensions of the algorithm. In order to increase the efficiency of the basic algorithm we endow it with a number of extra Monte Carlo moves. These include

1. Random rotations of the spatial and time axes in the dynamic representation of the Arak process. While enlarging the set of possible moves, this does not alter the stationary distribution of the process due to the isometry invariance of \mathcal{A}_D .
2. Repetitive velocity updates: a particle undergoing a velocity update in r-time, is allowed to ‘change its mind’ in the course of s-time and to randomly modify the previously performed update. This can also be done so that the stationary distribution is unaltered.
3. Rescaling, to guarantee better resolution at later stages of annealing.

6 Implementation and Examples

The algorithm was implemented in C++. General features are discussed briefly below.

Representation of polygonal configurations. A configuration of a polygonal Markov field is represented as a list of labelled vertices. The full description of a vertex is provided by

- the Cartesian coordinates of the vertex;
- two pointers to the neighbouring vertices;

- the *virtual lengths* of the segments that emanate from the given vertex; these are the lengths the segments would have if the corresponding particles were the only ones present in the system and evolved in an empty environment. The actual lengths of these segments are usually smaller due to collisions.

The list of vertices is sorted by increasing x -coordinate (r-time coordinate).

Generation of the initial configuration. The initial configuration for our MCMC procedure is generated according to the dynamic representation of the basic Arak process as discussed in Section 3. This is done in a single left-to-right sweep through the image domain, by successively updating in r-time two priority queues that store respectively

- the birth sites with r-time coordinate exceeding the current r-time, and *virtual end points* (with the distance from the respective initial point given by the corresponding virtual length) of segments generated so far, for which the r-time coordinate exceeds the current r-time;
- *virtual collision points* which are all possible pairwise intersection points of currently existing *virtual segments* (i.e. segments joining an initial point to its corresponding virtual end point) lying forward in r-time.

At each step of the algorithm, the next vertex to arise in the course of the r-time evolution is determined by choosing that vertex that minimises the r-time coordinate in both queues. Consequently, the contents of these queues can be regarded as a current collection of ‘candidates’ for the next vertex.

Updates. Our main configuration-modifying operations are adding a new birth site (disagreement loop birth (**DL:birth**)) and removing an existing one (disagreement loop death (**DL:death**)). To add a new birth site, we first choose its position uniformly at random within the image domain, and then we let the newborn particles evolve and interact with the existing ones according to the usual evolution rules. Likewise, when removing a given birth site, we let the remaining particles obey the usual evolution rules. Both these updates are implemented using the same data structures as when generating the initial configuration above.

Evaluation of the Hamiltonian. For binary images \bar{y} , $\mathcal{H}(\bar{y} \mid \hat{\gamma})$ requires the evaluation of the number of misclassified pixels upon each update proposal. To this end we apply the divergence theorem, constructing a real-valued vector field such that the input image data \bar{y} coincides with its divergence, and then computing appropriate flux integrals along the suitably oriented contours of the polygonal field. For grey level images, we resort to Monte Carlo sampling to calculate the L_1 distance between \bar{y} and $\hat{\gamma}$.

Examples. Here we present a few typical segmentations obtained by the approach. The data in the first example consist of a spray-traced image of a happy face. For the cooling schedule we used $1/T_n := 20.0 + 0.009 * n$. The result after approximately 1 150 000 steps is given in Figure 2a. The misclassification rate we achieved was 3 percent, as visualised in the corresponding graph [Fig.

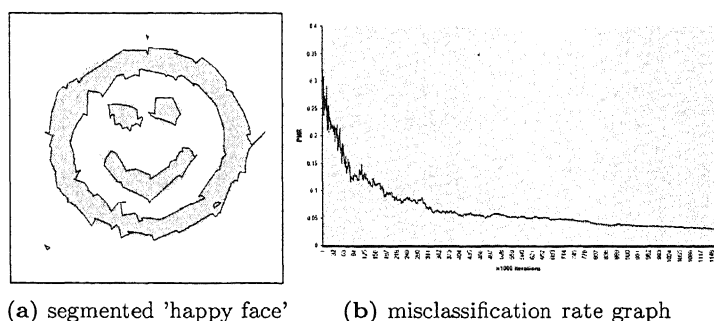


Fig. 2

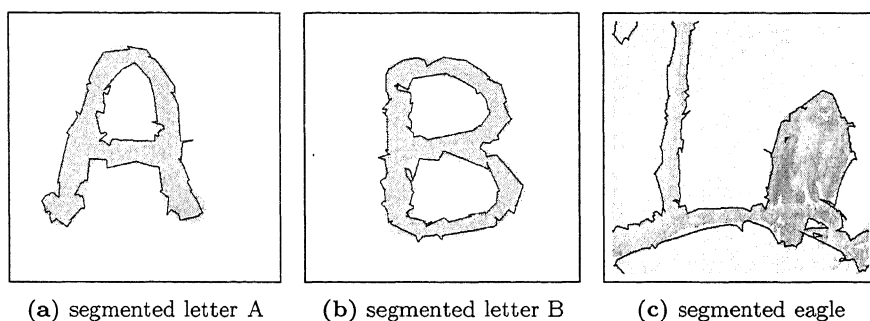


Fig. 3

2b]. Average execution time for a single iterate was 0.0061 second for 2 CPUs architecture with Intel Xeon 2.4 GHz processors and 2 GB RAM. The data of the second and third image are given by the first letters of the alphabet, 'A' and 'B'. The results under the same cooling regime are given in Figures 3a and 3b, and were achieved after 800 000 and 2 million iterations respectively. The misclassification rate is 3 percent in both cases. Finally, Fig. 3c presents a sample segmentation of a grey level eagle image from the Berkeley segmentation dataset and benchmark site. The misclassification rate reached after 860 000 iterations was again 3 percent.

7 Discussion and Future Work

Here, for brevity, we restricted ourselves to foreground/background segmentation. However, the general framework can easily be extended to allow for segmentation into $k > 2$ classes of grey level, colour, or textured images [9]. Suitable consistent polygonal field models are available with more flexibility in the type of intersections [4], but care is needed with respect to the Hamiltonian. Indeed, it is the object of our current work to develop software for such more complicated cases, and to evaluate the performance of our algorithm on benchmark data available at: <http://www.cs.berkeley.edu/projects/vision/grouping/segbench/> or <http://mosaic.utia.cas.cz>.

A preliminary conclusion is that, in contrast to model-based probabilistic image segmentation at the pixel level, the topology of the foreground object is preserved better. Our approach is also relatively robust with respect to noise. The price to pay is that fine details are not recovered, especially those whose sizes fall below the characteristic scale of the polygonal field. This problem could easily be solved in a post-processing smoothing phase. An alternative could be to gradually decrease the characteristic scale of the field (multi-resolution approach) or to build local updates in the spirit of [7] into the algorithm.

Acknowledgements

This research was supported by the EC 6th Framework Programme Priority 2 Information Society Technology Network of Excellence MUSCLE (Multimedia Understanding through Semantics, Computation and Learning; FP6-507752) and by the Foundation for Polish Science (FNP) and the Polish Minister of Scientific Research and Information Technology grant 1 P03A 018 28 (2005-2007).

References

1. ARAK, T. (1982) On Markovian random fields with finite number of values. In *4th USSR-Japan Symposium on Probability Theory and Mathematical Statistics, Abstracts of Communications*, Tbilisi.
2. ARAK, T. AND SURGAILIS, D. (1989) Markov fields with polygonal realisations. *Probability Theory and Related Fields* **80**, 543–579.
3. ARAK, T. AND SURGAILIS, D. (1991) Consistent polygonal fields. *Probability Theory and Related Fields* **89**, 319–346.
4. ARAK, T., CLIFFORD, P. AND SURGAILIS, D. (1993) Point-based polygonal models for random graphs. *Advances in Applied Probability* **25**, 348–372.
5. BADDELEY, A.J. AND VAN LIESHOUT, M.N.M. (1992) ICM for object recognition. In *Computational Statistics*, Y. Dodge and J. Whittaker (Eds.), volume 2, pp. 271–286. Physica/Springer, Heidelberg.
6. CLIFFORD, P. AND MIDDLETON, R.D. (1989) Reconstruction of polygonal images. *Journal of Applied Statistics* **16**, 409–422.
7. CLIFFORD, P. AND NICHOLLS, G.K. (1994) A Metropolis sampler for polygonal image reconstruction. Electronic version available at: http://www.stats.ox.ac.uk/~clifford/papers/met_poly.html.
8. HURN, M.A., HUSBY, O. AND RUE, H. (2003) Advances in Bayesian image analysis. In *Highly Structured Stochastic Systems*, P.J. Green, S. Richardson and N.L. Hjort (Eds.), Oxford Statistical Science Series **27**, 323–325. Oxford University Press, Oxford.
9. KLUSZCZYŃSKI, R., VAN LIESHOUT, M.N.M. AND SCHREIBER, T. (2004) Image segmentation by polygonal Markov fields. Electronic version available as CWI Research Report PNA-R0409 at: <http://www.cwi.nl/publications>.
10. ROSENFELD, A. AND KAK, A.C. (1982) *Digital picture processing*, second edition, volume 2. Academic Press, Orlando.
11. SCHREIBER, T. (2003) Mixing properties of polygonal Markovfields in the plane. Electronic version available at: <http://www.mat.uni.torun.pl/preprints>, 18-2003.
12. SCHREIBER, T. (2004) Random dynamics and thermodynamic limits for polygonal Markov fields in the plane. Electronic version available at: <http://www.mat.uni.torun.pl/preprints>, 17-2004.