Energy minimisation of Partitions and Discrete Length Estimators

Luc Brun†, Jacques Olivier Lachaud‡

†GreyC CNRS UMR 6072
Équipe Image - ENSICAEN
6, Boulevard du Maréchal Juin
14050 CAEN Cedex - France
‡LaBRI CNRS UMR 5800
Équipe Image et Sons
351, cours de la Libération
33405 Talence cedex

†{luc.brun}@greyc.ensicaen.fr,‡lachaud@labri.fr

Abstract. The scale set theory allows to define a hierarchy of segmentations according to a scale parameter. This theory closely related to the Bayesian and the Minimum description Length(MDL) frameworks describes the energy of a partition as the sum of two terms : a goodness to fit and a regularisation term. This last term may be interpreted as the encoding cost of the model associated to the partition. It usually includes the total length of the partition's boundaries and is simply computed as the number of lignels between the regions of the partition. We propose to use a better estimation of the total length of the boundaries by using discrete length estimators. We state the basic properties which must be fulfilled by these estimators and show their influence on the partitition's energy.

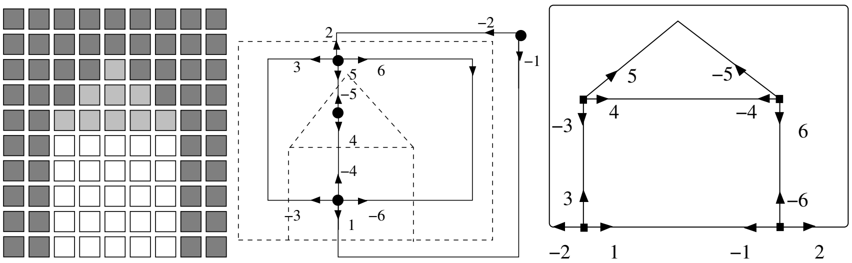
1 Introduction

1.1 Context

Different structures may be observered in a same image at different scales. The pioneer work of Witkins [] introduced the notion of continuous representation in scale. Using this representation a $1D$ signal is represented by a $2D$ function $f(t, \sigma)$. This *scale-space* representation enphase the fact that the value of a point t depends both on the position t of the observation and of the scale at which the observation is performed. The scale space representation representents thus the content of a signal or an image at different scales. From this point of view, the work of Witkins is quite from the the Gaussian pyramids or Wavelet [] representations. However, using a representation a partition of an image is deduced afterwards from the multi scale description of its content. In other worlds, the scale space representation do not readily allows to encode the evolution of a partition according to a scale parameter. Laurent Guigues [] introduced

2 Combinatorial Pyramids

2.1 Combinatorial maps



(a) Image (b) $G = (\mathcal{D}, \sigma, \alpha)$ (c) $\overline{G} = (\mathcal{D}, \varphi, \alpha)$

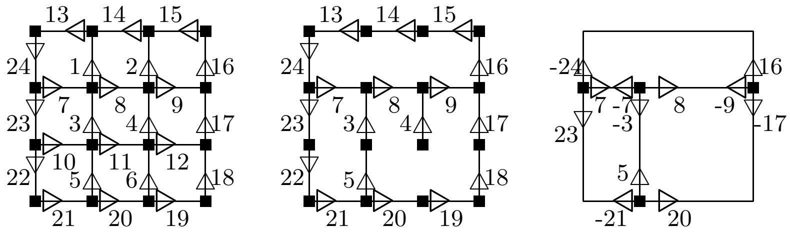
Fig. 1. The ideal segmentation of image (a) is encoded by the combinatorial map G (b). The borders of the partition are overlayed with the edge of the graph in figure (b). The dual combinarotial map $\overline{G}$ is provided in (c).

A combinatorial map $G = (\mathcal{D}, \sigma, \alpha)$ encodes a partition on an orientable surface without boundary. Combinatorial maps are used within the image processing and analysis framework to encode image's partitions. Using $2D$ images, combinatorial maps may be understood as a particular encoding of a planar graph where each edge is split into two half-edges called darts (e.g. darts 1 and -1 in Fig. 1(b)). Since each edge connects two vertices, each dart belongs to only one vertex. A $2D$ combinatorial map is formally defined by the triplet $G = (\mathcal{D}, \sigma, \alpha)$ where $\mathcal{D}$ represents the set of darts and σ is a permutation on $\mathcal{D}$ whose cycles correspond to the sequence of darts encountered when turning counter-clockwise around each vertex (e.g. cycle $(-4, -3, 1, -6)$ in Fig 1(b)). Finally α is an involution on $\mathcal{D}$ which maps each of the two darts of one edge to the other one (e.g. α maps 1 to -1 and -1 to 1 in Fig 1(b)). The cycles of α and σ containing a dart d will be respectively denoted by $\alpha^*(d)$ and $\sigma^*(d)$.

Each vertex of a combinatorial map $G = (\mathcal{D}, \sigma, \alpha)$ is implicitly encoded by its σ orbit. This implicit encoding may be transformed into an explicit one by using a vertex's labeling function [1]. Such a function, denoted by μ associates to each dart d a label which identifies its σ-orbit $\sigma^*(d)$. More precisely a vertex's labeling function should satisfy:

$$\forall (d, d')^2 \in \mathcal{D} \; \mu(d) = \mu(d') \Leftrightarrow \sigma^*(d) = \sigma^*(d')$$

For example, if darts are encoded by integers, the function $\mu(d) = min_{d' \in \sigma^*(d)}\{d'\}$ defines a valid vertex's labeling function.



(a) $\overline{G_0}$ (b) $\overline{G_1}$ contracted (c) $\overline{G_3}$ reduced

Fig. 2. A dual of a combinatorial map (a) encoding a 3×3 grid with the contracted combinatorial map (b) obtained by the contraction of the contraction kernel (CK) $K_1 = \alpha^*(1, 2, 10, 11, 12, 6)$. The reduced combinatorial map (c) is obtained by the removal of the empty self loops defined by the RKESL $K_2 = \alpha^*(4)$ and the removal kernel of empty double edges (RKEDE) $K_3 = \alpha^*(13, 14, 15, 19, 18, 22) \cup \{24, -16, 17, -20, 21, -23, 3, -5\}$.

Given a combinatorial map $G = (\mathcal{D}, \sigma, \alpha)$, its dual is defined by $\overline{G} = (\mathcal{D}, \varphi, \alpha)$ with $\varphi = \sigma \circ \alpha$(Fig 1(c)). The cycles of permutation φ encode the faces of the combinatorial map and may be interpreted as the sequence of darts encountered when turning clockwise around a face. The cycle of φ containing a dart d will be denoted by $\varphi^*(d)$.

Fig. 1 represents the encoding of the ideal segmentation of a house by a combinatorial map. The set of dart $\mathcal{D}$ of this combinatorial map is equal to $\{-6, \ldots, -1, 1, \ldots, 6\}$ and the involution α is implicitly encoded by the sign on this figure. The region corresponding to the background of the house is encoded by the vertex $\sigma^*(2) = (2, 3, 5, 6)$ (Fig. 1(b)). This vertex corresponds to a face of the dual combinatorial map(Fig.1(c)) where $\sigma^*(2)$ corresponds to the sequence of darts encoutered when turning counter-clockwize around the face. Note that a mapping such as $\mu(2) = \mu(3) = \mu(5) = \mu(6) = 2$ would be consistent for the vertex $\sigma^*(2)$. The φ cycle of the dart 2 is defined by : $\varphi(2) = \sigma(\alpha(2)) = \sigma(-2) = -1$, $\varphi(-1) = \sigma(1) = -6$ and $\varphi(-6) = \sigma(6) = 2$. We have thus $\varphi^*(2) = (2, -1, -6)$. This cycle is represented by the bottom right vertex in Fig. 1(c).

2.2 Combinatorial map encoding of a planar sampling grid

Fig. 2(a) describes a dual combinatorial map $\overline{G_0} = (\mathcal{D}_0, \varphi_0, \alpha_0)$ encoding a 3×3 4-connected planar sampling grid. Using this encoding the φ, σ and α cycles of each dart may be respectively understood as elements of dimensions 0, 1 and 2 and formally associated to a 2D cellular complex [3]. More precisely, each α_0 cycle may be associated to a lignel between two pixels. Each of the two darts of an α_0 cycle corresponds to an orientation along the lignel. For example, the cycle $\alpha_0^*(1) = (1, -1)$ is associated to the lignel encoding the right border of the top left pixel of the 3×3 grid (Fig. 2(a)). The darts 1 and -1 define respectively a bottom to top and top to bottom orientation along the lignel.

2.3 Construction of Combinatorial Pyramids

A combinatorial pyramid is defined by an initial combinatorial map successively reduced by a sequence of contraction or removal operations. Contraction operations are encoded by contraction kernels (CK). These kernels defined as a forest of the current combinatorial map may create redundant edges such as empty-self loops and double edges(Fig. 2(b)). Empty self loops (edge $\alpha_1^*(4)$ in Fig. 2(b)) may be interpreted as region's inner boundaries and are removed by an empty self loops removal kernel (RKESL) after the contraction step. The remaining redundant edges called double edges, belong to degree 2 vertices in $\overline{G}$ (e.g. $\varphi_1^*(13)$, $\varphi_1^*(14)$, $\varphi_1^*(15)$) in Fig. 2(b)) and are removed using a double edge removal kernel (RKEDE) which contains all darts incident to a degree 2 dual vertex. From a geometrical point of view, a RKEDE concatenates several boudaries between two regions which are connected by degree two dual vertices. The application of a kernel CK or RKESL induces thus the removal of some boundaries between regions while the application of a RKEDE concatenates boundaries between regions which become artificially divided after the application of a CK. Note that only the contraction kernel is application dependent. The application of the RKESL and the RKEDE may be interpreted as cleaning steps of the data structure and are automatically deduced from the combinatorial map created by the contraction kernel. Given an initial combinatorial map G_0, encoding a planar sampling grid, a first contraction kernel K_1 is applied on G_0 in order to produce a first reduced combinatorial map G_1. The combinatorial map G_1 is then further reduced using the RKESL K_2 which produces the reduced combinatorial map G_2 and the RKEDE K_3 which produces the combinatorial map G_3 which encodes the same partition then G_1 but without any redundant edges. This process may be applied recursively in order to produce a sequence $(G_0, \ldots, G_n)$ of reduced combinatorial maps definining the combinatorial pyramid. Further details about the construction scheme of a combinatorial pyramid may be found in [1, 2].

As mentioned in Section 2.1, if the initial combinatorial map encodes a planar sampling grid, the geometrical embedding of each initial dart corresponds to an oriented lignel. Moreover, each dart of a reduced combinatorial map which is not a self loop encodes a connected boundary between two regions. The embedding of the boudary associated to such a dart may be retrieved from the embedding of the darts of the initial combinatorial map G_0. Let us consider a reduced combinatorial map $G_i = (\boldsymbol{\mathcal{D}}_i, \sigma_i, \alpha_i)$ defined at level i and one dart $d \in \boldsymbol{\mathcal{D}}_i$ which is not a self loop. The sequence $d_1 \ldots, d_n$ of initial darts encoding the embedding of the dart d is retreived from the receptive field of d [1, 4] using the following equation:

$$d_1 = d \, , d_{j+1} = \varphi_0^{m_j}(\alpha_0(d_j)) \tag{1}$$

where $\overline{G_0} = (\boldsymbol{\mathcal{D}}_0, \varphi_0, \alpha_0)$ is the dual of the initial combinatorial map and m_j is the minimal integer such that $\varphi_0^q(\alpha_0(d_j))$ survives at level i or belongs to a double edge kernel. The dart d_n is the first dart defined by equation 1 which survives up to level i. This dart also satisfies $\alpha_0(d_n) = \alpha_i(d)$ by construction of the receptive fields [1, 4]. Note that the tests performed on $\varphi_0^q(\alpha_0(d_j))$, $q \in \{1, \ldots, m_j\}$ to determine if it is equal to d_{j+1} or d_n are performed in constant time using the implicit encoding of combinatorial pyramids [1].

If G_0 encodes the 4-connected planar sampling grid, each φ_0 cycle is composed of at most 4 darts (Fig. 2(b)). Therefore, the computation of d_{j+1} from d_j requires at most 4 iterations and the determination of the whole sequence of lignels composing a boundary between two regions is performed in a time proportional to the length of this boundary.

2.4 Region's boundaries : An Example

Let us consider the dart 16 in Fig. 2 (c). This dart belongs to the σ_3 cycle $\sigma_3^*(16) = (16, 7, 8)$ of G_3 and encodes the first row of the 3×3 grid (Fig. 2 (c)). The sequence of initial darts encoding the boundary of the dart 16 is retrived using equation 1 and is equal to: 16.15.14.13.24 (Fig. 2(b)). We have for example $15 = \varphi_0(\alpha_0(16)) = \varphi_0(-16)$ (Fig. 2 (c)). In the same way, the darts 7 and 8 are respectively associated to the sequences of initial darts 7 and 8.9 (Table 1 and Fig. 2(b)). Since each initial dart is associated to an oriented linel, one may associate a sequence of Freeman's code to each sequence of initial darts (third column of Table 1 and Fig. 2(b)). The sequence of Freeman's code encoding the boundary of $\sigma_3^*(16) = (16, 7, 8)$ is defined as the concatenation of the sequences associated to 16, 7 and 8 and is equal to : 1.2.2.2.3.0.0.0 (Fig. 2 (b)).

reduced dart	sequence of initial darts	Freeman's codes
16	16.15.14.13.24	1.2.2.2.3
7	7	0
8	8.9	0.0

Table 1. Sequences of Freeman's codes encoding the boundary of each of the darts $\sigma_3^*(16) = (16, 7, 8)$ within the reduced combinatorial map G_3(Fig. 2(c)).

3 Discrete Estimators

4 Energie computation

5 Conclusion

References

[1] L. Brun. *Traitement d'images couleur et pyramides combinatoires*. Habilitation à diriger des recherches, Université de Reims, 2002.

[2] L. Brun and W. Kropatsch. Combinatorial pyramids. In Suvisoft, editor, *IEEE International conference on Image Processing (ICIP)*, volume II, pages 33–37, Barcelona, September 2003. IEEE.

[3] L. Brun and W. Kropatsch. Receptive fields within the combinatorial pyramid framework. *Graphical Models*, 65:23–42, 2003.
[4] L. Brun and W. Kropatsch. Contains and inside relationships within combinatorial pyramids. *Pattern Recognition*, 39(4):515–526, April 2006.