A Multiscale Particle Filter and Winding Number Constrained for Contour Detection

Sonam Verma
M-Tech Student
Mittal Group of Institutions Bhopal, India

Achint Chugh
Assistant Professor
Mittal Group of Institutions Bhopal, India

ABSTRACT
This paper compares the basic contour detection algorithms. A contour detection algorithm which jointly tracks at two scales small pieces of edges called edgelets. This multiscaleedgelet structure naturally embeds semi-local information and is the basic element of the recursive Bayesian modeling. The underlying model is estimated using a sequential Monte Carlo approach, and the soft contour detection map is retrieved from the approximated trajectory distribution. The winding number constrained contour detection (WNCCD) is an energy minimization framework based on winding number constraints. In this framework, both region cues, such as color/texture homogeneity, and contour cues, such as local contrast and continuity, are represented in a joint objective function, which has both region and contour labels. This technique is based on the topological concept of winding number. Using a fast method for winding number computation, a small number of linear constraints are derived to ensure label consistency. Experiments conducted on the Berkeley Segmentation data sets show that the Multi Scale Particle Filter Contour Detector method performs a comparable result with the winding number constrained contour detection method.

Keywords
Particle filtering, sequential Monte Carlo methods, statistical model, multiscale contour detection, BSDS

1. INTRODUCTION
We ask that Detecting contours is a universal job in image processing, as it is often the basis of higher level applications, such as segmentation, recognition, tracking, etc. The intrinsic variability of natural images makes this task a proper challenge. In this paper, we describe a contour as a visually salient, well-defined chain of connected pixels. This definition may be understood in terms of the Gestalt Theory, which underlines the significance of perceptual grouping and continuation properties for human visual perception. Also, saliency is contextual, signifying that it conforms the Helmhotz principle, which discusses more importance to unusual geometric patterns. Properties of good continuation and saliency shall function as motivations of the proposedcontour detector. A connected pixel set is the atomic element of the proposed technique, and is named an edgelet. This term shall not be mistaken with the edgelet transform (an image representation method), however, our definition is similar to the ones in [1], [2]. The construction of an edgelet is learned offline by a shape database [3].

This paper attempts a more general, yet efficient approach that tightly integrates both region nods and contour cues. We consider the contour extraction problem within an energy minimization framework. Our objective function is designed to encode various regions and contour cues by explicitly introducing both 2D region labels and 1D contour labels. Then we use novel constraints to ensure the consistency of region and contour labels. There are other ways to use both region/and contour cues [8]. For example, we could first segment the objects using region methods and then refine the segmentation results by contours. However, using contour cues for post processing can be sub-optimal in the sense that it can only refine the results. In contrast, our framework involves both contour cues and region cues in global optimization. In addition, this framework allows the flexibility in choosing energy functions. The key to this framework is the design of constraints with the following three properties. First of all, these constraints must ensure the topological correctness of solutions. For example, regions with different labels should be separated by contours, and object boundaries should not be fragmented. Second, these constraints should not be too preventive such that there is no viable solution. Recent work [8], [19] argued that in order to ensure the closed ness condition, exponentially many constraints are desired. Other particle filtering skills have been used in the context of vessels and arteries detection in 3D CT data [3], [5], [6]. Like the Jet Stream algorithm, these techniques have been mainly dedicated for semi-automatic and/or single detection tasks. Contrary to the aforementioned methods, our particle filtering framework is fully automatic, semi-local, and contextually-dependent. Moreover, related to our initial model [7], the edgelets are defined at two scales, meaning that the algorithm locally tracks the edgelets along contours by sequentially operating the computations on each scale. This produces to our new Multi-Scale Particle Filter Contour Detector (MS-PFCD). This paper is systematized as follows. Segment 2 makes a study of previous works proposed in the contour detection literature. In Segment 3, we propose to learn the distributions that handle the multiscaleedgelets and define our Bayesian model. The Bayesian model is defined by our proposed method to acquire the distributions that handle the multiscaleedgelets..

2. LEARNING THE BAYESIAN MODEL
Before giving the motivations of the proposed section and further, the sequential Bayesian framework, we introduce some notations. The basic element of the framework is a two-scales set of connected points. Let $E = (E^1, \dots\dots, E^{M_E}) \in \Gamma'_E \subset \Omega_E^{M_E}$ be a set of M_E four-connected points defined at the coarse scale. Each point E^i is defined.

The proposed contour detection method is based on a spatial tracking approach. This means that we want to define an edgelet at a certain step, or time, of the tracking procedure.

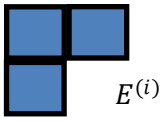
3. LEARNING THE BAYESIAN MODEL
The vectors e and e are small pieces of a contour, integrating more information than a classical pixel-wise formulation. Nevertheless, they remain semi-local, in order to be applied generally to most of the contours [3]. By learning their prior distributions, we avoid imposing mathematical constraints

that may decrease the detection efficiency of an algorithm, since it is impractical to define a mathematical model that captures every possible contour singularity.

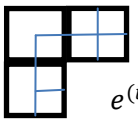
***Algorithm** 1: collecting the multiscaleedgelets*
***Input**: A shape database*
***Output**: Couple the edgelet sets* $\{\bar{B}_E, \bar{B}_e\}$
Begin
For s=1 . . . S_p *do*
Get an image I and a segmentation H at random
Extract a multiscaleedgelet$\{\bar{E}^s, \bar{e}^s\}$ *from* (I, H) *at random*
Center $\bar{E}^{(s)}$ *with respect to* $\bar{E}^{1.(s)}$
Return$\{\bar{E}^{(1)}, \ldots, \bar{E}^{(S_p)}\}, \{e^{(1)}, \ldots, \bar{e}^{(S_p)}\}$

Ofline

Prior



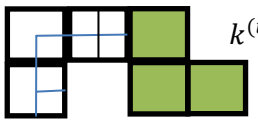
$E^{(i)}$ $p(E^{(i)})$



$e^{(i)}$ $p\left(e^{(i)}\middle|E^{(i)}\right)$

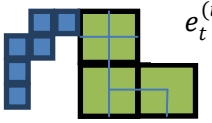
Transition

$k_{t-1}^{(i)}, e_{t-1}^{(i)}$



$k^{(i)}$ $p\left(E_t^{(i)}\middle|E_{t-1}^{(i)}, e_{t-1}^{(i)}\right)$

$E_t^{(i)}, e_{t-1}^{(i)}$



$e_t^{(i)}$ $p\left(e_t^{(i)}\middle|k_{t-1}^{(i)}, e_{t-1}^{(i)}\right)$

4. OFFLINE LEARNING: TRANSITION MODELS

We defined in the previous section a way to initialize edge-lets. Next, to randomly extract full contours with our tracking algorithm, we need to create a candidate multi-scale edge-let at a certain time t given the previous one at $t-1$. This is what the transition distributions $p(E_t \vee E_{t-1}, e_{t-1})$ and $p(e_t \vee E_{t,}e_{t-1})$ are designed to do. Both distributions areconditioned by e_{t-1}in order to assurance convexity from one time to another. Finally, in order to learn the distribution

$p(e_t \vee E_{t,}e_{t-1})$, we define $B_e^{-E^{(s)}}$ the set of the distinct elements of Be that are potential predecessors of $E^{(s)}$. Using the same shape data set as in Section 3.1, the approximation procedures of these two distributions are given in Algorithms 2 and 3. 3.3

Algorithm 1: Approximation of the transition distribution of an edgelet at the coarse scale

Input: Prior distribution, a shape database

Output: Approximation of $p(E_t|E_{t-1}, e_{t-1})$

begin

Online

Feature distribution:

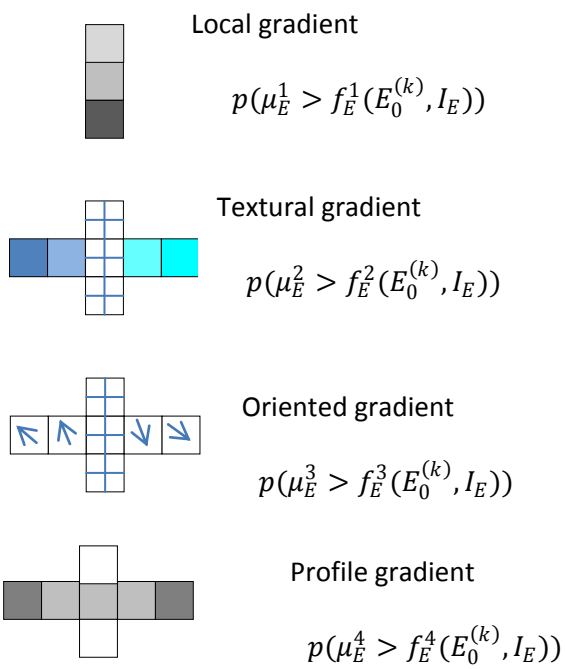


Fig. 1. The learning procedure is divided into two steps. The offline step estimates the prior and the transition distributions, which are used to generate samples in the contour tracking procedure. The online step is performed on the image to be tracked, and aims at learning: the feature distributions, in order to recognize the meaningful contours in the image; and the initialization distribution, in order to (re-)initialize the tracking in the contour detection procedure. (i) denotes a sample in the offline procedure, (k) denotes a sample in the online procedure.

for each $\left(E^{(s)}, e^{(r,s)}\right) \in B_E \times B_e^{E(s)}$ do

repeat

Get an image I and a segmentation H at random
Extract a multiscaleedgelet from (I, H) such that $(E_{t-1}, e_{t-1}) = \left(E^{(s)}, e^{(r,s)}\right)$ and let $E_t = \left(E^{(u)} \in B_E\right)$ be its successor at the coarse scale

Increment by $1/S_t$ the probability $p(E_t = E^{(u)}|E_{t-1} = E^{(s)}, e_{t-1} = e^{(r,s)})$
Until S_t times
Center $\bar{E}^{(s)}$ with respect to $\left(E^{(s)}, e^{(r,s)}\right)$ and let

return, $\left\{p\left(E_t^{(u)}\middle|E_{t-1}^{(s)}, e_{t-1}^{(r,s)}\right), \forall(u, s, r)\right\}$

5. ONLINE LEARNING: OBSERVATION MODEL

In this section, we define the observation model $p(Y,y|E_t$, $e_t$which measures the adequation between the data (x,y) and 1.actully, the prior distribution at the reference scale is $p(e,m|E) = p(e|E,m)\,p(m|E)$, with $m \in N$ the r.v. which denotes the length of the edgelet at the reference scale, $p(e|E,m) \triangleq p(e = (e^1,\dots,e^m)|E)$,and $p(m|E)$ a probability proportional to the number of edge lets at the reference scale of length m included in E . $p(e|E) \triangleq p(e|E,m)$ with e belonging to the set of all the possible edgelets of different lengths, the probability $p(m|E)$ is implicitly totalled , and m can thus be omitted. a multi-scale edgelet at a time t,(E_t,e_t):

$$P(Y,y|E_t,E_t) = p(Y|E_t,)\,p(y|e_t), \qquad (1)$$

where independence hypotheses have been assumed to simplify the estimation. Also, we consider a particular case in which the likelihoods $p(Y|E_t)$ and $p(y|e_t)$ are similarly defined [3]. Then, in order to lighten the content of the following section, we will only define the likelihood $p(y|e_t)$ and its associated features. The definitions at the coarse scale can simply be obtained by swapping et for et, the subscripts of e for e, and y for y.

6. LOCAL GRADIENT

This classical feature uses the 2×2 gradient norm $|\Delta I_e|$. The gradient feature f1 e is totaled along the edgelets e_τ:

$$f_e^1(e_t,I_e) = \emptyset\big((\nabla I_e(e_t^i)|)_{1\trianglelefteq i\trianglelefteq M}\big)$$

The tractability comes from the combination operator Φ. One can set $\Phi = \min, \Phi = \max$, or a weighted mean $\Phi(v^1,\dots\dots,v^M) = \sum_{i=1}^{M} W(i)v^i$, with $W:\{1,\dots\dots,M\} \to [0,1]$ a weighting function. Note that since the image I_e is multidimensional, we proceeds on each point the maximum gradient value among the different channels

7. TEXTURAL GRADIENT

The textural gradient feature goals at getting low reply values on texture locations, while getting high ones on object contours. For a point e_t^i of an edgelete$_t$, we consider its normal segment. The two sides of the normal segment of three consecutive points e_t^{i-1}; e_t^i; $e_t^{i+1}$are noted $\overleftarrow{n}(e_t^{i-1})$and $\vec{n}(e_t^i)$. In a texture, the intuition is that pixel values along the first segment should not really differ from the ones of the second segment. Let $h_T[a]$ f $\{h_T^r[a]\}_{r=1}^{R}1$ be the histogram of a set of pixels a, where r is the bin index of a histogram of length R. In the case of color images, the length R equals$R_T \times R_T \times R_T$, with R_T the number of bins by channel. Distances between pairs of histograms along normals of the curve are combined to form the textural gradient feature:

$$f_e^2(e_t,I_e) = \Psi\big((d_B(h_T[\overleftarrow{n}(e_t^i)],h_T[\vec{n}(e_t^i)])_{2\trianglelefteq i\trianglelefteq M-1}\big)$$

withΨ the fusion operator, and dB the Bhattacharyya distance between two histograms, i.e., $d_B(h[a],h[b])$ is the square root of $1-\sum_{r=1}^{n}\sqrt{h^r[a],h^r[b]}$. In order to reduce the spread of the histogram values, the widths of the bins are defined by the R_T −quantiles of each independent channel.

8. CONTOUR DETECTION BY TRACKING BASED ON PARTICLE FILTER

We defined in Section 3 several distributions that manipulate the edgelets$E_t$and e_t. In this section, we define the framework that handles these distributions by integrating them in a sequential Monte Carlo approach. Our goal is to estimate the distribution of the jointedgelets$X_{0:t} = (E_{0:t}, e_{0:t})$ conditioned by a set of jointobservations $z = (Y,y)$. Hence, at a time t, $e_{0:t}$ defines a contour map at the reference scale of $t+1$ edgelet elements. The estimation of the so-called trajectory distribution $p(x_{0:t}|z)$ is thus completed using a particle filtering technique that we present now.

Algorithm 5: Approximation of the initialization distribution at the coarse scale

Input: Prior and feature distribution, an image I_E Output: Output Approximation of $p(E_0|v)$

Begin

for$s = 1,\dots,S_{i_E}$ do

- Generate a string point $E_0^{1,(s)} \sim U|\Omega_E|$
- Generate the edgelet shape $E_0^{2:M,(s)}$approaching to its prior distribution$E_0^{2:M,(s)}|E_0^{1,(s)} \sim p(E_0^{1,(s)}|E^1$
- Compute the joint likelihood $w\left(E_0^{(s)}\right): w\left(E_0^{(s)}\right) \propto \prod_{j=1}^{J}\exp\left(\lambda_E^j P\left(\mu_E^j > f_E^j\left(E_0^{(s)}I_E\right)\right)\right)$ $s.t.\sum_{r=1}^{S_{i_E}} w\left(E_0^{(r)}\right) = 1$

return $\frac{1}{S_{i_E}}\sum_{s=1}^{S_{i_E}} w(E_0^{(s)})\,\delta_{E_0^{(s)}}^{E_0}$

9. ESTIMATING THE MULTI-OBJECT TRAJECTORY DISTRIBUTION

In this section, we study a special case of the Bayesian recursion, in which the observations are not indexed by time. 4.1.1 valuing the Trajectory Distribution Let$x_t \in \chi$ be the hidden state of a stochastic procedure at time t and $z \in Z$ be the measurement state. Under the Markovian hypothesis of the hidden states and the conditional independence hypothesis of the observations given the states, the trajectory distribution $p(x_{0:t}|z)$ is given by [4]:

$$p(x_{0:t}|z) \propto p(x_{0:t-1}|z)p(z|x_t)p(x_t|x_{t-1}) \qquad (2)$$

whose normalizing term is independent from $x_{0:t}$.

10. PARTITIONED SAMPLING

Estimating the posterior distribution $p(x_{0:t}|z)$ with the multi-scale nature of the state $x_{0:t} = (E_{0:t}, e_{0:t})$ may increase a problem. In fact, as described in Section 4.1.1, the particle filter makes use of an significance sampling procedure which involves the simulation and the weighting of the particles. This technique suffers from a expletive of the dimensionality: it has been shown in [15] that N^2 particles are necessary to reach the same level of estimation performance as when estimating a single scale with N particles. To alleviate this problem, MacCormick et al. proposed the Partitioned Sampling algorithm which decomposes the vector state by partitioning the state space, and then by handling one component, i.e., scale, at a time [15], [16]. We present here a basic version of this algorithm, adapted to our purpose.

First, we introduce the weighted resampling procedure. This transforms a particle set $\{X_{0:t}^{(n)}, w_t^{(n)}\}_{n=1}^{N}$ into another one $\{\widetilde{X}_{0:t}^{(n)}, w_t^{(n)}/g(X_{0:t}^{(n)})\}_{n=1}^{N}$ while keeping the distribution intact [27].The weighting function g is strictly positive and defined such that $\sum_{n=1}^{N} g(X_{n=1}^{(n)}) = 1$. The new particle set is obtained by simulating according to the empirical distribution defined by the weights $\{g(x_{0:t}^{(n)})\}_{n=1}^{N}$. We describe now the Partitioned

Sampling algorithm. From Section 3, we know that the edgelet propagations can be executed sequentially and that the possibility factorization permits to deal with each scale independently, hence each vector observation is related to one edgelet scale. By allowing for the state et first, the Partitioned Sampling algorithm consist in

1) Propagating the particles using the marginal importance function of et;
2) computing the weighting function such that $g(X_{0:t}^{(n)}) \propto P(Y|E_t^{(n)})P(E_t^{(n)}|E_{t-1}^{(n)}, e_{t-1}^{(n)})/q(E_t^{(n)}|E_{0:t-1}^{(n)}, e_{t-1}^{(n)}, Y)$;
3) Performing a weighted resampling procedure on the particle set $\{(E_{0:t}^{(n)}, e_{0:t-1}^{(n)}), w_t^{(n)}\}_{n=1}^{N}$,
4) Propagating the particles using the marginal importance function of e_t; and
5) Computing the particle weights $w_t^{(n)} \propto P(y|e_t^{(n)}) P(e_t^{(n)}|E_t^{(n)}, e_{t-1}^{(n)})/q(e_t^{(n)}|E_t^{(n)}, e_{t-1}^{(n)}, y$.

Defining the weighting function g in that way enables the generation of more samples for higher values of the marginal likelihood of e_t and greatly simplifies the computation in the last step of the Partitioned Sampling technique since the g term is present both at the numerator and denominator of the final particle weight computation.

11. PARTICLE FILTER CONTOUR DETECTION ALGORITHM

In this section, we describe our particle filter process devoted to the contour detection task. We present $c_t \in \{0,1\}$ a binary random variable of jump: if $c_t = 0$, the tracking of contour at time t goes on, else, i.e., if $c_t = 1$, the edgelet is initialized to a new contour. This is useful when the tracking of the current contour is lost or finished. The unseen state x_t is then composed of an edgelet et at the coarse scale, an edgelet et at the reference scale, and a jump variable c_t, yielding to $x_t = (E_t, e_t, c_t)$. A particle filter requires the definition of four distributions: a prior $p(x_0)$, to initialize particles; an importance function $q(x_t|x_{0:t-1}, z)$, to predict a particle at time t given the past states and observations; a trajectory prediction $p(x_t|x_{0:t-1})$, to describe the prior evolution of a particle at time t given the past states; and a likelihood $p(z|x_t)$, to weight the particles using the last known measure. While the prior and the probability are learned in Section 3, the importance function and the transition want to be defined. The tracking way is summarized in Fig. 2.

12. TRANSITION

First, we express the trajectory transition $p(x_t|x_{t-1})$such that the edgelet distribution rest on the jump variable. Also,we consider the edgelet transition at the coarse scale first, and make the edgelet transition at the reference scale dependent from the coarse scale edgelet, as proposed in Section 3:

$$P(X_t|X_{t-1}) = p(E_t|E_{t-1}, e_{t-1}, c_t)\, p(e_t|E_t, e_{t-1}, c_t)\, p(c_t) \quad (3)$$

The jump variable c_t is assumed independent from c_{t-1} by simplicity. This way, it is related to the length of a contour.

Let $p(c_t = 1) = \beta$ be the probability of jump. The edgelet transition of the coarse scale is a mixture of the prior and the transition distributions learned in Sections 3.1 and 3.2, respectively. It depends on the value of the switching variable ct in such a way that if it designates a jump, then the prior distribution is considered. Otherwise, it consists in a transition:

$$P(E_t|E_{t-1}, e_{t-1}, c_t) = c_t\, p(E_t^{2:M}|E_t^1)p(E_t^1) + (1-c_t)\, p(E_t|E_{t-1}, e_{t-1}), \quad (4)$$

13. IMPORTANCE FUNCTION

We now consider the importance function. Its role is to generate the particles. It is possible to set$q(x_t|x_{0:t-1})$ in order to propagate the particles using the previoustransition. However, a more sophisticated design can drastically improve the estimation efficiency by reducing the variance of the particle weights [14], [17].

14. STOPPING CRITERION

Although it is popular to perform tracking tasks, the particle filter does not embed a natural stopping procedure. In most of the cases, defining it is clearly not obvious. Here, we can take benefit of the feature tail distributions, which provide robust statistics of the data to be retrieved. We define the stopping criterion such that it depends on the meaningfulness of the last extracted contours. To this end, the probability of jump using the meaningfulness of an edgelet, and this probability grows with time. Hence, we stop the detection when the proportion of jumps reaches a fixed threshold:

$$\frac{1}{KN}\sum_{k=1}^{K}\sum_{n=1}^{N} c_{t-k+1}^{(n)} \geq Y. \quad (5)$$

Since it relies on the number of jump operated in the K last steps, this criterion also depends on the length of the retrieved contours, meaning that the repetition of small contour detections likely indicates that the detection is over.

15. DIVERSITY

The resampling technique does not alter the posterior distribution but impacts on the diversity of the particles, especially for the past states. In practice, this means that most of the particles share the same trajectory, which may degrade the quality of the estimator. To alleviate this effect, we propose to divide the N particles into L independent particle filters, leading to the following final posterior distribution:

$$P(X_{0:t}|Y,y) = \frac{1}{L}\sum_{l=1}^{L} p\,(X_{0:t,l}|Y,y). \quad (6)$$

Each particle filter approximates the trajectory distribution using $N_L = N/L$ particles. More elaborated techniques [28] aim at reducing the particle impoverishment effect, and at the cost of an increase of the algorithm computational complexity, might also be functional to our model.

16. CONTOUR DETECTOR

We consider here the contour detector at the reference scale, but the definition at the coarse scale can be equally obtained. The soft contour detector is an image $O: \Omega_e \rightarrow [0,1]$ with $O(z)$ the confidence value that the pixel z belongs to a contour. This is computed by an average of the estimations given by the L particle filters:

$$\forall z\ \Omega_e,\ O(z) = \frac{1}{L}\sum_{l=1}^{L} \max\ w_{tl,l}^{(n)}\ 1\ e_{0:tl,l}^{(n)}\ (z), \quad (7)$$

withtl the last step performed by the lth particle filter. An optional non-maximum suppression step may then be employed to produce thin contours [9], [18].

For interactive segmentation, [26] uses local constraints to achieve boundary-region consistency, In contrast, the winding number constraints used through our approach are global constraints. The winding number model not only leads to a smaller number of constraints but also delivers a clearer understanding about region contour interface. Winding numbers have recently been used for extracting volume representations [22]. Winding number is one of basic concepts in topology [21], [25]. Unfortunately, it is often confused with

another topological concept, rotation index, which is defined as the total rotation angle of tangent if one travels along a curve [27]. Rotation index is often called rotation number, even winding number in the literature [24]. In computer vision field, rotation index has been used for confirming contour topology in [20]. To the best of our knowledge, winding number has not been applied in perceptual grouping contexts. The winding number of a loop abouta point is demarcated as the number oftimes

the loop travels around the point in counter-clockwise direction. Here, the loop does necessity to be simple, i.e. it is allowed to intersect with itself. As shown in [21], any connected component of the plane has a constant and integral winding number. Therefore, we can assign a single winding number to a connected region. We further extend the winding number concept to a plane with a finite number of curves. The

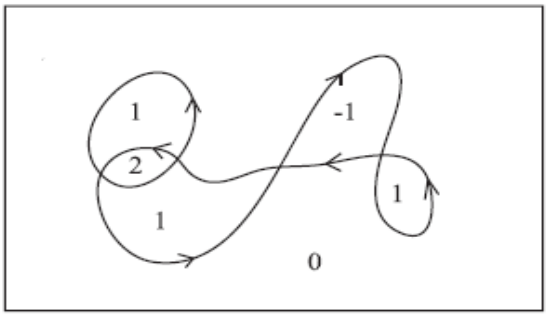


Fig. 2. Winding numbers induced by closed contours [8].

winding number of a region is defined as the sum of the winding numbers of all loops. Fig. 1 shows the winding numbers of different regions induced by two closed curves. Take the region labeled with a winding number 2 for example; both two curves travel around this region in counter-clockwise direction, and each curve induce a winding number 1.

17. REGION-BOUNDARY CONSISTENT CONTOUR EXTRACTION

An image can have many edges caused by various factors such as light, texture or color change. This paper focuses

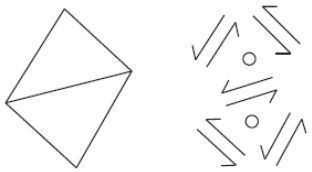


Fig. 3.Examples of region and edge hypotheses. The left shows two triangular regions and their edges. The right shows the edge and region hypotheses extracted from an image. Two circles denote two region variables; an arrow denotes an edge variable [8]

on detecting salient ones which bound objects in the scene. Psychophysical experiments show that human can effectively differentiate them from texture edges [23]. It is important to note that our winding number formulation is not restricted to the superpixel setup, but applies to general boundary-region graphs as well. An illustration of this setup is shown in Fig. 4. These two triangles give rise to two atomregions and eight directed edges. Let the variable $x = \{x_i \,|i = 1 \dots N_r\}$ denote the labels of Nr atom regions, and $y = \{y_j \,|\, j = 1 \dots N_e\}$ denote the binary labels of Ne edges. The edge label space is denoted as $Y = \{0,1\}^{N_e}$. The label space of all region variables is denoted as X. Although the winding number concept is potentially applicable for multiple-label segmentation problems, our work focuses on the binary-label figure/ground segmentation problem. In other words, we let $X = \{0,1\}^{N_r}$.

The energy function E(x, y) represents various cues, such as figure-background contrast and contour smoothness, depending on applications. Our main contribution is constraints which ensure the topological correctness ofsolutions. Together, the basic energy minimization problem has the following form:

$$\min_{x,y} E(x,y) \quad (8)$$

$$s.t.\ \emptyset_W(x,y) = 0 \quad (9)$$

$$\emptyset_C(x,y) = 0 \quad (10)$$

$$x \in \chi,\ y \in \Upsilon \quad (11)$$

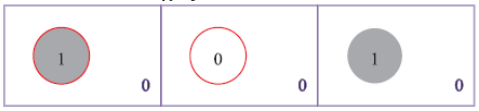


Fig. 4. The left image shows a consistent region and edge configuration. The middle figure is not consistent because two regions separated by a contour have the same label. The right figure is not consistent because two adjacent regions with different labels are not separated by any contour.

In order to guarantee the topological validity of labels, two sets of constraints are set up. The constraint set $\emptyset_c$is the edge continuity constraints, as follows:

$$\sum_{i=j_{in}} y_i = \sum_{i=j_{out}} y_i, \forall_j \in V, \quad (12)$$

where j is from the vertex index set V ; $j_{in}$and j_{out} denote edges indices heading into and moving out of the vertex j , respectively. These constraints say that the net flow at every vertex is zero. For a flow network without source and sink, all the flows can be decomposed into a set of cycles. Therefore, our method aims to extract a set of closed curves as contours. The constraints $\emptyset_w$ represent the winding number constraints which confirm the consistency of region and contour labels [8]. The specific consistency condition used in this paper is that:

If an edge is active, its adjacent (i.e. incident) regions must have different region labels; if two adjacent regions have different labels, one of the edge elements in-between must be active. This condition guarantees that every edge must be part of a region boundary, and every region is enclosed by contours. Fig. 5 shows one correct labeling and two incorrect cases that violate the condition. At first glance, this condition does not have anything to do with winding numbers. Instead, it can be formulated as follows:

$$|y_m - y_n| = 1(x_i - x_j = 0), \forall (i,j)G,$$

where xi and x j denote the labels of two adjacent regions, and G denotes the set of indices of adjacent regions. Variables y_m and y_n denote two conjugate edges separating these two regions. The function $1(\cdot)$ equals one if its argument is true, and equals zero otherwise. Although constraints (8) are sufficient for the consistency condition, they are expensive to implement due to their non-linearity. Even if the energy function E is convex, the whole energy minimization problem will generally turn out to be non-convex with these nonlinear constraints. Next section will show that this condition is guaranteed by linear constraints based on winding numbers. We realize that the winding number concept, from topological study, provides an elegant and effective means to

parameterize the region-contour consistency condition for image segmentation. We have reached the following key step called winding numbertechnique: The label of a region can be identified by its winding number induced by contours.

The winding number constraints guarantee the consistency between region labels and contour labels. The benefit of such winding number scheme also lies in that: it leads to a small number of linear constraints. This can be made evident by examining the fast computation procedure of winding number computation in Eq (10). To adapt it to our problem, we assume that every image is enclosed by a rectangular border, any region outside of the image has a label zero. Therefore, the winding number of region i is:

$$x_i = \sum_{\alpha \epsilon P_i} y_\alpha - \sum_{\beta \epsilon N_i} y_\beta , \forall_i \qquad (13)$$

where Pi and Ni are the edges crossing from right to left, and edges crossing from left to right, respectively. Eq (15) for all atom regions together can be represented as the following winding number constraint, denoted as $\emptyset_w$ in Eq (12):

$$x = My, \qquad (14)$$

where M is a matrix whose entries are 0, 1, or −1. Take the i-th row of M for example, $M(i,\alpha) = 1, \forall \alpha \in Pi$; $M(i,\beta) = -1, \forall \beta \in Ni$; and the rest of entries are zero. The number of these constraints is the similar as the number of atom regions.

Proposition 1: For any segmentation in which region labels can only be zero or one, there always exists a set of oriented boundaries such that the regional labels equal the winding numbers induced by these boundaries.

Proof: First of all, we assume that edges do not intersect with each other and each edge is only adjacent to two regions.

Fig.5. The left shows an image in BSDS dataset. The right shows paths by which the winding number of the superpixels (red dots) are calculated [8].

If this assumption is not valid, the edges can be divided into smaller segments to satisfy the guess. Then, for an atom region whose label is one, we set a cycle of its adjacent edges in counter clockwise direction to be active. This cycle of edges will bring a winding number one to this region, and a winding number zero to other regions. Since edges are not shared by more than two regions, this operation can be done to every atom region without conflict. Consequently, every atom region in the foreground has a winding number one. Last, conjugate edges which are both active can be removed without affecting winding numbers of any region. Therefore, the final contour map is consistent with the given segmentation.

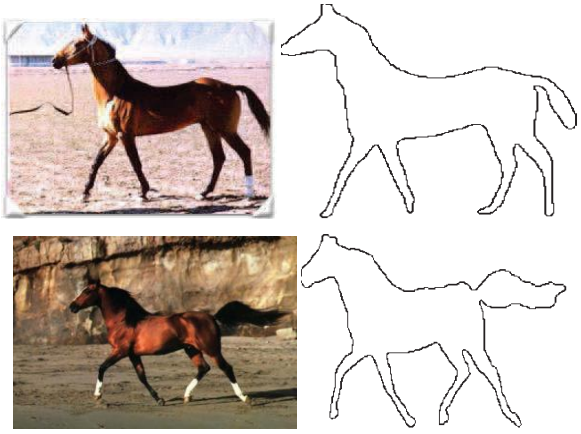


Fig. 6. Sample images from the Weizmann horse dataset. The horse images are in the first and third columns, and the corresponding ground truth contours are in the second and forth columns.

18. EXPERIMENTS

Full results present three measures. The optimal image scale (OIS) is the F-Measure score obtained using the optimal threshold on each image. The last measure is the average precision (AP) and corresponds to the area under the precision-recall curves of Fig.

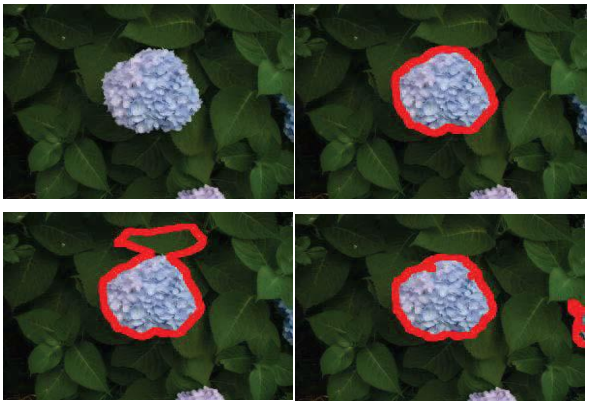


Fig. 7.Experimental results. Starting points are colored in purple. (Left) Forbidden points region tool. The contour detector runs without any interaction, unless the algorithm goes wrong, in which case forbidden points regions are drawn (in red). (Middle) Control points region tool.

The user manually defines a rough contour and the algorithm aims at extracting an accurate contour path from it. First, all the pixels that do not belong to the rough contour are considered forbidden. Then, taking advantage of the user interaction, we consider that the rough contour path is ordered and follows the true contour. Thus, intermediate control points regions are automatically spread at regular intervals within the rough contour. Using the rough contour tool has several advantages. First, the extraction process is efficient since the space problem is constrained. Second, it is certainly more convenient for the user since, unlike the first two tools that often result in a trial-and-error procedure; this one only needs one simple fast interaction.

19. CONCLUSIONS

The multiscale particle filters method to track contours in complex natural images. The basic division of this model is a pair of edgelets, i.e., sets of linked pixels demarcated at two

scales, which naturally embed semi-local information. While the winding number based method combining the regions and contour representations efficiently. This ideal is simple and interesting, as it only contains a compact set of linear controls to ensure the consistency of both representations. As an application of this method, region similarity cues and region-based user interface are added into our ratio-based contour detection context, and lead to improved results. In upcoming, the more refined design of region/contour cues could benefit to extract contours of complex objects.

20. REFERENCES

[1] B. Wu and R. Nevatia, "Detection of Multiple, Partially Occluded Humans in a Single Image by Bayesian Combination of Edgelet Part Detectors" Proc. IEEE 10th Int'l Conf. Computer Vision (ICCV), vol. 1, pp. 90-97, 2005.

[2] N. Widynski and M. Mignotte, "A Contrario Edge Detection with Edgelets," Proc. IEEE Int'l Conf. Signal and Image Processing Applications (ICSIPA), pp. 421-426, 2011.

[3] Nicolas Widynski and Max Mignotte "A MultiScale Particle Filter Framework for Contour Detection" IEEE Transactions on Pattern Analysis And Machine Intelligence, Vol. 36, No. 10, October 2014.

[4] P. Perez, A. Blake, and M. Gangnet, "Jetstream: Probabilistic Contour Extraction with Particles," Proc. IEEE Eighth Int'l Conf. Computer Vision (ICCV), vol. 2, pp. 524-531, 2001.

[5] C. Florin, N. Paragios, and J. Williams, "Globally Optimal Active Contours, Sequential Monte Carlo and On-Line Learning for Vessel Segmentation," Proc. Ninth European Conf. Computer Vision (ECCV), pp. 476-489, 2006.

[6] D. Lesage, E.D. Angelini, I. Bloch, and G. Funka-Lea, "Medial- Based Bayesian Tracking for Vascular Segmentation: Application to Coronary Arteries in 3D CT Angiography," Proc. IEEE Fifth Int'l Symp. Biomedical Imaging: From Nano to Micro (ISBI), pp. 268-271, 2008.

[7] N. Widynski and M. Mignotte, "A Particle Filter Framework for Contour Detection," Proc. 12th European Conf. Computer Vision (ECCV), pp. 780-794, 2012.

[8] Y. Ming, H. Li, and X. He, "Winding Number Constrained Contour Detection" IEEE Transactions on Image Processing, Vol. 24, No. 1, January 2015.

[9] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik, "Contour Detection and Hierarchical Image Segmentation," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. 33, no. 5, pp. 898-916, May 2011.

[10] X. Ren, "Multi-Scale Improves Boundary Detection in Natural Images," Proc. 10th European Conf. Computer Vision (ECCV), pp. 533-545, Jan. 2008.

[11] M. Maire, P. Arbel_aez, C. Fowlkes, and J. Malik, "Using Contours to Detect and Localize Junctions in Natural Images," Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), pp. 1-8, 2008.

[12] N. Dalal and B. Triggs, "Histograms of Oriented Gradients for Human Detection," Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR), vol. 1, pp. 886-893, 2005.

[13] J. Sun, Z. Xu, and H.-Y. Shum, "Gradient Profile Prior and Its Applications in Image Super-Resolution and Enhancement," IEEE Trans. Image Processing, vol. 20, no. 6, pp. 1529-1542, June 2011.

[14] Sequential Monte Carlo Methods in Practice, A. Doucet, N. De Freitas, and N. Gordon, eds., Springer, 2001.

[15] J. MacCormick and M. Isard, "Partitioned Sampling, Articulated Objects, and Interface-Quality Hand Tracking," Proc. Sixth European Conf. Computer Vision (ECCV), pp. 3-19, 2000.

[16] J. MacCormick and A. Blake, "A Probabilistic Exclusion Principle for Tracking Multiple Objects," Int'l J. Computer Vision, vol. 39, no. 1, pp. 57-71, 2000.

[17] Z. Chen, "Bayesian Filtering: From Kalman Filters to Particle Filters, and Beyond," technical report, McMaster Univ., 2003..

[18] J. Canny, "A Computational Approach to Edge Detection," IEEE Trans. Pattern Analysis and Machine Intelligence, vol. PAMI-8, no. 6, pp. 679-698, Nov. 1986.

[19] B. Andres, J. H. Kappes, T. Beier, U. Köthe, and F. A. Hamprecht, "Probabilistic image segmentation with closedness constraints," in Proc. IEEE ICCV, Nov. 2011, pp. 2611–2618.

[20] J. H. Elder and S. W. Zucker, "Computing contour closure," in Proc. 4th ECCV, 1996, pp. 399–412.

[21] J. Gallier and D. Xu, "The fundamental group, orientability," in A Guide to the Classification Theorem for Compact Surfaces (Geometry and Computing), vol. 9. New York, NY, USA: Springer-Verlag, 2013.

[22] A. Jacobson, L. Kavan, and O. Sorkine-Hornung, "Robust insideoutside segmentation using generalized winding numbers," in Proc. ACM SIGGRAPH, 2013, vol. 32, no. 4, p. 33.

[23] I. Kovács and B. Julesz, "A closed curve is much more than an incomplete one: Effect of closure in figure-ground segmentation," Proc. Nat. Acad. Sci. USA, vol. 90, no. 16, pp. 7495–7497, 1993.

[24] M. McIntyre and G. Cairns, "A new formula for winding number," GeometriaeDedicata, vol. 46, no. 2, pp. 149–159, 1993.

[25] T. Needham, Visual Complex Analysis. New York, NY, USA: Oxford Univ. Press, Feb. 1999.

[26] T. Schoenemann, F. Kahl, S. Masnou, and D. Cremers, "A linear framework for region-based image segmentation and inpainting involving curvature penalization," Int. J. Comput. Vis., vol. 99, no. 1, pp. 53–68, 2012.

[27] H. Whitney, "On regular closed curves in the plane," Compos. Math., vol. 4, pp. 276–284, 1937.

[28] A. Doucet, M. Briers, and S. S_en_ecal, "Efficient Block Sampling Strategies for Sequential Monte Carlo Methods," J. Computational and Graphical Statistics, vol. 15, no. 3, pp. 693-711, 2006.