Connected Contours: a New Contour Completion Model that Respects the Closure Effect

Yansheng Ming
Australian National University

Hongdong Li
Australian National University

Xuming He
NICTA, Australia

Abstract

Contour Completion plays an important role in visual perception, where the goal is to group fragmented low-level edge elements into perceptually coherent and salient contours. This process is often considered as guided by some middle-level Gestalt principles. Most existing methods for contour completion have focused on utilizing rather local *Gestalt laws such as good-continuity and proximity. In contrast, much fewer methods have addressed the* global *contour closure effect, despite that many psychological evidences have shown the usefulness of closure in perceptual grouping.*

This paper proposes a novel higher-order CRF model to address the contour closure effect, through local connectedness *approximation. This leads to a simplified problem structure, where the higher-order inference can be formulated as an integer linear program (ILP) and solved by an efficient cutting-plane variant. Tested on the BSDS benchmark, our method achieves a comparable precision-recall performance, a superior contour grouping ability (measured by Rand index), and more visually pleasing results, compared with existing methods.*

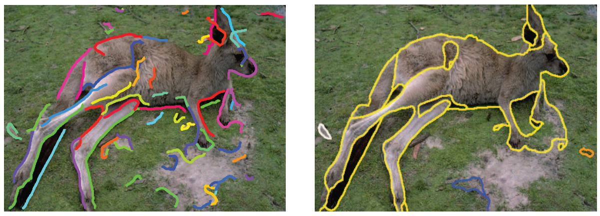

Figure 1. The goal of this paper is to extract perceptually-salient and closed contours from an image. **Left**: result of the *contour-cut* algorithm [10]. **Right**: this paper's result.

1. Introduction

Extracting perceptually-salient contours from images is a fundamentally important task for vision systems [16]. While much progress has been made recently, it remains a challenging task to automatically extract image contours with a computer such that the obtained results are consistent with human perception [2]. On the other hand, human vision system seems to be able to extract salient image contours effortlessly.

It is well recognized that human vision system has made substantial use of some mid-level perceptual grouping rules (known as Gestalt principles), such as "proximity", "good continuity" and "closure" *etc*., during contour grouping. A particular challenge in using Gestalt principles is how to effectively enforce these by-and-large non-local interactions in a coherent framework and to generate contours as human perceives.

Many computational models for contour completion have been proposed in the past decades, for example,[6, 8, 7, 13, 24, 10], to cite but a few. Most of the existing methods, however, only utilize a very small subset of the Gestalt rules, and primarily the *good-continuation* and *proximity* rules since they involve mostly relatively-local interactions. In contrast, fewer work have elaborated on the global effect of *contour-closure*.

The *closure effect* has been observed by numerous psychological studies (see [16] and references therein), that have confirmed the usefulness and importance of the closure effect in perceptual organization. Kovacs and Julesz once claimed [12]: *"a closed curve is much more than an incomplete one"*. Our work is aimed at addressing the close effect. In particular, we notice that, most existing methods (for computer contour completion) either produce isolated or disconnected curve segments, or fail to handle cases of occlusion (*e.g*. [13]). While some other methods do emphasize the closure effect, they often do so at individual contour level ([23]). Unlike the relatively-local Gestalt principles, the closure principle is "more global", making it not an easy task to enforce such condition.

In this paper, we propose a new higher-order Conditional Random Field (CRF) model for solving the contour grouping problem, based on a novel representation of image contours and their interactions. To approximate the closure effect, we introduce a set of *contour connectedness* condition, represented as some hard inequality constraints, in addition to a set of soft constraints capturing good-continuity and proximity *etc*. represented as potential energy terms.

Although our CRF model relies on higher-order cliques, it however does not suffer from solving otherwise a generally NP-hard inference task. By carefully exploiting the special connectedness constraints that we propose, we have simplified the overall structure of the higher-order potential function, reducing it to linear order subjecting to global constraints, which consequently enables us to formulate the CRF inference as an ILP (integer linear programming) problem which can be solved rather efficiently.

We have tested our method on both synthetic data and real images. Experiments show that it can extract multiple connected contours without loose end. On BSDS300 benchmark [14], our model achieves higher performance than previous CRF-based models, showing that the proposed method is able to complete local edges without sacrificing the accuracy. Moreover, the extracted contours are not only connected, but also clean and visually pleasing, perhaps suggesting that the results close to human's perception.

1.1. Related work

Learning-based methods have been used recently for low-level image edge detection, which achieve improved accuracy, and are getting closer to human's perception (*e.g*. [14, 17, 11]). Based on these local detectors, some mid-level contour grouping models have been proposed to improve the detection precision [25, 10, 2]. However, the contour topology is either overlooked or left to post-processing by these models. On the other hand, important attempts have been made to exploit the Gestalt grouping principles and to ensure the contour closure, *e.g*. [6, 13, 23, 24, 20]. However, their contour models are often coded for isolated curves, and not very suitable for cases with occlusion. Some of them do handle multiple contours, but via an iterative procedure in a greedy way(*e.g*. [13]). In paper [1], contour closedness is explicitly enforced in the domain of edgelets as the boundary of small regions (*i.e*. superpixels). However, our model is purely contour-based, an we deliberately avoid to use any region segmentation information such as used by the GPb or GPb-owt-ucm [2]. There are other methods prefer closed contours to open ones, *e.g*. [19] and [11], but they do so in a "soft" way, by *e.g*. introducing penalty to open curves whenever found. In contrast, our model allows a joint optimization over all contours simultaneously, subject to hard connectedness constraints.

This paper is organized as follows. In the next section (*i.e*. Section-2) we describe how to construct our new CRF model, and how to use it to represent multiple image contours. Section-3 explains the CRF's potential function design. Section-4 is devoted to the proposed inference algorithm. The final two sections are experiments and conclusion.

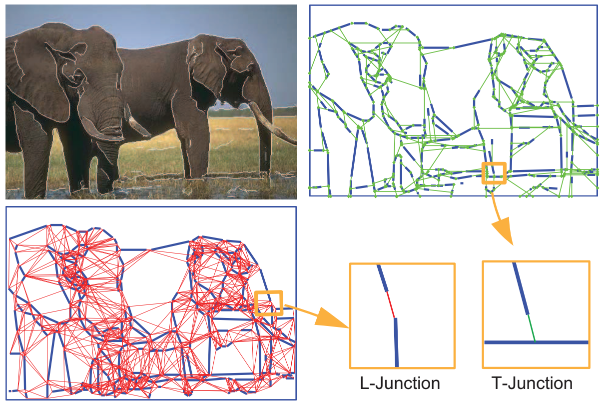

Figure 2. **Top left**: The original image overlayed with its thresholded Pb edges. **Top right**:The proposed T-junction completion edgelets shown in green, and the observed Pb gradient edgelets shown in blue. **Bottom left**:The proposed L-junction completion edgelets are shown in red. **Bottom right**: Zoomed-in view of the completion edgelets: L-junction(left) and T-junction(right). Best viewed in color.

2. Modeling multiple contours

To facilitate contour completion, we need to design a proper representation that allows mid-, and long-range interactions among local edge elements. For this purpose, we first construct a graph of boundary segments based on the output of local boundary detector, such as the Pb detector in [14]. On this graph, we then propose a higher-order CRF that integrates local evidences with global constraints based on the Gestalt properties of contours at a scene level.

2.1. Boundary segment graph

Our graph is built in two stages. We first form a set of short boundary segments using Pb detector followed by a line fitting process, as done in [19]. We refer to those line segments as the *gradient edgelets* or G-edgelet in short. Normally, there are many gaps among the G-edgelets, due to occlusions and miss-detections. So at the second stage, we introduce two new types of virtual *completion edgelets* (or C-edgelet in short) aiming to fill in the gaps and hence complete the contour.

In more details, we first shorten each gradient edgelet by several pixels at its two endpoints such that it does not touch other gradient edgelets. The first type of C-edgelets, called as the **L-junction** edgelet, is proposed to link two neighboring gradient edgelets accounting for the good-continuation principle. The second type of C-edgelets, called as the **T-junction** edgelets, is used to capture the occlusion relationship between contours. A T-junction edgelet is placed between an endpoint of a gradient edgelet and another gradient edgelet if the extension of the former intersects the latter without crossing other gradient edgelets. See Figure-2 for an illustration of gradient and completion edgelets. For consistency, we treat image borders as gradient edgelets.

To build a graph of boundary segments, we view each edgelet as a graph node (*i.e*. graph vertex). We use the same connectivity as we propose the completion edgelets. Note that in our graph the neighboring C-edgelet and G-edgelet always appear alternatingly. Compared with the CDT (Constrained Delaunay Triangulation) graph proposed in [19], our graph model accepts higher order of connectivity, and it also explicitly encodes richer types of junction relations (such as occluding/occluded) among contours. Moreover, we have experimentally confirmed that the proposed graph is able to recall 95% of the ground-truth boundaries on the BSDS300 dataset.

2.2. A higher-order CRF for contours

Based on the graph of boundary segments in Section 2.1, we build a CRF model to capture both local properties of individual contours, and the global interactions between neighboring contours. In particular, our model is focused on four aspects of contour properties, including 1) local image contrast; 2) smoothness and good continuity; 3) contour closure, and 4) overall model-complexity.

We formulate our model mathematically as follows. Let an image I have a boundary segment graph $G = (V, E)$ with edgelets as its node set V. We associate a binary label variable with each edgelet, and denote the label of the entire image as $\mathbf{Y} = \{y_i \in \{0,1\}, \forall i \in V\}$. The edgelet variables are divided into two sets: the gradient edgelets $\mathbf{Y}^g = \{y_i, i \in V_g\}$ and the completion edgelets $\mathbf{Y}^c = \{y_i, i \in V_c\}$. When we need to differentiate two types of completion edges, we use V_c^l and V_c^t to denote the sets of L-junction edgelets and T-junction edgelets, respectively. Each edgelet is connected to neighboring edgelets at the vicinity of its two endpoints. To capture the "good continuity" and "closure" rules, we need to consider both near-range and longer-range interactions between the edgelets, by introducing the following three types of cliques in the graph:

1. For every connected edgelet pair, we have a "pairwise clique", and denote the set of pairwise cliques as C^P;
2. We assign every completion edgelet (no matter it is L-type or T-type) a triple-node clique that includes itself and two neighboring gradient edgelets, and this type of clique is called "junction clique", and the entire set of such junction-cliques is denoted as C^J;
3. At an endpoint of a given gradient edgelet, all the edgelets connecting to this endpoint induce a higher-order clique, and we refer to the set of such higher-order cliques as C^H.

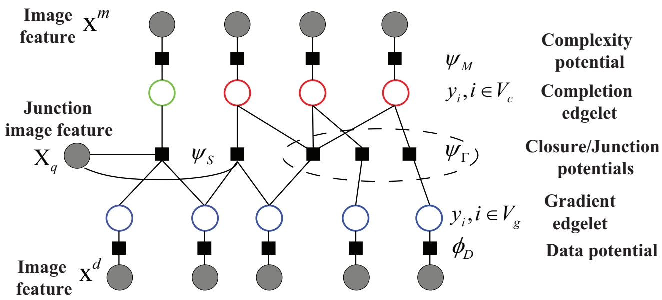

Figure 3. The factor graph representation of our CRF model. The circle node represents represent the variables in the model and the squares are potential functions (*i.e*. factors). The gradient edgelets are shown in blue and the completion edgelets are in red or green. Some connections are not shown for clarity and see text for details.

Our CRF defines a joint distribution of the labels $\mathbf{Y}$ given the input observation $\mathbf{X}$, denoted as $P_{\mathbf{Y}|\mathbf{X}}$ for short, which includes four types of potential functions,

$$P_{\mathbf{Y}|\mathbf{X}} = \frac{1}{Z_{\mathbf{X}}} \exp\Big\{ -\Big(\sum_{i \in V_g} \phi_D\left(y_i, \mathbf{x}_i^d\right) + \sum_{q \in C^J} \psi_J\left(\mathbf{Y}_q, \mathbf{X}_q\right) + \sum_{q \in C^P \cup C^H} \psi_\Gamma\left(\mathbf{Y}_q\right) + \sum_{i \in V_c} \psi_M\left(y_i, \mathbf{x}_i^m\right) \Big) \Big\}, \tag{1}$$

where $Z_{\mathbf{X}}$ is the partition function, and the potential functions ϕ_D (data term), ψ_J (junction term), ψ_Γ (global closure/connectedness effect term) and ψ_M (model-complexity term) are used to describe different aspects of desirable contour properties. We use $\mathbf{Y}_q$ to represent the label variables associated with clique q. A *factor graph* representation of the CRF model is shown in Figure 3.

3. Design of potential functions

3.1. Unary data term ϕ_D

The unary potential $\phi_D\left(y_i, \mathbf{x}_i^d\right)$ computes the likelihood score that the ith edgelet lies on a contour based on the output of local Pb detector. The data term is defined as a linear function of a boundary feature vector $\mathbf{x}_i^d$:

$$\phi_D\left(y_i, \mathbf{x}_i^d\right) = \alpha \mathbf{w}_d^T \mathbf{x}_i^d y_i, \tag{2}$$

where $\mathbf{w}_d$ is the weight for image features and α is an overall weight for the data term. In this work, we uses the lengths of edgelet l_i, the logarithm of the average Pb value of edgelet Pb_i, and their product as input features.

	image features	description
$\mathbf{X}_q^l(1)$	l_i	effective distance(smooth)
$\mathbf{X}_q^l(2)$	$\tilde{l}_i$	effective distance (corner)
$\mathbf{X}_q^l(3)$	$a_i(\theta_{i,j}+\theta_{i,k})^2$	angular completion
$\mathbf{X}_q^l(4)$	$b_i(\theta_{i,j}-\theta_{i,k})^2$	angular completion
$\mathbf{X}_q^t(1)$	l_i	effective distance
$\mathbf{X}_q^t(2)$	l_i/l_j	relative distance
$\mathbf{X}_q^t(3)$	$\frac{\min(l_k^a,l_k^b)}{l_k^a+l_k^b}$	intersection position
$\mathbf{X}_q^t(4)$	$(\theta_{i,k}-\pi/2)^2$	intersection angle
$\mathbf{x}_i^d(1)$	l_i	length of edgelets
$\mathbf{x}_i^d(2)$	$\log(Pb_i)$	average log(Pb) response
$\mathbf{x}_i^d(3)$	$l_i\log(Pb_i)$	sum of log(Pb) response

Table 1. Summary of image features used in the junction potential functions (c.f. Fig-4). For completeness, we also include the unary data term features $\mathbf{x}_i^d$.

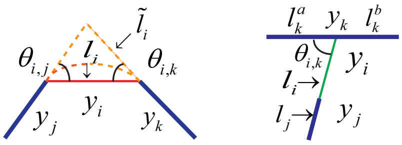

Figure 4. Two types of completion edgelet. **Left**: an L-junction edgelet and its image feature description; **Right**: a T-junction edgelet and its image feature description.

3.2. Junction potential ψ_J

For every completion edgelet $y_i, i \in V_c$ and the associated triple-node clique $q \in C^J$, we define a junction potential $\psi_J(\mathbf{Y}_q, \mathbf{X}_q)$ to encode the continuity property. For L-junctions, we design an L-potential to impose the principle of good-continuation; For the T-junctions, we design a T-potential to express the likelihood of the occluding/occluded relationships. For both cases, we define an image-dependent triplet potential function involving both of the central completion edgelet y_i and its two direct neighbors $\{y_j, y_k\}$, as shown below (and in fig-4):

$$\psi_J(\mathbf{Y}_q, \mathbf{X}_q) = \psi_J(y_i, y_j, y_k, \mathbf{X}_q) = \mathbf{w}_J^T \mathbf{X}_q y_i y_j y_k, \quad (3)$$

where $\mathbf{w}_J$ is the weighting coefficient for the corresponding junction feature vector $\mathbf{X}_q$ which is extracted from the neighborhood of clique q. Note that this potential assigns a score of $\mathbf{w}_J^T\mathbf{X}_q$ to the case that the whole triplet is on, and 0 otherwise.

The set of image feature vectors that we are using in this work is summarized in Table-1 (for notation please refer to Figure-4). For L-junction case, we denote the features by $\mathbf{X}_q^l$, and for T-junction case we use $\mathbf{X}_q^t$. This special design is inspired by [21].

3.3. Contour closure potential ψ_Γ

The contour closure effect is important to visual perception, but is also difficult to capture by local potential functions. In this paper we *approximate* the global closure effect by a sequence of (relatively local) *connectedness constraints*. At an endpoint of an edgelet we can identify two types of such connectedness constraints, each of them is formulated as a set of linear inequalities given follows:

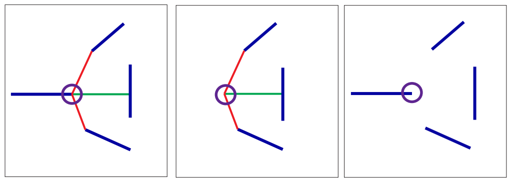

Figure 5. Examples of valid/invalid configurations w.r.t. the contour closure potential. Blue:Gradient edge; Red/Green: Completion edge. **Left**: A valid configuration satisfies the closure potential. **Middle**: A configuration that violates the completion constraint (*i.e.* Eq-(4)). **Right**: A configuration that violates the extension constraint (*i.e.* Eq-(5)).

(1) Completion Constraint ensures that no completion edgelet can turn on without its neighboring gradient edgelet being on. That is, at either endpoint of a completion edgelet $y_i, i \in V_c$, its neighboring gradient edgelet $y_j, j \in V_g$ should satisfy the inequality,

$$y_i \le y_j, \quad \forall i \in V_c, j \in V_g, (i,j) \in C^P. \quad (4)$$

(2) Extension Constraint ensures that if a gradient edgelet is on, then at least one of its neighboring completion edgelet should be on so that it can be extended. Formally, at either endpoint of a gradient edgelet $y_j, j \in V_g$, all the edgelets incident to that endpoint, which forms a clique $q \in C^H$, should satisfy the inequality,

$$y_j \le \sum_{i \in q \cap V_c} y_i, \quad \forall j \in V_g, q \in C^H \quad (5)$$

Figure-5 illustrates some configurations that either satisfies all the above constraints, or violates one of the constraints. Together, inequality-4 and -5 ensure the connectedness of contours, and hence in the solution space, no contour will be extracted with loose end.

We collectively represent all the connectedness constraints by the contour-closure potential as $\sum_{q \in C^P \cup C^H} \psi_\Gamma(\mathbf{Y}_q) =$

$$\mathtt{M}\Big(\sum_{\substack{(i,j)\in C^P \\ i \in V_c, j \in V_g}} (1-y_j)y_i + \sum_{\substack{q \in C^H \\ j \in V_g}} y_j \prod_{i \in q \cap V_c} (1-y_i)\Big),$$

where $\mathtt{M}$ is a very big positive number (*i.e.* the big-M trick). Note that $\psi_\Gamma(\mathbf{Y}_q) = 0$, if and only if the corresponding inequality is satisfied.

3.4. Model complexity potential ψ_M

In natural images, image contours often have multiple levels of details, depending on the scale at which a scene is perceived. To reflect this, we introduce a fourth-type potential function, which can be viewed as adding a preference towards reducing the total effective length of the completion edgelets, hence controls the overall model complexity:

$$\psi_M(y_i, \mathbf{x}_i^m) = \tau \mathbf{w}_m^T \mathbf{x}_i^m y_i, \tag{6}$$

where $\mathbf{x}_i^m$ is a feature representing the effective length of completion edgelet y_i, $\mathbf{w}_m$ the (negative valued) weighting coefficients, and τ is a user-specified global scalar controlling the overall model complexity. Details will be given in the following sections. Note that this term is simply a weighted "label cost" used in [5], which is a global higher-order term.

3.5. Energy function simplification

After combining the four potential functions together, we obtain a (very) higher-order CRF model. To directly solve (*i.e*. inference) on this higher-order CRF is very difficult. First of all, unlike previous CRF models, our graph construction allows much larger sized cliques to form. Therefore, it is non-trivial to apply message-passing algorithms, such as Loopy BP, to the problem. Secondly, it is easy to prove that the higher-order inequality constraints in ψ_Γ makes the whole energy function *non-submodular* under very mind conditions (a detailed proof can be found in the author's website). This non-submodularity property of our energy function, joint with the higher-order potential terms, makes it very difficult to apply graph-cut based [4] or QPBO method [9] to the inference.

Fortunately, due to the speciality of our potential functions, especially noting that the y_i in are 0-1 boolean variables, we realize that when Eq. (4) is true, then the cubic-term triplet junction potential in Eq. (3) can be reduced to linear terms, *i.e.*:

$$\mathbf{w}_J^T \mathbf{X}_q y_i y_j y_k = \mathbf{w}_J^T \mathbf{X}_q y_i, \quad \forall i \in (q \cap V_c),\ q \in C^J. \tag{7}$$

This reduction is significant, as it greatly simplifies our energy function. Now, except for the higher-order terms of ψ_Γ, all the other terms are reduced to be unary and linear. Moreover, even the higher-order terms ψ_Γ are in *linear* form, because they are nothing but the linear inequalities in Eq. (4) and -(5).

In summary, the inference problem become solving the following *integer linear program* (ILP), subject to a set of linear constraints:

$$\min_{\mathbf{Y}} \alpha \sum_{i\in V_g} \mathbf{w}_d^T \mathbf{x}_i^d y_i + \sum_{\substack{q\in C^J,\\ i\in(q\cap V_c)}} \mathbf{w}_J^T \mathbf{X}_q y_i + \tau \sum_{i\in V_c} \mathbf{w}_m^T \mathbf{x}_i^m y_i$$
$$s.t. \qquad \psi_\Gamma(\mathbf{Y}_q) = 0, \quad \forall q \in C^P \cup C^H. \tag{8}$$

4. Inference

We compute the MAP estimate of the model distribution to infer the surface contours in an image. Formally, we search the optimal configuration of boundary edgelets by solving the above ILP problem.

While there exist many general-purpose off-the-shelf ILP solvers, such as that based on branch-and-bound, they are *incapable* to handle large-scale problems such as our case (where there typically have thousands of variables and closure constraints to be solved). We therefore propose a tailored optimization approach, which combines the ideas of cutting-plane method and coordinate-descent.

Our inference algorithm goes as follows. We firstly solve a *linear relaxation* of the original integer program. Usually, after each round some variables have fractional values and we sequentially add more constraints such that those fractional variables have to be 0 or 1. Unlike the conventional cutting-plane method, we directly add the integral constraint to a small subset of fractional variables instead of searching for a cut. In particular, we randomly select N_{max} fractional variables, and add their integer constraints into Eq. (8). Then we solve a mixed integer linear programming (MILP) that generates integer solutions to the selected subset of fractional variables. Once a solution to the MILP is found, we fix the values of the selected subset of variables and search an optimal configuration for the remaining variables iteratively in a similar way.

It can be verified that the above procedure always converges, in finite number of iterations, to a feasible solution to the Eq.-(8). On the BSDS300 dataset, our algorithm often converged in no more than 10 iterations.

5. Experiments

Implementation Details. We have implemented the proposed contour completion model (specifically, the ILP based inference on our CRF) in `Matlab`, and tested it both on synthetic images and on the BSDS300 image sets. The parameters used in the inference, *i.e.* $\mathbf{w}_d, \mathbf{w}_J, \mathbf{w}_m$ (cf. Eq-(8)), are pre-learned by the logistic regression with piecewise learning strategy [22], based on a small subset of labeled BSDS300 ground-truth boundaries. Cross-validation is used to choose the global parameter α in the unary term (Eq-(8)). For the L-junction potential, we estimate a_i, b_i (in table-1) based on local features including the relative effective distance and the combined length of two gradient edgelets.

5.1. Tests on synthetic images

In order to validate our contour completion model, we test it on a number of purposely designed synthetic images. These images are commonly used in cognitive vision.

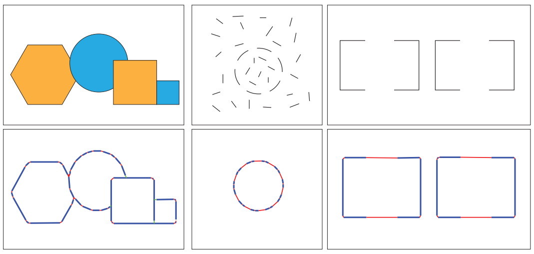

Figure 6. **Top row**: three synthetic test images (from left to right: occlusion, clutter, closure effect). **Bottom row**: Our results. The lines in blue indicate the "on" gradient-edges (G-edge); The lines in red indicate the "on" L-junction edges (L-edge), and the lines in green indicate the "on" T-junction edges (T-edge).

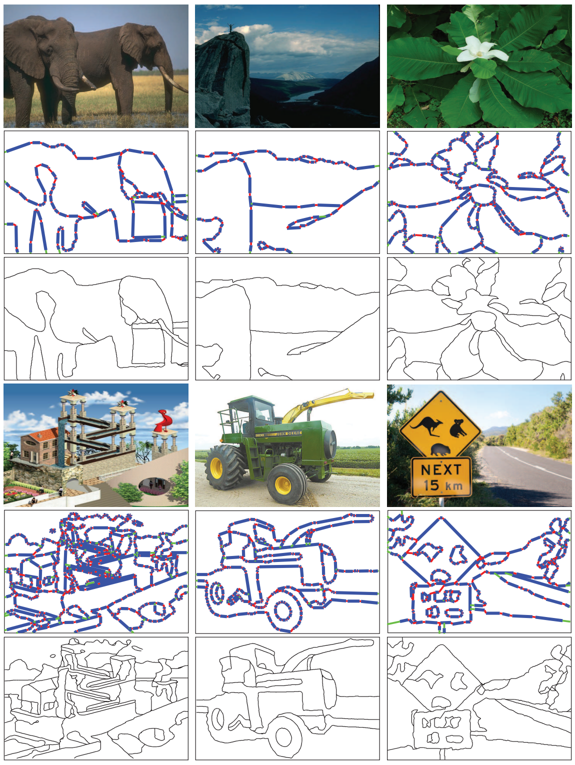

Figure 7. Sample results of our method on real images. For every three rows, **top row**: the input images; **middle row**: Our method's raw output (blue: G-edge; red: L-edge; green: T-edge); **bottom row**: Our method's final output. (**Better viewed on screen with zoom-in**).

Figure-6 shows some examples where our method performs nearly perfectly, extracting multiple closed contours regardless of occlusion (1st image), clutter (2nd image), and prefers closed contours (3rd image), just as we expect. In this figure, we give the raw output of our method, where different colors are used to indicate different type of edges: blue stands for gradient edge (G-edge); red stands for L-junction edge (L-edge); green stands for T-junction edge (T-edge). Note that the third image is often used for demonstrating Kanizsa's visual illusion contours. Our model shows an evident preference over closed contours which is similar to human's perception. The L-junction edgelets in-between are turned "on" because they lead to lower cost to the energy function.

5.2. Tests on natural images

Example results. We test our model on BSDS300, and also some other natural images (such as the Weizmann horse dataset [3]*etc.*). Satisfactory results are obtained. Figure-7 gives some sample results of our method, with both the raw outputs and the final outputs displayed. More results can be found in Figure-8 and Figure-10. It is seen that our method produces very clean, and also connected contours. Even though each of these contours does not necessarily correspond to a semantically meaningful surface region (which would require higher level vision processing such as figure-background segmentation), our current result does improve over Pb's output significantly, and will be helpful for later-stage high-level vision processing.

Our `Matlab` implementation running on a 2.1 GHz Intel Core Duo CPU, takes about 5 minutes to process a BSDS image (excluding the time used for Pb detection).

Model complexity. In this experiment, we aim to test the effect of our model-complexity parameter τ. By tuning this parameter, we obtain a series of results, each with a different level of details (complexity). Some examples are shown in Figure-8. Clearly, when τ is getting higher, the extracted contour image will have less and less details, and *vice versa*. It is worth noting that, in both cases, the connectedness of the obtained contours are always maintained, thanks to our hard closure constraints used in the inference. This is in sharp contrast to local methods such as Pb, for which increasing their threshold tends to yield more-fragmented contours.

The closure potential. This paper's key insight is about the closure effect. In this experiment, we deliberately exclude the two sets of inequality constraints and run inference again.[1] Without the inequities, our model reduces to a fully factorized CRF, and the probability of each edge is determined solely by its weight.

We have tested two cases, one with the closure potential, and one without, on all the 300 images in BSDS300. Figure-9 gives a statistical comparison using the precision-recall curve. From this figure, the effect (usefulness) of the

[1] This way of removing the inequalities is only a weak (approximate) form of excluding the contour closure constraint. We do so because otherwise we would have to minimize the original objective function containing cubic higher-order terms.

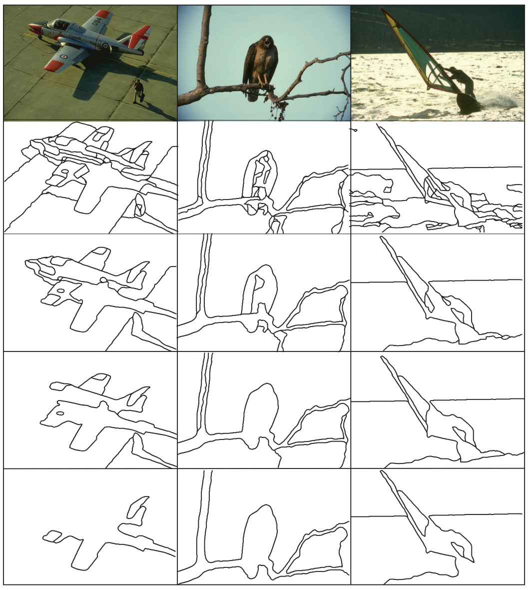

Figure 8. Effect of the model complexity parameter τ. **Row 1**: the original images. **Row 2∼5**: Our method's outputs at $\tau = 0, 1, 2, 3$. As τ increases, the extracted contour images contain less details, yet the connectedness is well maintained.

Figure 10. Methods comparison on real images: **Top row**: sample images from the BSDS dataset[2], Weizmann horse dataset [3], and baseball player dataset[15]. **Other rows** (from top to bottom): Pb detector, Ren's CRF (reproduced from [19]), contour-cut method, and our method ($\tau = 0.5$).

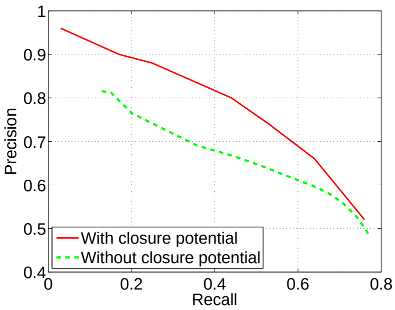

Figure 9. Effect of the closure-potential. It is clear that the closure-potential substantially boosts the model's overall performance.

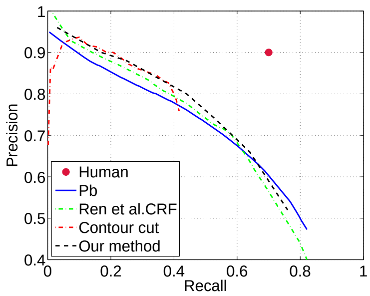

Figure 11. Precision-Recall curves for 4 methods on the BSDS300 dataset (Better viewed on screen).

closure potential is obvious (F-index increases from F=0.62 to F=0.65).

5.3. Benchmark with existing methods

We compared our method with other existing methods for contour extraction and completion. Some example results are shown in Figure-10 for visual evaluation. Compared with Pb, Ren *et al.*'s CRF and the contour-cut algorithm, our method seems produce better result in terms of the connectedness of the contours. Of course, such a comparison may be seen unfair, as it is not the other algorithms' intention to generate connected contours. Nevertheless, we find our result is visually more pleasing, which perhaps suggest that our result is closer to human's perception. We also did an overall statistical comparison among these algorithms, based on BSDS300 benchmark. The resulted precision-recall curves are plotted in Figure-11. Our new model outperforms Ren *et al.*'s CRF model by a clear margin. In the high-precision regime, our model achieves comparable result with the contour-cut algorithm (using its top 10 contours only), but in the high-recall regime (this regime is most useful for object recognition [11]) our method's performance is more consistent. We did not include Kokkinos' method [11] in the comparison, because he used a dedicated local edge detector (instead of Pb).

Contour Rand Index. It is interesting to compare our method with Global-Pb (GPb) method [2] which is the best performing contour extraction method up to date. On BSDS 300, our method has achieved F=0.65 on the precision-recall curve, while the GPb achieves F=0.70 which is far

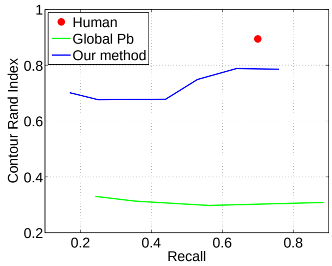

Figure 12. Performance comparison using our Contour Rand Index. Human performance is shown as the red dot.

better. This is not surprising, as the GPb makes a combined use of both contour and region information. However, it is generally recognized that the precision-recall (or the F-index) metric does not measure the grouping error well. To measure such grouping performance, we adapt the well-known *Rand Index* metric [18] to the contour case –we call our new metric the Contour Rand Index (CRI). Specifically, what we did is as follows. Firstly, the correspondence of contour points between the contour image and groundtruth are established by bipartite matching. Then contour points in both images are traced and grouped based on connectedness. We claim two contour points belong to the same group if they lie on the same contour. Finally, the grouping consistency of matched points is measured by the standard Rand index. Using this novel CRI, we compare our method against GPb on BSDS300 dataset. The result is given in Figure-12, which shows that our method is much better than GPb, and is very close to human's performance.

6. Closing remarks

Understanding the mechanism for contour-completion holds the promise to develop better-performing image segmentation and object recognition algorithms relying on object boundaries. This is however, a very challenging task, which necessarily involves the understanding and implementing many perceptual grouping principles. In this paper, we have presented a higher-order CRF model attempting to mimic the effect of contour closure (via first-order connectedness approximation). We also derived an efficient optimization method to perform approximate inference on the higher-order CRF. We demonstrated our model's superior performance for contour completion. We hope this work will provide useful ideas for future perceptual grouping research.

Acknowledgement: This work is supported by Australia Research Council via an ARC-DP grant to the second author. NICTA is a research center of excellence funded by the Australian Government, in part through ARC.

References

[1] B. Andres, J. H. Kappes, T. Beier, U. Köthe, and F. Hamprecht. Probabilistic image segmentation with closedness constraints. In *Proc. ICCV*, 2011. 2

[2] P. Arbelaez, M. Maire, C. Fowlkes, and J. Malik. Contour detection and hierarchical image segmentation. *IEEE TPAMI*, 33(5):898–916, 2011. 1, 2, 7

[3] E. Borenstein and S. Ullman. Class-specific, top-down segmentation. In *Proc. ECCV*, pages 109–124, 2002. 6, 7

[4] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *IEEE TPAMI*, 23:1222 –1239, 2001. 5

[5] A. Delong, A. Osokin, H. Isack, and Y. Boykov. Fast approximate energy minimization with label costs. In *CVPR'10*, pages 2173 – 2180, june 2010. 5

[6] J. H. Elder and S. W. Zucker. Computing contour closure. In *Proc. ECCV*, pages 399–412, 1996. 1, 2

[7] P. Felzenszwalb and D. McAllester. A min-cover approach for finding salient curves. In *CVPRW*, pages 185–185, 2006. 1

[8] G. Guy and G. Medioni. Inferring global perceptual contours from local features. In *Proc. CVPR*, pages 786–787, 1993. 1

[9] P. Hammer, P. Hansen, and B. Simeone. Roof duality, complementation and persistency in quadratic 0-1 optimization. *Mathematical Programming*, 28:121–155, 1984. 5

[10] R. Kennedy, J. Gallier, and J. B. Shi. Contour cut: identifying salient contours in images by solving a hermitian eigenvalue problem. In *Proc. CVPR*, pages 2520–2527, 2011. 1, 2

[11] I. Kokkinos. Highly accurate boundary detection and grouping. In *Proc. CVPR*, pages 2520–2527, 2010. 2, 7

[12] I. Kovacs and B. Julesz. A closed curve is much more than an incomplete one - effect of closure in figure ground segmentation. *P-NAS,USA*, 90(16):7495–7497. 1

[13] S. Mahamud, L. R. Williams, K. K. Thornber, and K. L. Xu. Segmentation of multiple salient closed contours from real images. *IEEE TPAMI*, 25(4):433–444, 2003. 1, 2

[14] D. R. Martin, C. C. Fowlkes, and J. Malik. Learning to detect natural image boundaries using local brightness, color, and texture cues. *IEEE TPAMI*, 26(5):530–549, 2004. 2

[15] G. Mori, X. Ren, A. A. Efros, and J. Malik. Recovering human body configurations: combining segmentation and recognition. In *Proc. CVPR*, pages 326–333, 2004. 7

[16] S. E. Palmer. *Vision science : photons to phenomenology*. MIT Press, Cambridge, Mass., 1999. 1

[17] S. B. Piotr Dollar, Zhuowen Tu. Supervised learning of edges and object boundaries. In *Proc. CVPR*, pages 1964–1971, 2006. 2

[18] W. M. Rand. Objective criteria for the evaluation of clustering methods. *Journal of the American Statistical Association*, 66(336):846–850, 1971. 8

[19] X. F. Ren, C. C. Fowlkes, and J. Malik. Learning probabilistic models for contour completion in natural images. *IJCV*, 77(1-3):47–63, 2008. 2, 3, 7

[20] T. Schoenemann, S. Masnou, and D. Cremers. The elastic ratio: Introducing curvature into ratio-based image segmentation. *IEEE TIP*, pages 2565–2581, 2011. 2

[21] E. Sharon, A. Brandt, and R. Basri. Completion energies and scale. *IEEE TPAMI*, 22(10):1117 – 1131, 2000. 4

[22] C. Sutton and A. McCallum. Piecewise training for structured prediction. *Machine Learning*, 77:165–194, 2009. 5

[23] S. Wang, T. Kubota, J. M. Siskind, and J. Wang. Salient closed boundary extraction with ratio contour. *IEEE TPAMI*, 27(4):546–561, 2005. 1, 2

[24] L. R. Williams and K. K. Thornber. A comparison of measures for detecting natural shapes in cluttered backgrounds. *IJCV*, 34(2-3):81–96, 1999. 1, 2

[25] Q. Zhu, G. Song, and J. Shi. Untangling cycles for contour grouping. In *Proc. ICCV*, pages 793–800, 2007. 2