Signal Processing: Image Communication 78 (2019) 159–170

Contents lists available at ScienceDirect

Signal Processing: Image Communication

journal homepage: www.elsevier.com/locate/image

Interactive image segmentation using geodesic appearance overlap graph cut[☆]

Zili Peng[a,b], Shaojun Qu[c], Qiaoliang Li[a,b,*]

[a] Key Laboratory of Computing and Stochastic Mathematics (Ministry of Education of China), Changsha, Hunan, 410081, People's Republic of China
[b] College of Mathematics and Statistics, Hunan Normal University, Changsha, Hunan, 410081, People's Republic of China
[c] College of Information Science and Engineering, Hunan Normal University, Changsha, Hunan, 410081, People's Republic of China

ARTICLE INFO

Keywords:
Graph-cut
Geodesic
Appearance overlap
Interactive image segmentation

ABSTRACT

Image segmentation is a fundamental step in many applications such as image editing, medical image analysis and processing. It is quite common to use graph cuts for image segmentation in recent years. Tang, Gorelick, Veksler, and Boykov proposed a new image segmentation model which uses the appearance overlap on unnormalized histograms and graph cut framework. Their model is highly effective for interactive segmentation but is prone to isolated points. To avoid that problem, we propose an effective interactive image segmentation method, that is appropriately incorporating geodesic distance information, appearance overlap information, and edge information together into the well-known graph-cut framework. Rather than a simple union of these information, the respective strengths of each information term can be tuned adaptively in our method. We utilize the user's scribbles to obtain the estimated foreground/background color models via fast kernel density estimation, and then get the appearance overlap intricateness according to the inferred color models. By taking comprehensive advantage of the geodesic distance and the global appearance overlap color clues, our method requires less user effort and achieves higher accuracy of segmentation than the latest interactive segmentation techniques, such as *Geodesic Graph Cut*, *GrabCut in One Cut*, *Semi-Supervised Normalized Cuts*, and *Convexity Shape Prior for Binary Segmentation*, as shown in our experiments.

1. Introduction

Image segmentation is the separation of the target region corresponding to the object of the real world from the background of the image. The target region is based on the needs of specific applications and usually conforms to the subjective cognition and experience of the operator. Without image segmentation, it is hard to switch into image analysis from image processing and get further image understanding. Specifically, in image editing tasks, we may need to replace the background of the image, which requires accurate segmentation of the foreground objects; in the analysis and processing of medical images, it may be necessary to segment the image of an organ to carry out subsequent medical data statistics and analysis; in the field of intelligent transportation, the identification of traffic signs is an indispensable step, which relies on accurate image segmentation. Because of its importance, image segmentation algorithm is developing rapidly, and new and efficient algorithms are emerging. However, real-world image is quite intricate and varying; there is no fully-automatic segmentation algorithm to achieve the same effect as the manual annotation does. Because completely manual segmentation is both time-consuming and laborious, interactive image segmentation [1–3] using computers with user intervention is necessary to obtain better segmentation results more efficiently. Interactive segmentation emphasizes the accurate extraction of the real objects of interest that are consistent with the human subjective perceptions (high-level perception). The most significant advantage of interactive segmentation is that it can effectively combine the high-level perception of people with the tirelessness of the machine interactively, making the ill-posed problem of image segmentation clear and reducing the difficulty of segmentation.

In the image processing communities, Boykov et al. [4] proposed a technique that optimizes boundary and region energies of objects in $N-$dimensional images in 2001, they exploit the min-cut/max-flow algorithm in graph optimization to solve the minimum energy problem. In addition, they proposed a graph-cut framework for interactive image segmentation methods [5–8], and obtained good segmentation results. Their framework has greatly stimulated the development of researches and applications in the field of computer vision and image processing. Many image segmentation methods based on graph cut was proposed, e.g., *GrabCut* [9], *Lazy Snapping* [10], and *Graph-Cut with*

[☆] No author associated with this paper has disclosed any potential or pertinent conflicts which may be perceived to have impending conflict with this work. For full disclosure statements refer to https://doi.org/10.1016/j.image.2019.06.012.

[*] Corresponding author at: College of Mathematics and Statistics, Hunan Normal University, Changsha, Hunan, 410081, People's Republic of China.
E-mail addresses: pengzili0603@gmail.com (Z. Peng), qshj@hunnu.edu.cn (S. Qu), liqiaoliang@hunnu.edu.cn (Q. Li).

https://doi.org/10.1016/j.image.2019.06.012
Received 24 April 2019; Received in revised form 27 June 2019; Accepted 27 June 2019
Available online 3 July 2019

Prior Constraints [11–14]. Presently, image segmentation using graph-cut is very popular, e.g., *Geodesic Graph Cut (GeoGC)* [15], *GrabCut in One Cut (OneCut)* [16], *Convexity Shape Prior for Binary Segmentation (TRC)* [17], and *Interactive Segmentation on RGBD Images via Cue Selection (ISCS)* [18] mentioned in Section 2 and so on.

Among the above methods, OneCut [16] is a model proposed by Tang, Gorelick, Veksler and Boykov, which used the appearance overlap on unnormalized histograms and graph cut framework. This model is highly effective for interactive segmentation but is prone to isolated points. In this paper, we develop an improved interactive image segmentation algorithm. The following is the main contribution in this paper. Firstly, we find that the number of isolated points in segmentation results is positively correlated to the weight of the appearance overlap penalty term under certain conditions. Secondly, we incorporate geodesic distance information, edge information, and appearance overlap information together properly in the traditional graph-cut optimization framework, and propose our geodesic appearance overlap graph cut (GeoAGC) algorithm, solving the problem of the OneCut algorithm's [16] tendency to produce isolated points. Our method is superior to the recently developed algorithms (e.g., *Semi-Supervised Normalized Cuts (SSNCuts)* [19] and the first three works (GeoGC [15], OneCut [16], TRC [17]) mentioned in Section 2.1) in segmentation effect, and is higher or closer to the recently developed algorithms in segmentation efficiency. Finally, we propose a reasonable weighting scheme of the appearance overlap penalty.

The remainder of this paper is organized as follows: in Section 2, we introduce several interactive image segmentation methods; in Section 3, we present the proposed method in detail; Section 4 presents the experimental results; finally, Section 5 concludes the paper and proposes future work.

2. Related work

This section briefly reviews several related image segmentation techniques that involves the graph-cut optimization algorithms, including "traditional" approaches[1]: GeoGC [15], GrabCut [9], OneCut [16], TRC [17], TouchCut [20], Interactive Image Segmentation Using Adaptive Constraint Propagation [21], LooseCut [22], Weakly Supervised Foreground Segmentation Based on Superpixel Grouping [23], ISCS [18], Probabilistic Diffusion (PD) [24], Label Propagation Through Complex Networks (LPTCN) [25], Adaptive Appearance Separation (AAS) [26], Graph Reduction Framework (GRF) [27] and some methods incorporating Deep Learning, such as A Joint Convolutional Neural Networks and Context Transfer for Street Scenes Labeling (JCNNCT) [28].

2.1. Interactive segmentation based on graph-cut

To improve the traditional graph-cut based image segmentation methods [4], Price et al. [15] proposed the GeoGC method in 2010. Their approach included the geodesic distance, color, and simple edge terms. The first two terms measure the regional properties. For the boundary term, they used the reciprocal of the color contrast among adjacent pixels. In addition, they considered both the global weight and the local weight for the regional and boundary terms. The quality of the estimated color model influences the global weight, and the local weight reflects the relative importance of the geodesic regional and boundary terms [3]. Thus, their approach not only overcomes the short-cutting problem but also increases segmentation accuracy.

In [9], Rother, Kolmogorov and Blake proposed GrabCut algorithm. First, a user draws a rectangle around the foreground, and then the initial appearance model parameters are computed by the given rectangle box. The parameters are then used to optimized the energy function of the segmentation by graph cuts. After the optimization is obtained, the user can get an initial segmentation. Then the segmentation is used to compute the new appearance parameters, iteratively perform the above procedures, we can get more accurate segmentation. Since the energy optimization problem is NP-hard, it is needed to perform iterative procedure many times. As a result, the algorithm is very slow.

In OneCut [16], Tang et al. found out that when the appearance model in the segmentation energy of GrabCut are replaced by the L_1 norm of the difference between the unnormalized color histograms, the user can achieve higher accuracy comparing to other forms of approximations. The new segmentation energy can be solved by using graph cut only once. Furthermore, the OneCut method can guarantee global minimum.

By using the convexity of shape as a priori knowledge, Gorelick et al. [17] proposed the TRC algorithm to overcome the OneCut algorithm's [16] isolated points problem, thereby increasing the segmentation accuracy. Once the convex structure is violated, they introduce a specific potential function as a penalty. The overall segmentation energy of their approach included but not limited to the prior of shape convexity, user's input, color separation [16] and boundary information [29]. However, it is a huge challenge to optimize the convexity energy; the reasons include that the number of the non-convex cliques in an image may be vast. Moreover, the energy is a non-submodular function [17]. To overcome the difficulty, Gorelick et al. designed the dynamic programming to efficiently count the number of the non-convex cliques. Besides, an iterative approximate optimization [30,31] was used to achieve the global minimum energy.

In Touch [20], the authors also used appearance model fusing edge, region texture and geometric information of the pixels marked by users, but they used the level set framework. Yu et al. [21] proposed a LooseCut model to deal the case that the input bounding box only loosely cover the foreground object by emphasizing the appearance difference between the foreground and background models. Jian et al. [22] employed adaptive constraint propagation for semi-supervised kernel matrix learning which adaptively propagates the interactive information into the whole image. In [23], the authors integrated the watershed algorithm and mean-shift clustering algorithm to obtain reliable initial foreground and background labels for simple linear iterative clustering superpixels.

The ISCS method [18] also weighs the choice of color and depth cues in RGBD images. The algorithm is fast and accurate. Different from previous RGBD image segmentation, the ISCS method only used normalized map which is derived from the depth information as an additional cue. Only one single cue is selected for each pixel in the image for graph cuts framework. The authors of ISCS [18] believe that a single cue in an RGBD image already has sufficient advantages in local regions to discern the foreground and background. They argued that if poor predictive cues is used in the algorithm, the advantages may be diluted. The ISCS method also used the geodesic distance and the graph-cut optimization method. Different from our method, the geodesic distance information in ISCS is mainly used for processing various cues and for the purpose of selecting which cue to use for segmentation. In our algorithm, we incorporate the cues as separate sub-items into the graph-cut framework.

2.2. Other related prominent segmentation methods

The problem of image segmentation is essentially the problem of assigning category labels to each pixel. In the labeling problem of foreground objects in the street scenes, JCNNCT method [28] points out that traditional Markov random field (MRF) inference is prone to excessive smoothing, so JCNNCT defines a soft-restricted MRF energy function to improve the labeling accuracy of the priori superpixel-CNNs model (features and priori location information learned at the superpixel level) while alleviating the over-smoothness. Both JCNNCT and our method use the graph-cut optimization algorithm [6–8]

[1] The "traditional" approach is relative to the "new" approach incorporating Deep Learning developed in recent years.

to achieve MRF inference, and adaptively adjust the weight of the boundary term or smoothing term for the specific application scenario to improve segmentation results. Besides, JCNNCT and our method contain both appearance information and spatial location information regarding feature representation. The difference between JCNNCT and our method are: (1) JCNNCT uses powerful CNNs to extract features, and also uses contextual cues from the training set in the segmentation process; while the feature descriptors (such as color model, geodesic model, and approximate appearance overlap) in our method are hand-crafted. (2) JCNNCT is a multi-object semantic segmentation solution for dealing with the imbalance of foreground and background distributions in street scenes data, while our method is a simple binary interactive segmentation, which is relatively less difficult. Additionally, our method performs individual weight adjustment on several sub-items of the energy function, so the segmentation accuracy is usually much higher. The PD method [24] proposed by Wang et al. turned the segmentation problem into a probabilistic estimation problem based on prior seeds information, which is essentially different from the graph cut framework. PD [24] and our approach aimed to improve the performance of the segmentation by combining local features and global features of the image. The difference between the two methods is as follows: Firstly, PD [24] used a probability framework based on the pairs of pixels and the pairs of labels to obtain more accurate unary potential at a limited quantity of seeds. This is similar to the way we calculate the geodesic distance in our method: to ensure labels consistencies within the homogeneous region and differences within heterogeneous regions. Secondly, PD method [24] used a likelihood learning method to distribute local probabilities in order to obtain intrinsic global structural features, making the probability diffusion process simpler and more efficient. This scheme is similar to color histograms we used in our method to reflect the global features of the entire image, the purpose of both is to make use of global information, but our method is simpler. Finally, PD method [24] established the equivalence relation between likelihood learning and likelihood diffusion, and proposed an optimization strategy based on iterative diffusion to reduce calculation time. Our approach directly utilizes an efficient graph cut optimization scheme (mini-cut/max-flow algorithm) [6–8]. Breve's LPTCN [25] method represented image segmentation as a two-stage label propagation process: the first phase was the underlying fast label propagation based on the pixel characteristics of the 9 features; the second phase was the label propagation on the reconstructed graph so that the label on the object boundary can be fixed. LPTCN method [25] did not particularly emphasize the label consistency in the same area, which caused the user to input more scribbles even for the homogeneous region, otherwise it may result in poor segmentation due to the distance limitation of the label propagation. The AAS approach [26] proposed by Peng and Li is based on the efficient dense conditional random field (Dense CRF) framework [32,33], in which Gaussian mixture models (GMMs) and geodesic distance features were incorporated, but the global features of the histogram were not utilized. Moreover, because of the many parameters in the Dense CRF framework, it was difficult to set the optimal segmentation parameter values. Subgraphhi and Mukhopadhyay's GRF [27] has contributed to reducing the storage space and computation time for high-resolution images segmentations. Their GRF [27] extracted texture features by using a specific wavelet transform and divided the image into sub-blocks for processing. Different from the superpixel-based graph cuts methods, GRF method [27] merged together the segmentation regions on the sub-block boundaries to ensure the segmentation precision of the full-resolution image.

In recent years, Deep Learning has demonstrated its dominant performance in image and video processing. Among them, Mask R-CNN [34], Path Aggregation Network (PANet) [35] and Mask Scoring R-CNN [36] methods are particularly prominent in image segmentation. Some researchers have tried to combine deep learning with graph-cut, e.g., [37–40]. Ullah et al. [39] use ResNet [41] to obtain good boundary detection results, and then they used the obtained results (the boundary level information) to construct the boundary terms. After combining with the traditional regional items [4], Ullah et al. used the graph-cut to optimize the energy function, claiming that they obtain the most advanced results. It is well-known that compared with hand-crafted feature descriptors, deep neural networks have an unparalleled advantage in multi-level feature description, but this advantage is based on powerful computing resources, well-designed neural network architecture (e.g., ResNet), and time-consuming training. For most Deep Learning methods (e.g., supervised learning), a large number of data samples are also indispensable. For the specific task of image segmentation, in addition to requiring large representative image datasets to participate in the training, accurate annotated truth images corresponding to the training images are required. Besides, it is necessary for experienced researchers to finetune the neural network. However, we are concerned with the interactive segmentation algorithm of a single image that does not depend on the dataset. The interactive segmentation does not need to annotate the image in advance (only as a reference for evaluating the segmentation accuracy). By integrating the subjective judgment and experience of the operator, we are able to segment the foreground objects accurately and efficiently after adding the appropriate amount of interaction. Our method needs no complicated and time-consuming training process. Therefore, we have not experimentally compared our method with the methods of combining Deep Learning with graph-cut in this paper.

3. Geodesic appearance overlap graph cut (GeoAGC)

3.1. Motivation

The TRC algorithm [17] provides an approach for optimization of segmentation energies with non-submodule terms using graph-cut. However, through our experiments and analyses, we find that the TRC algorithm is not high efficient, for objects with holes or with non-convex shapes, its segmentation results lack of accuracy. Besides, multiple iterations are needed to converge the minimum energy, particularly for images with complex textures. GrabCut [9] and TRC algorithm belong to the iterative algorithm, whereas we are trying to seek an efficient segmentation scheme without iterations, which is similar to OneCut. Note that OneCut has the following three shortcomings:

(1) It is too sensitive to the user's scribbles.
(2) The problem of isolated points: when the appearance models are not distinct (i.e., the foreground and background appearance models are ambiguous and intricate), or the number of seeds (or scribbles) is not enough, the irrational β value (i.e., the weight of the appearance overlap penalty term) used in OneCut [16] can easily lead to the problem of isolated points, holes, and fake object blocks, and the larger the β value is, the more serious of the problem will be.
(3) It is hard to directly decide how many penalties should be assigned to the appearance overlap if only based on user's scribbles.

The isolated points problem produced by OneCut is mainly due to the lack of distance information between pixels and seeds, as shows in Fig. 1(b). Besides, the next interaction is likely to cause severe damages to the previous segmentation result by using OneCut, see Fig. 2. From the construction of OneCut's appearance overlap approximation, the OneCut algorithm essentially uses the difference between the color histograms to approximate the actual overlap of the foreground and the background appearance. However, the color histogram reflects the global feature of the entire image, which is the proportion of pixels of different colors in the entire image. The histogram is irrelevant to the spatial position information of each color in the image, so it is impossible to accurately describe the position of the specific objects in the image; this is the root cause of the problem that OneCut is prone to

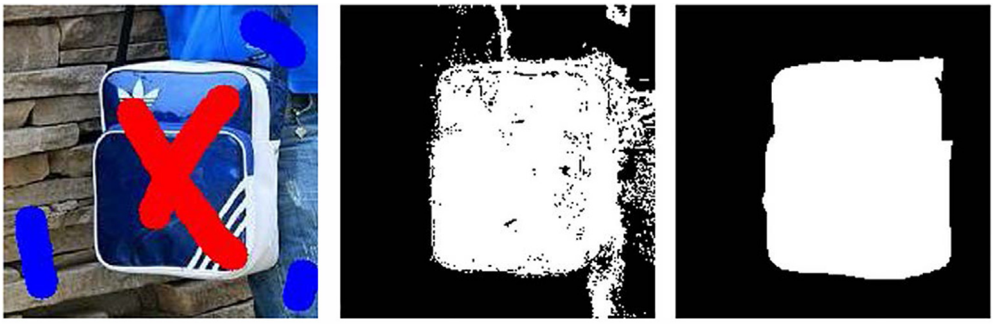

(a) (b) (c)

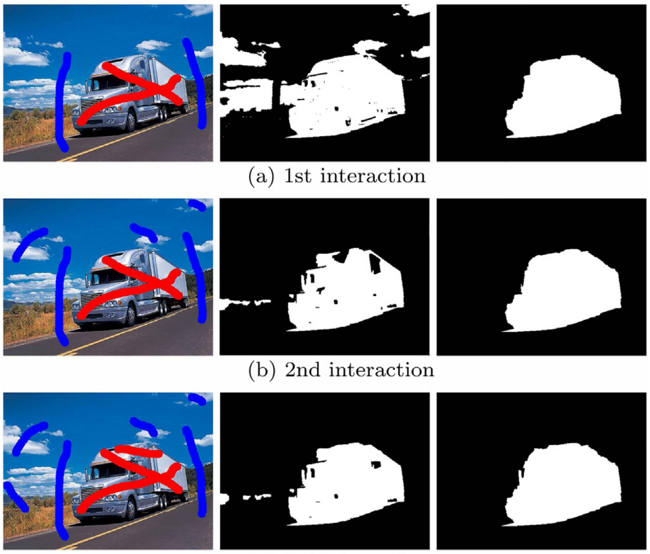
Fig. 1. The problem of isolated points from OneCut. (a) is the original image with user scribbles, (b) and (c) are the segmentation masks of OneCut and our method respectively.

isolated points. Moreover, by adding subsequent user interactions, one may break the existing segmentation results. The appearance overlap approximation in OneCut have used only the color feature of the pixels at which the seeds are located, and it is precisely the position information of the seeds that clarifies the specific position of the target in the image. Although the smoothness term in OneCut can suppress the occurrence of isolated points to a certain extent, the controllability is not promising. Therefore, we prefer the geodesic distance which takes into account the weighted shortest distance from one point to another, rather than the straight line connecting the two pixels. Compare to Euclidean distance, geodesic distance has inherent advantages in maintaining the internal geometry of data globally. The GeoGC [15] is better than the traditional graph-cut [4] and the conventional geodesic method [42,43], because of the geodesic distance and reasonable trade-off between regional and boundary term weights. However, GeoGC only considers the geodesic distance in the LUV color space, and lacks consideration of the global color characteristics of the image; the object cannot be segmented precisely in the case of a small number of seeds and blurry object boundaries, which increases the user's workload. Regarding the boundary term, GeoGC merely sets it to the reciprocal value of the color contrast of neighboring pixels.

To weaken the adverse effects brought by the drawbacks of the GrabCut, TRC, OneCut, and GeoGC methods mentioned above, at the same time, exert their respective advantages, our method makes comprehensive use of color information and geodesic distance information; the method does not need iterative processes, and can give good results in many cases. In addition, after getting poor segmentation in the condition of inadequate scribbles, our approach can continue to add extra interaction to improve the previous segmentation results, instead of destroying the existing result.

3.2. The proposed method

In our method, geodesic distance and appearance overlap constraints are incorporated in the graph-cut optimization framework: $E(L) = \lambda \cdot \sum_{x_i \in I} R_i(L_i) + \sum_{(x_i,x_j)\in N} B_{i,j}(L_i, L_j)$. Like being used in other methods (such as the algorithms in Section 2) based on the graph-cut framework, $E(L)$ in the formula is the total energy function of the segmentation problem, which is the sum of all regional terms R_i and all boundary terms $B_{i,j}$. The set of labels L that minimizes the total energy function $E(L)$ is the segmentation result, and the BK (Boykov–Kolmogorov) graph-cut algorithm [6–8] provides an efficient solution for the segmentation problem. The coefficient $\lambda \geq 0$ specifies a proportion of the importance of regional term R_i to the importance of boundary term $B_{i,j}$. $L = (L_i)$ is a binary index vector and its components L_i specify labels to pixels x_i in set I (i.e., $L_i = 0$ if x_i is a background pixel and $L_i = 1$ if x_i is a foreground object pixel). I is the set of the image pixels, and N is the set of all adjacent pixel pairs.


(a) 1st interaction

(b) 2nd interaction

(c) 3rd interaction

Fig. 2. Interactive image segmentation using OneCut and our method. The first column is the original image with user scribbles; the second and the third columns are the segmentation results of OneCut and our method separately. From top (a) to bottom (c), we show the OneCut's effects of the first, second, and third interaction separately. For comparison, the results of our method after the corresponding interactions are showed in the third column.

To weigh the relative significance of each sub-item respectively, we express the regional term as:

$$R_l(x_i) = s_l(x_i) + \alpha_1 M_l(x_i) + \alpha_2 G_l(x_i), \tag{1}$$

where x_i represents a pixel in an image, α_1 and α_2 are used to specify the importance of $M_l(x_i)$ and $G_l(x_i)$ separately; the meaning of each sub-item is consistent with the expression of GeoGC: $s_l(x_i)$ is a hard-constraint term from user strokes, $M_l(x_i)$ is a color model term from global color features, and $G_l(x_i)$ is based on the geodesic distance [15]. Namely,

$$s_l(x_i) = \begin{cases} \infty, & \text{if } x_i \in \Omega_{\bar{l}} \\ 0, & \text{otherwise,} \end{cases} \tag{2}$$

$$M_l(x_i) = P_{\bar{l}}(C(x_i)), \tag{3}$$

$$G_l(x_i) = \frac{D_l(x_i)}{D_F(x_i) + D_B(x_i)}, \tag{4}$$

where Ω_l is the seeds set assigned by users, labels $l \in \{F, B\}$; $C(x_i) = c$ is the color intensity at pixel x_i; $P_l(C(x_i))$ is the color model at x_i for the label l, and can be computed by normalizing the color probability density function (PDF) $Pr(c|l)$ of Ω_l, mathematically,

$$P_l(c) = \frac{\Pr(c|l)}{\Pr(c|F) + \Pr(c|B)}, \tag{5}$$

specifically, $Pr(c|l)$ is the probability that the pixel of color c belongs to the foreground or background, which is determined by the color distribution generated by the seeds given by users, so for the entire image, $Pr(c|l)$ can be directly represented by the PDF, and can be estimated using kernel density estimation (KDE). The improved Fast Gaussian Transform (iFGT) algorithm proposed by Yang et al. [44] has successfully accelerated the computing of KDE in image processing. Thus, the color PDF $Pr(c|l)$ can be efficiently obtained via the iFGT [44]. Note that the color model term $M_l(x_i)$ is a sub-item of the region term (or data item) R_l in Eq. (1), and $P_l(C(x_i))$ as is shown in Eq. (5), it is obtained by normalizing the likelihood functions. As we know, the likelihood function is numerically equal to the probability that the pixel x_i is the foreground or the background under the condition of the given color model parameter, that is, the color PDF $Pr(c|l)$. The regional term expresses the cost of assigning a label to a pixel [6–8]. Thus, given the color model parameters, the greater the probability that the pixel x_i is the foreground, the smaller the cost of assigning a foreground label to the pixel x_i; likewise, the greater the probability that the pixel x_i is the background, the smaller the cost of assigning a background label to the pixel x_i. Thus, the color model term is expressed as $M_l(x_i) = P_{\bar{l}}(C(x_i))$ instead of $M_l(x_i) = P_l(C(x_i))$. $D_l(x_i)$ is the geodesic distance that is defined as the smallest geodesic distance $d_l(s, x_i)$ from pixel x_i to the seeds s (i.e. foreground seeds(F) or background seeds (B)) in the form of:

$$D_l(x_i) = \min_{s \in \Omega_l} d_l(s, x_i), \tag{6}$$

$$d_l(x_i, x_j) = \min_{\mathfrak{R}_{x_i,x_j}} \int_0^1 \left| W_l(\mathfrak{R}_{x_i,x_j}(p)) \cdot \dot{\mathfrak{R}}_{x_i,x_j}(p) \right| dp, \tag{7}$$

where $\mathfrak{R}_{x_i,x_j}$ is a path from x_i to x_j on the weighted graph $W_l(x_i)$, and is parameterized by $p \in [0,1]$. In essence, given the label l, $W_l(x_i)$ is a gradient map of the color model $P_l(c)$, namely, $W_l(x_i) = \nabla P_l(c)$ [15,42].

The boundary term of our method is expressed by using the combination of the appearance overlap penalty in OneCut [16] with the usual smoothness terms:

$$B(x_i, x_j) = -\beta \left\| \theta^L - \theta^{\bar{L}} \right\|_{L_1} + |\partial S|, \tag{8}$$

where $L \subset \Omega$ is a segment, and $\overline{L} = \Omega \backslash L$, θ^L and $\theta^{\bar{L}}$ are appearance models represented by the unnormalized foreground and background color histograms respectively, $\|\cdot\|_{L_1}$ is the L_1 norm, β is the penalty factor of the approximate appearance overlap; the second addend in Eq. (8) is a commonly used contrast-sensitive smoothness term, that is to say,

$$|\partial S| = \sum w_{x_i,x_j} \left| l_{x_i} - l_{x_j} \right|, \\ \text{with } w_{x_i,x_j} = \exp \frac{-\Delta I^2}{2\sigma^2} \Big/ d, \tag{9}$$

where l_{x_i} is the label assignment for pixel x_i, ΔI is the color difference between pixel x_i and x_j, σ^2 is the average ΔI^2 over the whole image, d is the distance between x_i and x_j.

According to GeoGC, OneCut, and [8], our objective energy function is submodular that can be minimized utilizing the efficient BK graph-cut [6–8]. Based on the idea of augmenting paths, the BK graph-cut algorithm constructs two search trees S and T with no common child node to detect the augmenting paths, and the roots of the trees are located at "source" (s) and "sink" (t), respectively. Based on the definition of various nodes and edges of the constructed graph [8,15,16], the BK graph-cut algorithm can be simply described as the sequential loops of three stages: the search tree growth, path augmentation, and storage adoption. The specific process of the algorithm is described in [7]. After the end of the loop in the BK graph-cut algorithm, a max-flow path from the "source" (s) to the "sink" (t) can be obtained. We can get the min-cut according to the definition that the pixels connected to the "source" (s) are the segmented foreground objects; the pixels connected to the "sink" (t) are the inferred background.

3.3. The setting of related parameters

3.3.1. The penalty factor of appearance overlap β

Irrational appearance overlap penalty factor (β) used in the OneCut algorithm is easy to cause the problem of isolated points, so we need to adjust the value of β properly according to the ambiguity degree of the estimated appearance model. If the appearance is more cluttered and the appearance overlap is more serious, then the approximation of the appearance is more difficult. The KDE result $Pr(c|l)$ obtained by using the iFGT may have large deviations from the actual distribution. In order to reduce the risk of segmentation errors caused by such deviations, our color model uses a posterior probability consisting of the color PDF $Pr(c|l)$ as in Eq. (5), (color model is based on a priori assumption that a pixel of color c belonging to the foreground and to the background are both 0.5 in probability). Then, the degree of appearance clutter and the severity of overlap can be expressed by estimating the classification error rate of the Bayesian classifier. So, in order to set the relevant parameters, we can introduce an error term ε as in the GeoGC algorithm [15]:

$$\varepsilon = \frac{1}{2} \left[\frac{\sum_{x \in F} P_B(C(x))}{|\Omega_F|} + \frac{\sum_{x \in B} P_F(C(x))}{|\Omega_B|} \right]. \tag{10}$$

With the "mirror" (the error term ε) that respond to the appearance, the correlation weight coefficient can be adjusted in conjunction with the degree of appearance clutter and the severity of the overlap. When there is minor error ($\varepsilon \approx 0$), we would like to give the appearance overlap term a big weight; when the color appearance models are unreliable any more (i.e., the color appearance models will be cluttered more and more), the error ε will grow, and it is wise to deliver less weight to the approximate appearance overlap:

$$\beta_{t+1} = \begin{cases} 0.25 \times \beta_t, & \text{if } \varepsilon > 0.9 \\ (1-\varepsilon) \times \beta_t, & \text{otherwise,} \end{cases} \tag{11}$$

with $\beta_0 = 1.4$.

The formula for the value of β is written as a recursive form because we may dissatisfy with the segmentation results after the first interaction, and it is likely that the estimated appearance models will be improved if we continue to provide seeds. The β value is updated along with the estimated appearance model.

We can see from Fig. 3 that in the first interaction (placing seeds with greater randomness to specify the foreground and background according to the requirements of the specific application), we improved the segmentation result after continuing the joining of seeds at the position where the segmentation error occurred (the lady's heel, knee and right hand, and the upper left corner in the image). In contrast, it is difficult to correct the errors, and may even destroy the original segmentation results for OneCut to continue adding seeds where the segmentation results are wrong, as shown in Fig. 2. Because our method is more comfortable for the user to grasp the regularity when placing the seeds for improving the previous segmentation result, it minimizes blindly or randomly placing seeds and reduces the user's interaction workload.

3.3.2. The constraints of color models and geodesic distance α

In order to enhance the robustness to the placement of seeds and overcome the short-cutting defect like GeoGC [15], we set α according to the error term ε computed in the previous section:

$$\kappa = \begin{cases} 1 - 2\varepsilon, & \text{if } \varepsilon < 0.5 \\ 0, & \text{otherwise,} \end{cases} \tag{12}$$

$$\alpha_1 = 80 \times (1-\kappa), \ \alpha_2 = 800 \times \kappa. \tag{13}$$

where κ can be viewed as the confidence of the estimated appearance model.

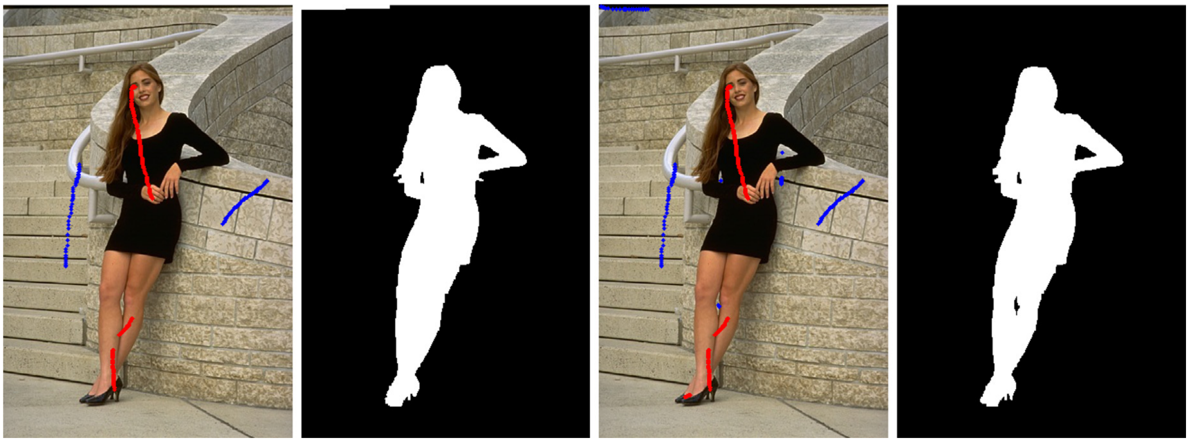


Fig. 3. Our method improves the segmentation effect continuing the joining of scribbles at the position where the segmentation error occurred (the lady's heel, knee and right hand, and the upper left corner in the image).

Table 1
OneCut's error rate and mean runtime under the condition of different numbers of bins.
Source: From Table 2 in [16].

Bins	Error rate	Mean runtime (s)
8^3	9.98%	18
16^3	8.10%	5.8
32^3	6.99%	2.4
64^3	6.67%	1.3
128^3	6.71%	0.8
256^3	7.14%	0.8

3.3.3. The number of bins

Table 1 is a table of error rates and mean runtime for OneCut (from Table 2 in OneCut [16]). Our method using the L_1 norm between non-normalized histograms to approximate the objective appearance overlap is almost identical to OneCut; therefore, the choice of bins' quantity and its impact on the segmentation results should be consistent between the two methods, and the error rates and mean runtime for OneCut in Table 1 can be used to represent those of our method.

If the value of bins parameter is chosen to be 8, then for all color images with RGB color channels, all the colors in the image quantified into the histogram will generate 8^3 bins in theory, though not every bin is occupied. For the cases where the value of bins parameter selected is 16, 32, etc., the corresponding numbers of bins can be derived similarly. The larger the bins parameter value, the more detailed the quantified color histogram, the more the slight difference between different colors can be reflected. Therefore, as the number of bins increases, the error rate is gradually decreasing overall, as shown in Table 1. The increased error rate with the value of bins parameter of 256, may be due to the fact that the color quantization is too detailed, resulting in a bias on the fit of objective appearance overlap features while using approximate appearance overlap. The optimization method of the energy function we use is the BK graph-cut algorithm [6–8] as OneCut does. When constructing the graph corresponding to the segmentation problem, the image pixels and the occupied bins in histogram will become the nodes in the graph. As the bins parameter value increases, the number of theoretically generated bins will increase significantly, and the quantified color histogram will be more detailed. However, in practical applications, the number of occupied bins may not increase much. More importantly, since the edges connected the auxiliary nodes (corresponding to the occupied bins) in the graph to the regular nodes (corresponding to the pixels) are reduced, the density of the edges in the graph decreases. So, the BK graph-cut algorithm can get the results faster, and the mean runtime decreases, as shown Table 1. If the selected bins parameter value is large, and the resolution of the image is higher, the probability of appearing richer color in the image may increase accordingly, resulting in a significant increase in the number of occupied bins in the histogram, thus, the number of nodes in the graph can be huge and requires more memory resources.

Combining the above analyses, after weighing the runtime, error rate, and memory resources, we set the number of bins parameter automatically when using L_1 norm to approximate the appearance overlap term [16]: 128 for small images (images with resolutions below 400 × 300); 64 for medium-sized images (images with resolution between 400 × 300 and 640 × 480, the main images used in our experiments); and 16 for large images (images with resolutions above 640 × 480).

3.4. Framework and implementation

In order to make our method clearer and easier for readers to understand, we have drawn a flowchart of the method, as shown in Fig. 4. The steps of the algorithm implementation can be summarized as follows:

Input: The image to be segmented and the scribbles.

Stage 1: Prepare the data.

1: Based on the image and the scribbles, use the iFGT [44] to compute the foreground/background color PDF (in LUV space): $Pr(c|l)$;

2: Compute $M_l(x_i)$, $D_l(x_i)$, and $G_l(x_i)$ according to Eqs. (3)–(7);

3: Get ε according to Eq. (10), then specify β, α_1 and α_2 according to Eqs. (11)–(13);

4: Specify the value of Bins parameter based on the size of the image, as described in Section 3.3.3. Then the color of each pixel in the image is then quantified into a histogram to approximate the appearance overlap penalty term;

5: Compute the common smooth term $|\partial S|$ according to Eq. (9).

Stage 2: Compute the result using the BK graph-cut algorithm [6–8] .

6: Consider scribbles as a hard constraint, combine the previous subitems with the relevant weight coefficients to get the energy function of the segmentation problem, and construct the graph corresponding to the energy function;

7: Use the graph-cut to optimize the energy function, and the result of the segmentation of the graph is obtained;

8: Take the pixels corresponding to the regular nodes as the object, where the nodes are labeled as Source in the result of the segmentation of the graph.

Output: Segmentation result.

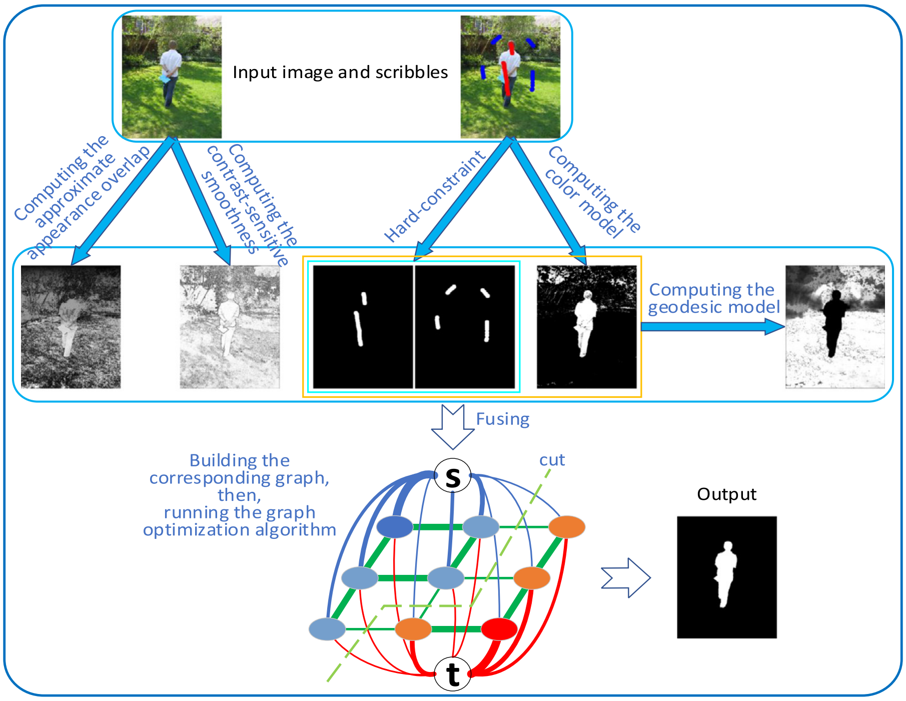

Input image and scribbles
Computing the approximate appearance overlap
Computing the contrast-sensitive smoothness
Hard-constraint
Computing the color model
Computing the geodesic model
Fusing
Building the corresponding graph, then, running the graph optimization algorithm
cut
s
t
Output

Fig. 4. The flowchart of our proposed GeoAGC. After inputting the image and scribbles, the approximate appearance overlap term, the common smoothness term, the color model, and the geodesic model can be calculated according to the relevant formula, and the scribbles are treated as a hard constraint. Then, these sub-items are combined according to the correlation weight coefficients to obtain the energy optimization function of the segmentation problem. Finally, the corresponding graph is constructed and solved by the BK graph optimization algorithm [6–8] to output the segmentation result.

3.5. Why introduce the geodesic distance information and appearance overlap information Can improve the performance of the graph-cut?

Price et al. [15] has explained in GeoGC that under the premise that the foreground/background color model is accurate, when the geodesic distance between adjacent pairs of pixels that are both on the same side of the boundary of the target to be segmented (both inside or outside the target) is shorter than the geodesic distance between adjacent pairs of pixels that go through the target boundary, the normal geodesic segmentation effect will be very good; however, when the color model is unclear, and affected by the inevitable image noise, the probability map $P_l(c)$ may be highly unstable, and in some extreme cases, the geodesic segmentation may degenerate into a simple segmentation of color-based distance, a slight difference in the location of the seeds will have a significant impact on the segmentation results. The above research by Price et al. shows that the introduction of geodesic distance information under certain conditions (precise foreground/background color model) can improve the performance of graph-cut. When the appearance overlap is severe or the appearance is cluttered, the color model is usually ambiguous and cannot accurately represent the segmentation target. In this case, in order to reduce the negative impact of the degradation of the geodesic segmentation, we introduce the appearance overlap punishment. OneCut [16] has made detailed reports on the efficiency of graph-cut using appearance overlap, but simply introducing an appearance overlap penalty can cause isolated points. According to the estimated ambiguity of the appearance model, we weigh the introduced geodesic distance information and appearance overlap information to achieve the advantages of using both geodesic distance information and appearance overlap information and to avoid their respective shortcomings. As a result, the segmentation performance is improved.

4. Experiments

We test our interactive image segmentation algorithm, GeoAGC, on a synthesis color–texture image and three image benchmarks and compare it with the other four latest methods. Images for evaluation include the natural images and medical images selected from the GrabCut dataset [9], the Convexity Database [17], and the BSD [45], altogether 550 test images, of which 152 with human segmentations as the ground-truth results, and its sizes are from 178 × 127 to 640 × 480. All the segmentation results used for this comparison are derived from official released code, and the default parameter values are not changed.

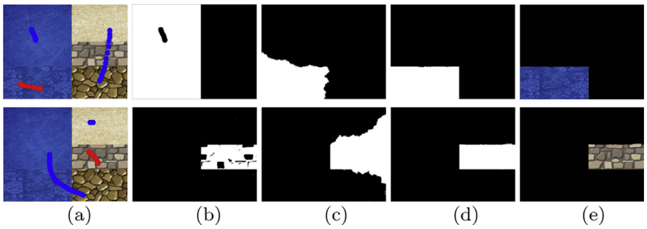


Fig. 5. Comparison of segmentation results among our approach, OneCut, and GeoGC, on synthesis color–texture images with some specific designs. (a) A challenging synthesis image with five different color–texture types. We mark the foreground with red scribbles and blue for background parts of images. (b)–(d) show the object mask separated by OneCut, GeoGC, and our method, respectively, after providing the same scribbles. (e) shows the image covered with the mask in (d) for our method.

4.1. Experiments on synthesis color–texture images

Our goal in the first row of Fig. 5(a) which is Task 1, is to segment the texture out in the bottom left of the synthesis image; in the second row of Fig. 5(a) which is Task 2, the segment target is in the middle right of the image.

In Task 1, since the colors of the top part and the bottom one on the left hand are very close to each other, and the texture difference is small, OneCut directly treats the two parts on the left as the object, removing the background labels only; GeoGC failed to segment the texture out accurately too, because the object texture and the background

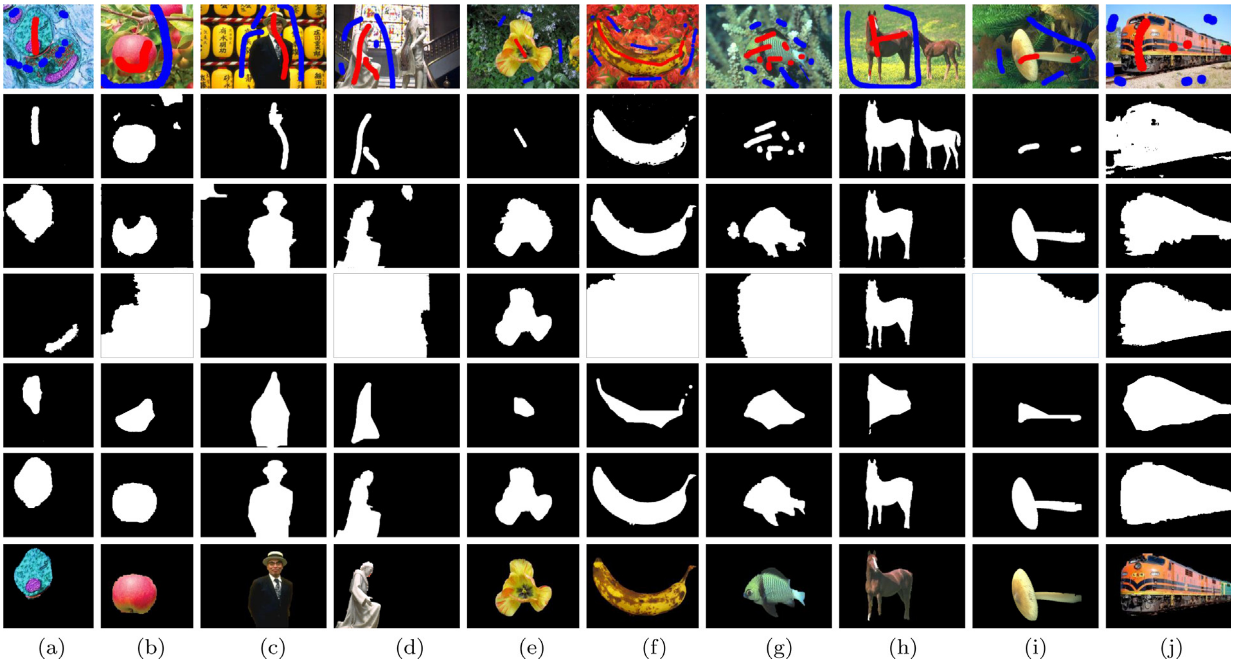


Fig. 6. Comparison results with the same user scribbles. The first row is the original images with user scribbles (red scribbles specify the foreground target and blue for background parts of images), 2–6 rows are the segmentation masks of OneCut, GeoGC, SSNCuts, TRC, and Ours respectively, the last row is the segmentation results corresponding to our segmentation mask.

share lots of same colors; our method takes three critical factors into account: the small global differences of colors; the position relations among the pixels, the seeds, and the edges; a trade-off among all the sub-items. Therefore, the object texture is segmented out successfully. In Task 2, because of the same reason, we get similar segmentation results as in Task 1.

4.2. Qualitative and quantitative comparison

4.2.1. Qualitative comparison

Because the seeded positions information and object boundary information are not specially considered in OneCut, the segmentation may fail, as Fig. 6(a), (c)–(e), (g) and (i) show. The irrational β value used in OneCut trend to the isolated points problem, while we set β adaptively, and obtain good results, as shown in Fig. 6(b), (f) and (j).

GeoGC only takes into account the relative information between the pixels in the LUV color space and the incomplete global color characteristics of the image, so it gets inaccurate segmentation. Our method adds the information of appearance overlap and makes extensive use of the color information (both locally and globally), better segmentations are obtained, as shown in Fig. 6(a)–(g), (i), (j).

Based on Normalized Cuts [46], the SSNCuts proposed by Chew et al. [19] adds the interactive constraints, and its segmentation results are not bad when the texture is simple, boundaries between the object and the background are not blurry, and the seeds are placed in proper positions. In contrast to SSNCuts, our method can easily make the object segmented, as shown in Fig. 6(a)–(d), (f), (g) and (i).

TRC is sensitive to the position of seeds, and the segmentation results are basically the approximations of the objects with convex polygons, its segmentation is not very accurate. However, our method makes comprehensive use of the geodesic distance information and the global foreground/background appearance overlap information, can segment the objects successfully with the same user scribbles, as shown in Fig. 6(a)–(j).

More examples of comparison are shown in Fig. 7.

4.2.2. Quantitative comparison and error estimation

To quantify the segmentation results, we follow common practice [47] with four criteria: (1) Dice similarity coefficient (DSC) [48], reflects the similarity between the two samples participating in the comparison after balancing the size differences; (2) Jaccard distance (JD), computes the average displacement between the segmentation and the ground truth; (3) Modified Hausdorff distance (MHD) [49], measures dissimilarity between sample sets; (4) Mean error rate (MER) [47], a statistical accuracy metrics. A segmentation is better if the JD, the MHD, and the MER are smaller when compared to the ground truth.

When evaluating the segmentation results, we should not only see how many object pixels are successfully segmented but also deal with possible classification imbalances. (In the segmentation task, the size of the foreground may be much smaller than the size of the background, that is, the foreground/background classification is imbalanced. At this point, if we merely calculate the proportion of pixels in the segmentation result that is not foreground objects relative to all pixels in the image, then even if the segmentation result is poor, the evaluation metric that does not consider the classification balance will indicate that the segmentation is not bad.) The Dice's coefficient (DSC) [48] not only can be used to compare the similarity of two samples but also considers the classification balance. It measures the ratio of the size of the overlap between the foreground segmented by the algorithm and the ground truth foreground to the total foreground size. The calculation formula is:

$$DSC = \frac{2 \times |GT \cap O|}{|GT| + |O|}, \tag{14}$$

where $|GT|$ and $|O|$ represent the area of ground truth foreground and the area of the foreground segmented by the algorithm, respectively, $|GT \cap O|$ is the common area of the two areas. The first chart in Fig. 8 presents the DSC between the segmentation result and the ground truth of each algorithm. It can be seen from the chart that the DSC of our method is the highest, followed by the GeoGC, TRC and OneCut algorithms respectively, and the segmentation effect of SSNCuts is the worst.

The Jaccard coefficient (JC) [50] calculates the ratio of the intersection size and the union size of the sample sets:

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|}. \tag{15}$$

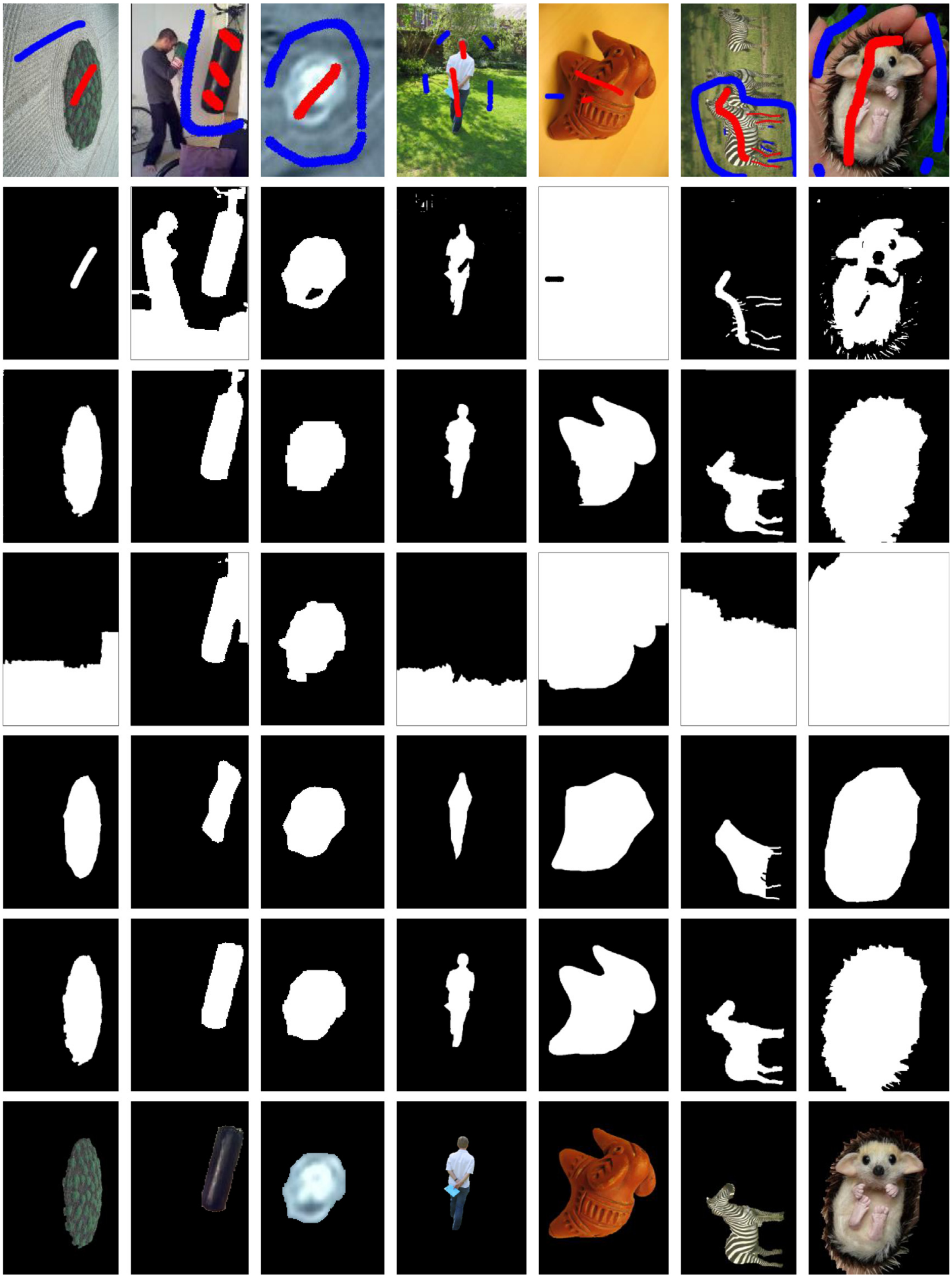


Fig. 7. Examples of qualitative results on the three image datasets with the same user strokes. Foreground strokes are colored red and background strokes are blue, and each column is an example, from top to bottom: input image with fg/bg scribbles, output from OneCut, output from GeoGC, output from SSNCuts, output from TRC, output from Ours, and the segmentation results corresponding to our segmentation mask, respectively.

When set A and set B are both empty, the union size is zero, then let $J(A, B) = 1$. Obviously, $0 \leq J(A, B) \leq 1$. The value of JC will increase as the similarity of A and B increases. Then the JD (d_J) can be computed by: $d_J(A, B) = 1 - J(A, B)$. The second chart in Fig. 8 reports the JD results. As is depicted in the chart, our segmentation results are similar to the ground truth, and almost all of the results of ours are better than others, followed by GeoGC, TRC and OneCut, and SSNCuts is the worst.

Let $A = \{a_1, \ldots, a_p\}$ and $B = \{b_1, \ldots, b_p\}$ denote two finite point sets, then the Hausdorff distance (HD) [51] can be defined as:

$$H(A, B) = \max(h(A, B), h(B, A)), \tag{16}$$

where $h(A, B) = \max_{a \in A} \min_{b \in B} \|a - b\|$, and $\|\cdot\|$ is a quantification of the norm on the two points (e.g., the 2-*norm*). A MHD brings in two dozens of possible distance measurements based on the HD between two sample sets, better measures the displacement between the segmentation

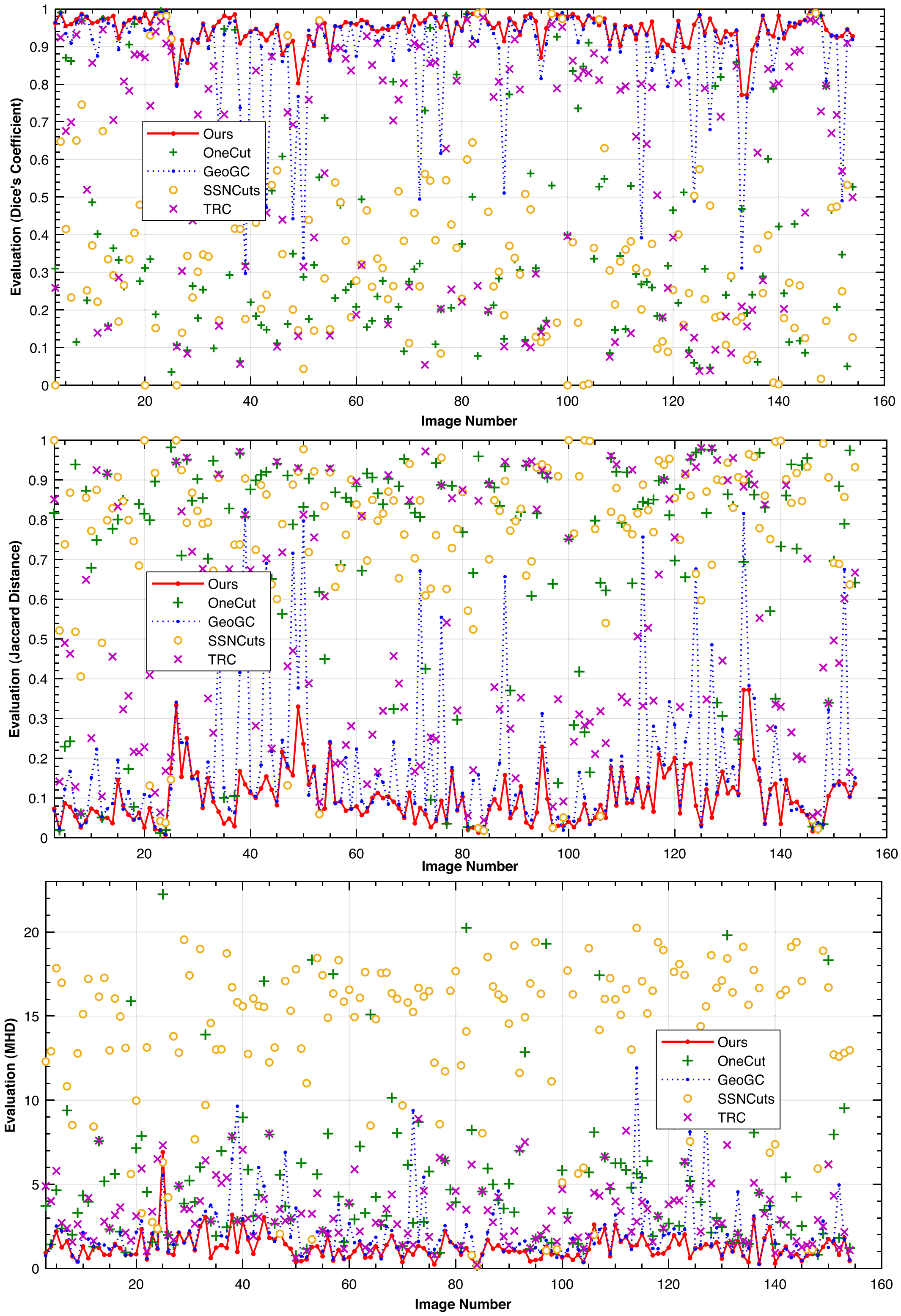


Fig. 8. Comparison between the other four approaches and our method with Dice's coefficient, Jaccard Distance, and MHD, separately.

and the ground truth than HD. The results of our method and the other four algorithms compared with the ground truth are showed in the third chart in Fig. 8. Similar to the comparison results of the JD, almost all of the results of ours are better than the other four methods.

Based on the two segmentation sets, the MER [47] is defined as the following form:

$$MER = 1 - \frac{p_b + p_o}{r \times c}, \tag{17}$$

where p_b represents the size of correctly segmented backgrounds, similarly, p_o counts the size of correctly segmented objects. The denominator $r \times c$ represents the size of the image to be segmented. Table 2 shows the details of the five algorithms with MER in minimum, maximum and average; our method is the lowest in MER relative to the other four.

Table 2
MER for Ours, OneCut, GeoGC, SSNCuts, and TRC.

Method	Min.	Max.	Avg.
Ours	**0.08%**	11.93%	**1.57%**
OneCut	0.13%	84.89%	15.81%
GeoGC	0.11%	32.19%	3.70%
SSNCuts	0.11%	**91.05%**	58.54%
TRC	0.28%	36.81%	8.33%

Table 3
Runtime for Ours, OneCut, GeoGC, SSNCuts, and TRC.

Method	Min.	Max.	Avg.
Ours	0.141	2.032	0.587
OneCut	0.233	16.019	1.914
GeoGC	**0.104**	1.671	**0.384**
SSNCuts	0.169	**182.060**	15.819
TRC	0.477	130.345	19.134

4.2.3. Time analysis

Our algorithm is very close to GeoGC in segmentation time, approximately 1 s or less for 550 testing images. Table 3 shows the details of segmentation time of each algorithm on the test images, and the SSNCuts algorithm does not take into account the time of computing the *globalized probability of boundary* (gPb, [52]). For most of the images, the segmentation time of our algorithm is only 0.3 s longer than that of GeoGC, but a few seconds less than that of OneCut. This is because our algorithm combines OneCut and geodesic distance to graph-cut, and the number of bins parameter (default 64) is selected automatically based on the size of the image. According to OneCut [16], the segmentation time is almost 1/5 of 16 bins by default in OneCut when taking the value of bins parameter to 64.

5. Conclusions and future work

We proposed an excellent interactive image segmentation using geodesic appearance overlap graph cut, which incorporates the advantages of GeoGC and OneCut; rather than a fixed combination, our method tries to take advantage of the knowledge of geodesic distance and the rich color information locally and globally, and to leverage the respective strengths based on the distinctiveness of appearance overlap, which is inferred from the user's scribbles; in many cases, our approach works even better than the two methods, with little increased time. The experiments demonstrated that our method achieves significant improvement in both speed and accuracy compared to OneCut, SSNCuts, and TRC methods. Compared with GeoGC, our method not only achieves better segmentation results with little additional runtime, but also further reduces the user's interactive workload.

In the future, we plan to explore the soft border to achieve slender target segmentation, to further decrease the sensitiveness to seeds placement. In addition, the algorithm could be extended in order to segment more than one object in the images, i.e., multi-objects segmentation.

Acknowledgment

The authors would like to sincerely thank Brian L. Price in Adobe company for unselfish guidance on our code, and the anonymous reviewers for helpful comments.

Funding

This work was supported by the National Natural Science Foundation of China (NSFC No. 11471002); Hunan Provincial Science and Technology Plan, China (No. 2013FJ4052); Platform project of the Education Department of Hunan Province, China (No. 16K054).

References

[1] A. Blake, C. Rother, M.A. Brown, P. Pérez, P.H.S. Torr, Interactive image segmentation using an adaptive GMMRF model, in: T. Pajdla, J. Matas (Eds.), Computer Vision - ECCV 2004, 8th European Conference on Computer Vision, Prague, Czech Republic, May 11-14, 2004. Proceedings, Part I, in: Lecture Notes in Computer Science, vol. 3021, Springer, 2004, http://dx.doi.org/10.1007/978-3-540-24670-1_33.

[2] J. Ning, L. Zhang, D. Zhang, C. Wu, Interactive image segmentation by maximal similarity based region merging, Pattern Recognit. 43 (2) (2010) 445–456, http://dx.doi.org/10.1016/j.patcog.2009.03.004.

[3] J. He, C. Kim, C.J. Kuo, Interactive Segmentation Techniques - Algorithms and Performance Evaluation, in: Springer Briefs in Electrical and Computer Engineering, Springer, 2014, http://dx.doi.org/10.1007/978-981-4451-60-4.

[4] Y. Boykov, M. Jolly, Interactive graph cuts for optimal boundary and region segmentation of objects in N-D images, in: ICCV, 2001, pp. 105–112, http://dx.doi.org/10.1109/ICCV.2001.10011.

[5] Y. Boykov, M. Jolly, Interactive organ segmentation using graph cuts, in: S.L. Delp, A.M. DiGioia, B. Jaramaz (Eds.), Medical Image Computing and Computer-Assisted Intervention - MICCAI 2000, Third International Conference, Pittsburgh, Pennsylvania, USA, October 11-14, 2000, Proceedings, in: Lecture Notes in Computer Science, vol. 1935, Springer, 2000, pp. 276–286, http://dx.doi.org/10.1007/978-3-540-40899-4_28.

[6] Y. Boykov, O. Veksler, R. Zabih, Fast approximate energy minimization via graph cuts, IEEE Trans. Pattern Anal. Mach. Intell. 23 (11) (2001) 1222–1239, http://dx.doi.org/10.1109/34.969114.

[7] Y. Boykov, V. Kolmogorov, An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision, IEEE Trans. Pattern Anal. Mach. Intell. 26 (9) (2004) 1124–1137, http://dx.doi.org/10.1109/TPAMI.2004.60.

[8] V. Kolmogorov, R. Zabih, What energy functions can be minimized via graph cuts? IEEE Trans. Pattern Anal. Mach. Intell. 26 (2) (2004) 147–159, http://dx.doi.org/10.1109/TPAMI.2004.1262177.

[9] C. Rother, V. Kolmogorov, A. Blake, "Grabcut": interactive foreground extraction using iterated graph cuts, ACM Trans. Graph. 23 (3) (2004) 309–314, http://dx.doi.org/10.1145/1015706.1015720.

[10] Y. Li, J. Sun, C. Tang, H. Shum, Lazy snapping, ACM Trans. Graph 23 (3) (2004) 303–308, http://dx.doi.org/10.1145/1015706.1015719.

[11] D. Freedman, T. Zhang, Interactive graph cut based segmentation with shape priors, in: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2005), 20-26, June 2005, IEEE Computer Society, San Diego, CA, USA, 2005, pp. 755–762, http://dx.doi.org/10.1109/CVPR.2005.191.

[12] O. Veksler, Star shape prior for graph-cut image segmentation, in: D.A. Forsyth, P.H.S. Torr, A. Zisserman (Eds.), Computer Vision - ECCV 2008, 10th European Conference on Computer Vision, Marseille, France, October 12-18, 2008, Proceedings, Part III, in: Lecture Notes in Computer Science, vol. 5304, Springer, 2008, pp. 454–467, http://dx.doi.org/10.1007/978-3-540-88690-7_34.

[13] S. Vicente, V. Kolmogorov, C. Rother, Graph cut based image segmentation with connectivity priors, in: 2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2008), 24-26, June 2008, IEEE Computer Society, Anchorage, Alaska, USA, 2008, pp. 1–8, http://dx.doi.org/10.1109/CVPR.2008.4587440.

[14] V. Gulshan, C. Rother, A. Criminisi, A. Blake, A. Zisserman, Geodesic star convexity for interactive image segmentation, in: The Twenty-Third IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2010, San Francisco, CA, USA, 13-18, June 2010, IEEE Computer Society, 2010, pp. 3129–3136, http://dx.doi.org/10.1109/CVPR.2010.5540073.

[15] B.L. Price, B.S. Morse, S. Cohen, Geodesic graph cut for interactive image segmentation, in: The Twenty-Third IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2010, San Francisco, CA, USA, 13-18, June 2010, IEEE Computer Society, 2010, pp. 3161–3168, http://dx.doi.org/10.1109/CVPR.2010.5540079.

[16] M. Tang, L. Gorelick, O. Veksler, Y. Boykov, Grabcut in one cut, in: IEEE International Conference on Computer Vision, ICCV 2013, Sydney, Australia, December 1-8, 2013,, IEEE Computer Society, 2013, pp. 1769–1776, http://dx.doi.org/10.1109/ICCV.2013.222.

[17] L. Gorelick, O. Veksler, Y. Boykov, C. Nieuwenhuis, Convexity shape prior for binary segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 39 (2) (2017) 258–271, http://dx.doi.org/10.1109/TPAMI.2016.2547399.

[18] J. Feng, B.L. Price, S. Cohen, S. Chang, Interactive segmentation on RGBD images via cue selection, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016, IEEE Computer Society, 2016, pp. 156–164, http://dx.doi.org/10.1109/CVPR.2016.24.

[19] S.E. Chew, N.D. Cahill, Semi-supervised normalized cuts for image segmentation, in: 2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015, IEEE Computer Society, 2015, pp. 1716–1723, http://dx.doi.org/10.1109/ICCV.2015.200.

[20] T. Wang, B. Han, J.P. Collomosse, Touchcut: Fast image and video segmentation using single-touch interaction, Comput. Vis. Image Underst. 120 (2014) 14–30, http://dx.doi.org/10.1016/j.cviu.2013.10.013.

[21] H. Yu, Y. Zhou, H. Qian, M. Xian, S. Wang, Loosecut: Interactive image segmentation with loosely bounded boxes, in: 2017 IEEE International Conference on Image Processing, ICIP 2017, Beijing, China, September (2017) 17-20, IEEE, 2017, pp. 3335–3339, http://dx.doi.org/10.1109/ICIP.2017.8296900.

[22] M. Jian, C. Jung, Interactive image segmentation using adaptive constraint propagation, IEEE Trans. Image Process. 25 (3) (2016) 1301–1311, http://dx.doi.org/10.1109/TIP.2016.2518480.

[23] W. Yu, Z. Hou, P. Wang, X. Qin, L. Wang, H. Li, Weakly supervised foreground segmentation based on superpixel grouping, IEEE Access 6 (2018) 12269–12279, http://dx.doi.org/10.1109/ACCESS.2018.2810210.

[24] T. Wang, J. Yang, Z. Ji, Q. Sun, Probabilistic diffusion for interactive image segmentation, IEEE Trans. Image Process. 28 (1) (2019) 330–342, http://dx.doi.org/10.1109/TIP.2018.2867941.

[25] F.A. Breve, Interactive image segmentation using label propagation through complex networks, Expert Syst. Appl. 123 (2019) 18–33, http://dx.doi.org/10.1016/j.eswa.2019.01.031.

[26] Z. Peng, Q. Li, Adaptive appearance separation for interactive image segmentation based on dense CRF, IET Image Processing 13 (1) (2019) 142–151, http://dx.doi.org/10.1049/iet-ipr.2018.5073.

[27] P. Subudhi, S. Mukhopadhyay, An efficient graph reduction framework for interactive texture segmentation, Sig. Proc.: Image Comm. 74 (2019) 42–53, http://dx.doi.org/10.1016/j.image.2019.01.010.

[28] Q. Wang, J. Gao, Y. Yuan, A joint convolutional neural networks and context transfer for street scenes labeling, IEEE Trans. Intell. Transp. Syst. 19 (5) (2018) 1457–1470, http://dx.doi.org/10.1109/TITS.2017.2726546.

[29] Y. Boykov, V. Kolmogorov, Computing geodesics and minimal surfaces via graph cuts, in: 9th IEEE International Conference on Computer Vision (ICCV 2003), 14-17 October 2003, Nice, France, IEEE Computer Society, 2003, pp. 26–33, http://dx.doi.org/10.1109/ICCV.2003.1238310.

[30] L. Gorelick, Y. Boykov, O. Veksler, I.B. Ayed, A. Delong, Local submodularization for binary pairwise energies, IEEE Trans. Pattern Anal. Mach. Intell. 39 (10) (2017) 1985–1999, http://dx.doi.org/10.1109/TPAMI.2016.2630686.

[31] L. Gorelick, F.R. Schmidt, Y. Boykov, Fast trust region for segmentation, in: 2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, June 23-28, 2013, IEEE Computer Society, 2013, pp. 1714–1721, http://dx.doi.org/10.1109/CVPR.2013.224.

[32] P. Krähenbühl, V. Koltun, Efficient inference in fully connected CRFs with gaussian edge potentials, in: J. Shawe-Taylor, R.S. Zemel, P.L. Bartlett, F.C.N. Pereira, K.Q. Weinberger (Eds.), Advances in Neural Information Processing Systems 24: 25th Annual Conference on Neural Information Processing Systems 2011. Proceedings of a Meeting Held 12-14 December 2011, Granada, Spain, 2011, pp. 109–117.

[33] P. Krähenbühl, V. Koltun, Parameter learning and convergent inference for dense random fields, in: Proceedings of the 30th International Conference on Machine Learning, ICML 2013, Atlanta, GA, USA, 16-21 June 2013, in: JMLR Workshop and Conference Proceedings, vol. 28, JMLR.org, 2013, pp. 513–521.

[34] K. He, G. Gkioxari, P. Dollár, R.B. Girshick, Mask. R.-C.N.N, Mask R-CNN, in: IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017, IEEE Computer Society, 2017, pp. 2980–2988, http://dx.doi.org/10.1109/ICCV.2017.322.

[35] S. Liu, L. Qi, H. Qin, J. Shi, J. Jia, Path aggregation network for instance segmentation, in: 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018, IEEE Computer Society, 2018, pp. 8759–8768, http://dx.doi.org/10.1109/CVPR.2018.00913.

[36] Z. Huang, L. Huang, Y. Gong, C. Huang, X. Wang, Mask scoring R-CNN, in: CVPR, 2019.

[37] X. Wu, Z. Zhong, J.M. Buatti, J. Bai, Multi-scale segmentation using deep graph cuts: Robust lung tumor delineation in MVCBCT, in: 15th IEEE International Symposium on Biomedical Imaging, ISBI 2018, Washington, DC, USA, April 4-7, 2018, IEEE, 2018, pp. 514–518, http://dx.doi.org/10.1109/ISBI.2018.8363628.

[38] D. Mahapatra, Semi-supervised learning and graph cuts for consensus based medical image segmentation, Pattern Recognit. 63 (2017) 700–709, http://dx.doi.org/10.1016/j.patcog.2016.09.030.

[39] M. Ullah, A. Iltaf, Q. Hou, F. Ali, C. Liu, A foreground extraction approach using convolutional neural network with graph cut, in: 2018 IEEE 3rd International Conference on Image, Vision and Computing (ICIVC), 2018, pp. 40–44, http://dx.doi.org/10.1109/ICIVC.2018.8492887.

[40] Y. Hu, A. Soltoggio, R. Lock, S. Carter, A fully convolutional two-stream fusion network for interactive image segmentation, Neural Netw. 109 (2019) 31–42, http://dx.doi.org/10.1016/j.neunet.2018.10.009.

[41] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: 2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016, IEEE Computer Society, 2016, pp. 770–778, http://dx.doi.org/10.1109/CVPR.2016.90.

[42] X. Bai, G. Sapiro, A geodesic framework for fast interactive image and video segmentation and matting, in: IEEE 11th International Conference on Computer Vision, ICCV 2007, Rio de Janeiro, Brazil, October 14-20, 2007, IEEE Computer Society, 2007, pp. 1–8, http://dx.doi.org/10.1109/ICCV.2007.4408931.

[43] A. Criminisi, T. Sharp, A. Blake, Geos: Geodesic image segmentation, in: D.A. Forsyth, P.H.S. Torr, A. Zisserman (Eds.), Computer Vision - ECCV 2008, 10th European Conference on Computer Vision, Marseille, France, October 12-18, 2008, Proceedings, Part I, in: Lecture Notes in Computer Science, vol. 5302, Springer, 2008, pp. 99–112, http://dx.doi.org/10.1007/978-3-540-88682-2_9.

[44] C. Yang, R. Duraiswami, N.A. Gumerov, L.S. Davis, Improved fast gauss transform and efficient kernel density estimation, in: 9th IEEE International Conference on Computer Vision (ICCV 2003), 14-17 October 2003, Nice, France, IEEE Computer Society, 2003, pp. 464–471, http://dx.doi.org/10.1109/ICCV.2003.1238383.

[45] D.R. Martin, C.C. Fowlkes, D. Tal, J. Malik, A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics, in: ICCV, 2001, pp. 416–425, http://dx.doi.org/10.1109/ICCV.2001.937655.

[46] J. Shi, J. Malik, Normalized cuts and image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 22 (8) (2000) 888–905, http://dx.doi.org/10.1109/34.868688.

[47] S. Qu, Q. Li, The human image segmentation algorithm based on face detection and biased normalized cuts, in: H. Zha, X. Chen, L. Wang, Q. Miao (Eds.), Computer Vision - CCF Chinese Conference, CCCV 2015, Xi'an, China, September 18-20, 2015, Proceedings, Part I, in: Communications in Computer and Information Science, vol. 546, Springer, 2015, pp. 134–143, http://dx.doi.org/10.1007/978-3-662-48558-3_14.

[48] L.R. Dice, Measures of the amount of ecologic association between species, Ecology 26 (3) (1945) 297–302, http://dx.doi.org/10.2307/1932409.

[49] M. Dubuisson, A.K. Jain, A modified hausdorff distance for object matching, in: 12th IAPR International Conference on Pattern Recognition, Conference A: Computer Vision & Image Processing, ICPR 1994, Jerusalem, Israel, 9-13 October, 1994, Vol. 1, IEEE, 1994, pp. 566–568, http://dx.doi.org/10.1109/ICPR.1994.576361.

[50] A.K. Jain, R.C. Dubes, Algorithms for Clustering Data, Prentice-Hall, 1988.

[51] D.P. Huttenlocher, G.A. Klanderman, W. Rucklidge, Comparing images using the hausdorff distance, IEEE Trans. Pattern Anal. Mach. Intell. 15 (9) (1993) 850–863, http://dx.doi.org/10.1109/34.232073.

[52] P. Arbelaez, M. Maire, C.C. Fowlkes, J. Malik, Contour detection and hierarchical image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 33 (5) (2011) 898–916, http://dx.doi.org/10.1109/TPAMI.2010.161.