

Symmetry Constraint for Foreground Extraction

Huazhu Fu, Xiaochun Cao, Zhuowen Tu, Dongdai Lin

Abstract—Symmetry as an intrinsic shape property is often observed in natural objects. In this paper, we discuss how explicitly taking into account the symmetry constraint can enhance the quality of foreground object extraction. In our method, a symmetry foreground map is used to represent the symmetry structure of the image, which includes the symmetry matching magnitude and the foreground location prior. Then, the symmetry constraint model is built by introducing this symmetry structure into the graph-based segmentation function. Finally, the segmentation result is obtained via graph cuts. Our method encourages objects with symmetric parts to be consistently extracted. Moreover, our symmetry constraint model is applicable to weak symmetric objects under the part-based framework. Quantitative and qualitative experimental results on benchmark datasets demonstrate the advantages of our approach in extracting the foreground. Our method also shows improved results in segmenting objects with weak, complex symmetry properties.

Index Terms—symmetry constraint, foreground segmentation, saliency detection, graph cuts

I. INTRODUCTION

The task of foreground segmentation [1]–[3] is to separate a given image into two distinct regions: foreground and background. Over the past years, methods employing the graph to minimize binary pairwise energy functions have become the popular standard for segmenting foreground. However, most existing methods [4]–[7] generate the graph model based on the low-level appearances, such as color, texture, and edge. They neglect some basic structural properties of the object

itself, such as the symmetry, which is commonly observed and can be used to enhance the segmented result. The advantage of symmetry structure is that it represents an intrinsic shape prior without heavy learning, which has the properties of being simple and general. Many existing works employ the geometry property into their framework to improve the processing results [8]–[13]. However, foreground segmentation with the symmetry constraint is a relatively under-explored area. In this paper, we introduce the symmetry constraint into the graph-based segmentation model, and demonstrate that the high-level geometry property could significantly improve the low-level segmentation results.

There exist previous works introducing the concept of symmetry constraints [14]–[18]. For example, [16] finds a cutting curve to divide the shape into two parts under some transformation constraints, including the symmetry one. In [17], a symmetry affinity matrix is used in a region growing algorithm. However, these methods do not focus on foreground extraction problem and the advantages of using symmetry on foreground extraction remain unclear. [18] employs the symmetry boundary as the dynamic shape prior in a level-set model. But it only deals with the symmetric objects, and is invalid for general non-symmetric objects. In contrast, our method differs from the above approaches on the following aspects: first, our method focuses on “foreground object” rather than “symmetric region”, where the symmetry constraint is only employed as the award or punishment on the foreground objects. Second, a symmetry foreground map is defined in our model to preserve the symmetry structure of image based on weak geometry constraint. Third, we provide a symmetry constraint model, which the symmetry constraint is easily integrated into the graph-based model. Our symmetry constraint model is a submodular function, which can be minimized optimally by the graph cuts.

In this paper, we begin by describing our algorithm with bilateral reflection symmetry, since the reflection symmetry is simple and common. More importantly, most natural objects could be considered as the weak bilateral reflection with the medial axis transform [19]. At the end of this paper, we also extend our model for the general non-symmetric cases based on a part-based framework.

This work was supported by National Basic Research Program of China (2013CB329305), National High-tech R&D Program of China (2013AA01A601), 100 Talents Programme of The Chinese Academy of Sciences, and Tianjin Key Technologies R&D program (No.11ZCKFGX00800). Z. Tu was supported by NSF CAREER Award IIS-0844566, NSF Award IIS-1216528, and NIH R01 MH094343.

H. Fu is with School of Computer Science and Technology, Tianjin University, Tianjin 300072, China (E-mail: hzfu@tju.edu.cn).

X. Cao and D. Lin are with State Key Laboratory Of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, China (E-mail: {caoxiaochun, ddlin}@iie.ac.cn).

Z. Tu is with Laboratory of Neuro Imaging, Department of Neurology, University of California, Los Angeles, CA 90095, USA (E-mail: ztu@loni.ucla.edu).

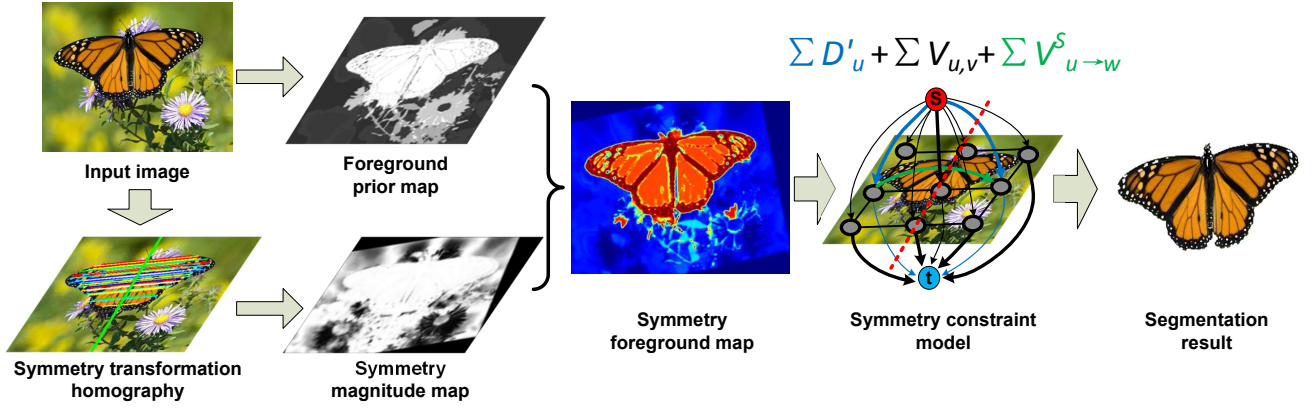


Fig. 1. The framework of foreground extraction with the symmetry constraint. Our method employs two basic maps, foreground prior map and symmetry magnitude map, to generate the symmetry foreground map. The symmetry constraint model is constructed by integrating this symmetry foreground map into a graph-based function, which can be solved via the graph cuts.

A. Related works

Symmetry detection: symmetry has been an active research area in recently [20], [21]. For example, in [22], the axis of symmetry histogram is defined to accumulate the weighted votes for all possible symmetry axes. In [23], the angular correlation, measuring the correlation between images in the angular direction, is employed to effectively detect the rotational and reflectional symmetries. [24] provides a symmetry-growing method to detect symmetry features. A descriptor of mirror SIFT (MIFT) is adopted in [25] to detect the mirror reflection. However, these local descriptors are sparse and insufficient to support the pixel-level segmentation.

On the other hand, the reflection transformation homography, which is computed with moderate accuracy by these matched descriptors, provides a global symmetry relationship for each pixel. The reflection transformation homography is linear and represented by a non-singular 3×3 matrix \mathbf{H}_T [26]. Let Λ be the lattice of an input image. For each pixel $u \in \Lambda$, the geometric symmetry coordinates $\mathbf{x}_{u'}$ is given by:

$$\mathbf{x}_{u'} = \mathbf{H}_T \mathbf{x}_u, \quad (1)$$

where $\mathbf{x}_u = (x_u, y_u, 1)^T$ denotes the homogeneous coordinate of the pixel u . This matrix provides a global symmetric relation. However, it does not address the robust matching for the weak/complex symmetry cases. In our method, we propose a symmetry foreground map to represent the geometry structure of the image. The proposed map combines the global geometry constraint and the local feature matching relationship.

Graph-based segmentation: foreground segmentation problem is often formulated as a graph-based optimization problem [7], [27]. The standard graph formu-

lation is defined as :

$$E(L) = \sum_{u \in \Lambda} D_u(l_u) + \sum_{(u,v) \in \mathcal{N}} V_{u,v}(l_u, l_v), \quad (2)$$

where $L = \{l_u | u \in \Lambda\}$ denotes the binary-valued label of the pixel. D_u is called the data term, $V_{u,v}$ is the smoothness term, and \mathcal{N} denotes the set of neighboring pixel pairs. In the segmentation problem, the data penalties D_u indicate individual label-preferences of pixels based on observed intensities and prespecified foreground function. The smoothness penalties $V_{u,v}$ encourage spatial coherence by penalizing discontinuities between the similar neighboring pixels. This function can be justified by the well-known Markov Random Fields (MRF) formulation [28], which can be solved optimally via the graph cuts [29]–[32]. In this paper, we integrate the symmetry constraint into this graph-based function, and construct the symmetry constraint model to boost the performance of the foreground segmentation.

B. Overview and contributions

Fig. 1 gives a brief overview of our approach. First, a symmetry transformation homography is obtained by the symmetry detection algorithm. In our method, we are not constrained to specific choices of the detection methods, and herein MIFT [25] is used. Then, we generate two basic maps to represent the foreground and symmetry structure. The foreground prior map is computed by saliency measure, and provides the initial foreground location. The symmetry magnitude map offers the symmetry relationship of the image. Next, the symmetry foreground map is generated to inherit the merits of these two basic maps. Finally, our symmetry constraint model is constructed based on this symmetry foreground map.

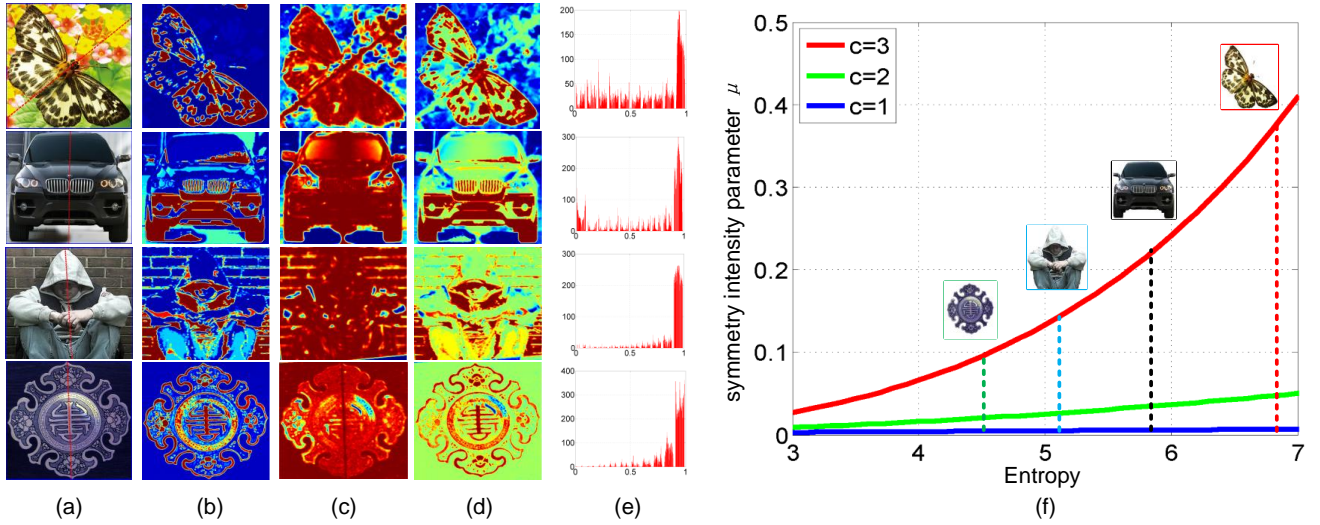


Fig. 2. (a) Four typical symmetry cases with various foregrounds and backgrounds. (b) The foreground prior map. (c) The symmetry magnitude map. (d) The symmetry foreground map. (e) The histogram of symmetry magnitude map. (f) The curve of symmetry intensity parameter μ to the entropy $E_n(\mathbf{M})$.

The main contributions of this paper are that we show studying the symmetry property significantly improves the results of foreground extraction on objects with strong, weak, and complex symmetry. This makes the symmetry constraint a general and desirable term to be included in object segmentation. In detail, we introduce a symmetry foreground map to represent the intrinsic symmetry structure of an object in pixel-level. And a symmetry constraint model is also proposed to integrate the symmetry constraint into the graph-based segmentation function. Our symmetry constraint model is a submodular function, which can be minimized optimally by the graph cuts. Moreover, we extend the symmetry constraint model into the part-based framework for the more general cases.

This paper is organized as follows: Sec. II gives the details of our symmetry foreground map. Sec. III proposes the symmetry constraint model. The extended part-based framework is provided in Sec. IV. Then the quantitative and qualitative experimental results are shown in Sec. V. Finally, a concluding remark is presented in Sec. VI.

II. SYMMETRY FOREGROUND MAP

In our work, we propose a symmetry foreground map (SFM) to represent the symmetry and foreground priors of the image. SFM includes two basic maps, the foreground prior map and the symmetry magnitude map. SFM provides three main information: the initial foreground location, the pixel-level symmetry intensity, and the geometry matching relationship.

A. Foreground prior map

The first basic map is foreground prior map. The foreground is often defined as a special object, which has distinct properties different from the background [2], [33], [34]. Most existing methods employ the user interactions to reduce the ambiguity caused by complex appearances and weak boundaries [5]–[7]. Recently, some works release the burden of user interactions to automatically detect the foreground proposals via the objectness [35]–[37] and saliency [38]–[42]. For example, [43] produces a bag of regions and ranks them to select the good segmentations of general objects. [33] employs the visual attention as the initial input to extract the foreground. In this paper, we employ the saliency concept to automatically generate the foreground prior map, which offers the initial foreground location.

An effective saliency method [42] employs the color contrast of the pixel to all other pixels in the image as the saliency cue. In [42], all the pixels are quantified into 1728 bins. However, the practical bin number of one image is less than 1728 bins, and the estimated color histogram has discontinuities at the bin edges. To obtain a highly cohesive global constraint of the image, we modify this method by measuring the color contrast on cluster rather than on histogram. Firstly, we group the pixels of the image into C clusters by K-means. The clusters are denoted by a set of RGB appearance $\{Z^c\}_{c=1}^C$, in which Z^c denotes the prototype (cluster center) associated with the cluster c . We then define the

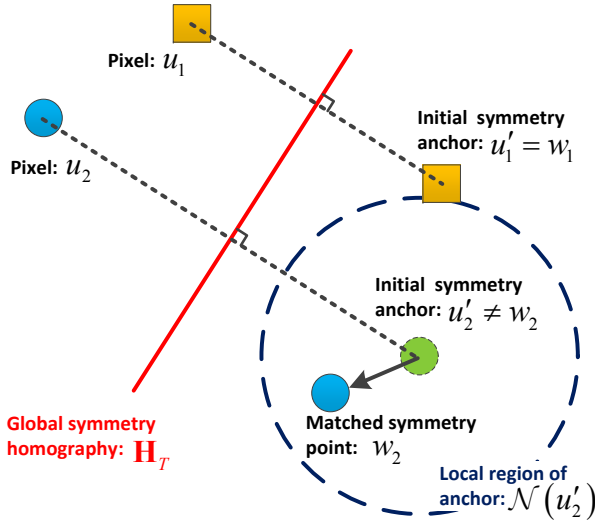


Fig. 3. The illumination between the initial symmetry anchor and matched symmetry point. The global symmetry homography \mathbf{H}_T only provides the initial symmetry anchor u'_i of pixel u_i . Our symmetry magnitude map considers the local matching quality, and achieves an optimal matched symmetry point w_2 of pixel u_2 . The region $\mathcal{N}(u'_2)$ denotes the searching region of the anchor u'_2 .

contrast saliency value of cluster c as follows:

$$s(c) = \sum_{i=1, i \neq c}^C \left(\frac{m^i}{M} \|Z^c - Z^i\|_2 \right), \quad (3)$$

where a L_2 norm is used to compute the distance on the feature space, m^i represents the pixel number of cluster i , and M denotes the image pixel number. Eq. (3) means the saliency of a pixel is defined using its cluster contrast to all other clusters.

Fig. 2 (b) shows the foreground prior map $\mathbf{S} = [\tilde{s}(c)]$, where $\tilde{s}(c)$ means the normalized saliency value. The saliency detection is based on the bottom-up cue, and produces an initial foreground prior. But it performs poor for the complex or textured background. The main contribution of the foreground prior map focuses on providing the initial location of the foreground and depressing the backgrounds.

B. Symmetry magnitude map

The second basic map is the symmetry magnitude map. As described above, the global symmetry mapping is computed by symmetry homography \mathbf{H}_T . However, the global symmetry mapping lacks the local robustness. Therefore, we generate a symmetry magnitude map, which combines the global geometry constraint and the local feature matching. Firstly, a initial symmetry anchor u' of pixel u is computed by Eq. (1). This initial anchor u' satisfies the global geometry of the pixel u . Then the

symmetry weight $W(v')$ for each neighboring point v' of the symmetry anchor u' is computed by:

$$W(v') = \frac{-\|z_u - z_{v'}\|_2}{\mathcal{G}[\text{dis}(\mathbf{x}_{u'}, \mathbf{x}_{v'}) \mid 0, \sigma^2]}, \quad (4)$$

where $\mathbf{x}_u = (x_u, y_u, 1)^T$ denotes the homogeneous coordinates of the pixel u , z_u denotes the RGB appearance of pixel u , a L2 norm is used to measure the matching quality on color space, Gaussian kernel \mathcal{G} plays distance penalty on Euclidean distance $\text{dis}()$, and σ is the radius of neighborhood. This formula represents the symmetry property on two aspects: the global geometry constraint and local matching quality. In other words, it is considered as a cost function for the searching of the optimal symmetrically matched peer. As a result, the optimal matched symmetry point w is obtained by maximizing the symmetry weight W in the neighboring region of the initial symmetry anchor u' :

$$u \rightarrow w = \arg \max_{w \in \mathcal{N}(u')} \{W(w)\}, \quad (5)$$

where $\mathcal{N}(u')$ is the local region of the anchor u' . The initial symmetry anchor u' only presents the initial symmetric mapping of u . The matched symmetry point w is the local optimal symmetry matching of u . This local searching provides more robustness to the weak symmetry case, as shown in Fig. 3.

Finally, we define the symmetry magnitude value $m(u)$ of pixel u as:

$$m(u) = W(w), \text{ with } u \rightarrow w. \quad (6)$$

Fig. 2 (c) show the samples of symmetry magnitude map $\mathbf{M} = [\tilde{m}(u)]$, where $\tilde{m}(u)$ means the normalized symmetry magnitude in $[0, 1]$. The symmetry magnitude map produces a dense pixel-level symmetry matching relationship. However, its power degrades in the situation of the similar background, such as the last two rows of Fig. 2 (c).

C. Symmetry foreground map

Either foreground prior map or symmetry magnitude map, if used independently, has its advantages and, of course, disadvantages. Therefore, we define a symmetry foreground map to combine these two basic maps. A common fusion method is formulated as a pixel-wise summation:

$$\mathbf{H}'_s = \frac{\mathbf{S} + \mathbf{M}}{2}, \quad (7)$$

where \mathbf{S} and \mathbf{M} denote the foreground prior and symmetry magnitude maps, respectively.

There are four typical symmetry cases with various foregrounds and backgrounds. Fig. 2 (a) gives an illustration. The first row observes a symmetric foreground, i.e. the butterfly, and non-symmetric background. The second row displays a symmetric foreground which is compact inside the region. The third row has a weak symmetric foreground and symmetric background. The last row demonstrates symmetric foreground and background. Fig. 2 (d) show the results of SFM, which represents the local symmetry and foreground priors. However, SFM could not deal with the cases of symmetric background, such as the last two rows of Fig. 2. For these cases, the constraint of SFM in segmentation model should be weakened. Therefore, we introduce a symmetry intensity parameter to control the constraint strength of SFM.

In practice, we observe that the symmetry magnitude map is easily corrupted by the symmetric backgrounds. We formulate the values of symmetry magnitude map into a 256 bin histogram, such as Fig. 2 (e). It observes that the sample of non-symmetric background has more scattering histogram distribution than that of symmetric background. For simplification, we utilize the entropy to represent the global distribution. The symmetry intensity parameter μ is defined as:

$$\mu = \exp \left\{ \frac{E_n(\mathbf{M})}{a} \right\} - b, \quad (8)$$

where $E_n(\mathbf{M})$ denotes the entropy of the symmetry magnitude map \mathbf{M} . The domain of entropy E_n with 256 bins is $[0, 8]$. The hyperparameters (a, b, c) control the distribution of parameter μ and normalize it into $(0, 1)$. In our experiment, $a = 1000$ and $b = 1$. Fig. 2 (f) shows the curve of symmetry intensity parameter μ with respect to the entropy $E_n(\mathbf{M})$. We also plot the curves of Eq. (8) with various values of c in Fig. 2 (f). It can be seen that the curve with $c = 3$ has the better gradient variation.

Finally, we reform SFM \mathbf{H}_s with symmetry intensity parameter μ as:

$$\mathbf{H}_s = 2\mu \frac{\mathbf{S} + \mathbf{M}}{2} + 1 - \mu. \quad (9)$$

The domain of SFM \mathbf{H}_s is limited into $(1 - \mu, 1 + \mu)$ in Eq. (9). In general, the parameter μ controls the symmetry constraint in the symmetry constraint model. If the background observes strong symmetry, $E_n(\mathbf{M})$ can help in deemphasizing the symmetry constraint by assigning the intensity parameter μ to a small value. And therefore, the MRF penalty could be dominant in the symmetry constraint model.

III. SYMMETRY CONSTRAINT MODEL

In this section, we integrate our SFM into the graph-based model to account for the symmetry property of the foreground object, which is called ‘‘symmetry constraint model’’.

A. Data term

In the traditional graph-based formulation as shown in Eq. (2), the data term D_u can be obtained based on the foreground and background seeds, which are usually described as the source s and the sink t :

$$D_u(u) = \mathcal{V}(u; l_u), \quad \text{with } l_u = \{s, t\}, \quad (10)$$

where $\mathcal{V}(\cdot)$ evaluates the fit of the opacity distribution seeds s and t to the pixel u .

In our model, we employ SFM as a tuning parameter to enhance the symmetry foreground extraction. The modified data penalty D'_u integrating symmetry constraint is rewritten as:

$$D'_u(u) = \begin{cases} \mathcal{V}(u; l_u) H_s(u) & , \text{if } l_u = s \\ \mathcal{V}(u; l_u) [2 - H_s(u)] & , \text{if } l_u = t \end{cases}, \quad (11)$$

where $H_s(u)$ is SFM value of pixel u . In our modified data term, the symmetric pixels are encouraged to be assigned to the foreground, while non-symmetric pixels are more likely to be labeled as the background. For the non-symmetric foreground and the symmetric background, the value of μ tends to be 0 and the modified data penalty degenerates to the traditional segmentation model.

B. Smoothness term

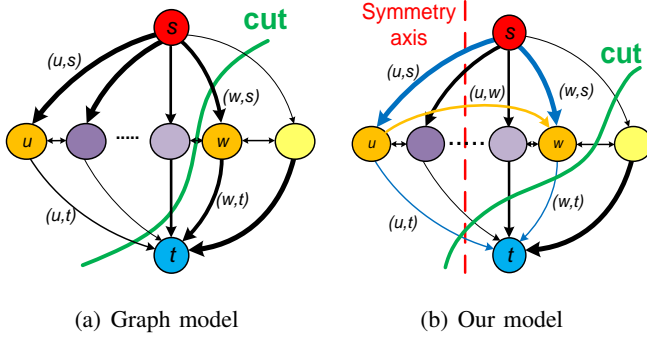
For smoothness term, we use a linear combination to integrate the symmetry constraint, which is represented as:

$$\sum_{\{u,v\} \in \mathcal{N}} V_{u,v}(l_u, l_v) + \sum_{u \in \Lambda} V_{u \rightarrow w}^S(l_u), \quad (12)$$

where $u \rightarrow w$ denotes that the pixel w is the optimal symmetry matching of pixel u obtained by Eq. (5). The smoothness term $V_{u,v}$ is basically unchanged from Eq. (2). The symmetry smoothness term $V_{u \rightarrow w}^S$ presents the constraint of symmetry and ensures the symmetric pairs to be assigned the same labels:

$$V_{u \rightarrow w}^S(l_u) = \mu H'_s(u) \cdot V_{u,w}(l_u, l_w), \quad (13)$$

where $H'_s(u)$ denotes the initial SFM value of pixel u in Eq. (7), and μ is the symmetry intensity parameter in Eq. (8). Different from the domain of SFM $(1 - \mu, 1 + \mu)$ in Eq. (9), the domain of term $\mu H'_s(u)$ in Eq. (13) is $[0, \mu]$. The meanings of this weight $\mu H'_s(u)$ are as following:



Edge	Graph model	Symmetry constraint model
(u, s)	$\mathcal{V}(u; s)$	$\mathcal{V}'(u; s) = \mathcal{V}(u; s)H_s(u)$
(u, t)	$\mathcal{V}(u; t)$	$\mathcal{V}'(u; t) = \mathcal{V}(u; t)[2 - H_s(u)]$
(w, s)	$\mathcal{V}(w; s)$	$\mathcal{V}'(w; s) = \mathcal{V}(w; s)H_s(w)$
(w, t)	$\mathcal{V}(w; t)$	$\mathcal{V}'(w; t) = \mathcal{V}(w; t)[2 - H_s(w)]$
(u, w)	N/A	$V_{u \rightarrow w}^S(l_u) = \mu H'_s(u) \cdot V_{u,w}(l_u, l_w)$

Fig. 4. The example of the graph-based model. In the graph, the node u has the optimal symmetry matching node w . The red dotted line denotes the symmetry relationship. The thickness of line denotes the penalty of this edge. The newly introduced yellow edge in (b) is the symmetry smoothness penalty from u to w . The table shows the change of the edges.

first, it awards the symmetric foreground and background simultaneously, due to the \mathbf{M} term in Eq. (7). Second, the foreground pixels is encouraged to have the same label due to the \mathbf{S} term in Eq. (7). Third, the parameter μ weakens the contribution of this smooth term, when the background shows strong symmetry.

C. Segmentation energy function

Finally, our symmetry constraint model joints the traditional graph-based term and the symmetry constraint as:

$$\mathbf{E}_s(L) = \sum_{u \in \Lambda} D'_u(l_u) + \sum_{\{u,v\} \in \mathcal{N}} V_{u,v}(l_u, l_v) + \sum_{u \in \Lambda} V_{u \rightarrow w}^S(l_u). \quad (14)$$

The segmentation can be estimated as:

$$\mathbf{L}^* = \arg \min \mathbf{E}_s(L). \quad (15)$$

Fig. 4(a) shows the structure of typical graph-based model. Fig. 4(b) shows our symmetry constraint model, where the node u has a symmetry matching to node w . The red dotted line denotes the symmetry relationship. Comparing with Fig. 4(a), the blue edges are the data penalties involving SFM \mathbf{H}_s , and a new yellow edge is the symmetry smoothness penalty of $V_{u \rightarrow w}^S$. Our SFM \mathbf{H}_s is not bijective, and the symmetry smoothness penalty is a directed edge from u to w .

Property: The energy function $\mathbf{E}_s(L)$ defined in Eq. (14) is submodular, and hence the proposed energy minimization can be solved via the graph cuts.

The domain of SFM \mathbf{H}_s is $[1 - \mu, 1 + \mu]$, and the intensity parameter $\mu \in (0, 1)$. For the data term of symmetry constraint model, the added coefficients $H_s(u)$ and $(2 - H_s(u))$ in Eq. (11) do not change the sign of data term in MRF, and this term is regular. For the smoothness term, the symmetry smoothness penalty $\sum V_{u \rightarrow w}^S$ in Eq. (14) is treated as a special smoothness term connected two non-adjacent nodes with the scale $\mu H'_s(u)$. So the energy function (14) is submodular and can be solved optimally via the graph cuts.

IV. PART-BASED FRAMEWORK EXTENSION

Now, we discuss how our method is extendable to the general objects. We address this problem from two aspects.

For the weak symmetric objects: as described early, our model provides a weak constraint, which encourages symmetric points to be assigned as the foregrounds, and simultaneously, suppresses the non-symmetric points rather than discards them. With the intensity parameter μ in Eq. (9), our model depresses the effect of symmetry constraint for the worst non-symmetry cases. Moreover, due to the smoothness term of the graph-based model, the non-symmetric pixels (node) with strong neighboring edge could be segmented correctly as usual.

For the complex/non-symmetric objects: in general, most real objects can be decomposed into a set of skeletal parts by the medial axis transform [19], [44]. For example, the curved symmetry pattern [12] with a non-linear symmetry transformation is approximately represented by integrating the local symmetry. The objects, without obvious symmetry structure, could be partitioned into a set of symmetric parts with skeletal skeleton and their connections [45]. Inspired by this part-based method, we extend our method into the local symmetry, which is suitable for the complex/non-symmetry cases, as shown in Fig. 5. The part-symmetry detection (or medial axis detection) [19] is employed to obtain the local symmetry structure. For each local symmetry region $\{R_k\}_{k=1}^K$, we compute the corresponding local symmetry magnitude map $\{M_k(u)\}_{k=1}^K$, where K denotes the number of symmetry regions. The global symmetry magnitude map \mathbf{M}^G is generated by merging the multiple local maps:

$$M^G(u) = \begin{cases} \max\{M_k(u)\} & , \text{ if } u \in \{R_k\}_{k=1}^K, \\ 0 & , \text{ otherwise.} \end{cases} \quad (16)$$

With the global symmetry magnitude map \mathbf{M}^G , the other steps of our method is the same, and foreground is extracted.

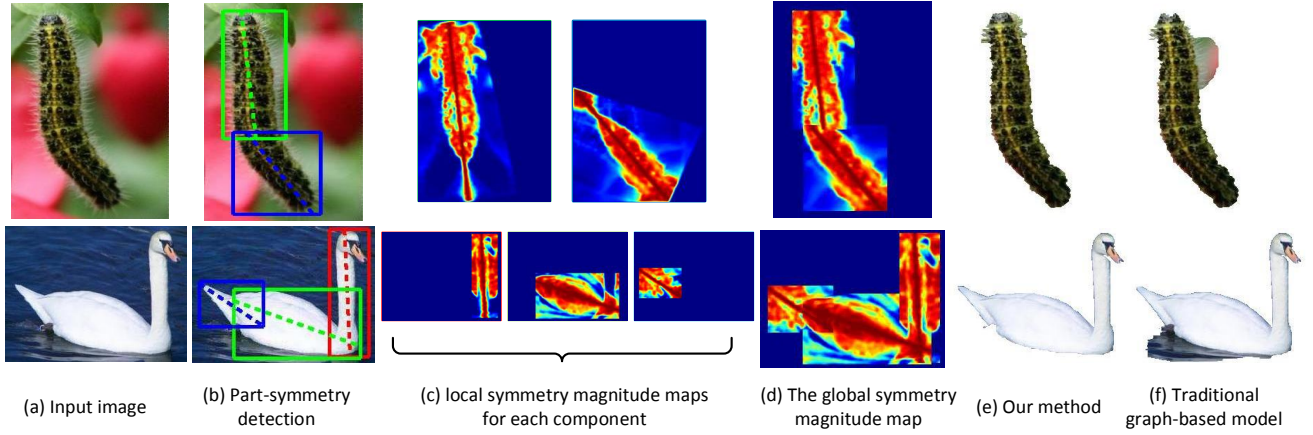


Fig. 5. The part-based framework for complex/non-symmetric objects. (b) The local symmetry axis (or medial axis) is detected, where the local symmetry region is marked by rectangle. The symmetry axis is denoted by dotted line. (c) The local symmetry magnitude map is computed for each symmetry component. (d) The global symmetry magnitude map is obtained by merging the local symmetry magnitude maps. With the part-based framework, our symmetry constraint method (e) achieves better result than traditional graph-based model (f).

V. EXPERIMENTS

In this section, our experiments address three main aspects: 1) evaluating the effectiveness of each component in our model, 2) comparing our method with traditional graph-based model and other foreground segmentation methods [18], [33], [43], 3) discussing the robustness of our method on incorrect symmetry detection.

The experiments employ our part-based framework, since the symmetric object is considered as a special one-part case. The practical procedure of our method is that: firstly, it seeks a global symmetry axis of the object using [20]. If that fails, we employ the axes of part-based symmetry using [19]. The number of parts is determined by [19]. Then, our SFM is computed, and the symmetry constraint model is constructed using Eq. (9). Finally, the foreground is extracted via graph cuts [27], [29].

A. Effectiveness of each component

We evaluate the effectiveness of each component. Our symmetry constraint model has two components: the data penalty term integrating SFM and the symmetry smoothness penalty. Fig. 6 (c) shows the segmentation results using traditional graph-based model without any symmetry constraint, which is denoted as $D_v + V_{u,v}$. In this case where the background and foreground are similar, the results are typically unsatisfactory. Under-segmentations (the shadow and partial ceiling in the first image, and flowers in the second image) and over-segmentations (the headlight of car) might happen. Fig. 6 (d) shows the segmentation results by only combining the SFM with the data penalty ($D'_u + V_{u,v}$). The under-segmentations are relieved, because they are non-symmetric regions, which have lower data penalties to foregrounds. Fig. 6

(e) shows that the symmetry smoothness penalty itself does not obtain satisfactory segmentation results. This is expected, since the merit of our symmetry smoothness penalty is to enhance the relation between the symmetry pair and encourage them to be assigned the same labels. Adopting $(D'_u + V_{u,v} + V_{u \rightarrow w}^S)$ shown in Fig. 6 (f), the foregrounds are best segmented under our complete symmetry constraint model in Eq. (14). The over-segmentations (the motorcade of car, and left wing of butterfly) in Fig. 6 (d) are now corrected.

B. Qualitative performance

In our model, the symmetry constraint enables most images to achieve the pleasing segmentation without iteration, so we employ the traditional MRF model as the baseline (MRF model). In this experiment, the Gaussian Mixture Model (GMM) is employed to describe the data cost of foreground/background seeds [7]. The data term of MRF model [5] is defined as:

$$D_u = \log p(z_u | GMM_b) - \log p(z_u | GMM_f), \quad (17)$$

where GMM_f/GMM_b are the GMMs of foreground/background, and z_u denotes the appearance of pixel u . We implement our model into the this framework, and the data term is modified as:

$$D'_u = (2 - H_s(u)) \log p(z_u | GMM_b) - H_s(u) \log p(z_u | GMM_f), \quad (18)$$

where H_s is the symmetry foreground map. The smoothness term is defined the same as in [5]:

$$V_{u,v} = \frac{\lambda \exp(-\beta \|z_u - z_v\|^2)}{\text{dis}(u, v)}, \quad (19)$$

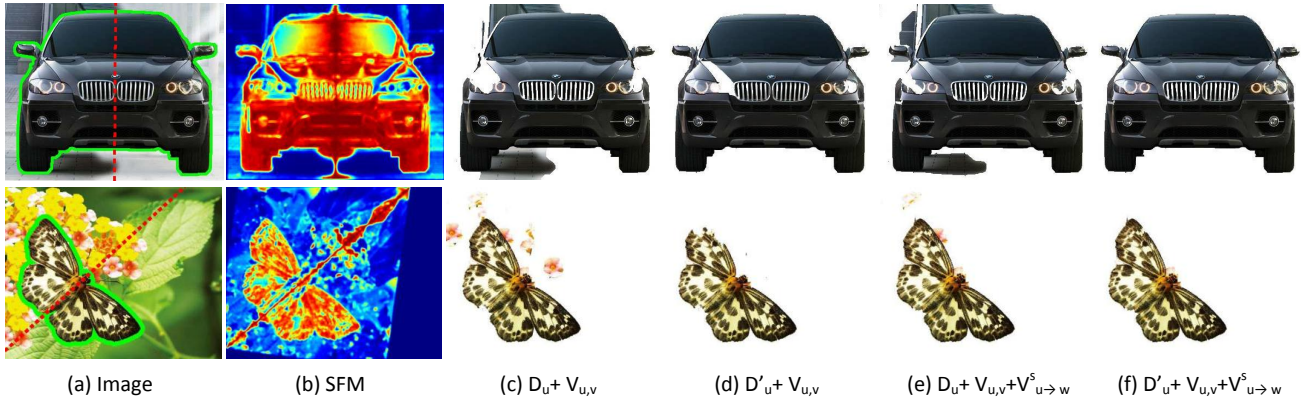


Fig. 6. The results using different combinations of components in our model. From left to right: the input image with labelled symmetry axis (red dotted line) and ground truth (green line), our SFM, the segmentation using traditional graph-based model, using the symmetry constraint in data penalty, using the symmetry smoothness penalty, and using the full power of our symmetry constraint model.

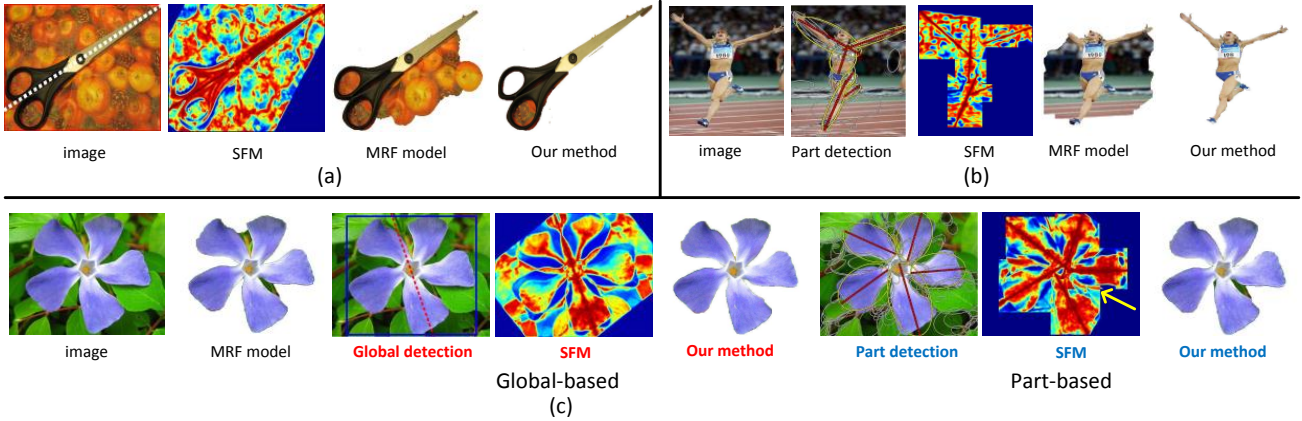


Fig. 7. Segmentation results on various symmetry cases. (a) The simple reflection symmetry is expressed by one symmetry axis. (b) The object without reflection symmetry is decomposed into a set of symmetric components. (c) The rotation case can be represented by both single symmetry and part-based models.

where $dis(\cdot)$ is the Euclidean distance of neighboring pixels. $\beta = \left(2 \langle (z_u - z_v)^2 \rangle\right)^{-1}$ where $\langle \cdot \rangle$ denotes expectation over the image. λ is the weight for the contrast sensitive term. Because in this experiment, we focus on whether the high-level symmetry term could improve the traditional low-level segmentation. The baseline in this experiment is different to other experiments (e.g. [5], [7]), in that we only employ a single rough bounding box as the input to initialize the data cost rather than user scribbles.

Fig. 7 shows three segmentation results. The first is the symmetric object, which is easily expressed by a single symmetry axis, as shown in Fig. 7 (a). Compared with MRF model, our symmetry constraint model significantly outperforms on penalizing the non-symmetric background. Fig. 7 (b) shows the object without global symmetry structure. Thanks to the part-based symmetry detection [19], the global SFM is obtained. Our method

rewards the symmetric parts, such as the arm of the athlete, which are over-segmented by MRF model. Furthermore, Fig. 7 (c) shows the object can be represented by both single and part-based models, e.g. the rotation symmetry. It can be seen that both single and part-based models outperform baseline. In this sample, the single model obtains a better segmentation than the part-based model. One possible reason is that the background between leaves appears the local symmetry structure, which hurts the local symmetry foreground map (see the yellow arrows in part-based SFM of Fig. 7 (c)).

C. Performance on datasets

For quantitative experiments, we evaluate our method and baseline (MRF model) on two benchmark datasets. We also compare our method with other two foreground segmentation methods [33], [43]. [43] produces a set of regions and ranks them, where the top-ranked regions

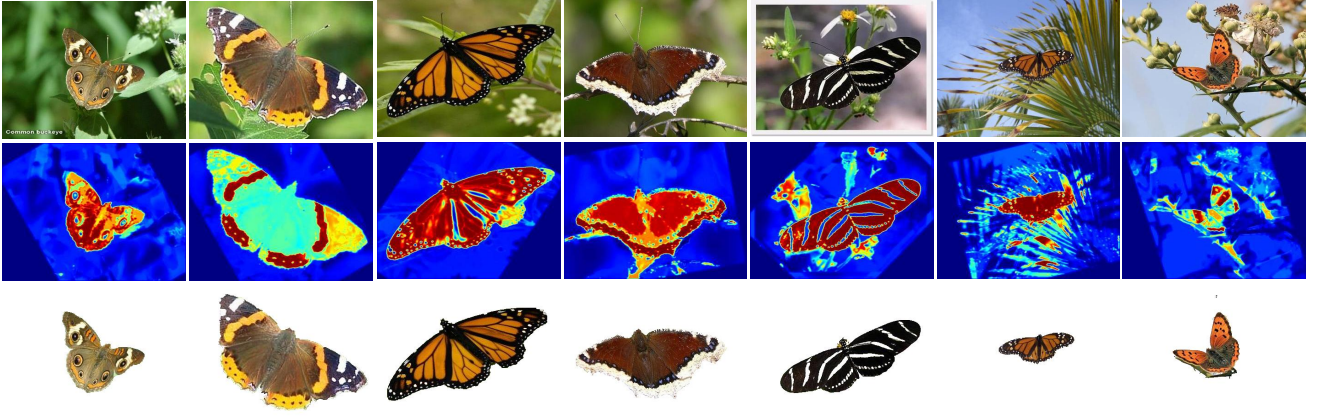


Fig. 8. Segmentation results on butterfly dataset. From top to bottom: The input image, SFM, and our segmentation result.

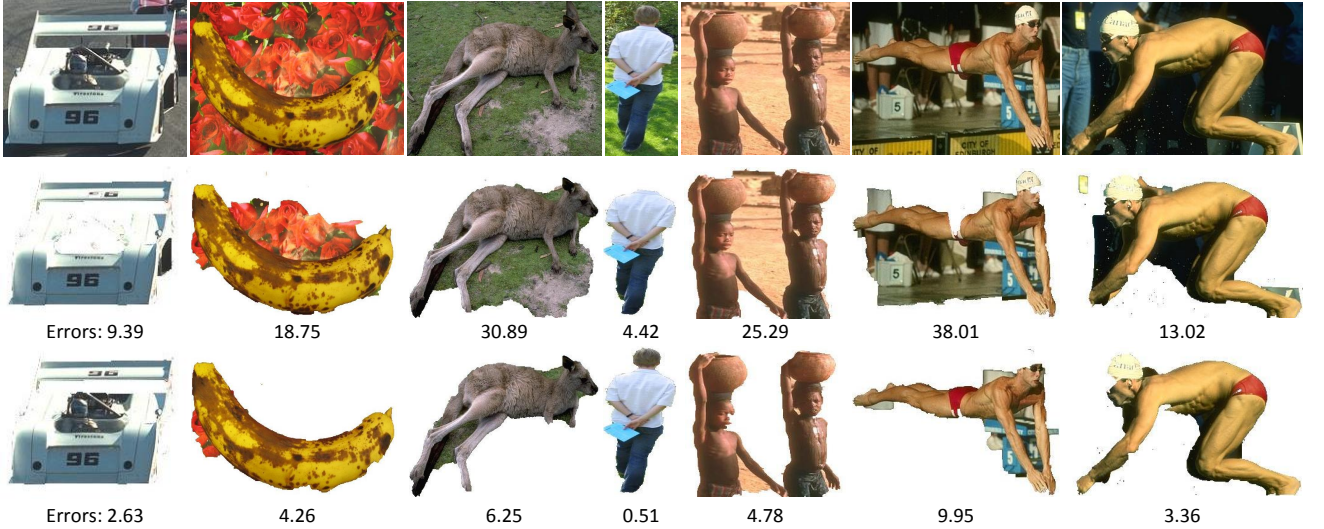


Fig. 9. Some results of MRF model (middle) and our method (bottom), where the error rate is given under each image. The symmetry constraint improves the segmentation accuracy.

TABLE I
AVERAGE SEGMENTATION ERROR RATES (%) OF OUR METHOD AND OTHER METHODS ON TWO DATASETS. THE BEST ACHIEVED RESULTS ARE INDICATED WITH BOLD FONT IN EACH ROW.

	MRF model	[33]	[43]	Out method		
				$V_{u \rightarrow w}^S$	D'_u	$D'_u + V_{u \rightarrow w}^S$
Butterfly Dataset	13.73	12.84	14.78	12.56	10.74	8.83
GrabCut Dataset	11.39	10.58	12.40	12.14	9.81	9.21

are likely to be good segmentations of the objects. We select the top rank region as the output of [43]. [33] employs the fixation point as an identification marker on the object, and segments the foreground by finding the optimal closed contour in the polar space. We employ the center of the foreground groundtruth as the fixation point in [33] to extract the foreground. Moreover, the error rate in the experiments uses the percentage of mislabeled

pixels, as in previous works [5], [7].

The first benchmark dataset is the Leeds Butterfly Dataset [46], which comprises 832 images, with the groundtruth masks for each image. The butterfly is a common symmetric object in the nature. Fig. 8 shows the segmentation results. Our SFM represents the structure of symmetry, and offers the accurate segmentation mask for the symmetry constraint model. Without surprise, for the tiny objects or complex backgrounds in images, such

as the last three columns in Fig. 8, our SFM has some noises. However, for the symmetry constraint model integrating SFM as the weak constraint, the final result eliminates these disturbances and obtains the pleasing segmentation results. The performance comparison on this dataset is shown in the first row of Tab. I, where from left to right are: the results of MRF model, [33], [43], only using the symmetry smoothness penalty ($V_{u \rightarrow w}^S$), only using the symmetry constraint in data penalty (D'_u), and our complete symmetry constraint model ($D'_u + V_{u \rightarrow w}^S$). The error rate shows that our method has the obvious improvements on this benchmark, and exceeds 5% with respect to the MRF model. The performance of each component agrees with the observation in Sec. V-A. [43] performs the worst. The reason may be that [43] is based on a generally learning function. The best object segmentation may be ranked well, but is not necessarily the first rank, which is used as the output result in our experiment.

The second benchmark dataset employed in our experiment is the GrabCut dataset [5], which is a widely used dataset including complex/non-symmetric objects and real world images. Some results are shown in Fig. 9, where the error rate is given below each image. With the symmetry constraint, our segmentation obtains better error rates. The second row of Tab. I shows the quantitative results, where our symmetry constraint model achieves 2% less error rate than the MRF model. The results of using only the symmetry smoothness penalty ($V_{u \rightarrow w}^S$) has degeneration with respect to the MRF model on this benchmark. One possible reason is that the strong symmetry mapping without data term constraint leads to over-segmentation on the weak symmetric parts.

D. Comparison with symmetry level-set segmentation

The method based on level-set using symmetry constraint is provided in [18], which uses a level-set-based curve evolution technique with the symmetry counterpart to provide a dynamic shape prior. As described in Sec. I-A, our method is different from [18]. Our symmetry constraint model deals with “foreground object” rather than “symmetric object”. Thus, the symmetry constraints are only employed as the award/punishment to the symmetric/non-symmetric objects. Thanks to this weak constraint, our method also enhances the segmentation for the objects with complex/non-symmetric properties, which is invalid for [18]. Furthermore, for the symmetry, our method obtains a better segmentation, as shown in Fig. 10, where we test the images used in [18] for accurate comparison. In the first sample, our method achieves accurate segmentation of the spots on the butterfly wing,

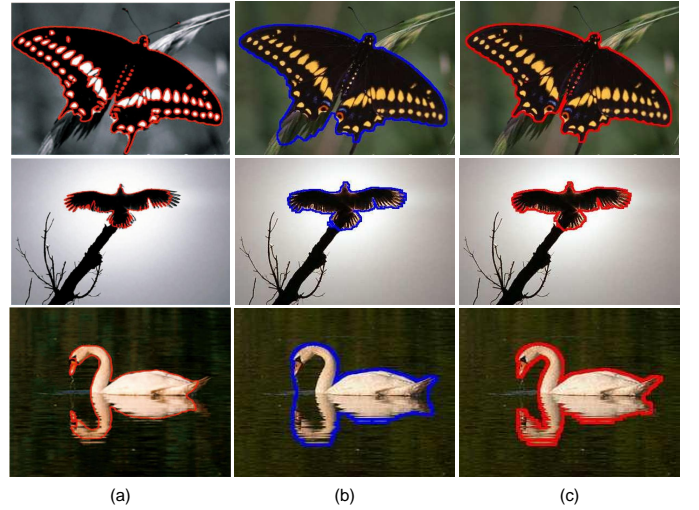


Fig. 10. The comparison between our algorithm and symmetry level-set segmentation [18]. (a) Segmentation results in [18]. (b) Segmentation results by MRF model. (c) Our segmentation results.

which are over-segmented in [18]. In the second image, the right wing of eagle is over-segmented by [18]. In contrast, our SFM has sufficient robustness for weak symmetry. And in the last sample, our method preforms better than MRF model on the neck of swan, and is comparable to [18].

E. Robustness discussion

The symmetry detection method (e.g. MIFT [25]) is employed as the prior in our method. When the symmetry detection has errors, our SFM may produce unmatched regions. Fig. 11 shows some samples of non-symmetric objects with incorrect medial axes (without part-based symmetry detection), where the incorrect medial axes (red dotted lines) result in unmatched pixels in SFM. However, thanks to the symmetry intensity parameter μ , the range of symmetry foreground map in Eq. (9) is limited, and the symmetry constraint is depressed. Simultaneously, the weak symmetry constraint on the symmetric parts obtains the satisfying discrimination, which guarantees that our method still achieves the better result than MRF model.

VI. CONCLUSION

We have presented a novel symmetry constraint model for extracting the foreground object. The symmetry constraint has been addressed on three aspects: the global symmetry constraint, the local feature matching quality, and the symmetry intensity parameter. The encouraging experiments demonstrated that the high-level geometry structure property significantly improves the low-level segmentation results. The introduction of symmetry

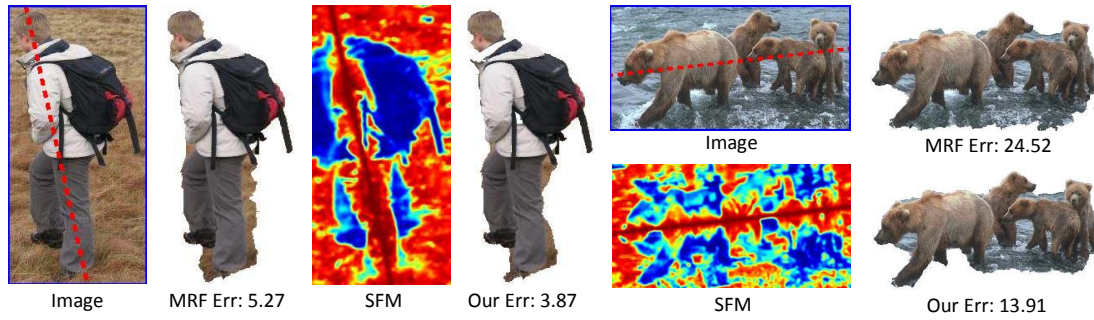


Fig. 11. The incorrect medial axes (red dotted lines) lead to lots of unmatched pairs in SFM, which affects the power of the symmetry constraint. However, our symmetry constraint model still achieves the better result than MRF model.

foreground map into the data term typically has better contribution to the foreground segmentation, comparing to the symmetry smoothness penalty.

REFERENCES

- [1] Z. Tu and X. Bai, "Auto-context and its application to high-level vision tasks and 3d brain image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 10, pp. 1744–1757, 2010. **1**
- [2] M. Kumar, P. Torr, and A. Zisserman, "OBJCUT: Efficient segmentation using top-down and bottom-up cues," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 3, pp. 530–545, 2010. **1, 3**
- [3] M. Maire, S. Yu, and P. Perona, "Object detection and segmentation from joint embedding of parts and pixels," in *ICCV*, 2011, pp. 2142–2149. **1**
- [4] J. Shi and J. Malik, "Normalized cuts and image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 8, pp. 888–905, 2000. **1**
- [5] C. Rother, V. Kolmogorov, and A. Blake, "GrabCut: interactive foreground extraction using iterated graph cuts," *ACM Transactions on Graphics*, vol. 23, no. 3, pp. 309–314, 2004. **1, 3, 7, 8, 9, 10**
- [6] Y. Li, J. Sun, C. Tang, and H. Shum, "Lazy snapping," *ACM Transactions on Graphics*, vol. 23, no. 3, pp. 303–308, 2004. **1, 3**
- [7] A. Blake, C. Rother, M. Brown, P. Perez, and P. Torr, "Interactive image segmentation using an adaptive gmmrf model," in *ECCV*, 2004, pp. 428–441. **1, 2, 3, 7, 8, 9**
- [8] A. Yang, K. Huang, S. Rao, W. Hong, and Y. Ma, "Symmetry-based 3-d reconstruction from perspective images," *Computer Vision and Image Understanding*, vol. 99, no. 2, pp. 210–240, 2005. **1**
- [9] Y. Liu, H. Hel, C. Kaplan, and L. Gool, "Computational symmetry in computer vision and computer graphics," *Foundations and Trends® in Computer Graphics and Vision*, vol. 5, no. 1-2, pp. 1–195, 2010. **1**
- [10] N. Jiang, P. Tan, and L. Cheong, "Multi-view repetitive structure detection," in *ICCV*, 2011, pp. 535–542. **1**
- [11] P. Zhao and L. Quan, "Translation symmetry detection in a fronto-parallel view," in *CVPR*, 2011, pp. 1009–1016. **1**
- [12] S. Lee and Y. Liu, "Curved glide-reflection symmetry detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 2, pp. 266–278, 2012. **1, 6**
- [13] D. Tang, H. Fu, and X. Cao, "Topology preserved regular superpixel," in *IEEE International Conference on Multimedia and Expo*, 2012, pp. 765–768. **1**
- [14] A. Yang, S. Rao, K. Huang, W. Hong, and Y. Ma, "Geometric segmentation of perspective images based on symmetry groups," in *CVPR*, 2003, pp. 1251–1258. **1**
- [15] H. Park, G. Martin, and A. Bhalerao, "Structural texture segmentation using affine symmetry," in *IEEE International Conference on Image Processing*, 2007, pp. 49–52. **1**
- [16] Y. Elor, D. Shaked, and A. Bruckstein, "Crazy-Cuts: From theory to app," *The Mathematical Intelligencer*, vol. 34, no. 2, pp. 50–55, 2012. **1**
- [17] Y. Sun and B. Bhanu, "Symmetry integrated region-based image segmentation," in *CVPR*, 2009, pp. 826–831. **1**
- [18] T. Riklin-Raviv, N. Sochen, and N. Kiryati, "On symmetry, perspectivity, and level-set-based segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 8, pp. 1458–1471, 2009. **1, 7, 10**
- [19] A. Levinstein, S. Dickinson, and C. Sminchisescu, "Multiscale symmetric part detection and grouping," in *ICCV*, 2009, pp. 2162–2169. **1, 6, 7, 8**
- [20] G. Loy and J. Eklundh, "Detecting symmetry and symmetric constellations of features," in *ECCV*, 2006, pp. 508–521. **2, 7**
- [21] S. Lee and Y. Liu, "Skewed rotation symmetry group detection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 9, pp. 1659–1672, 2010. **2**
- [22] V. Prasad and B. Yegnanarayana, "Finding axes of symmetry from potential fields," *IEEE Trans. Image Process.*, vol. 13, no. 12, pp. 1559–1566, 2004. **2**
- [23] Y. Keller and Y. Shkolnisky, "A signal processing approach to symmetry detection," *IEEE Trans. Image Process.*, vol. 15, no. 8, pp. 2198–2207, 2006. **2**
- [24] M. Cho and K. Lee, "Bilateral symmetry detection via symmetry-growing," in *British Machine Vision Conference*, 2009, pp. 1–11. **2**
- [25] X. Guo and X. Cao, "MIFT: A framework for feature descriptors to be mirror reflection invariant," *Image and Vision Computing*, vol. 30, no. 8, pp. 546–556, 2012. **2, 10**
- [26] R. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge University Press, 2004. **2**
- [27] Y. Boykov and V. Kolmogorov, "An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 9, pp. 1124–1137, 2004. **2, 7**
- [28] V. Kolmogorov and R. Zabini, "What energy functions can be minimized via graph cuts?" *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 2, pp. 147–159, 2004. **2**
- [29] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 11, pp. 1222–1239, 2001. **2, 7**
- [30] P. Felzenszwalb and D. Huttenlocher, "Efficient graph-based

- image segmentation,” *Int. J. Comput. Vision*, vol. 59, no. 2, pp. 167–181, 2004. [2](#)
- [31] R. Szeliski, R. Zabih, D. Scharstein, O. Veksler, V. Kolmogorov, A. Agarwala, M. Tappen, and C. Rother, “A comparative study of energy minimization methods for markov random fields with smoothness-based priors,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 6, pp. 1068–1080, 2008. [2](#)
- [32] M. Szummer, P. Kohli, and D. Hoiem, “Learning CRFs using graph cuts,” in *ECCV*, 2008, pp. 582–595. [2](#)
- [33] A. Mishra, Y. Aloimonos, L. Cheong, and A. Kassim, “Active visual segmentation,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 4, pp. 639–653, 2012. [3](#), [7](#), [8](#), [9](#), [10](#)
- [34] A. Mishra, A. Shrivastava, and Y. Aloimonos, “Segmenting “simple” objects using RGB-D,” in *IEEE International Conference on Robotics and Automation*, 2012, pp. 4406–4413. [3](#)
- [35] J. Carreira and C. Sminchisescu, “Constrained parametric min-cuts for automatic object segmentation,” in *CVPR*, 2010, pp. 3241–3248. [3](#)
- [36] J. Feng, Y. Wei, L. Tao, C. Zhang, and J. Sun, “Salient object detection by composition,” in *ICCV*, 2011, pp. 1028–1035. [3](#)
- [37] B. Alexe, T. Deselaers, and V. Ferrari, “Measuring the objectness of image windows,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 11, pp. 2189–2202, 2012. [3](#)
- [38] L. Itti, C. Koch, and E. Niebur, “A model of saliency-based visual attention for rapid scene analysis,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 11, pp. 1254–1259, 1998. [3](#)
- [39] X. Hou and L. Zhang, “Dynamic visual attention: Searching for coding length increments,” in *NIPS*, 2008, pp. 681–688. [3](#)
- [40] R. Achanta, S. Hemami, F. Estrada, and S. Süsstrunk, “Frequency-tuned salient region detection,” in *CVPR*, 2009, pp. 1597–1604. [3](#)
- [41] T. Liu, Z. Yuan, J. Sun, J. Wang, N. Zheng, X. Tang, and H.-Y. Shum, “Learning to detect a salient object,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 2, pp. 353–367, 2011. [3](#)
- [42] M. Cheng, G. Zhang, N. Mitra, X. Huang, and S. Hu, “Global contrast based salient region detection,” in *CVPR*, 2011, pp. 409–416. [3](#)
- [43] I. Endres and D. Hoiem, “Category independent object proposals,” in *ECCV*, 2010, pp. 575–588. [3](#), [7](#), [8](#), [9](#), [10](#)
- [44] H. Blum, “A transformation for extracting new descriptors of shape,” *Models for the perception of speech and visual form*, vol. 19, no. 5, pp. 362–380, 1967. [6](#)
- [45] X. Bai, X. Wang, L. Latecki, W. Liu, and Z. Tu, “Active skeleton for non-rigid object detection,” in *ICCV*, 2009, pp. 575–582. [6](#)
- [46] J. Wang, K. Markert, and M. Everingham, “Learning models for object recognition from natural language descriptions,” in *British Machine Vision Conference*, 2009, pp. 1–11. [9](#)