SymmSLIC: Symmetry Aware Superpixel Segmentation

Rajendra Nagar, Shanmuganathan Raman
Indian Institute of Technology Gandhinagar, Gujarat, India, 382355
{rajendra.nagar,shanmuga}@iitgn.ac.in

Abstract

Over-segmentation of an image into superpixels has become an useful tool for solving various problems in computer vision. Reflection symmetry is quite prevalent in both natural and man-made objects. Existing algorithms for estimating superpixels do not preserve the reflection symmetry of an object which leads to different sizes and shapes of superpixels across the symmetry axis. In this work, we propose an algorithm to over-segment an image through the propagation of reflection symmetry evident at the pixel level to superpixel boundaries. In order to achieve this goal, we exploit the detection of a set of pairs of pixels which are mirror reflections of each other. We partition the image into superpixels while preserving this reflection symmetry information through an iterative algorithm. We compare the proposed method with state-of-the-art superpixel generation methods and show the effectiveness of the method in preserving the size and shape of superpixel boundaries across the reflection symmetry axes. We also present an application called unsupervised symmetric object segmentation to illustrate the effectiveness of the proposed approach.

1. Introduction

Superpixels. A superpixel is a collection of spatially proximal and similar pixels [38]. Similarity could be defined in terms of color, texture, etc. The superpixels are known to preserve the local image features such as object boundaries, their regular shape and size, simple connectivity, and reduce the cost of computation of many computer vision problems. This is due to the fact that superpixel over-segmentation effectively reduces the number of units to be processed in an image. The superpixel segmentation has been used in applications such as segmentation [38], image parsing [48], tracking [55], and 3D reconstruction [12].

Symmetry. The symmetry present in real world objects is proven to play a major role in object detection and object recognition processes in humans and animals [51]. Therefore, detecting the symmetry evident in the objects has be-

come an important area of research. The major types of symmetry are: reflection symmetry, rotation symmetry, and translation symmetry. The most commonly occurring symmetry in nature is the reflection symmetry. We mainly focus on preserving the reflection symmetry present in the image in this work. The reflection symmetry present in natural images has been used to solve many problems in computer vision such as object detection, image matching [11], facial images analysis [33], real-time attention for robotic vision [41], tumour segmentation in medical images [25], 3D reconstruction [14, 44], shape manipulation, model compression, and symmetrization [30]. The common way to represent reflection symmetry is through a set of pairs of pixels which are mirror reflections of each other and the axis of symmetry.

Motivation. The perceptual grouping of local object features is a major cue in understanding objects in the human visual system [58]. The symmetry present in real world objects is proven to play a major role in object detection and object recognition processes in humans as well as animals [51]. Therefore, the symmetry present in objects should be preserved even after perceptual grouping in order to perceive the objects efficiently from the perceptually similar groups. The existing superpixel algorithms do not attempt to preserve the symmetry present in the image. The main motivation behind preserving symmetry at superpixel level is that the time complexity of algorithms using symmetry such as [11], can be reduced significantly by working at superpixels level. However, without preserving the symmetry at superpixels, their performance might get degraded. There have been attempts in preserving structure [56, 22]. However, no emphasis has been made on preserving symmetry. In this work, we propose an algorithm to partition an image into superpixels while preserving the reflection symmetry. At the superpixel level, we would like to represent the symmetry as a set of pairs of superpixels which are mirror reflections of each other. We develop a modified version of SLIC algorithm [1] to achieve this task. In Figure 1, we show an example output generated by the proposed approach along with another recent superpixel segmentation method [22] for illustration. The main contributions of this
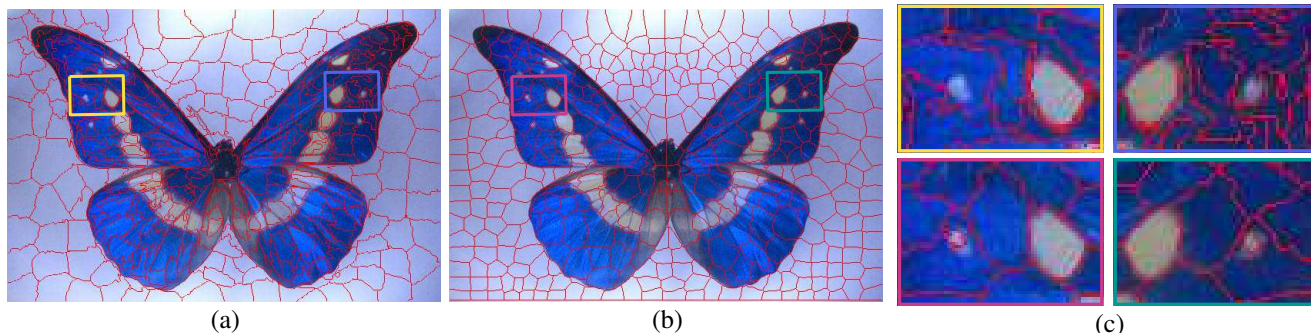
Figure 1. (a) The results for the Manifold SLIC [22], (b) results for the proposed algorithm SymmSLIC, and (c) zoomed mirror symmetric windows for both the methods (top: MSLIC, bottom: ours). We observe that the symmetry at the superpixel level is preserved in ours.

work are the following.

1. We propose an algorithm, termed *SymmSLIC*, in order to partition an image into superpixels such that the reflection symmetry present at the pixel level is preserved at the superpixel level.

2. We also propose a novel algorithm to detect pairs of pixels which are mirror reflections of each other.

3. We introduce an application of SymmSLIC called unsupervised symmetric object segmentation.

We organize remainder of the paper as follows. In Section 2, we review the existing works on superpixel segmentation and reflection symmetry detection. In Section 3.1, we discuss the proposed method for the detection of mirror symmetry point pairs. In Section 3.2, we develop the SymmSLIC algorithm to achieve reflection symmetry aware superpixel over-segmentation. In Section 4, we present the results for reflection symmetry aware superpixel segmentation. In Section 5, we introduce a novel application called unsupervised symmetric object segmentation. In Section 6, we conclude the paper with limitations and future directions.

2. Related Works

Superpixel segmentation and symmetry detection problems have been studied thoroughly and are active research problems in computer vision and computer graphics. To the best of our knowledge, there have not been any previous attempt on the problem of symmetry preserving superpixel segmentation. We discuss the state-of-the-art methods for superpixel segmentation and symmetry detection methods. **Superpixels.** There are two major categories of algorithms for superpixel segmentation - graph based and clustering based. Following are the major graph based approaches. Shi and Malik proposed normalized cut algorithm to over-segment an image [43]. Felzenszwalb proposed a graph-based image segmentation approach [8]. Li and Chen used

linear spectral clustering approach [18]. [54] and [52] used optimization techniques for superpixel segmentation. Zhang *et al.* proposed a boolean optimization framework for superpixel segmentation [61]. Moore *et al.* posed the problem of superpixel detection as lattice detection [35]. Duan and Lafarge used shape anchoring techniques on the set of detect line segments in order to partition the image into convex superpixels [7]. The key idea in the clustering based approaches is to first initialize some cluster centers and then refine these cluster centers using various techniques. Achanta *et al.* proposed a k-means clustering based approach called simple linear iterative clustering (SLIC) [1]. They initialize cluster centers at the centers of equally spaced squares. They perform clustering by assigning each pixel to the nearest center based on the color and location similarity. Levinshtein *et al.* proposed a geometric flow based approach [17]. Wang *et al.* proposed a content sensitive superpixel segmentation approach where the distance between the cluster center and a pixel is the geodesic distance [56]. Liu *et al.* proposed a fast algorithm to get structure sensitive superpixels, where authors perform the SLIC on a 2-dimensional manifold [22]. Liu *et al.* used the entropy rate for homogeneous and compact superpixels [19]. **Reflection Symmetry.** The problem of detecting reflection symmetry present in images have been thoroughly studied recently [21, 30]. The existing approaches for symmetry detection in images can be categorized in four categories - *direct approach* [53, 15], *voting based approaches* [36, 59, 31, 24, 2], *basis function based approaches* [60], and *moment based approaches* [26, 42]. Loy and Eklundh mirrored the scale invariant feature transform (SIFT [23]) descriptors in order to get the reflection invariant SIFT descriptors. In order to match two points they used SIFT descriptor for one point and mirrored-SIFT descriptor for the other point. Then, they detect the symmetry axis using Hough transform based line detection algorithm [24]. Kondra *et al.* proposed a kernel based approach [13]. Patraucean *et al.* used affine invariant edge features and a *contrarion* validation scheme [37]. Michaelsen *et al.* used

Gestalt algebra [29]. Atadjanov and Lee detected symmetry axes using appearance of structure features [2]. The works [49, 40, 62, 45, 4, 3, 30, 34, 57, 5, 50, 47] present good methods for symmetry detection. There also have been works in detecting symmetry in 3D geometric models [32].

3. Proposed Approach

3.1. Approximate and Partial Reflection Symmetry Detection

Let $I : \mathcal{W} \times \mathcal{H} \to \mathbb{R}^3$ be a color image with width w and hight h, where $\mathcal{W} = \{1, 2, \dots, w\}$ and $\mathcal{H} = \{1, 2, \dots, h\}$. Most of the real images exhibit only the *partial reflection symmetry*, which means that the mirror symmetric pixel exists for only a fraction of pixels. This is due to the fact that the images have square boundaries and the boundaries of real object are not necessarily square and furthermore there could be occlusions and missing parts. Since the real objects are not perfectly mirror symmetric, we attempt to detect the *approximate reflection symmetry*. Our goal is to detect the partial and approximate reflection symmetry present in the input image I. In order to do so, we detect the pairs of pixels which are mirror reflections of each other. We represent the reflection symmetry present in the image by two subsets, $\mathcal{L} \subset \mathcal{W} \times \mathcal{H}$ and $\mathcal{R} \subset \mathcal{W} \times \mathcal{H}$, satisfying the following property. For each pixel $\mathbf{x}_i \in \mathcal{L}, \exists\, \mathbf{x}_{i'} \in \mathcal{R}$ such that

$$\mathbf{x}_{i'} = \mathbf{R}_{ii'} \mathbf{Q} \mathbf{R}_{ii'}^\top (\mathbf{x}_i - \mathbf{t}_{ii'}) + \mathbf{t}_{ii'}, \text{ and } I(\mathbf{x}_i) = I(\mathbf{x}_{i'}).$$

Here, the point $\mathbf{t}_{ii'} = \frac{\mathbf{x}_i + \mathbf{x}_{i'}}{2}$, the matrix $\mathbf{Q} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$, and the matrix $\mathbf{R}_{ii'} = \begin{bmatrix} \cos\theta_{ii'} & -\sin\theta_{ii'} \\ \sin\theta_{ii'} & \cos\theta_{ii'} \end{bmatrix}$. The angle $\theta_{ii'}$ is the slope of the symmetry axis which is a line perpendicular to the vector $\mathbf{x}_i - \mathbf{x}_{i'}$ and passes through the mid-point $\mathbf{t}_{ii'}$. In order to determine the sets $\mathcal{L}$ and $\mathcal{R}$, representing the reflection symmetry, we use the profile and orientations of edges present in the image. We first extract the edges from the image using [6] and represent them as curves. Let $E : \mathcal{W} \times \mathcal{H} \to \{0, 1\}$ be the image representing the edges present in the image I. Let $\mathcal{E} = \{\mathbf{x} : E(\mathbf{x}) = 1\}$ be the set of pixels lying on the edges. Now, for each pixel $\mathbf{x}_i \in \mathcal{E}$, we extract an edge of length p pixels passing through the pixel $\mathbf{x}_i$ such that the pixel $\mathbf{x}_i$ lies at equal distance from the end points of the edge. We determine with how much confidence two pixels are mirror reflections of each other based on the following observation.

Let $\mathbf{x}_i$, $\mathbf{x}_{i'} \in \mathcal{E}$ be any two edge pixels, where $i \in \{1, 2, \dots, |\mathcal{E}|\}$, $i' \in \{1, 2, \dots, |\mathcal{E}|\}$, and $|\mathcal{E}|$ is the cardinality of the set $\mathcal{E}$. Let $c_i(\alpha) : [0, 1] \to \mathbb{R}^2$ be an edge or a curve such that $c_i(0.5) = \mathbf{x}_i$. Let $c_{i'}(\alpha) : [0, 1] \to \mathbb{R}^2$ be another curve such that $c_{i'}(0.5) = \mathbf{x}_{i'}$. If the pixels $\mathbf{x}_i$ and $\mathbf{x}_{i'}$ are mirror reflections of each other and the image I is

mirror symmetric in the proximity of the pixels $\mathbf{x}_i$ and $\mathbf{x}_i'$, then the following equalities hold true.

$$\begin{aligned} c_{i'}(\alpha) &= \mathbf{R}_{ii'} \mathbf{Q} \mathbf{R}_{ii'}^\top c_i(\alpha) - \mathbf{R}_{ii'} \mathbf{Q} \mathbf{R}_{ii'}^\top \mathbf{t}_{ii'} + \mathbf{t}_{ii'} && (1) \\ c_i(\alpha) &= \mathbf{R}_{ii'} \mathbf{Q} \mathbf{R}_{ii'}^\top c_{i'}(\alpha) - \mathbf{R}_{ii'} \mathbf{Q} \mathbf{R}_{ii'}^\top \mathbf{t}_{ii'} + \mathbf{t}_{ii'} && (2) \end{aligned}$$

$\forall \alpha \in [0, 1]$. Now let us define the following two quantities using equations (1) and (2).

$$o_{ii'} = \int_{\alpha \in [0,1]} \| c_{i'}(\alpha) - \mathbf{R}_{ii'} \mathbf{Q} \mathbf{R}_{ii'}^\top (c_i(\alpha) - \mathbf{t}_{ii'}) - \mathbf{t}_{ii'} \|_2^2 d\alpha$$
$$(3)$$
$$o_{i'i} = \int_{\alpha \in [0,1]} \| c_i(\alpha) - \mathbf{R}_{ii'} \mathbf{Q} \mathbf{R}_{ii'}^\top (c_{i'}(\alpha) - \mathbf{t}_{ii'}) - \mathbf{t}_{ii'} \|_2^2 d\alpha.$$
$$(4)$$

For the case of perfect symmetry, we require $o_{ii'} = 0$ and $o_{i'i} = 0$. However, due to the presence of noise and illumination variations, this might not hold true. Therefore, in order to determine the pairs of mirror symmetric pixels, we use the Hough transform. We determine the symmetry axis, $\rho_{ii'} = x \cos\phi_{ii'} + y \sin\phi_{ii'}$, defined by each pair $(\mathbf{x}_i, \mathbf{x}_{i'})$ of pixels $\mathbf{x}_i$ and $\mathbf{x}_{i'}$, which is a line passing through the mid-point $\mathbf{t}_{ii'}$ and perpendicular to the vector $\mathbf{x}_i - \mathbf{x}_{i'}$. In Figure 2 (a), we show a graphical illustration. For this axis, we define the weight equal to $e^{-\frac{o_{ii'} + o_{i'i}}{4\sigma^2}}$. We choose $\sigma = 24$. In the transformation space, we determine the dominant clusters and obtain the pairs of mirror symmetric pixels corresponding to these clusters. Using these pairs, we form the sets $\mathcal{L}$ and $\mathcal{R}$ by picking randomly one pixel of a pair and include it in the set $\mathcal{L}$ and the other pixel in the set $\mathcal{R}$. We further remove the outlier pairs using the following property of a symmetric function. Let the points $\mathbf{x}_i$ and $\mathbf{x}_{i'}$ be mirror reflections each other. Therefore,

$$I(\mathbf{x}_i) = I(\mathbf{x}_{i'}) \Rightarrow \nabla_{\mathbf{x}_i} I(\mathbf{x}_i) = \nabla_{\mathbf{x}_i} I(\mathbf{x}_{i'})$$
$$\nabla_{\mathbf{x}_i} I(\mathbf{x}_i) = \nabla_{\mathbf{x}_i} I(\mathbf{R}_{ii'} \mathbf{Q} \mathbf{R}_{ii'}^\top \mathbf{x}_i - \mathbf{R}_{ii'} \mathbf{Q} \mathbf{R}_{ii'}^\top \mathbf{t}_{ii'} + \mathbf{t}_{ii'})$$
$$\nabla_{\mathbf{x}_i} I(\mathbf{x}_i) = \mathbf{R}_{ii'} \mathbf{Q} \mathbf{R}_{ii'}^\top \nabla_{\mathbf{x}_{i'}} I(\mathbf{x}_{i'}). \quad (5)$$

We only keep those pairs satisfying

$$\nabla_{\mathbf{x}_i} I(\mathbf{x}_i)^\top \mathbf{R}_{ii'} \mathbf{Q} \mathbf{R}_{ii'}^\top \nabla_{\mathbf{x}_{i'}} I(\mathbf{x}_{i'}) > 1 - \epsilon.$$

Where, $0 < \epsilon < 1$. Since the number of pixels, $|\mathcal{E}|$, lying on the edges is very high, it results in huge number $\frac{|\mathcal{E}|(|\mathcal{E}|-1)}{2}$ of pairs. Therefore, we randomly pick pairs and vote in order to reduce the computational complexity. For each edge pixel, we select $h << |\mathcal{E}|$ pixels which results in a total number of $h|\mathcal{E}|$ pairs. We now show that the probability of selecting the correct mirror reflection pixel of a pixel using the proposed randomization scheme is very high for $h << |\mathcal{E}|$. Since the symmetry present in the image is approximate symmetry, we consider a pixel to be a mirror reflection even if it is shifted in a square of width u from its ideal location. Now, the probability of selecting the approximate mirror reflection pixel of a pixel under consideration
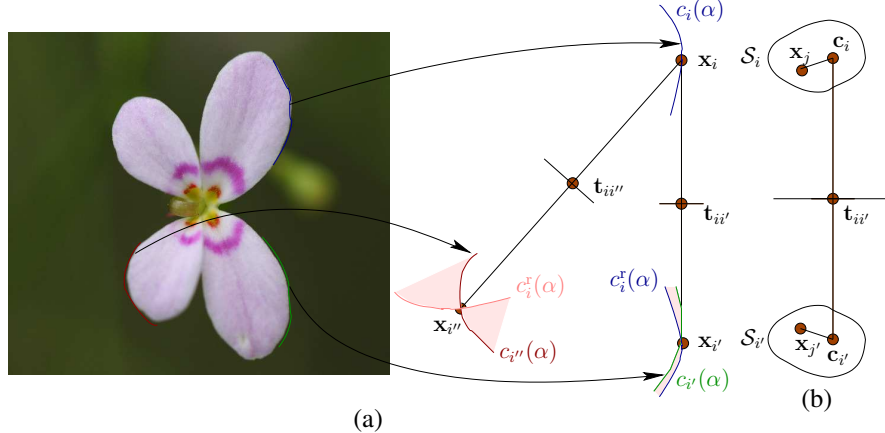
(a)

Figure 2. (a) Detection of pairs of mirror symmetric pixels. The pair $(\mathbf{x}_i, \mathbf{x}_{i'})$ is more symmetric pair than the pair $(\mathbf{x}_i, \mathbf{x}_{i''})$ since the curves $c_i^{\mathrm{r}}(\alpha)$ and $c_{i'}(\alpha)$ are more aligned. (b) The symmetric assignment: if the superpixels $\mathcal{S}_i$ and $\mathcal{S}_{i'}$ are mirror reflections of each other, pixels $\mathbf{x}_i$ and $\mathbf{x}_{i'}$ are mirror reflections of each other, and the pixel $\mathbf{x}_j$ is assigned to the center $\mathbf{c}_i$, then we assign the pixel $\mathbf{x}_{j'}$ to the center $\mathbf{c}_{i'}$.

in one attempt is $\frac{u^2}{|\mathcal{E}|-1}$. Therefore, the probability of not selecting in one attempt is $1 - \frac{u^2}{|\mathcal{E}|-1}$. Therefore, probability of not selecting the approximate mirror reflection pixel of a pixel under consideration in h attempts is $\left(1 - \frac{u^2}{|\mathcal{E}|-1}\right)^h$. Hence, the probability of selecting the approximate mirror reflection pixel of a pixel in h attempts is $1 - \left(1 - \frac{u^2}{|\mathcal{E}|-1}\right)^h$. For example, for $|\mathcal{E}| = 3000$, $h = 200$, and $u = 5$, this probability is 0.8124 which is quite high.

3.2. Symmetry Aware SLIC

We extend the SLIC algorithm to preserve reflection symmetry at the superpixel level. In order to preserve the reflection symmetry present in the input image, represented by the sets $\mathcal{L}$ and $\mathcal{R}$, we have to make sure that for each pair $(\mathbf{x}_i, \mathbf{x}_{i'})$ of pixels $\mathbf{x}_i$ and $\mathbf{x}_{i'}$ which are mirror reflections of each other, there should be a pair $(\mathcal{S}_i, \mathcal{S}_{i'})$ of superpixels, $\mathcal{S}_i$ and $\mathcal{S}_{i'}$, which are mirror reflections of each other. We define two superpixels, $\mathcal{S}_i$ and $\mathcal{S}_{i'}$, to be mirror reflections of each other if for each $\mathbf{x}_j \in \mathcal{S}_i$, $\exists \mathbf{x}_{j'} \in \mathcal{S}_{i'}$ such that the pixels $\mathbf{x}_j$ and $\mathbf{x}_{j'}$ are mirror reflections of each other. We extend the SLIC algorithm proposed in [1] to estimate symmetry aware superpixels. The SLIC algorithm is based on the k-means clustering algorithm. Cluster centers are initialized on the center of equally spaced squares of sizes $s \times s$. In order to update the cluster centers, the distances between a center and all pixels within the square of size $2s \times 2s$ centered at these pixels is computed. Then a pixel is assigned the nearest center. Then the center is updated using its new neighboring pixels by taking the average location and the average color. This process is continued till convergence. The distance between the pixels and the centers is defined as below. Let $\begin{bmatrix} l_i & a_i & b_i \end{bmatrix}^\top$ be the color at the pixel $\mathbf{x}_i$ in the *CIELab* color space. The distance between

any two pixels $\mathbf{x}_i = \begin{bmatrix} x_i & y_i \end{bmatrix}^\top$ and $\mathbf{x}_{i'} = \begin{bmatrix} x_{i'} & y_{i'} \end{bmatrix}^\top$ is defined as $d(\mathbf{x}_i, \mathbf{x}_{i'}) = \sqrt{d_c^2 + \left(\frac{d_s}{s}\right)^2 m^2}$, where $d_c = \sqrt{(l_i - l_{i'})^2 + (a_i - a_{i'})^2 + (b_i - b_{i'})^2}$, $d_s = \sqrt{(x_i - x_{i'})^2 + (y_i - y_{i'})^2}$. Here m is the compactness factor and generally chosen in range $[1, 40]$ [1]. Higher values of m result in compact superpixels and poor boundary adherence and lower values m result in poor compactness of superpixels and better adherence to boundaries.

Symmetric Initialization. In order to preserve the reflection symmetry represented in the sets $\mathcal{L}$ and $\mathcal{R}$, we have to make sure that for a pair of pixels which are mirror reflections of each other, there should be a corresponding pair of superpixels which are mirror reflections of each other. Let the pixels $\mathbf{x}_i \in \mathcal{L}$ and $\mathbf{x}_{i'} \in \mathcal{R}$ be mirror reflections of each other. We initialize the centers $\mathbf{c}_i$ and $\mathbf{c}_{i'}$ of two superpixels $\mathcal{S}_i$ and $\mathcal{S}_{i'}$ at the pixels $\mathbf{x}_i$ and $\mathbf{x}_{i'}$. We observe that the symmetric object present in the image might not cover the full image and therefore in the non-symmetric region, we follow the same initialization strategy as used in SLIC. We first find the convex hull, $\mathcal{C}$, of the set $\mathcal{L} \cup \mathcal{R}$ which represent the symmetric region. Now, in the non-symmetric region $\{\mathcal{W} \times \mathcal{H}\} \setminus \mathcal{C}$, we initialize the centers at the centers of equally spaced squares and in the region $\mathcal{C}$, we do the symmetric initialization. We observe that the reflection symmetric pixels obtained lie on the edges. Therefore, according to [1], it is an unstable initialization. However, we observe that these pairs of reflection symmetric pixels exhibit high accuracy. Therefore, we transfer each pair to a new location such that the image gradient at both the pixels of the new pair is minimum in the local vicinity and are mirror reflections of each other.

Number of superpixels. If we want k superpixels, then we partition the image into square windows of sides equal to

Algorithm 1 *SymmSLIC*

1: **Input**: Image I, number of superpixels k, and parameter m.
2: Initialize the label matrix $\mathbf{L}(\mathbf{x}_i) = -1$, and the distance matrix $\mathbf{D}(\mathbf{x}_i) = \infty$, $\forall \mathbf{x}_i \in \mathcal{W} \times \mathcal{H}$.
3: Determine the sets $\mathcal{L}$ and $\mathcal{R}$ as discussed in Section 3.1.
4: Initialize the cluster centers as discussed in Section 3.2.
5: **while** not converged **do**
6:     **for** each cluster center $\mathbf{c}_i$ in $\mathcal{L}$ **do**
7:         Determine $\mathbf{R}_{ii'}$ and $\mathbf{t}_{ii'}$ using $\mathbf{c}_i$ and $\mathbf{c}_{i'}$.
8:         **for** each pixel $\mathbf{x}_j$ in the $2s \times 2s$ square around $\mathbf{c}_i$ **do**
9:             Compute the distance $d(\mathbf{x}_j, \mathbf{c}_i)$ between $\mathbf{c}_i$ and $\mathbf{x}_j$.
10:           **if** $d(\mathbf{x}_j, \mathbf{c}_i) < \mathbf{D}(\mathbf{x}_j)$ **then**
11:              $\mathbf{D}(\mathbf{x}_j) \leftarrow d(\mathbf{x}_j, \mathbf{c}_i)$ and $\mathbf{L}(\mathbf{x}_j) \leftarrow i$
12:              $\mathbf{x}_{j'} \leftarrow \mathbf{R}_{ii'}\mathbf{Q}\mathbf{R}_{ii'}^{\top}\mathbf{x}_j - \mathbf{R}_{ii'}\mathbf{Q}\mathbf{R}_{ii'}^{\top}\mathbf{t}_{ii'} + \mathbf{t}_{ii'}$
13:              $\mathbf{D}(\mathbf{x}_{j'}) \leftarrow d(\mathbf{x}_{j'}, \mathbf{c}_{i'})$ and $\mathbf{L}(\mathbf{x}_{j'}) \leftarrow i'$
14:           **end if**
15:         **end for**
16:     **end for**
17:     **for** each cluster center $\mathbf{c}_{i'}$ in $\mathcal{R}$ **do**
18:         Determine $\mathbf{R}_{ii'}$ and $\mathbf{t}_{ii'}$ using $\mathbf{c}_i$ and $\mathbf{c}_{i'}$.
19:         **for** each pixel $\mathbf{x}_{j'}$ in the $2s \times 2s$ square around $\mathbf{c}_{i'}$ **do**
20:             Compute the distance $d(\mathbf{x}_{j'}, \mathbf{c}_{i'})$ between $\mathbf{c}_{i'}$ and $\mathbf{x}_{j'}$.
21:           **if** $d(\mathbf{x}_{j'}, \mathbf{c}_{i'}) < \mathbf{D}(\mathbf{x}_{j'})$ **then**
22:              $\mathbf{D}(\mathbf{x}_{j'}) \leftarrow d(\mathbf{x}_{j'}, \mathbf{c}_{i'})$ and $\mathbf{L}(\mathbf{x}_{j'}) \leftarrow i'$
23:              $\mathbf{x}_j \leftarrow \mathbf{R}_{ii'}\mathbf{Q}\mathbf{R}_{ii'}^{\top}\mathbf{x}_{j'} - \mathbf{R}_{ii'}\mathbf{Q}\mathbf{R}_{ii'}^{\top}\mathbf{t}_{ii'} + \mathbf{t}_{ii'}$
24:              $\mathbf{D}(\mathbf{x}_j) \leftarrow d(\mathbf{x}_j, \mathbf{c}_i)$ and $\mathbf{L}(\mathbf{x}_j) \leftarrow i$
25:           **end if**
26:         **end for**
27:     **end for**
28:     **for** all the centers in the region $\left(\{\mathcal{W} \times \mathcal{H}\} \setminus \text{convexhull}(\mathcal{L} \cup \mathcal{R})\right)$ **do**
29:         Perform SLIC.
30:     **end for**
31:     Update the cluster centers.
32: **end while**

$\sqrt{\frac{wh}{k}}$. If the number of pairs of mirror symmetric points $|\mathcal{L}|$ in the symmetric region $\mathcal{C}$ is greater than $(k-s)$, then we randomly select $(k-s)$ pairs from the $|\mathcal{L}|$ pairs. Here, s is the number square windows in the non-symmetric region $\{\mathcal{W} \times \mathcal{H}\} \setminus \mathcal{C}$. If the number pairs of mirror symmetric points $|\mathcal{L}|$ in the symmetric region $\mathcal{C}$ is less than $(k-s)$ then we randomly select $(k-s-|\mathcal{L}|)$ points in the symmetric region and then reflect them using the symmetry axis

defined by their nearest pair.

Symmetric Assignment. Now, we propose an assignment strategy in order to achieve pairs of reflection symmetric superpixels with equal areas and similar boundaries. We assign pixels to their nearest centers such that each pair of superpixels $(\mathcal{S}_i, \mathcal{S}_{i'})$ remains mirror reflection of each other in all iterations. Let us consider the centers $\mathbf{c}_i$ and $\mathbf{c}_{i'}$ of two superpixels, $\mathcal{S}_i$ and $\mathcal{S}_{i'}$, which are mirror reflections of each other. Let $\mathbf{x}_j$ be a pixel inside the square of size $2s \times 2s$ around the center $\mathbf{c}_i$. If the nearest center to the pixel $\mathbf{x}_j$ is $\mathbf{c}_i$, then we assign the center $\mathbf{c}_{i'}$ as the nearest center to the pixel $\mathbf{x}_{j'} = \mathbf{R}_{ii'}\mathbf{Q}\mathbf{R}_{ii'}^{\top}(\mathbf{x}_j - \mathbf{t}_{ii'}) + \mathbf{t}_{ii'}$. Here, $\mathbf{t}_{ii'} = \frac{\mathbf{c}_i + \mathbf{c}_{i'}}{2}$ and $\mathbf{R}_{ii'}$ is the rotation matrix with angle $\theta_{ii'}$ equal to the slope of the line passing through the pixels $\mathbf{c}_i$ and $\mathbf{c}_{i'}$. Fig. 2 (b) graphically illustrates this concept. We prove that, using this assignment strategy, a pair of reflection symmetric superpixels remains a pair of reflection symmetric superpixels after one iteration.

Claim 1. *Let $\mathcal{S}_i^t$ and $\mathcal{S}_{i'}^t$ be two superpixels which are mirror reflections of each other at the iteration t. Let $\mathbf{c}_i^t$ and $\mathbf{c}_{i'}^t$ be their centers respectively. Then, at the iteration $t+1$, the updated superpixels $\mathcal{S}_i^{t+1}$ and $\mathcal{S}_{i'}^{t+1}$ will also be the mirror reflections of each other.*

Proof. Using assignment strategy, if we assign $\mathbf{x}_i$ to $\mathcal{S}_i$, then we assign the pixel $\mathbf{R}_{ii'}\mathbf{Q}\mathbf{R}_{ii'}^{\top}\mathbf{x}_j - \mathbf{R}_{ii'}\mathbf{Q}\mathbf{R}_{ii'}^{\top}\mathbf{t}_{ii'} + \mathbf{t}_{ii'}$ to $\mathcal{S}_{i'}$. Let us assume that we assign n_i pixels to the superpixel $\mathcal{S}_i$, and $\mathcal{J}_i = \{i_1, i_2, \ldots, i_{n_i}\}$, $\mathcal{J}_{i'} = \{i_1', i_2', \ldots, i_{n_i}'\}$ be the sets of indices of pixels belonging to the superpixels $\mathcal{S}_i$ and $\mathcal{S}_{i'}$, respectively. The center of the superpixel $\mathcal{S}_i$ is $\mathbf{c}_i^{t+1} = \frac{1}{n_i}\sum_{j \in \mathcal{J}_i}\mathbf{x}_j$. Now, the center of the superpixel $\mathcal{S}_{i'}$ is

$$\mathbf{c}_{i'}^{t+1} = \frac{1}{n_i}\sum_{j' \in \mathcal{J}_i}\mathbf{x}_{j'} =$$

$$\frac{1}{n_i}\sum_{j \in \mathcal{J}_i}\mathbf{R}_{ii'}\mathbf{Q}\mathbf{R}_{ii'}^{\top}\mathbf{x}_j - \frac{1}{n_i}\sum_{j \in \mathcal{J}_{i'}}\mathbf{R}_{ii'}\mathbf{Q}\mathbf{R}_{ii'}^{\top}\mathbf{t}_{ii'} + \frac{1}{n_i}\sum_{j \in \mathcal{J}_{i'}}\mathbf{t}_{ii'}$$

$$= \mathbf{R}_{ii'}\mathbf{Q}\mathbf{R}_{ii'}^{\top}\frac{1}{n_i}\sum_{j \in \mathcal{J}_i}\mathbf{x}_j - \mathbf{R}_{ii'}\mathbf{Q}\mathbf{R}_{ii'}^{\top}\mathbf{t}_{ii'} + \mathbf{t}_{ii'}$$

$$= \mathbf{R}_{ii'}\mathbf{Q}\mathbf{R}_{ii'}^{\top}\mathbf{c}_i^{t+1} - \mathbf{R}_{ii'}\mathbf{Q}\mathbf{R}_{ii'}^{\top}\mathbf{t}_{ii'} + \mathbf{t}_{ii'}.$$

Therefore, the center of superpixels at the iteration $t+1$ and the updated superpixels $\mathcal{S}_i^{t+1}$ and $\mathcal{S}_{i'}^{t+1}$ will also be mirror reflections of each other. $\square$

We further observe that the centers of mirror symmetric superpixels follow the curves that are mirror reflections of each other. It is easy to prove this claim from the Claim 1. In Algorithm 1, we present all the steps involved in the proposed *SymmSLIC* algorithm.

4. Results and Evaluation

In Figure 3, we show the major steps of the proposed approach using an example image where we detect two symmetric regions. We observe that the symmetry present at
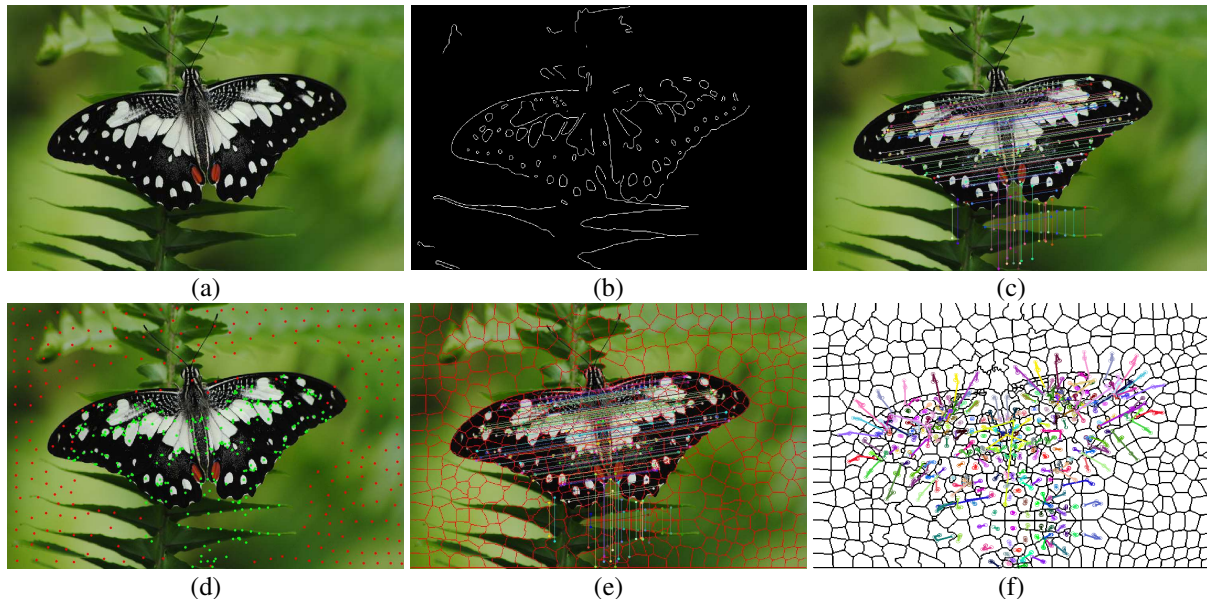
Figure 3. (a) Input image I, (b), detected edges, (c) the pairs of mirror symmetric points ($\mathcal{L}$ and $\mathcal{R}$), (d) the initialized centers, the red centers are in the non-symmetric regions and the green centers are in the symmetric regions, (e) the symmetric superpixel segmentation with the pairs of mirror symmetric superpixels, and (f) The movement of the centers. The mirror symmetric centers move on the mirror symmetric paths (shown using the same color).
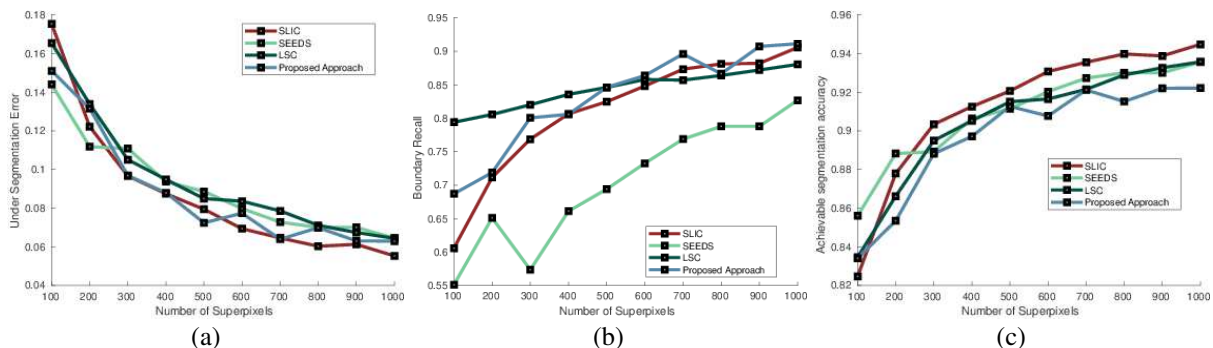

Figure 4. (a) Under segmentation error, (b) Boundary recall, and (c) Achievable segmentation accuracy vs the number of superpixels plots.

Figure 5. A failure case: Input, Edges, and SymmSLIC

the pixel level, as shown in Figure 3 (c), is well preserved at the superpixel level, as shown in Figure 3 (e). In Figure 3 (f), we show the movement of the mirror symmetric centers. We observe that the mirror symmetric centers move along the mirror symmetric paths. In Figure 5, we present a failure case. The SymmSLIC fails due to the improper detection of the edges across the symmetry axis. However, we observe that the resulting over-segmentation is near to the SLIC superpixels.

For the quantitative evaluation, we measure the boundary recall and the under-segmentation error for the proposed approach and the approaches [1], [18], and [52] on the dataset BSDS500 [27]. We do evaluation only on the images containing symmetric objects from this dataset. In order to measure the performance, we use the three metrics: under segmentation error [17], boundary recall [28], and achievable segmentation accuracy [19]. The under segmentation error measures the leakage of estimated superpixels. The higher values of boundary recall represents better adherence to the ground truth boundaries. The achievable segmentation accuracy measures the object segmentation accuracy which can be achieved through the estimated superpixels. In Figure 4, we plot the under segmentation error, boundary recall, and achievable segmentation accuracy as the function of number of superpixels for SLIC [1], LSC [18], SEEDS [52] and the proposed approach. We observe that the performance of the proposed method is comparable to that of

the SLIC [1], since SymmSLIC works similar to SLIC algorithm and on average almost 60 % region of the images is non-symmetric. Therefore, in these regions only SLIC is performed.

In Figure 7, we show the results for the proposed approach. We compare the results of the proposed approach to the results of the methods TURBO [17], SLIC [1], ERS [19], LSC [18], SEEDS [52], and MSLIC [22]. In each image, we zoom two windows which are mirror reflections of each other. In odd numbered rows, we show the results obtained on images for all the methods. In the even numbered rows, we show two zoomed-in mirror symmetric windows from the images in the odd numbered rows. There does not exist any dataset in which, for an image containing symmetric objects, the ground truth pairs of reflection symmetric pairs of superpixels are present. Therefore, we only report the results obtained through all the methods on images containing symmetric objects. We choose the length, p, of the curve $c_\mathbf{x}(\alpha)$ to be equal to 64 pixels, the threshold $\epsilon = 0.2$, the variable $m \in [1, 40]$, and $\sigma = 24$. We observe that, using the proposed algorithm *SymmSLIC*, we are able to generate pairs of mirror symmetric superpixels in the symmetric regions. In some cases, our algorithm partitions an perceptually uniform region into many superpixels and due to the symmetric assignment strategy, we achieve a similar segmentation in the mirror symmetric counterpart. We implemented SymmSLIC in MATLAB on a 2.90GHz×4, 8GB RAM machine. The average time is $\sim 10s$ for an image of size 640×480 for 500 superpixels including the detection of pairs of pixels which are mirror reflections of each other.
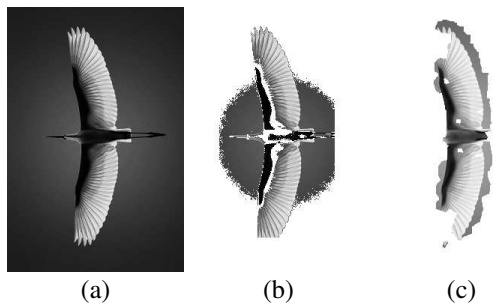

(a)   (b)   (c)
Figure 6. (a) Input, (b) Kernel Cut [46], and (c) Proposed.

5. Application: Unsupervised Symmetric Object Segmentation

Object segmentation is a challenging problem which is generally performed either with user interaction and graph cuts ([39], [9], [46]) or with supervised learning [10]. Our approach also differ from [16] in the sense that they perform local symmetry grouping whereas we perform global symmetry grouping. We would like to demonstrate how object segmentation can be performed in an unsupervised manner

using SymmSLIC. This section is meant to provide an illustration as to how the algorithm developed in this paper can be used to solve this classic computer vision application. The application is however limited to images containing objects exhibiting reflection symmetry.

We use the SymmSLIC superpixels to segment a symmetric object. This approach is clearly an unsupervised object segmentation approach. The proposed segment is the area, $\cup_{i\in\mathcal{I}}\mathcal{S}_i \cup \mathcal{S}_{i'}$, occupied by the pairs $(\mathcal{S}_i, \mathcal{S}_{i'})$ of the superpixels $\mathcal{S}_i$ and $\mathcal{S}_{i'}$, which are mirror reflections of each other. Here $\mathcal{I}$ is the set of indices of the pairs of mirror symmetric superpixels. We compare our method with the state-of-the-art interactive method [46] on this challenge dataset [20]. The method in [46] assumes that the bounding box around the symmetric object is given. Whereas, our method does not require any such user interaction. In this dataset [20], each 2D reflection symmetric image contains a symmetric object. We manually created the ground truth segmentations. We compute the error rate defined as the ratio of the mis-classified pixels to the total number of pixels. The averaged error rate on all the images from [20] for the method in [46] is 0.3622 and for the proposed approach is 0.527. In Figure 6, we show the results on an example image from [20]. We observe that our method does not require any user interaction and still we get a comparable error rate. The performance of our method depends on how well the edges are extracted in the given image.

6. Conclusion

In this work, we have proposed an algorithm to partition an image into superpixels, such that the symmetry present at the pixel level is preserved at the superpixel level. We first detect the symmetry present at the pixel level, presented as pairs of mirror symmetric pixels and then extend the SLIC algorithm to preserve the symmetry at the superpixel level by proposing a novel symmetric initialization and symmetric pixel center assignment strategies. We observe that we are able to achieve mirror symmetric superpixels in the symmetric regions. The main limitations of the proposed algorithm are that it is applicable only to the fronto-parallel views containing reflection symmetry and heavily depends on the performance of the edge detection algorithms. As a future work, we would like to extend the proposed method for the rotation symmetry, translation symmetry, and curved reflection symmetry. We also would like to prepare a dataset containing the set of ground truth pairs of superpixels for benchmarking the performance of the algorithm developed.

Figure 7. Results for the approaches TURBO [17], SLIC [1], ERS [19], LSC [18], SEEDS [52], MSLIC [22], and the proposed method. For each image, we show two zoomed-in mirror symmetric windows to visualize whether the superpixels are mirror reflections of each other or not. Window border represents its location in the image. The tilted windows are aligned horizontally.

References

[1] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Süsstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *IEEE transactions on pattern analysis and machine intelligence*, 34(11):2274–2282, 2012. 1, 2, 4, 6, 7, 8

[2] I. R. Atadjanov and S. Lee. Reflection symmetry detection via appearance of structure descriptor. In *ECCV*, pages 3–18. Springer, 2016. 2, 3

[3] M. Bokeloh, A. Berner, M. Wand, H.-P. Seidel, and A. Schilling. Symmetry detection using feature lines. In *Computer Graphics Forum*, volume 28, pages 697–706. Wiley Online Library, 2009. 3

[4] M. Cho and K. M. Lee. Bilateral symmetry detection via symmetry-growing. In *BMVC*, pages 1–11. Citeseer, 2009. 3

[5] H. Cornelius and G. Loy. Detecting bilateral symmetry in perspective. In *2006 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'06)*, pages 191–191. IEEE, 2006. 3

[6] P. Dollár and C. L. Zitnick. Structured forests for fast edge detection. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1841–1848, 2013. 3

[7] L. Duan and F. Lafarge. Image partitioning into convex polygons. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3119–3127, 2015. 2

[8] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient graph-based image segmentation. *International journal of computer vision*, 59(2):167–181, 2004. 2

[9] H. Fu, X. Cao, Z. Tu, and D. Lin. Symmetry constraint for foreground extraction. *IEEE transactions on cybernetics*, 44(5):644–654, 2014. 7

[10] B. Hariharan, P. Arbeláez, R. Girshick, and J. Malik. Simultaneous detection and segmentation. In *European Conference on Computer Vision*, pages 297–312. Springer, 2014. 7

[11] D. C. Hauagge and N. Snavely. Image matching using local symmetry features. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 206–213. IEEE, 2012. 1

[12] D. Hoiem, A. A. Efros, and M. Hebert. Automatic photo pop-up. *ACM transactions on graphics (TOG)*, 24(3):577–584, 2005. 1

[13] S. Kondra, A. Petrosino, and S. Iodice. Multi-scale kernel operators for reflection and rotation symmetry: further achievements. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 217–222, 2013. 2

[14] K. Köser, C. Zach, and M. Pollefeys. Dense 3d reconstruction of symmetric scenes from a single image. In *Pattern Recognition*, pages 266–275. Springer, 2011. 1

[15] J. L. Krahe. Detection of symmetric and radial structures in images. In *International Conference on Pattern Recognition*, pages 947–950, 1986. 2

[16] A. Levinshtein, C. Sminchisescu, and S. Dickinson. Multi-scale symmetric part detection and grouping. *International journal of computer vision*, 104(2):117–134, 2013. 7

[17] A. Levinshtein, A. Stere, K. N. Kutulakos, D. J. Fleet, S. J. Dickinson, and K. Siddiqi. Turbopixels: Fast superpixels using geometric flows. *IEEE transactions on pattern analysis and machine intelligence*, 31(12):2290–2297, 2009. 2, 6, 7, 8

[18] Z. Li and J. Chen. Superpixel segmentation using linear spectral clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1356–1363, 2015. 2, 6, 7, 8

[19] M.-Y. Liu, O. Tuzel, S. Ramalingam, and R. Chellappa. Entropy rate superpixel segmentation. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 2097–2104. IEEE, 2011. 2, 6, 7, 8

[20] Y. Liu and S. Dickinson. Detecting symmetry in the wild. In *International Conference on Computer Vision (ICCV) Workshops*. IEEE, 2017. 7

[21] Y. Liu, H. Hel-Or, C. S. Kaplan, and L. Van Gool. Computational symmetry in computer vision and computer graphics. *Foundations and Trends in Computer Graphics and Vision*, 5(1-2):1–195, 2010. 2

[22] Y.-J. Liu, C.-C. Yu, M.-J. Yu, and Y. He. Manifold slic: A fast method to compute content-sensitive superpixels. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 651–659, 2016. 1, 2, 7, 8

[23] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004. 2

[24] G. Loy and J.-O. Eklundh. Detecting symmetry and symmetric constellations of features. *Computer Vision–ECCV 2006*, pages 508–521, 2006. 2

[25] M. Mancas, B. Gosselin, and B. Macq. Fast and automatic tumoral area localisation using symmetry. In *Proceedings.(ICASSP'05). IEEE International Conference on Acoustics, Speech, and Signal Processing, 2005.*, volume 2, pages 725–728. IEEE, 2005. 1

[26] G. Marola. On the detection of the axes of symmetry of symmetric and almost symmetric planar images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(1):104–108, 1989. 2

[27] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, volume 2, pages 416–423. IEEE, 2001. 6

[28] D. R. Martin, C. C. Fowlkes, and J. Malik. Learning to detect natural image boundaries using local brightness, color, and texture cues. *IEEE transactions on pattern analysis and machine intelligence*, 26(5):530–549, 2004. 6

[29] E. Michaelsen, D. Muench, and M. Arens. Recognition of symmetry structure by use of gestalt algebra. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 206–210, 2013. 3

[30] Y. Ming, H. Li, and X. He. Symmetry detection via contour grouping. In *2013 IEEE International Conference on Image Processing*, pages 4259–4263. IEEE, 2013. 1, 2, 3

[31] N. J. Mitra, L. J. Guibas, and M. Pauly. Partial and approximate symmetry detection for 3d geometry. In *ACM Transactions on Graphics (TOG)*, volume 25, pages 560–568. ACM, 2006. 2

[32] N. J. Mitra, M. Pauly, M. Wand, and D. Ceylan. Symmetry in 3d geometry: Extraction and applications. In *Computer Graphics Forum*, volume 32, pages 1–23. Wiley Online Library, 2013. 3

[33] S. Mitra and Y. Liu. Local facial asymmetry for expression classification. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 2, pages II–889. IEEE, 2004. 1

[34] Q. Mo and B. Draper. Detecting bilateral symmetry with feature mirroring. In *CVPR Workshops*, 2011. 3

[35] A. P. Moore, S. J. Prince, J. Warrell, U. Mohammed, and G. Jones. Superpixel lattices. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008. 2

[36] S. K. Parui and D. D. Majumder. Symmetry analysis by computer. *Pattern Recognition*, 16(1):63–67, 1983. 2

[37] V. Patraucean, R. Grompone von Gioi, and M. Ovsjanikov. Detection of mirror-symmetric image patches. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 211–216, 2013. 2

[38] X. Ren and J. Malik. Learning a classification model for segmentation. In *ICCV*, volume 1, pages 10–17, 2003. 1

[39] C. Rother, V. Kolmogorov, and A. Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. In *ACM transactions on graphics (TOG)*, volume 23, pages 309–314. ACM, 2004. 7

[40] R. Scognamillo, G. Rhodes, C. Morrone, and D. Burr. A feature-based model of symmetry detection. *Proceedings of the Royal Society of London B: Biological Sciences*, 270(1525):1727–1733, 2003. 3

[41] G. Sela and M. D. Levine. Real-time attention for robotic vision. *Real-Time Imaging*, 3(3):173–194, 1997. 1

[42] D. Shen, H. H. Ip, K. K. Cheung, and E. K. Teoh. Symmetry detection by generalized complex (gc) moments: a close-form solution. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 21(5):466–476, 1999. 2

[43] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on pattern analysis and machine intelligence*, 22(8):888–905, 2000. 2

[44] S. N. Sinha, K. Ramnath, and R. Szeliski. Detecting and reconstructing 3d mirror symmetric objects. In *Computer Vision–ECCV 2012*, pages 586–600. Springer, 2012. 1

[45] C. Sun. Symmetry detection using gradient information. *Pattern Recognition Letters*, 16(9):987–996, 1995. 3

[46] M. Tang, I. Ben Ayed, D. Marin, and Y. Boykov. Secrets of grabcut and kernel k-means. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1555–1563, 2015. 7

[47] Z. Tang, P. Monasse, and J.-M. Morel. Reflexive symmetry detection in single image. In *International Conference on Curves and Surfaces*, pages 452–460. Springer, 2014. 3

[48] J. Tighe and S. Lazebnik. Superparsing: scalable nonparametric image parsing with superpixels. *Computer Vision–ECCV 2010*, pages 352–365, 2010. 1

[49] S. Tsogkas and I. Kokkinos. Learning-based symmetry detection in natural images. In *European Conference on Computer Vision*, pages 41–54. Springer, 2012. 3

[50] T. Tuytelaars, A. Turina, and L. Van Gool. Noncombinatorial detection of regular repetitions under perspective skew. *IEEE TPAMI*, 25(4):418–432, 2003. 3

[51] C. W. Tyler. *Human symmetry perception and its computational analysis*. Psychology Press, 2003. 1

[52] M. Van den Bergh, X. Boix, G. Roig, and L. Van Gool. Seeds: Superpixels extracted via energy-driven sampling. *International Journal of Computer Vision*, 111(3):298–314, 2015. 2, 6, 7, 8

[53] A. Vasilier. Recognition of symmetrical patterns in images. In *International Conference on Pattern Recognition*, pages 1138–1140, 1984. 2

[54] O. Veksler, Y. Boykov, and P. Mehrani. Superpixels and supervoxels in an energy optimization framework. *Computer Vision–ECCV 2010*, pages 211–224, 2010. 2

[55] J. Wang, L. Ju, and X. Wang. Image segmentation using local variation and edge-weighted centroidal voronoi tessellations. *IEEE Transactions on Image Processing*, 20(11):3242–3256, 2011. 1

[56] P. Wang, G. Zeng, R. Gan, J. Wang, and H. Zha. Structure-sensitive superpixels via geodesic distance. *International journal of computer vision*, 103(1):1–21, 2013. 1, 2

[57] Z. Wang, Z. Tang, and X. Zhang. Reflection symmetry detection using locally affine invariant edge correspondence. *Image Processing, IEEE Transactions on*, 24(4):1297–1301, 2015. 3

[58] M. Wertheimer. Laws of organization in perceptual forms. 1938. 1

[59] R. K. Yip. A hough transform technique for the detection of reflectional symmetry and skew-symmetry. *Pattern Recognition Letters*, 21(2):117–130, 2000. 2

[60] E. Yodogawa. Symmetropy, an entropy-like measure of visual symmetry. *Perception & Psychophysics*, 32(3):230–240, 1982. 2

[61] Y. Zhang, R. Hartley, J. Mashford, and S. Burn. Superpixels via pseudo-boolean optimization. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 1387–1394. IEEE, 2011. 2

[62] T. Zielke, M. Brauckmann, and W. Vonseelen. Intensity and edge-based symmetry detection with an application to car-following. *CVGIP: Image Understanding*, 58(2):177–190, 1993. 3