

SymmMap: Estimation of the 2-D Reflection Symmetry Map and its Applications

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

Abstract

Detecting the reflection symmetry axis present in an object has been an active research problem in computer vision and computer graphics due to its various applications such as object recognition, object detection, modelling, and symmetrization of 3D objects. However, the problem of computing the reflection symmetry map for a given image containing objects exhibiting reflection symmetry has received a very little attention. The symmetry map enables us to represent the pixels in the image using a score depending on the probability of each of them having a symmetric counterpart. In this work, we attempt to compute the 2-D reflection symmetry map. We pose the problem of generating the symmetry map as an intra-image dense symmetric pixels correspondence problem, which we solve efficiently using a randomized algorithm by observing the reflection symmetry coherency present in the image. We introduce an application of symmetry map called symmetry preserving image stylization.

1. Introduction

Reflection symmetry is the most prevalent symmetry observed in many natural as well as man made objects. Reflection symmetry is essential for physical stability of objects such as humans and it makes some objects readily recognizable. Furthermore, for some objects, it helps them look more beautiful. Symmetry has been studied in various branches of science such as mathematics, physics, and chemistry. Due to the presence of reflection symmetry in many objects, it has been one of the most active research topics in visual neuroscience as well as in computer vision.

Detecting the reflection symmetry axis of an object present in a given image has been an active research problem in computer vision and graphics due to its various applications such as low-level object features, modelling and editing of 3D models, shape database retrieval, shape matching, shape segmentation [26, 38], symmetrization [37], texture synthesis and manipulation [20], and depth from symmetry [23]. However, the problem of determining

which pixel is the mirror reflection of a given pixel and with how much confidence they qualify to be mirror reflections of each other without explicitly knowing the symmetry axis present in the image, has not been studied in the past. In this work, we introduce the concept of *reflection symmetry map* and design an algorithm to compute it. In the computation of the reflection symmetry map (referred also as symmetry map), we determine the mirror reflection for each pixel (*Nearest Mirror Reflection Field (NMRF)*) along with the confidence by which they are mirror reflections of each other (*Symmetry Score Map (SSM)*). Given an image with a symmetric object exhibiting reflection symmetry along k distinct axes, there will be k NMRF and k SSM for that image. We compute the symmetry score in addition to determining the mirror reflection of each pixel because two pixels may not be exact reflections of each other due to illumination variation, viewpoint change, and imaging noise.

The main contributions of this work are listed below.

1. We introduce the idea of 2-D *reflection symmetry map* which is a collection of two maps: NMRF which represents the mirror reflection of each pixel and SSM which represents a confidence score for each pixel which is a measure of similarity between the pixel and its mirror reflection.
2. We exploit the *reflection symmetry coherency* present in the image in order to effectively compute the symmetry map using an approximate nearest neighbor search randomized algorithm.
3. We show the importance of SSM by using it in image stylization where we try to preserve details in the symmetric region and remove details in the other regions.

2. Related Work

Computation of the reflection symmetry map for an input image has never been attempted directly before. However, there exist some works which explicitly calculate partial maps. Most of the existing reflection symmetry detection methods either detect reflection symmetry axis [41, 5, 4, 7, 15, 21, 24, 28, 32, 35, 39, 43, 48, 54] or

the reflection symmetry region [9, 49, 56, 33, 51, 12]. In [28], the authors proposed Hough transform and a voting based approach to detect reflection symmetry axes through the matching of SIFT and mirrored SIFT descriptors as features. In [9], the authors localized and segmented bilaterally symmetric object based on a symmetry growing method which uses photometric similarity and geometric consistency present in the image. They propagated the pairs of symmetrically matched local features to their neighbors. However, they only compute the symmetric regions which are just binary maps. In [15], the authors used local symmetries such as bilateral and rotational symmetries in order to detect and describe the local keypoints for the task of matching urban or architectural scenes. They have also provided a symmetry distance measure for a given image only for rotational, horizontal, and vertical symmetries. However, the symmetry present in the image may not just be horizontal or vertical. In [47], the authors segmented a symmetric object using a level-set based approach which is a binary symmetry map. However, they do not compute the continuous symmetry map and the NMRF. In [13], the authors computed the continuous symmetry map using a graph-cut based approach. However, they do not compute the NMRF. In this paper, we compute the complete symmetry map (NMRF and SSM) for multiple objects exhibiting different symmetry orientations. There have also been significant works in detecting reflection in 3D geometric models [38, 36, 45, 34, 19]. We pose the problem of finding reflection symmetry map as an intra-image pixel correspondence problem, which involves searching for the mirror reflection of a pixel within the image itself. There exists a rich body of work on inter-image correspondence [14]. In order to get faster correspondence, there exist many methods for approximate nearest neighbor (ANN) search such as coherency sensitive hashing [22], PatchMatch [6], an optimal algorithm for ANN searching [3], and fast approximate nearest neighbors [40]. We propose a strategy similar to PatchMatch [6] to solve this problem.

3. Estimation of Symmetry Map

Let $w \times h$ be the size of the input image I , let $\mathcal{W} = \{1, \dots, w\}$ and $\mathcal{H} = \{1, \dots, h\}$ be two sets. We define the reflection symmetry map to be a collection of two maps SSM (\mathbf{S}) and NMRF (\mathbf{f}). Here, $\mathbf{f} : \mathcal{W} \times \mathcal{H} \rightarrow \mathcal{W} \times \mathcal{H}$ and $\mathbf{S} : \mathcal{W} \times \mathcal{H} \rightarrow [0, 1]$. If $\mathbf{f} : Q \mapsto Q^m$, then the points Q and Q^m are mirror reflections of each other with symmetry score $\mathbf{S}(Q)$, where $Q, Q^m \in \mathcal{W} \times \mathcal{H}$. We define $\mathbf{S}(Q) = 1 - \|\mathbf{d}_{Q^m}^m - \mathbf{d}_Q\|$, where \mathbf{d}_Q is the normalized (unit norm) SIFT descriptor [27] of the point Q , and $\mathbf{d}_{Q^m}^m$ is the normalized (unit norm) mirrored SIFT descriptor [28] of the point Q^m . Given the input image I , our goal is to compute and visualize the pair (\mathbf{S}, \mathbf{f}) . If we follow a brute force approach to find the mirror reflection of each point in the set

$\mathcal{W} \times \mathcal{H}$, then the computational complexity is $O(d_0 w^2 h^2)$, where d_0 is the dimensionality of the local descriptor. The challenge here is to estimate the symmetry map using an algorithm with lesser computational time. We follow the randomized PatchMatch algorithm [6] to get the symmetry map in $O(ed_0 wh)$ time, where $e \ll \min\{w, h\}$ and e is the number of randomly sampled points in each iteration. We use the reflection symmetry coherency present in the image along with the PatchMatch algorithm to compute the symmetry map (\mathbf{S}, \mathbf{f}) as explained in this section.

3.1. Reflection Symmetry Coherency

In order to compute the reflection symmetry map, we exploit the *reflection symmetry coherency* present in the image. If two points Q_i and Q_i^m represent a mirror symmetry, then any other point in the image should have a mirrored counterpart on its location reflected across the axis/line through $\frac{Q_i^m + Q_i}{2}$ with a direction perpendicular to $Q_i^m - Q_i$. However, we observe that an image may not be fully symmetric, since most of the objects do not have a square or a rectangular boundary. This is due to the fact that there might be a background region present in the image. Due to this fact, symmetry coherency property will not hold true at all the pixels.

3.2. Symmetric PatchMatch

In order to compute the reflection symmetry map SSM and NMRF (\mathbf{S}, \mathbf{f}) , we have to determine both \mathbf{S} and \mathbf{f} at every pixel location. In order to determine \mathbf{f} , we need to determine the mirror reflection point $Q^m \in \mathcal{W} \times \mathcal{H}$ for each point $Q \in \mathcal{W} \times \mathcal{H}$. We pose the problem of determining \mathbf{f} as an intra-image dense pixel correspondence problem. We propose a *PatchMatch* [6] based randomized algorithm in order to solve this correspondence problem. There are three steps involved in the proposed algorithm. As the first step, for most of the points ($\approx 99\%$), we initialize the mirror reflection points randomly except a very few points for which we are able to accurately estimate the mirror reflection points. In the second step, we propagate the candidate pairs of points which are mirror reflections of each other to their neighbors. In the third step, we improve the estimate of mirror reflection for each point by searching for better estimate from a set of randomly sampled points in the neighborhood region of the current estimated mirror reflection point. We discuss each of these steps in detail below.

3.2.1 Initialization

In order to initialize the mirror reflection point to each point, we assign a random mirror reflection points to most of the points except a few points for which we determine approximate mirror reflection points. We find the candidate pairs of points which are mirror reflections of each

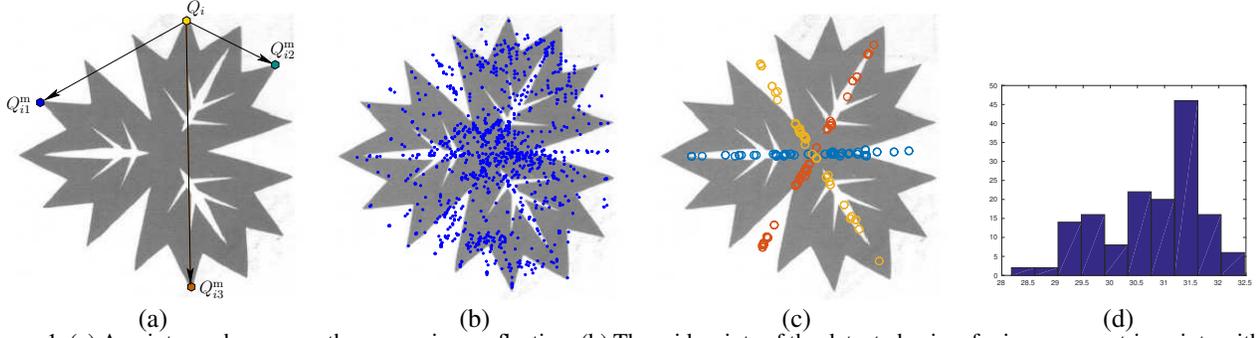


Figure 1. (a) A point may have more than one mirror reflection, (b) The mid-points of the detected pairs of mirror symmetric points without outlier rejection, (c) The three clusters of inlier pairs of mirror symmetric points after performing the outlier rejection and clustering, and (d) The histogram of direction of vectors joining mirror symmetric points in the orange colored cluster.

other as proposed in [28]. Let $\mathcal{Q} = \{Q_i\}_{i=1}^n$ be the set of n keypoints detected using the SIFT algorithm [27]. Let $\{\mathbf{d}_{Q_i}\}_{i=1}^n$ and $\{\mathbf{d}_{Q_i^m}\}_{i=1}^n$ be the sets which consist of the original SIFT descriptors and the mirrored SIFT descriptors of the points in the set \mathcal{Q} respectively, where $\mathbf{d}_{Q_i}, \mathbf{d}_{Q_i^m} \in \mathbb{R}^{128}$, $Q_i \in \mathbb{R}^2$, $\forall i \in \{1, 2, \dots, n\}$. For each point $Q_i \in \mathcal{Q}$, we determine the neighbors having distances less than a predefined threshold (we set to 0.1). Let $\{Q_{i1}^m, Q_{i2}^m, \dots, Q_{in_i}^m\}$ be the set of nearest (distance between descriptors) mirror reflections of the point Q_i , then this point Q_i produces n_i pairs of mirror symmetric points, $\{(Q_i, Q_{i1}^m), (Q_i, Q_{i2}^m), \dots, (Q_i, Q_{in_i}^m)\}$. We consider more than one nearest reflection point as the point can participate in symmetry over more than one axis. For example, consider Figure 1(a) in which the point Q_i has three mirror reflection points $\{Q_{i1}^m, Q_{i2}^m, Q_{i3}^m\}$. In Figure 1(b), we show the mid-points of the detected pairs of mirror symmetric points. We observe that a significant number of pairs are outliers. For the example image considered (having three symmetry axes) in Figure 1, ideally the mid-points of all such pairs should lie on three lines but they are distributed almost throughout the image. We follow the *RansaCov* approach proposed in [31] for outlier rejection and clustering of the detected pairs of mirror symmetric points.

Clustering pairs of mirror symmetric points: Let $\mathcal{M} = \{(Q_i, Q_{ij}^m) : j \in \{1, \dots, n_i\}, i \in \{1, \dots, n\}\} = \{\mathcal{M}_1, \mathcal{M}_2, \dots, \mathcal{M}_p\}$ be the set of detected candidate pairs of mirror symmetric points, where \mathcal{M}_i is a pair of mirror symmetric points. Each symmetry axis in an image gives us one reflection symmetry map. Hence, in order to find the reflection symmetry map for all the symmetry axes present in the image, we cluster these pairs such that the mid-points of all the pairs within a cluster lie on a line and the vectors joining the mirror symmetric points of the pairs present in a cluster have the same direction. We observe that most of the pairs in the set \mathcal{M} are outliers (see Figure 1(b)). Hence, a robust model fitting is inevitable.

There exist many methods for multiple model fitting,

such as energy based method [16], sequential RANSAC [52], J-Linkage [50], T-Linkage [29], robust preference analysis [30], and random cluster model simulated annealing [44]. We follow the best suitable method for our problem given by [31] and discuss the framework below.

For each pair $\mathcal{M}_i = (Q_i, Q_i^m)$, we find the symmetry axis, $\ell_i : a_i x + b_i y + c_i = 0$, which is a line passing through the mid-point $\frac{Q_i + Q_i^m}{2}$ and perpendicular to the vector $Q_i - Q_i^m$. Here the mid-point and the vector joining the mirror symmetric points are known and thereby determine the symmetry axis fully. Now, we find the pairs in the set $\mathcal{M} \setminus \{\mathcal{M}_i\}$ which agree with the symmetry axis ℓ_i with some tolerance ($\epsilon_d, \epsilon_\theta$). A pair (Q_j, Q_j^m) agrees to the symmetry axis ℓ_i if it satisfies the inequalities (1) and (2).

$$\arccos \left(\frac{[a_i \ b_i] (Q_j - Q_j^m)}{\|[a_i \ b_i]^T\|_2 \|Q_j - Q_j^m\|_2} \right) < \epsilon_\theta \quad (1)$$

$$\frac{\min\{d_1, d_2\}}{\max\{d_1, d_2\}} > 1 - \epsilon_d \quad (2)$$

where d_1 and d_2 are the distances of the points Q_j and Q_j^m from the line ℓ_i and ϵ_θ and ϵ_d are the tolerance parameters. We set ϵ_d to 0.03 and ϵ_θ to 2° in our experiments. See Figure 2 for a graphical illustration.

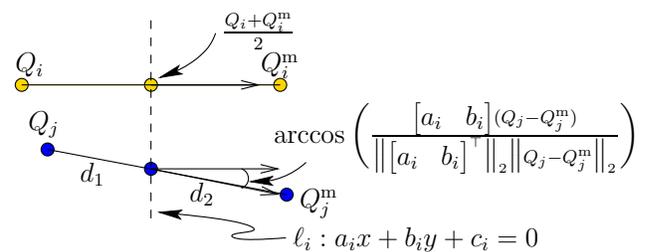


Figure 2. Illustration of a pair (Q_j, Q_j^m) agreeing to the symmetry axis $a_i x + b_i y + c_i = 0$ defined by the mid-point and the vector joining the points of the pair (Q_i, Q_i^m) .

Let \mathcal{I}_i be the set of those pairs which satisfy the inequalities (1) and (2) and let $\mathcal{I} = \{\mathcal{I}_1, \mathcal{I}_2, \dots, \mathcal{I}_p\}$ be the set

containing such sets for all the pairs \mathcal{M}_i . Here, each set \mathcal{I}_i represents a cluster of pairs agreeing with the symmetry axis defined by the pair \mathcal{M}_i . However, out of these p sets only k are desired, where k is the number of symmetric objects present in the image. The observation is that the number of pairs in each set are only a few. A few sets, as exceptions, contain a sufficient number of pairs because most of them are outliers and an outlier may not agree to the symmetry axis defined by another outlier. Therefore, we frame this problem of selecting k out of p sets as the maximum set coverage problem where we try to cover as many possible pairs as possible only using k sets.

This *maximum set coverage* problem can be formulated as an integer linear program as below.

$$\begin{aligned}
& \text{maximize} && \sum_{i=1}^p y_i \\
& \text{subject to} && \sum_{j=1}^p z_j \leq k \\
& && \sum_{j: \mathcal{I}_j \ni \mathcal{M}_i} z_j \geq y_i, \forall \mathcal{M}_i \in \mathcal{M} \\
& && 0 \leq y_i \leq 1, z_j \in \{0, 1\}. \quad (3)
\end{aligned}$$

Here the variable y_i equals 1 if the pair \mathcal{M}_i is in one of the returned subsets and 0 otherwise. The first constraint indicates that the number of returned subsets cannot be more than k . The second constraint indicates that if $y_i \geq 0$ then at least one subset \mathcal{I}_j is selected such that $\mathcal{M}_i \in \mathcal{I}_j$. To solve the above integer linear program, we use [31].

Let $\mathcal{C}_1, \mathcal{C}_2, \dots, \mathcal{C}_k$ be the resulting k clusters containing pairs of mirror symmetric points. Let \mathcal{L}_i and \mathcal{R}_i be the sets of left and right side points of pairs in the cluster $\mathcal{C}_i, \forall i \in \{1, 2, \dots, k\}$, respectively. We assume that the number of symmetric objects, k , present in the image is known. In Figure 1(c), we show the resulting clusters. Each cluster represents a unique symmetry axis. Therefore, we find the symmetry map (S_h, \mathbf{f}_h) for each cluster $\mathcal{C}_h, h \in \{1, \dots, k\}$, separately. We initialize the mirror reflections of the points in the set \mathcal{L}_h to the points in the set \mathcal{R}_h and vice versa. We randomly initialize the mirror reflections for all the points in the set $\{\mathcal{W} \times \mathcal{H}\} \setminus \{\mathcal{L}_h \cup \mathcal{R}_h\}$. We further observe while doing experiment that $|\mathcal{L}_h \cup \mathcal{R}_h| \ll |\mathcal{W} \times \mathcal{H}|$. Here, $|\cdot|$ represents the cardinality of a set.

3.2.2 Propagation

Since we assign the correct mirror reflection points to some points in the initialization step, our goal is to propagate such initializations to the neighboring points. If these inlier pairs were ideal, we could have directly found the mirror reflection of a point by reflecting it through any of the inlier pair. This situation is shown graphically in light gray color in the

Figure 3(a). However, we observe that the inlier pairs of the mirror symmetric points are not that ideal as the mid-points of all the mirror symmetric points are not collinear and the vectors joining these mirror symmetric points do not point in an identical direction. For example, consider the histogram, shown in Figure 1(d) of the directions of the vectors joining mirror symmetric points in the orange cluster where all the directions are not identical. Therefore, we consider more than one pair to find the correct mirror reflection point.

Let \mathcal{C}_h be the cluster under consideration and Q_i be the point under consideration. We randomly select two points $Q_j, Q_k \in \mathcal{L}_h$. Let d_{ij} and d_{ik} be the distances of the points Q_j and Q_k from the point Q_i , respectively. Let Q_j^m, Q_k^m , and Q_i^m be the current estimated mirror reflection points of the points Q_i, Q_j and Q_k , respectively. Let θ_{ij} be the angle between the vectors $Q_i - Q_j$ and $Q_j^m - Q_j$ and θ_{ik} be the angle between the vectors $Q_i - Q_k$ and $Q_k^m - Q_k$. Now, we consider the two points Q_j^m and Q_k^m such that the point Q_{ij}^m is located at distance d_{ij} and the vector $Q_{ij}^m - Q_j^m$ makes an angle θ_{ij} with the vector $Q_j - Q_j^m$. Similarly, the point Q_{ik}^m is located at distance d_{ik} and the vector $Q_{ik}^m - Q_k^m$ makes an angle θ_{ik} with the vector $Q_k - Q_k^m$. Consider the Figure 3(b) for a graphical illustration of this concept.

We search for a better estimate of the mirror reflection point of the point Q_i by exploiting the symmetry coherency property. According to symmetry coherency property, if (Q_j, Q_j^m) and (Q_k, Q_k^m) are good matches, then (Q_i, Q_{ij}^m) and (Q_i, Q_{ik}^m) should also be good matches. Now, we have three choices Q_{ij}^m, Q_{ik}^m , and Q_i^m for the mirror reflection point of the point Q_i . We choose the best reflection out of these three choices as given in equation (4).

$$Q_i^{\text{m,p}} = \underset{Q \in \{Q_{ij}^m, Q_{ik}^m, Q_i^m\}}{\text{arg min}} \|\mathbf{d}_{Q_i} - \mathbf{d}_Q\|_2. \quad (4)$$

3.2.3 Random Search

We observe that the mirror reflections of a point through all the pairs in the nearest cluster are not identical. For example, consider Figure 3(c). The mirror reflection of a point Q_i through all the inlier pairs are shown. In the zoomed-in regions, it is clearly visible that the mirror reflections are not concentrated on a single point but are rather spread out spatially.

Therefore, after estimating a better mirror reflection point $Q_i^{\text{m,p}}$ of the point Q_i in the propagation step, we try to improve the match further. We randomly select e points in the circular region of radius r (shown in Figure 3(b)) around the point $Q_i^{\text{m,p}}$ as $\mathcal{S} = \{Q_i^{\text{m,p}} + R_i\}_{i=1}^e \cup \{Q_i^{\text{m,p}}\}$. Here, $R_i = [\rho \cos(\psi) \quad \rho \sin(\psi)]^T$, and ρ and ψ are random variables following uniform probability distributions, that is $\rho \sim \mathcal{U}(1, r)$, $\psi \sim \mathcal{U}(0, 2\pi)$, where r is the radius. In

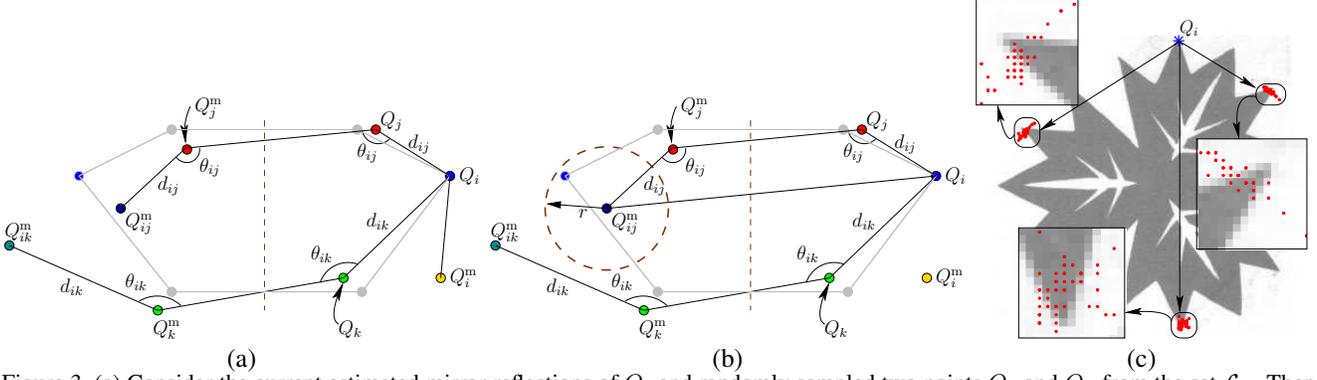


Figure 3. (a) Consider the current estimated mirror reflections of Q_i and randomly sampled two points Q_j and Q_k from the set \mathcal{L}_h . Then we search for the best mirror reflection for the point Q_i among the points Q_i^m , Q_{ij}^m and Q_{ik}^m , (b) We improve the current estimate by searching for better reflection point from the randomly sampled points from a circle around the best estimate among the points Q_i^m , Q_{ij}^m and Q_{ik}^m , and (c) the mirror reflection of the point Q_i through all the inlier pairs of cluster \mathcal{C}_h mirror symmetric points.

order to update the estimate Q_i^m , we choose the best mirror reflection point from the set \mathcal{S} as given below.

$$Q_i^{m,r} = \arg \min_{Q \in \mathcal{S}} \|\mathbf{d}_{Q_i} - \mathbf{d}_Q^m\|_2. \quad (5)$$

Finally, we update the reflection symmetry map as per the equations below.

$$\mathbf{f}_h(Q_i) = Q_i^{m,r}, \quad \mathbf{f}_h(Q_i^{m,r}) = Q_i, \quad (6)$$

$$S_h(Q_i) = e^{-\|\mathbf{d}_{Q_i} - \mathbf{d}_{Q_i^{m,r}}^m\|_2^2} e^{-\frac{d_{Q_i}^2}{(w^2+h^2)\sigma_d^2}} \quad (7)$$

$$S_h(Q_i^{m,r}) = e^{-\|\mathbf{d}_{Q_i} - \mathbf{d}_{Q_i^{m,r}}^m\|_2^2} e^{-\frac{d_{Q_i^{m,r}}^2}{(w^2+h^2)\sigma_d^2}}. \quad (8)$$

Here, d_{Q_i} and $d_{Q_i^{m,r}}$ are the distances of the points Q_i and $Q_i^{m,r}$ from the set \mathcal{I} , respectively. Here, we select the radius r to be equal to the radius of the region in which the mirror reflections of point Q_i through all the inlier pairs lie. It can be observed that r remains the same for all the points Q_i . Therefore, we calculate it once for the whole process. We found this value on an average to be equal to 10. However, it depends on the value of the tolerance parameters ($\epsilon_d, \epsilon_\theta$) (r increases if we increase the value of any of them or both). We set $e = 15$.

The random search step is an essential part in the whole algorithm as it helps in getting out of the local minima and it also helps in improving the current estimate of the mirror reflection of a point. In the worst case, the current estimate may not get improved but it can not get worse because the current estimate is also in the set \mathcal{S} .

Other Schemes: Instead of the proposed iterative random search scheme, we could use the below schemes. Let \mathcal{C}_h be the cluster under consideration and Q_i be the point under consideration. Let $\mathcal{Q}_h^i = \{Q_{i1}^m, Q_{i2}^m, \dots, Q_{i|\mathcal{C}_h|}^m\}$ be the set containing mirror reflections of the point Q_i from all the pairs (Q_j, Q_j^m) , $Q_j \in \mathcal{L}_h$, $Q_j^m \in \mathcal{R}_h$, and

$j \in \{1, 2, \dots, |\mathcal{C}_h|\}$. Here, Q_{i2}^m is the mirror reflection of the point Q_i through the axis defined by the pair (Q_j, Q_j^m) . **Mean mirror pixel (MMP):** Assign mirror reflection of the point Q_i to the mean of the points $\{Q_{i1}^m, Q_{i2}^m, \dots, Q_{i|\mathcal{C}_h|}^m\}$. **Best mirror pixel (BMP):** Assign mirror reflection of the point Q_i to $\arg \max_{Q \in \mathcal{Q}_h^i} \|\mathbf{d}_i - \mathbf{d}_P^m\|_2$. We compare our main approach (termed Iterative Random Search (IRS)) with these schemes in the Section 4 and show that IRS performs better.

Algorithm 1 Symmetric PatchMatch

Input: Image I containing an object with k symmetry axes.

- 1: Find the set \mathcal{M} and the clusters $\mathcal{C}_1, \mathcal{C}, \dots, \mathcal{C}_k$ as discussed in Section 3.
- 2: **for** all the clusters ($h \in \{1, 2, \dots, k\}$) **do**
- 3: Initialize the mirror reflections of points in set \mathcal{L}_h to the points in the set \mathcal{R}_h and vice versa.
- 4: Randomly initialize the mirror reflections for the points in set $\{\mathcal{W} \times \mathcal{H}\} \setminus \{\mathcal{L}_h \cup \mathcal{R}_h\}$.
- 5: **while** not converged **do**
- 6: For the point $Q_i \in \mathcal{W} \times \mathcal{H}$ randomly select two points Q_j, Q_k from the cluster \mathcal{C}_h .
- 7: For the point Q_i , select the best mirror image estimate $Q_i^{m,p}$. $Q_i^{m,p} \leftarrow \arg \min_{Q \in \{Q_{ij}^m, Q_{ik}^m, Q_i^m\}} \|\mathbf{d}_{Q_i} - \mathbf{d}_Q^m\|_2$.
- 8: Randomly sample e points from a circle around point $Q_i^{m,p}$.
 $\mathcal{S} \leftarrow \{Q_i^{m,p}\} \cup \{Q_i^{m,p} + R_i\}_{i=1}^e$.
- 9: Improve $Q_i^{m,p}$: $Q_i^{m,r} \leftarrow \arg \min_{Q \in \mathcal{S}} \|\mathbf{d}_{Q_i} - \mathbf{d}_Q^m\|_2$.
- 10: Update the symmetry map (S_h, \mathbf{f}_h) using Equations (6), (7), and (8).
- 11: **end while**
- 12: **end for**

Output: Symmetry maps $(S_1, \mathbf{f}_1), \dots, (S_k, \mathbf{f}_k)$.

4. Results and Evaluation

4.1. Results

In Algorithm 1, we present the complete symmetric-PatchMatch algorithm. We represent the NMRF (f) by a HSV color image where at each pixel Q , we set the hue equal to the angle made by vector $Q - f(Q)$ from the origin, saturation equal to the magnitude of the vector $Q - f(Q)$, and value equal to the symmetry score $S(Q)$. Equivalently, mirror image of the pixel Q is located in the the direction equal to the hue (H) at pixel Q , at a distance equal to the saturation (S) at pixel Q , and with a symmetry score equal to the value (V) at pixel Q . We represent the SSM (S) by a gray-scale image. In Figure 4, we show an improvement in the computed symmetry map over the iterations. We observe that even in one iteration we get many good matches, but still it improves more as more number of iterations are completed. In order to measure the quality, we generate a new image, I_m , by replacing each pixel by its estimated mirror reflection. We also weigh the intensity with the symmetry score. Therefore, $I_m(Q) = S(Q)f(Q), \forall Q \in \mathcal{W} \times \mathcal{H}$. For the image given in Figure 4(a), we represent the SSM S in the first row and the NMRF f in the second row for the iterations $\frac{1}{2}, 2, 4, 6$, and 10. In the third and the fourth row, we present the zoomed-in window (shown in red color in (a)) from the SSM and NMRF. And in the row five, we represent the zoomed-in window from the image I_m . Now we measure the improvement over the iterations by measuring the structural similarity (SSIM) [53] and PSNR between the window from the image from I_m and the original window from the original image. We report the SSIM and PSNR against the iteration number in the Figure 4(b). We observe that both SSIM and PSNR increase as the number of iterations increases. We observe that while doing experiment on more images, 6 iterations are sufficient for the convergence. In Figure 5, we show the computed reflection symmetry maps for five example images from the dataset provided in this challenge [25].

4.2. Evaluation of NMRF

Correspondences Rate. Let Q_i^{em} be an estimated mirror reflection of the point Q_i and let Q_i^{gm} be the ground-truth mirror reflection of the point Q_i . We decide whether the estimated mirror reflection Q_i^{em} of the point Q_i or the correspondence (Q_i, Q_i^{em}) is correct based on a distance threshold τ . If the distance $\|Q_i^{em} - Q_i^{gm}\|_2$ between the points Q_i^{em} and Q_i^{gm} is less than the distance threshold τ , then the correspondence (Q_i, Q_i^{em}) is correct and otherwise incorrect. For a given threshold τ , we count the correspondences (Q_i, Q_i^{em}) for which the condition $\|Q_i^{em} - Q_i^{gm}\|_2 < \tau$ holds true. We manually extracted 130 ground truth pairs of mirror symmetric points from an image in the dataset of this challenge [25]. In Figure 6, we show the correspondences

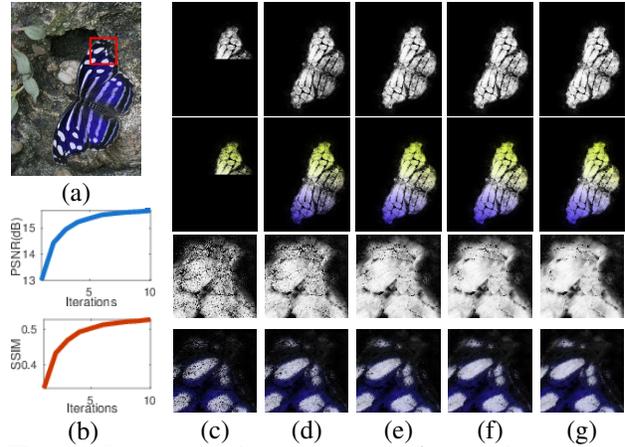


Figure 4. Improvement in symmetry map (first row: f , second row: S) over the iterations. (c) Iteration $\frac{1}{2}$ (d) Iteration 2, (e) iteration 4, (f) iteration 6, and (g) iteration 10. In the third and the fourth rows, we show the zoomed-in window in S and I_m , respectively. Plots in (b) represent the SSIM and PSNR vs iteration number curves for the window in I_m shown in the fourth row.

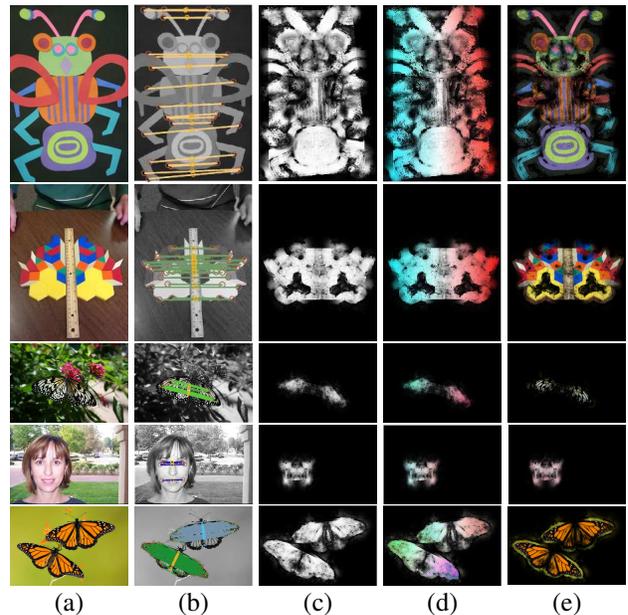


Figure 5. (a) Images, (b) Pairs of mirror symmetric points (different colors represent different clusters), (c) S , (d) f , and (e) I_m .

rate against the distance threshold τ for the schemes: Iterative random search (IRS), mean mirror pixel (MMP), and best mirror pixel (BMP). We observe that the correspondences rate for IRS is higher than that of the MMP and BMP.

SSIM and PSNR. The similarity between the images I and I_m represents the quality of estimated NMRF. The more similar the images I and I_m , the better will be the quality of NMRF. In order to measure the similarity between the

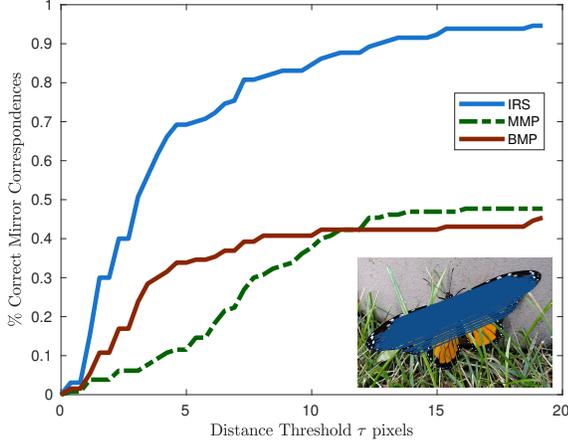


Figure 6. Fraction of correct mirror symmetric correspondences vs the distance threshold τ curve for the methods IRS (iterative random search), MMP (Mean Mirror Pixel), and BMP (Best Mirror Pixel).

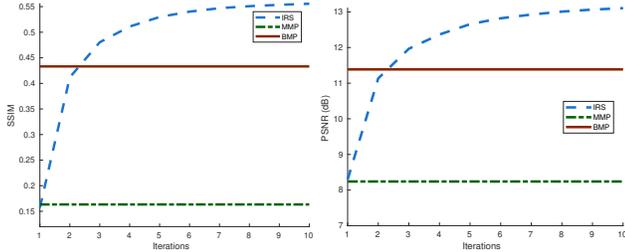


Figure 7. SSIM and PSNR vs the iteration number curve for the methods IRS, MMP, and BMP.

images I and I_m , we use SSIM and PSNR. In Figure 7, we plot SSIM and PSNR against the iteration number for the schemes IRS, MMP, and BMP from the images with single symmetry axis taken from the dataset [25]. Since there are no iterations for the schemes MMP and BMP, SSIM and PSNR remain constant for them. For IRS, both SSIM and PSNR increase as the number of iterations increase.

4.3. Evaluation of SSM

In order to evaluate the SSM, we follow a method similar to what is adopted to evaluate a saliency algorithm. First, to get the ground truth symmetry map (binary), we manually select 76 images which contain objects exhibiting reflection symmetry in the dataset [8], which also has the ground truth binary segmentation maps. Let G denote a ground truth map. We also manually segment 64 images from [25] containing symmetric objects to augment this dataset with more images. To compare with the binary symmetry map, denoted as Φ , we threshold the estimated symmetry score map with different thresholds (with a step size of 12.8 in the range $[0, 255]$) as proposed in [1]. We find the precision and recall rates ($\frac{|\Phi \wedge G|}{|\Phi|}$ and $\frac{|\Phi \wedge G|}{|G|}$, respectively). In Figure 8, we plot the precision vs recall curve. We also compute the AUC

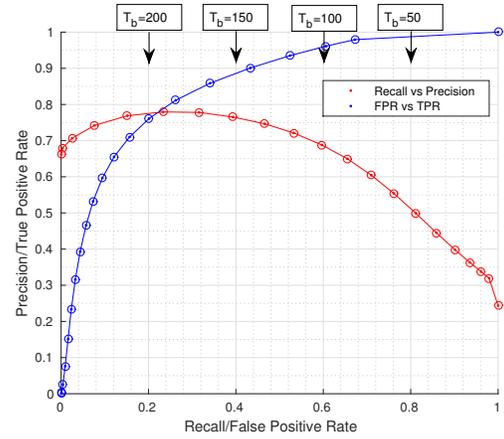


Figure 8. The Precision Vs Recall curve (red) and the ROC curve (blue) for the threshold with step size 12.8 in the range 0 to 255, on 140 images.

for the false positive rate (FPR) vs true positive rate (TPR) curve. The true positive rate measures the fraction of symmetric pixels which are correctly labelled $\frac{|\Phi \wedge G|}{|G|}$. The false positive rate measures the fraction of non-symmetric pixels which are classified as symmetric pixels $\frac{|\Phi \wedge \neg G|}{|\neg G|}$. The AUC and the mean precision, mean recall, and F_β scores are reported in Table 1 for the threshold value $T_f = 100$. Our method takes on an average 2.5 minutes for an image (of size approximately equal to 640 in both the dimensions) to compute the reflection symmetry map (NMRF and SSM) in Intel Core i5 processor with 8 GB RAM.

Table 1. Precision, Recall and F_β ($\beta = 0.3$) at $T_b = 100$, and the area under the FPR vs TPR curve.

Measure	Precision	Recall	F_β	AUC
Score	0.6880	0.60	0.6794	0.8531

4.4. Evaluation of Symmetry Axes Detection

We detect the symmetry axes using the estimated correspondences between the mirror symmetric pixels. We use the framework proposed in Section 3 to cluster the correspondences and estimate the axes parameters. We compare our results with the methods given in [5] and [28] by computing the precision and the recall values. In Figure 9 we represent the precision and the recall values. Our performance is comparable to that of [5] and [28].

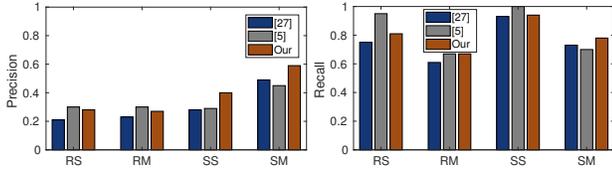


Figure 9. The Precision and the Recall values on the dataset [46]. Here, RS: real images single axis, RM: real images with multiple axes, SS: synthetic images with single axis, and SM: synthetic images with multiple axes.

5. Application: Symmetry Preserving Stylization

Energy guided or structure preserving stylization of images has been an interesting application in computer graphics, especially in non-photorealistic rendering. DeCarlo and Santella stylized image by preserving more details in the region where the eye-tracker responses were high and fewer details in the regions where eye-tracker responses were low [10]. There are various methods preserving structures such as saliency, shape, flow, texture, and gradient while stylizing ([11, 55, 18, 42, 17, 57]). We propose a simple method to stylize the input image while preserving the abstract details in symmetric regions and removing details in the non-symmetric regions. We process only the SSM component of the reflection symmetry map for this application. We first over-segment the image into superpixels by using SLIC [2]. We merge the superpixel (let us denote it by s_i) having the minimum average symmetry score to its adjacent superpixel (let us denote it by s_j) which has the next higher average symmetry score. In order to do this, we first construct a graph where each superpixel is a vertex and there is an edge between the two vertices if the corresponding superpixels are neighbors of each other. The superpixel merging is equivalent to *edge contraction* in the resulting graph. Let u_i be the vertex corresponding to the superpixel s_i and let the set \mathcal{N}_{u_i} be the set of neighborhoods of the vertex u_i , and u_j be the vertex corresponding to the superpixel s_j and let the set \mathcal{N}_{u_j} be the set of neighborhoods of the vertex u_j . Here, we observe that $u_i \in \mathcal{N}_{u_j}$ and $u_j \in \mathcal{N}_{u_i}$. Contracting the edge $e_{ij} = (u_i, u_j)$ results in a new vertex w and the set of neighbors of the resulting vertex w becomes $\mathcal{N}_w = \{\mathcal{N}_{u_i} \setminus \{u_j\}\} \cup \{\mathcal{N}_{u_j} \setminus \{u_i\}\}$. After merging the superpixels s_i and s_j , we set the average symmetry score of the new superpixel to be the average of the symmetry scores of s_i and s_j . We keep merging superpixels till the minimum average symmetry score over the remaining superpixels is above a predefined threshold. We then smooth each superpixel by replacing the color of each pixel by the average color within the superpixel.

In Figure 10, we present the results of the stylization of few images. We also present the results of other popular structure preserving stylization methods ([42],[55]) to vi-

sually compare our results. We are able to preserve more details in the symmetric regions and fewer details in the non-symmetric regions as compared to these methods. We observe from the results of these methods (shown in Figure 10 (e) and (f)) that these methods retain a significant amount of details in the non-symmetric regions, but our method removes most of the details from the non-symmetric regions.

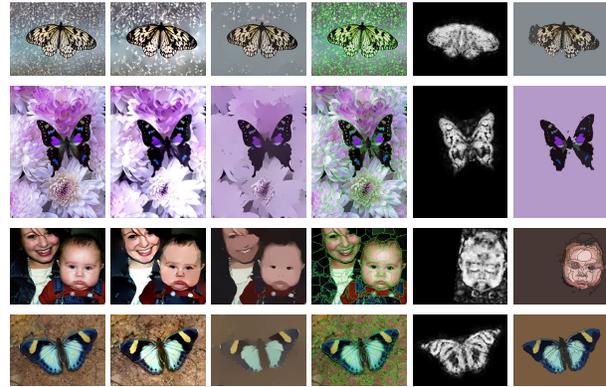


Figure 10. Results of stylization (a) Input images, (b) [42], (c) [55], (d) Superpixel over-segmentation (green colored boundaries), (e) Computed reflection symmetry map, (f) Proposed method.

6. Conclusion

We have presented a novel definition and description of the concept of reflection symmetry map for a given image containing a symmetric object. This concept has been realized using a randomized algorithm which enables us to compute the reflection symmetry map. In this work, for a given image containing an object exhibiting reflection symmetry, we have computed the symmetry map which is a collection of two maps: NMRF and SSM. To estimate the symmetry map, we have exploited the reflection symmetry coherency present in the image and efficient approximate nearest neighbor search methods. We have shown the importance of symmetry map by using it in an important application - symmetry preserving image stylization. As a future work, we would like to compute the symmetry map without any computation of candidate pairs of points which are mirror reflections of each other. We also would like to use the symmetry map for more computer vision applications.

Acknowledgement

Rajendra Nagar was supported by TCS Research Scholarship. Shanmuganathan Raman was supported by SERB-DST.

References

- [1] R. Achanta, S. Hemami, F. Estrada, and S. Susstrunk. Frequency-tuned salient region detection. In *CVPR*, pages 1597–1604. IEEE, 2009. 7
- [2] R. Achanta, A. Shaji, K. Smith, A. Lucchi, P. Fua, and S. Susstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *IEEE TPAMI*, 34(11):2274–2282, 2012. 8
- [3] S. Arya, D. M. Mount, N. S. Netanyahu, R. Silverman, and A. Y. Wu. An optimal algorithm for approximate nearest neighbor searching fixed dimensions. *Journal of the ACM (JACM)*, 45(6):891–923, 1998. 2
- [4] I. Atadjanov and S. Lee. Bilateral symmetry detection based on scale invariant structure feature. In *ICIP*, pages 3447–3451. IEEE, 2015. 1
- [5] I. R. Atadjanov and S. Lee. Reflection symmetry detection via appearance of structure descriptor. In *European Conference on Computer Vision*, pages 3–18. Springer, 2016. 1, 7
- [6] C. Barnes, E. Shechtman, A. Finkelstein, and D. Goldman. Patchmatch: a randomized correspondence algorithm for structural image editing. *ACM TOG*, 28(3), 2009. 2
- [7] 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. 1
- [8] M.-M. Cheng, N. J. Mitra, X. Huang, and S.-M. Hu. Salienshape: Group saliency in image collections. *The Visual Computer*, 30(4):443–453, 2014. 7
- [9] M. Cho and K. M. Lee. Bilateral symmetry detection via symmetry-growing. In *BMVC*, pages 1–11, 2009. 2
- [10] D. DeCarlo and A. Santella. Stylization and abstraction of photographs. In *ACM transactions on graphics (TOG)*, volume 21. ACM, 2002. 8
- [11] Z. Farbman, R. Fattal, D. Lischinski, and R. Szeliski. Edge-preserving decompositions for multi-scale tone and detail manipulation. In *ACM Transactions on Graphics (TOG)*, volume 27, page 67. ACM, 2008. 8
- [12] S. A. Friedberg. Finding axes of skewed symmetry. *Computer Vision, Graphics, and Image Processing*, 34(2):138–155, 1986. 2
- [13] H. Fu, X. Cao, Z. Tu, and D. Lin. Symmetry constraint for foreground extraction. *IEEE transactions on cybernetics*, 44(5):644–654, 2014. 2
- [14] T. Hassner and C. Liu. *Dense Image Correspondences for Computer Vision*. Springer, 2015. 2
- [15] D. C. Hauage and N. Snavely. Image matching using local symmetry features. In *CVPR*, pages 206–213. IEEE, 2012. 1, 2
- [16] H. Isack and Y. Boykov. Energy-based geometric multi-model fitting. *IJCV*, 97(2):123–147, 2012. 3
- [17] H. Kang and S. Lee. Shape-simplifying image abstraction. In *Computer Graphics Forum*, volume 27, pages 1773–1780. Wiley Online Library, 2008. 8
- [18] H. Kang, S. Lee, and C. K. Chui. Flow-based image abstraction. *IEEE transactions on visualization and computer graphics*, 15(1):62–76, 2009. 8
- [19] M. Kazhdan, B. Chazelle, D. Dobkin, A. Finkelstein, and T. Funkhouser. A reflective symmetry descriptor. In *European Conference on Computer Vision*, pages 642–656. Springer, 2002. 2
- [20] V. G. Kim, Y. Lipman, and T. A. Funkhouser. Symmetry-guided texture synthesis and manipulation. *ACM Trans. Graph.*, 31(3):22, 2012. 1
- [21] S. Kondra, A. Petrosino, and S. Iodice. Multi-scale kernel operators for reflection and rotation symmetry: further achievements. In *CVPR Workshops*, pages 217–222. IEEE, 2013. 1
- [22] S. Korman and S. Avidan. Coherency sensitive hashing. In *ICCV*, pages 1607–1614. IEEE, 2011. 2
- [23] K. Koser, C. Zach, and M. Pollefeys. Dense 3d reconstruction of symmetric scenes from a single image. In *Joint Pattern Recognition Symposium*. Springer, 2011. 1
- [24] J. Liu, G. Slota, G. Zheng, Z. Wu, M. Park, S. Lee, I. Rauschert, and Y. Liu. Symmetry detection from real-world images competition 2013: Summary and results. In *CVPR Workshops*, pages 200–205. IEEE, 2013. 1
- [25] Y. Liu and S. Dickinson. Detecting symmetry in the wild. In *International Conference on Computer Vision (ICCV) Workshops*. IEEE, 2017. 6, 7
- [26] Y. Liu, H. Hel-Or, and C. S. Kaplan. *Computational symmetry in computer vision and computer graphics*. Now publishers Inc, 2010. 1
- [27] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004. 2, 3
- [28] G. Loy and J.-O. Eklundh. Detecting symmetry and symmetric constellations of features. In *ECCV*, pages 508–521. Springer, 2006. 1, 2, 3, 7
- [29] L. Magri and A. Fusiello. T-linkage: A continuous relaxation of j-linkage for multi-model fitting. In *CVPR*, pages 3954–3961, 2014. 3
- [30] L. Magri and A. Fusiello. Robust multiple model fitting with preference analysis and low-rank approximation. In *Proceedings of the British Machine Vision Conference (BMVC)*, pages 20.1–20.12. BMVA Press, 2015. 3
- [31] L. Magri and A. Fusiello. Multiple model fitting as a set coverage problem. In *CVPR*, June 2016. 3, 4
- [32] M. Mancas, B. Gosselin, B. Macq, et al. Fast and automatic tumoral area localisation using symmetry. In *ICASSP 2005*, 2005. 1
- [33] 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
- [34] A. Martinet, C. Soler, N. Holzschuch, and F. X. Sillion. Accurate detection of symmetries in 3d shapes. *ACM Transactions on Graphics (TOG)*, 25(2), 2006. 2
- [35] Y. Ming, H. Li, and X. He. Symmetry detection via contour grouping. In *ICIP*, pages 4259–4263. IEEE, 2013. 1
- [36] 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. ACM, 2006. 2
- [37] N. J. Mitra, L. J. Guibas, and M. Pauly. Symmetrization. *ACM Transactions on Graphics (TOG)*, 26(3), 2007. 1

- [38] 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. 1, 2
- [39] Q. Mo and B. Draper. Detecting bilateral symmetry with feature mirroring. In *CVPR Workshops*, 2011. 1
- [40] M. Muja and D. G. Lowe. Scalable nearest neighbor algorithms for high dimensional data. *IEEE TPAMI*, 36(11), 2014. 2
- [41] R. Nagar and S. Raman. Reflection symmetry axes detection using multiple model fitting. *IEEE Signal Processing Letters*, 24(10):1438 – 1442, 2017. 1
- [42] A. Orzan, A. Bousseau, P. Barla, and J. Thollot. Structure-preserving manipulation of photographs. In *Proceedings of the 5th international symposium on Non-photorealistic animation and rendering*, pages 103–110. ACM, 2007. 8
- [43] V. Patraucean, R. Gioi, and M. Ovsjanikov. Detection of mirror-symmetric image patches. In *CVPR Workshops*, pages 211–216, 2013. 1
- [44] T. T. Pham, T.-J. Chin, J. Yu, and D. Suter. The random cluster model for robust geometric fitting. *IEEE TPAMI*, 36(8), 2014. 3
- [45] J. Podolak, P. Shilane, A. Golovinskiy, S. Rusinkiewicz, and T. Funkhouser. A planar-reflective symmetry transform for 3d shapes. *ACM Transactions on Graphics (TOG)*, 25(3):549–559, 2006. 2
- [46] I. Rauschert, J. Liu, and Y. Liu. Symmetry detection competition: A summary of how the competition is carried out. In *Proc. IEEE CVPR Workshop Symmetry Detection Real World Images*, pages 1–66, 2011. 8
- [47] T. Riklin-Raviv, N. Sochen, and N. Kiryati. On symmetry, perspectivity, and level-set-based segmentation. *IEEE transactions on pattern analysis and machine intelligence*, 31(8):1458–1471, 2009. 2
- [48] S. N. Sinha, K. Ramnath, and R. Szeliski. Detecting and reconstructing 3d mirror symmetric objects. In *ECCV*, pages 586–600. Springer, 2012. 1
- [49] Y. Sun and B. Bhanu. Reflection symmetry-integrated image segmentation. *IEEE TPAMI*, 34(9):1827–1841, 2012. 2
- [50] R. Toldo and A. Fusiello. Robust multiple structures estimation with j-linkage. In *ECCV*, pages 537–547. Springer, 2008. 3
- [51] L. Van Gool, T. Moons, and M. Proesmans. Mirror and point symmetry under perspective skewing. In *Computer Vision and Pattern Recognition, 1996. Proceedings CVPR’96, 1996 IEEE Computer Society Conference on*, pages 285–292. IEEE, 1996. 2
- [52] E. Vincent and R. Laganière. Detecting planar homographies in an image pair. In *Proceedings of the 2nd International Symposium on Image and Signal Processing and Analysis*, pages 182–187, 2001. 3
- [53] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004. 6
- [54] 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. 1
- [55] L. Xu, C. Lu, Y. Xu, and J. Jia. Image smoothing via l0 gradient minimization. In *ACM Transactions on Graphics (TOG)*, volume 30. ACM, 2011. 8
- [56] H. Zabrodsky, S. Peleg, and D. Avnir. Symmetry as a continuous feature. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 17(12):1154–1166, 1995. 2
- [57] H. Zhao, X. Mao, X. Jin, J. Shen, F. Wei, and J. Feng. Real-time saliency-aware video abstraction. *The Visual Computer*, 25(11):973–984, 2009. 8