Multiple instance subspace learning via partial random projection tree for local reflection symmetry in natural images

Wei Shen [a], Xiang Bai [b,*], Zihao Hu [b], Zhijiang Zhang [a]

[a] *Key Laboratory of Specialty Fiber Optics and Optical Access Networks, Shanghai University, Shanghai, China*
[b] *School of Electronic Information and Communications, Huazhong University of Science and Technology, Wuhan, China*

ARTICLE INFO

ABSTRACT

Local reflection symmetry detection in nature images is a quite important but challenging task in computer vision. The main obstacle is both the scales and the orientations of symmetric structure are unknown. The multiple instance learning (MIL) framework sheds lights onto this task owing to its capability to well accommodate the unknown scales and orientations of the symmetric structures. However, to differentiate symmetry vs non-symmetry remains to face extreme confusions caused by clutters scenes and ambiguous object structures. In this paper, we propose a novel multiple instance learning framework for local reflection symmetry detection, named multiple instance subspace learning (MISL), which instead learns a group of models respectively on well partitioned subspaces. To obtain such subspaces, we propose an efficient dividing strategy under MIL setting, named partial random projection tree (PRPT), by taking advantage of the fact that each sample (bag) is represented by the proposed symmetry features computed at specific scale and orientation combinations (instances). Encouraging experimental results on two datasets demonstrate that the proposed local reflection symmetry detection method outperforms current state-of-the-arts.

1. Introduction

As a popular low-level representation, reflection symmetry axis, also called *medial axis* in other literatures, has successfully been employed in many applications, such as shape-based object recognition [17,37,52,6], biological shape analysis [41], human gait/pose recognition [45], and topological analysis in sensor networks [12]. Although symmetry-based research community has embodied its remarkable success in such applications, it struggles when being applied to the mainstream issues like object detection and recognition in natural images. The reason is it has a restrictive assumption that the silhouette of the foreground object has been well-segmented, which is unrealistic for natural images. Since foreground segmentation and contour detection in natural images have not been fully solved yet, this limitation seriously restricts the potential of the symmetry-based object recognition.

To broaden the range of potential of symmetry-based research, in this paper, we aim to address the problem: Can we perform symmetry detection directly in the natural images, without pre-segmentation process? There are actually four primitive symmetry types in 2D Euclidean geometry: reflection, rotation, translation and glide-reflection [46,50]. In this paper, we focus on reflection symmetry, which is most ubiquitous in the world around us [22]. So we term "reflection symmetry" by "symmetry" for short in the rest of paper for simplicity. The symmetry axis is the centerline of a ribbon-like object part. Such object parts are common in the nature images, such as the trunk or limbs of humans or animals, roads, rivers and elongated man-made objects. Detecting such symmetry can deliver important hypotheses about the existence of objects, signalling that "there might be something there of about that size" [26]. Therefore, symmetry detection can offer the low-level ridge feature for object part localization [21], the initial seeds for image segmentation [43], and the templates for contour grouping [1]. we refer to [39] and [28] for more thorough presentation about the importance and the usefulness of symmetry in computer vision.

Symmetry detection in natural images is a quite challenging problem, due to the large variation in objects and the cluttered scene. With the growing interests in low-level feature extraction with learning techniques [18,31,43], we believe that symmetry detection with machine learning techniques is the right direction.

To our knowledge, [43] is the first work for learning-based symmetry detection, which defines the symmetry features under multiple scales and orientations and formulate symmetry detection as a multiple instance learning (MIL) problem. Another contribution of them is constructing a dataset for evaluation of

* Corresponding author. Tel.: +86 13297073017.
E-mail addresses: wei.shen@shu.edu.cn (W. Shen), xbai@hust.edu.cn (X. Bai).

Fig. 1. Some examples in the symmetry detection benchmark SYMMAX300 [43]. The groundtruths of symmetries are in yellow. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)
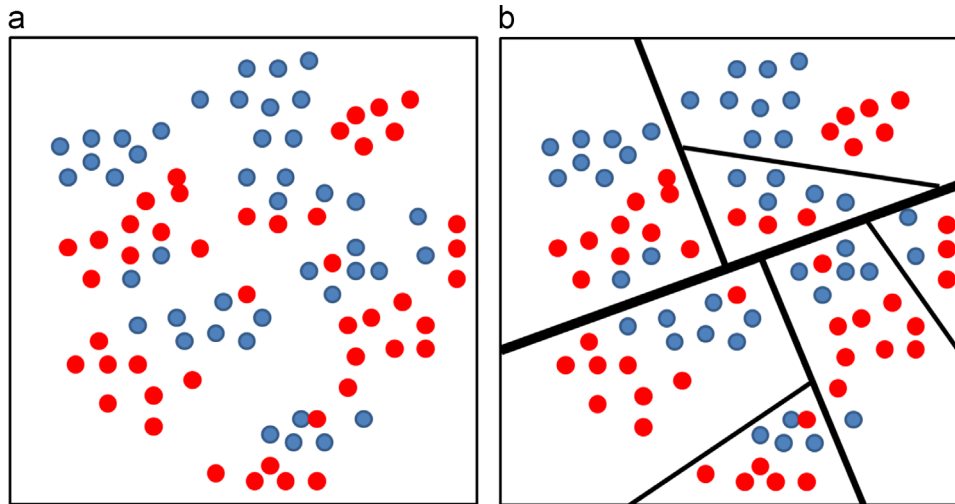

Fig. 2. (a) A group of samples which are hard to classify. Positive samples and negative samples are marked by the blue points and the red points respectively. (b) A proper partition leads to ease of the difficulty of classification. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

symmetry detection (named SYMMAX300), which is converted from the well-known Berkeley Segmentation Benchmark (BSDS300) [32]. Fig. 1 illustrates some examples in this dataset. Their learning-based framework serves as a promising direction for symmetry detection, and gives us some fundamental observations: (1) feature extraction at multiple scales and orientations is necessary for symmetry detection, since a symmetry is essentially related to a object part with the unknown scale and orientation; and (2) how to accommodate the unknown scale and orientation of the symmetry features is the key to learn a effective detector.

In this paper, we develop a novel multiple instance learning framework to accommodate the unknown scale and orientation of the symmetry features for symmetry detection. In training procedure, we first partition the training data into subsets, then train a group of MIL classifiers, one for each subset. In the testing procedure, a testing sample is assigned into one subset according to the partition rule learned in the training procedure, then classified by the MIL model learned on this subset. Our method is motivated by the phenomenon that the symmetry and non-symmetry pixels are quite confusing and difficult to differentiate in the feature space (state-of-the-art algorithm only reaches an *F*-measure of 0.434 on the SYMMAX300 dataset [43]). It is extremely difficult to fit a global model to the training data. Instead, in the spirit of divide-and-conquer strategy [8], partitioning the data into subsets often leads to easier subproblems.

Fig. 2 illustrates our motivation: the blue points and the red points in Fig. 2(a) denote the positive samples and negative samples respectively. Obviously, it is extremely difficult to find an accurate decision boundary by learning a single discriminative model. However, after a proper partition, the samples in each partitioned cell are easy to be classified as shown in Fig. 2(b). To

partition the training data into confident subsets, we propose an algorithm named partial random projection tree (PRPT), in which we recursively partition the training data into left and right subtrees according to the randomized projection of the symmetry features at a certain scale and orientation combination. Such a tree is first learned from the positive samples. After that, each negative sample starts at the root, then recursively branches left or right, and reaches a leaf node finally. The samples in such a leaf node represent a local subspace, and we train the MIL classifier in each subspace. Here we do not partition all the training samples in a single step, because such a strategy will result in meaningless subsets due to the confusion between symmetry (positive samples) and non-symmetry (negative samples). We refer to this method as MISL, for multiple instance subspace learning. The flowchart of MISL is illustrated in Fig. 3.

Note that the proposed PRPT is distinct from the standard random projection tree (RPT) [16,19] in that ours is mainly designed for a special case of MIL setting. PRPT performs unsupervised learning to cluster samples represented by multiple instances, which is known as the multiple instance clustering (MIC) problem [49,15]. Generally, MIC is an intrinsically ambiguous problem due to the mixture of positive and negative instances in each sample. Current MIC methods require a good initialization for cluster centers, otherwise may result in a low performance [51]. However, our special setting (the predefined arrangement of the symmetry features according to scale and orientation combinations) eases the burden of MIC. Here we benefit from this setting that each sample (bag), i.e. image pixel, is represented by features computed at specific scales and orientations (instances) and each positive sample, i.e. symmetry pixel, corresponds one true scale and orientation combination. Therefore, we can divide the positive samples into subsets, where the
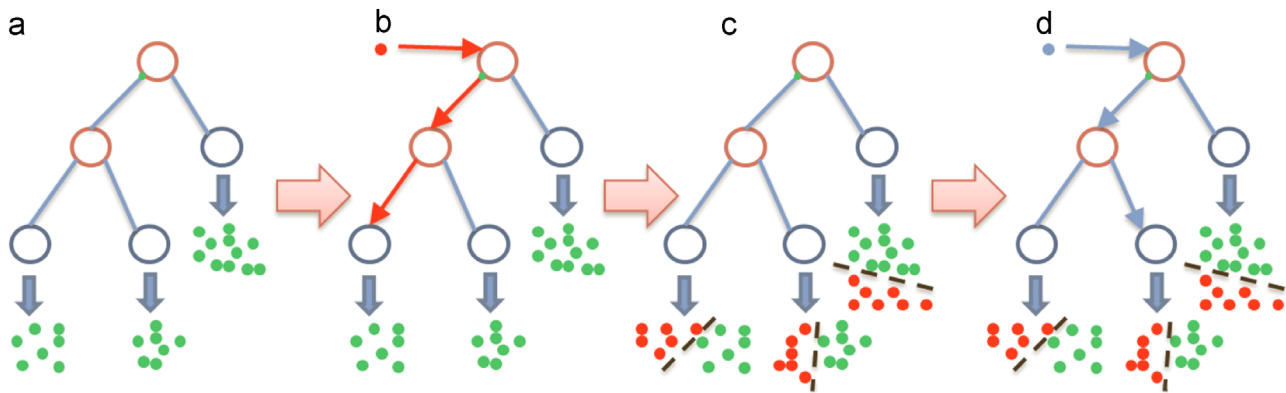
Fig. 3. The flowchart of the proposed multiple instance subspace learning. (a) By constructing a tree structure, the positive samples (denoted by green dots) are divided into leaf nodes. (b) Each negative sample (denoted by red dots) starts at the root, then recursively branches left or right, and reaches a leaf node finally. (c) A group of MIL models are trained on the subsets in leaf nodes respectively. (d) A testing sample (denoted by a blue dot) is classified by the MIL model of the leaf node where it falls into. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)
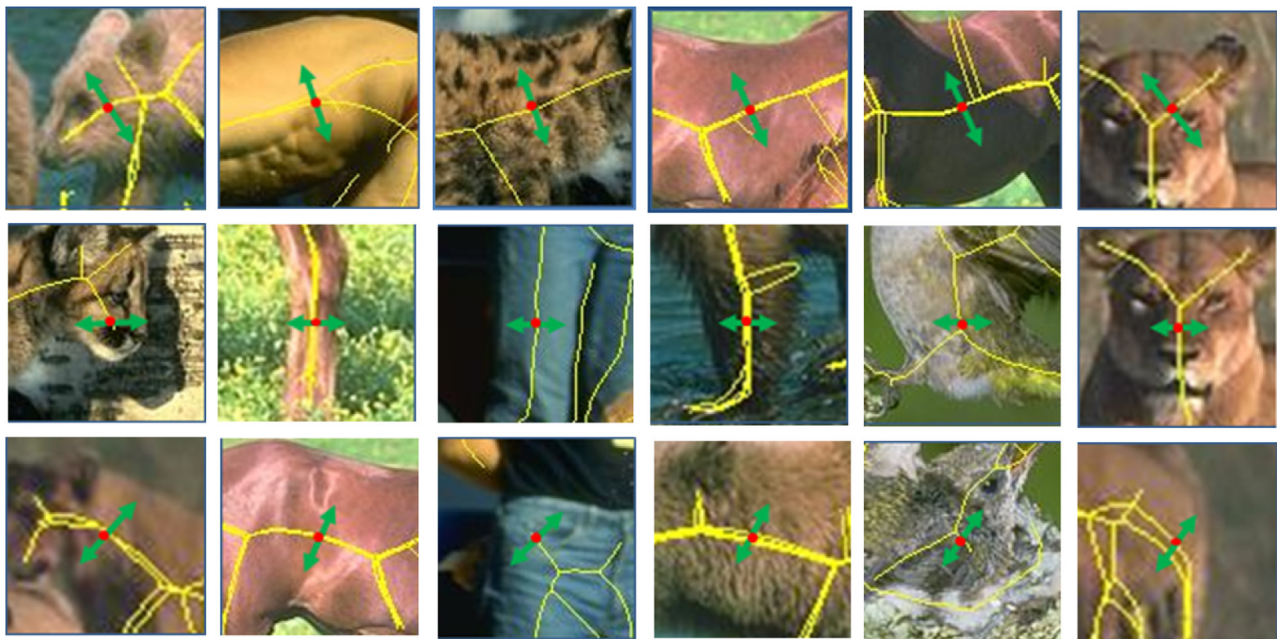

Fig. 4. The positive samples in lead nodes of the proposed partial random projection tree. Each row corresponds one leaf node. The sampled pixels are marked in red and the their hypothetical orientations are marked by green arrows. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

hypothetical scale and orientation combination of the samples are similar by looking up the features computed at one scale and orientation combination recursively. Fig. 4 illustrates the meaningful partitions obtained by PRPT, in each of which the hypothetical orientations of the positive samples are similar. Training a MIL model specifically in a local subspace leads to two advantages: one is the improvement of discrimination, as the positive samples and negative samples in one leaf node are close in the feature space, the other is reduction of computational burden in training.

We also propose two types of symmetry features to describe the intrinsic property of symmetry structures. Our first symmetry feature is defined on pairs of boundaries by multi-scale representation, which aims to search the hypotheses of pairs of boundaries with consistent high strengths; the second symmetry feature is defined on the regions at each sides of symmetries, which aims to ensure the region represented by every symmetry, i.e. the internal region bounded by the hypothesis of a pair of boundary, belongs to one object part. Our features are not just a simple extension of the features used in Tsogkas and Kokkinos's

work [43], as the information embedded in them would boost the performance under the proposed framework.

At last, we construct a new public dataset, named Weizmann Horse SYMMAX (WH-SYMMAX), for symmetry detection, which is converted from the well known Weizmann Horse dataset [10]. By applying skeletonization [38] on provided human-annotated foreground segments, we generate the groundtruth skeletons of horses. Some examples in WH-SYMMAX are shown in Fig. 5. We hope this dataset can be further used to evaluate the methods for symmetry-based class-specific object localization and recognition in the future.

Our main contributions lies in two folds: one is that we propose an algorithm named partial random projection tree to efficiently and robustly partition the training samples of multiple scales and multiple orientations into subsets, which leads to a more compact symmetry representation in each subspace. (We will show that learning classifiers on the subsets obtained by PRPT achieves favorably better performance than other clustering strategy for symmetry detection in Section 4.) The other is that we train the classifiers in each subspace to improve the discrimination
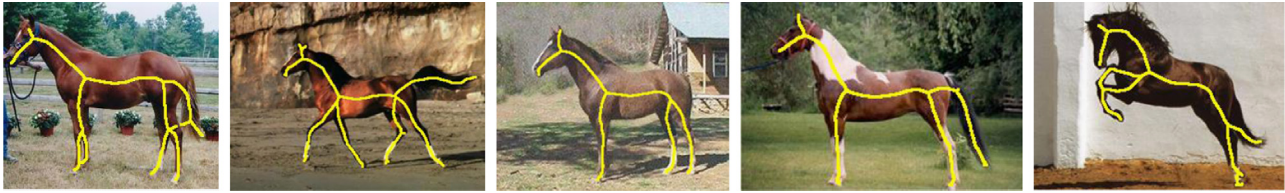
Fig. 5. Some examples in the Weizmann Horse SYMMAX dataset. The groundtruth symmetries of horses are in yellow. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

of symmetry detector, which leads to a performance improvement on symmetry detection.

The rest of the paper is organized as follows. Section 2 briefly reviews the related works about symmetry detection as well as the skeletonization approaches for binary and gray-scale images. In Section 3, our method for symmetry detection in natural images is proposed, including symmetry feature extraction, subspace partition and the multiple instance subspace learning learning strategy. The experimental results are demonstrated in Section 4 and the conclusion is drawn in Section 5.

2. Previous work

Symmetry detection is closely related to the concept named skeletonization. Skeletonization also deals with reflection symmetries, while to be emphasized, it follows the definition of the *medial axis transform* (MAT) given by Blum [9], which detects the centers of the maximal disks inside an object. Instead, the notation of symmetry in this paper is according to a analogous model named *smoothed local symmetries* (SLS) [11], which detects the centers of the lines connecting pairs of symmetry points on an object boundary. Therefore, SLS is a relax version of MAT, so it is more applicable for complex natural scenes.

Skeletonization in binary shape images has been fully studied in previous decades. A large number of skeletonization approaches [3,7,40,14,5,38,36] have been proposed to generate quite "good" skeletons. As we mentioned before, these approaches fail to be applied to natural images as the foreground segmentation techniques often cannot provide a confident object contour.

Skeletonization in gray-scale images [39,29] is much more challenging. Several researchers try to extract the skeleton from the edge strength map, which is generally obtained by applying directional derivative operators to a gray-scale image smoothed by Gaussian kernel. Morse et al. [33] compute the medial response of each image pixel measured by the edge strengths in its local region, then define the skeleton as the peak of medial responses. Lindeberg [27] define the skeleton as the points for which the intensity assumes a local maximum (minimum) in the direction of the main principal curvature, and provide a promising mechanism for automatic selection of the scale of Gaussian smooth kernel. Jang and Hong [20] extract the skeleton from the pseudo-distance map which is obtained by iteratively minimizing an object function defined on the edge strength map. The pseudo-distance map is actually a variant of Euclidean distance map, in which the distance value is small at the location with high edge strength, and vice versa. To reduce the influence of undesirable biased skeletons, Yu and Bajaj [48] extract the skeleton by tracing the ridges of the skeleton strength map calculated from the diffused vector field of the edge strength map. Due to the lack of object prior, these methods cannot handle the cluttered images with large variations and complex backgrounds.

Alternatively, others attempt to discover symmetry axes from the binary edge map, which is obtained by applying non-maximum suppression, thresholding, edge linking and other

post-processing to the edge strength map. Adluru et al. [1] group the object contours according to the local symmetry by walking along the skeleton paths of a given reference shape. The local symmetry is measured with symmetrical characteristic with respect to a pair of adjacent edge segments. Stahl and Wang [42] try to link the edge fragments with pair-wise symmetry to form a closed contour and extract the symmetry axis of the contour concurrently. They formulate such a contour grouping as a graph optimization problem. The computational cost of these methods is high due to the large number of the edge fragments in clutter images. Besides, the performance of these methods seriously depends on the quality of the edge maps.

Symmetry detection based on multi-scale segmentation seems a promising direction. Levinshtein et al. [24] propose a symmetry detection method based on multi-scale superpixel segmentation. In their method, an ellipse is fit to each symmetric part formed by grouping the adjacent superpixels with high shape and appearance similarities, then the major axis of the ellipse is treated as the symmetry axis. In their latest work, they extend the former model to allow the symmetric part to bend and taper [23].

Widynski et al. [47] propose an interesting approach to detect local symmetries in natural images, in which symmetry detection is formulated as a tracking problem and addressed by particle filtering. They use an adaptive geometric model to fit local symmetric structures which is able to detect the axes and contours of the symmetric structures jointly. However, tracing symmetries in natural images is quite difficult as symmetries are often intersected or broken by background clutters.

Tsogkas and Kokkinos's learning-based symmetry detection framework [43] serves as a confident direction and provide state-of-the-art performance on the SYMMAX300 dataset. However, to further improve the accuracy of symmetry detection, the more informative symmetry features and more discriminative detector are badly required.

Our work is somehow inspired by the recent object boundary detection work named *Sketch Token* [25], which clusters the contour data to form a mid-level representation. However, clustering the symmetry data is much more challenging. First, both the scales and the orientations of the symmetric parts are unknown and have large variations. Therefore, the feature space of symmetries is more scattered than the one of object boundaries. In addition, partitioning the samples represented by multiple instances, i.e. multiple instance clustering, is more difficult. Our method provides an effective way to cluster the symmetry samples represented by multiple instances, which offers compact subspaces to build a strong symmetry detector and achieves encouraging performances on both two datasets.

3. Methodology

In this section, we introduce our approach to detect symmetries in natural images. First, we describe the scale- and orientation-sensitive features that we used to represent symmetric structures. Then, we discuss how to train an effective

symmetry detector by the proposed multiple instance subspace learning strategy.

3.1. Feature extraction

Boundaries can be effectively detected by the features extracted from the local image patches around them [31,18,25]. However, the local image patches around symmetry points have no such discrimination. To extract informative and useful symmetry features, we need to consider the intrinsic properties of symmetries. According to the definition of symmetry, it is straightforward that it should be symmetrical to a pair of object boundaries. Therefore, to perform symmetry detection, two types of information can be leveraged: (1) symmetry detection can benefit from symmetrical boundary detection and (2) the object region represented by a symmetry axis should somewhat be self-similar. Based on these two observations, we propose two types of features for symmetry detection in this section. One is the boundary strength feature, the other is the self-similarity feature. Intuitively, symmetries have intrinsic scales and orientations, which are related to the pairs of symmetrical boundaries. Thus, we compute the proposed features at multiple scales and orientations.

To present our symmetry features clearly, we give some basic notations first. Taking Fig. 6 as the reference, let $R(P, a, \theta)$ denote the rectangle centered at the image location P with $a \times a$ size and θ orientation, $R^T(P, a, \theta)$ and $R^B(P, a, \theta)$ denote the top half and bottom half of $R(P, a, \theta)$ respectively, $H_{R(P,a,\theta)}$ denotes the histogram representation of the empirical distribution of some local cue values (e.g. brightness, color and texture) in the pixels included in the rectangle $R(P, a, \theta)$ and $D(\cdot, \cdot)$ denotes the distance measure between two histograms.

3.1.1. Boundary strength feature

To compute the boundary strength feature at the image location P, for each scale s and orientation θ, we consider a pair of locations P^T and P^B such that they have equal distance s to P and the two straight lines $\overline{P^T P}$ and $\overline{P^B P}$ are both perpendicular to orientation θ, as illustrated in in Fig. 6(a). The boundary strength feature at P is defined by the contrasts between rectangles with multiple sizes $[\frac{\sigma}{2}, \sigma, 2\sigma, 2s]$ (the heuristic scale s is also considered to ensure symmetric structures have high dissimilarity when

compared to their surroundings [43]) centered at P^T and P^B:

$$\mathbf{f_b}(P, s, \theta) = \left(f_b^a(P, s, \theta); a = \frac{\sigma}{2}, \sigma, 2\sigma, 2s \right)^T, \qquad (1)$$

where

$$f_b^a(P, s, \theta) = \min(D(H_{R^T(P^T,a,\theta)}, H_{R^B(P^T,a,\theta)}), \quad D(H_{R^T(P^B,a,\theta)}, H_{R^B(P^B,a,\theta)})). \qquad (2)$$

Instead of directly using both of the two boundary strengths at P^T and P^B, we use the simple min operator to effectively ensure both the boundary strength at P^T and P^B are strong, which guarantees the robustness of our boundary strength feature against clutter. Combining boundary information collected from the detection window of multiple sizes can improve the performance of boundary detection, since the detector of large size is robust to the false positive boundaries caused by clutter and the small one captures the detailed structures of boundaries [34,2].

3.1.2. Self-similarity feature

A symmetry axis can divide the object region into two symmetrical parts, which should be more or less similar to each other in local cues (e.g. brightness, color and texture). We refer to this as foreground self-similarity. Besides, as pointed out in [43], the two background regions on the two sides of the object boundary are usually uniform, which means the dissimilarity between these two such regions should be low. We refer to this as background self-similarity. Taking Fig. 6(b) as the reference and following the notation in Section 3.1.1, we measure the foreground self-similarity by

$$f_{s_f}(P, s, \theta) = D(H_{R^T(P,s,\theta)}, H_{R^B(P,s,\theta)}). \qquad (3)$$

Similarly, the background self-similarity is measured by

$$f_{s_b}(P, s, \theta) = D(H_{R^T(P^T,s,\theta)}, H_{R^B(P^B,s,\theta)}). \qquad (4)$$

We concatenate these two features as the self-similarity feature:

$$\mathbf{f_s}(P, s, \theta) = (f_{s_f}(P, s, \theta), f_{s_b}(P, s, \theta))^T. \qquad (5)$$

In this paper, we use the χ^2-distance function [35] to measure the difference between two histograms. The symmetry features are extracted based on three types of local cues: brightness, color and texture. Such features are computed at thirteen scales and
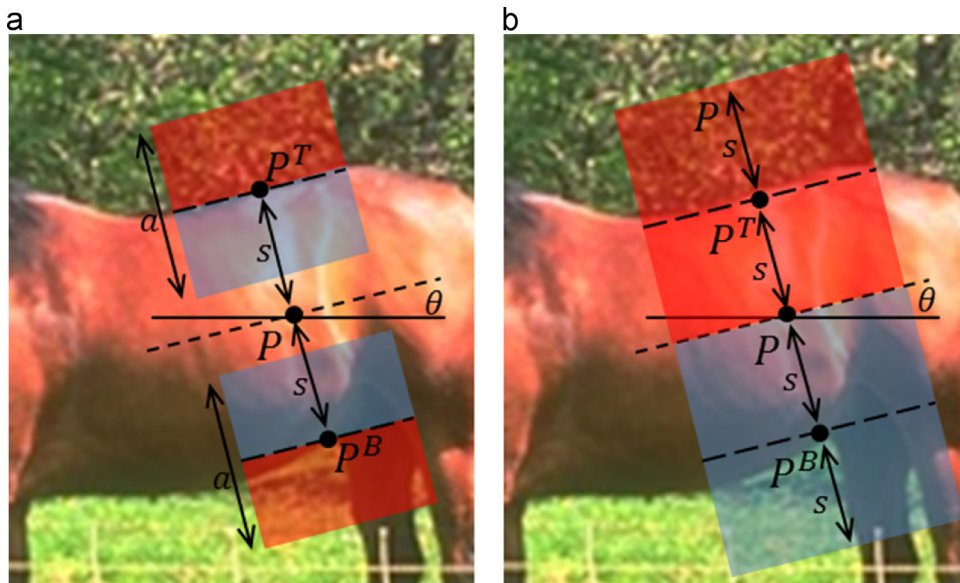

Fig. 6. The templates used to compute (a) boundary strength feature and (b) self-similarity feature. Consider a skeleton point P, given a proper scale s and a proper orientation θ, (a) both the similarities between the red and blue rectangles surrounding P^T and P^B are low, (b) the bottom half of the red rectangle shows high similarity to the top half of the blue rectangle. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)

eight orientations per scale. As all these features are rectangle-based and can be computed directly on integral images, they induce little additional computational time cost, compared to the features used in Tsogkas and Kokkinos's work [43].

3.1.3. Feature vector combinations

For each pixel, we obtain 24-dimensional feature vector at each scale and each orientation: sixteen for the boundary strength feature and eight for the self-similarity feature computed on the four channels: brightness $L*$, color $a*$, $b*$ and texture $T*$:

$$\mathbf{x}(P,s,\theta) = (\mathbf{f}_{\mathbf{b}_{ch}}^{\mathrm{T}}(P,s,\theta), \mathbf{f}_{\mathbf{s}_{ch}}^{\mathrm{T}}(P,s,\theta); ch = L*, a*, b*, T*)^{\mathrm{T}}. \qquad (6)$$

3.2. Multiple instance subspace learning

As the defined symmetry features are scale- and orientation-sensitive for each image pixel, it is nontrivial to train an effective detector to handle these parameters (i.e. scales, orientations) as latent variables during learning. Tsogkas and Kokkinos [43] adopt the MIL framework, in which each image pixel is a bag, and features computed at multiple scale and orientation combinations are the instances. In this section, we first review a simple setting of MIL briefly, then propose our learning framework.

3.2.1. Multiple Instance Learning

In the framework of MIL, samples are presented by bags, and labels are provided for the bags rather than individual instances. If a bag is labeled positive then it contains at least one positive instance, whereas if a bag is labeled negative then its instances are all negative. Formally, the training data has the form $\{(X_i, y_i)\}_{i=1}^n$, where $X_i = \{\mathbf{x}_{ij}\}_{j=1}^m$ is a bag, $\mathbf{x}_{ij} \in \mathbb{R}^d$ is an instance, and $y_i \in \{0,1\}$ is a binary bag label. The subscripts i and j are the bag index in the training dataset and the instance index in a bag, respectively. In general, the bag label can be expressed by $y_i = \max_j(y_{ij})$, where $y_{ij} \in \{0,1\}$ is the instance label, which is unknown during training. In this case, the negative log-likelihood cost function should be defined over bags but not instances, which can be written as:

$$\mathcal{L} = -\sum_i^n (y_i \log(p(y_i|X_i)) + (1-y_i)\log(1-p(y_i|X_i))), \qquad (7)$$

where $p(y_i|X_i)$ is the bag probability that should acquire label y_i. In MIL setting, the probability of a bag is expressed in terms of its instances. A *Noisy-OR* (NOR) rule is proposed in [44] to do this:

$$p(y_i|X_i) = 1 - \prod_j(1 - p(y_{ij}|\mathbf{x}_{ij})). \qquad (8)$$

By modeling the instance probability as a logistic function:

$$p(y_{ij}|\mathbf{x}_{ij}) = (1 + \exp(-\mathbf{w}^{\mathrm{T}}\mathbf{x}_{ij}))^{-1}, \qquad (9)$$

we can obtain the derivative of the negative log-likelihood (Eq. (7)):

$$\nabla\mathcal{L} = -\sum_i^n \left(\frac{y_i - p(y_i|X_i)}{p(y_i|X_i)}\sum_j^m p(y_{ij}|\mathbf{x}_{ij})\mathbf{x}_{ij}\right), \qquad (10)$$

and use gradient ascent algorithm to optimize the log-likelihood. For more details about MIL, we refer to [44,4].

3.2.2. Multiple instance subspace learning

As we described in Section 1, it is often difficult to fit a single MIL model to some confusing data such as the symmetry features in natural images. We present a novel MIL framework, which instead learns a group of models on the subsets of the training data respectively. The key step of this framework is how to partition the training data properly.

Recall that the features of each image pixel are computed at the predefined scale and orientation combinations, thus we can concatenate them into one feature vector according to a certain order of the combinations. In this case, we can rewire the training data as $\{(\mathbf{X}_i, y_i)\}_{i=1}^n$, where $\mathbf{X}_i = (\mathbf{x}_{ij}^{\mathrm{T}}; j = 1, ..., m)^{\mathrm{T}}$. Note that, the instance index j also indicates the order of the scale and orientation combination here. We define a indexer $[\cdot]_j$ operated on $\mathbf{X}_i$ to get back the features of jth instance: $[\mathbf{X}_i]_j = \mathbf{x}_{ij}$. To partition the training data, we first construct a tree structure based on the positive samples $\mathcal{S} = \{(\mathbf{X}_i, y_i) | i \in \mathcal{I}\}$, where $\mathcal{I} = \{i | y_i = 1\}_{i=1}^n$ is the set of the indexes of the positive samples, by the following algorithm:

1. Randomly propose a set of splitting parameter candidates $\phi = (j, \mathbf{b})$, where j is the instance index and $\mathbf{b}$ is a random unit direction.
2. For each ϕ, compute

$$\mu(\phi) = \frac{1}{|\mathcal{I}|}\sum_{i \in \mathcal{I}}\mathbf{b}^{\mathrm{T}}[\mathbf{X}_i]_j,$$

$$\sigma(\phi) = \sqrt{\frac{\sum_{i \in \mathcal{I}}(\mathbf{b}^{\mathrm{T}}[\mathbf{X}_i]_j - \mu(\phi))^2}{|\mathcal{I}|}}, \qquad (11)$$

and find $\phi^* = (j^*, \mathbf{b}^*)$ that maximize $\sigma(\phi)$.
3. Partition the set of samples $\mathcal{S}$ into left and right subsets by ϕ^*:

$$\mathcal{S}_l(\phi^*) = \{(\mathbf{X}_i, y_i) | i \in \mathcal{I}_l(\phi^*)\}, \mathcal{S}_r(\phi^*) = \mathcal{S}\backslash\mathcal{S}_l(\phi^*), \qquad (12)$$

where

$$\mathcal{I}_l(\phi^*) = \{i | \mathbf{b}^{*\mathrm{T}}[\mathbf{X}_i]_{j^*} < \mathrm{median}\{\mathbf{b}^{*\mathrm{T}}[\mathbf{X}_k]_{j^*} | k \in \mathcal{I}\}\}. \qquad (13)$$

4. If the depth of the tree is below a maximum $\mathbb{d}$ and all fractions of the sizes of subsets are above a minimum ρ, then recurse for left and right subsets $\mathcal{S}_l(\phi^*)$ and $\mathcal{S}_r(\phi^*)$, respectively.

As the dimension of our combined feature $\mathbf{X}_i$ is quite high $(24 \times 13 \times 8 = 2496)$, it is more proper to apply RPT to handle such high dimensional data rather than typical clustering algorithms like K-means. However, different from the standard RPT [16], we partition the data according to the randomized projection of partial dimensions of features. For this reason, we named ours as partial random projection tree (PRPT). The partial dimensions selected in our algorithm just correspond to an instance computed at a certain scale and a certain orientation. Recall that each positive sample only corresponds to one true scale and orientation combination. Therefore by looking up such partial dimensions, we can fast coarsely divide positive samples into subsets according to their hypothetical scale and orientation combination. At each splitting node of the tree, we choose a unit direction from the parameter pool generated randomly along which the variance of projections is maximum (Eq. (11)). Such a direction is actually approaching the direction of the principle component of the data. Along this direction, we split the samples into two equal-sized sets (Eq. (13)). This splitting strategy is actually inspired by PCA-tree [19]. Such tree based splitting methods aim to separate dissimilar data recursively into different nodes. In PCA-tree, the principal eigenvector of the covariance of the data is selected to project the data and split the data. The direction of the principal eigenvector is along the major axis of variation. In our case (multiple instances setting), we cannot directly apply PCA-tree, while similarly we search the direction leading to the projections which has the maximum variance, as we have the assumption that the features of positive instances and negative instances are quite different. To avoid the overfitting, we restrict the tree depth and the number of samples in leaf nodes.

Once a tree model is learned by PRPT, we partition the negative samples guided by such a tree. For each negative sample $\mathbf{X}_i$, it
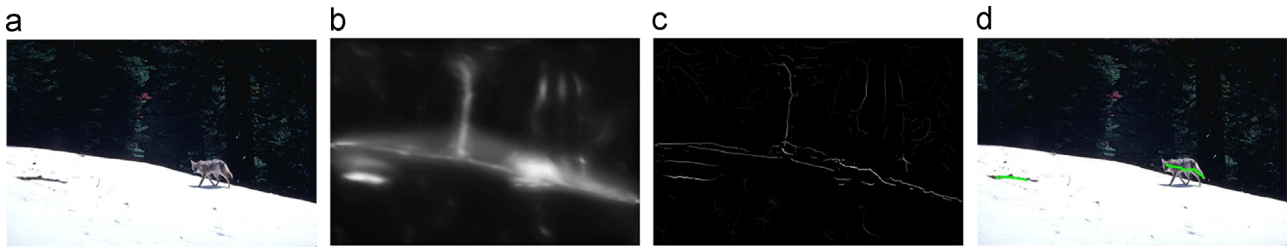
Fig. 7. A typical pipeline of our symmetry detection. (a) The input image. (b) The detector response. (c) The non-maximum suppression. (d) The thresholded binary symmetry map (in green). (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)
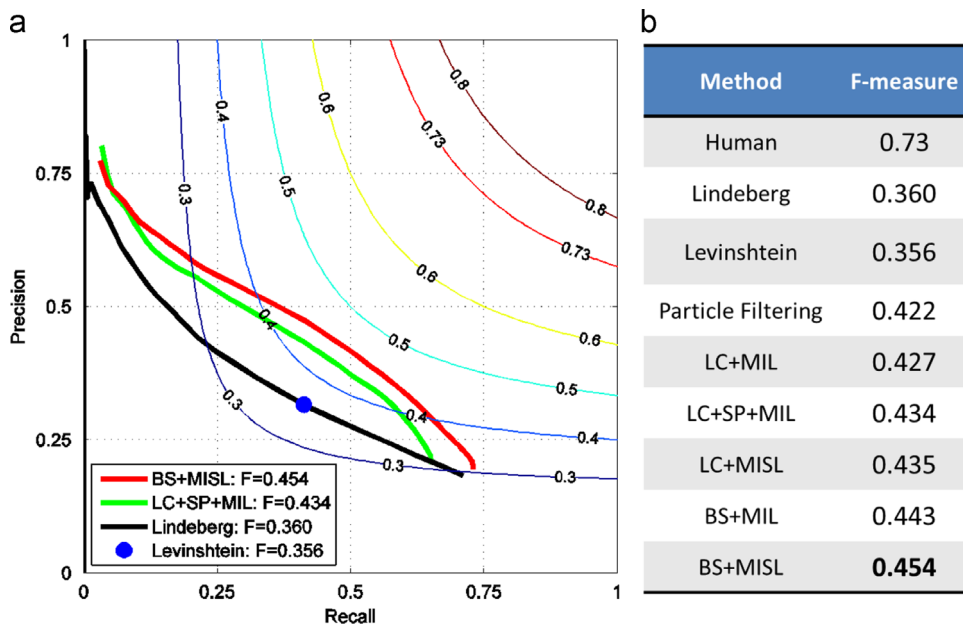

Method	F-measure
Human	0.73
Lindeberg	0.360
Levinshtein	0.356
Particle Filtering	0.422
LC+MIL	0.427
LC+SP+MIL	0.434
LC+MISL	0.435
BS+MIL	0.443
BS+MISL	**0.454**

Fig. 8. Evaluation of symmetry detectors on the SYMMAX300 dataset [43]. (a) The precision–recall curves. (b) Leading symmetry detection approaches are ranked according to their F-measure with respect to human groudtruth. Our approach performs favorably better than others (Levinshtein [24], Linderberg [27], Particle Filtering [47], LC+MIL [43] and LC+SP+MIL [43]). For clear visualization, we do not plot the precision–recall curves of all the approaches.

starts at the root, then recursively branches left or right according to the splitting rule (Eqs. (12) and (13)) learned during the above tree construction procedure. Finally, it reaches a leaf node L of whom index is denoted by $\mathbb{l}(\mathbf{X}_i)$. Assume that there are K leaf nodes in the tree, for each leaf node $L_k (k = 1, \ldots, K)$, we rewrite the samples in it as the form: $\{(X_i, y_i) | \mathbb{l}(\mathbf{X}_i) = k\}_{i=1}^{n}$. The samples in each leaf node L_K observe a local subspace. A MIL model is train on each subspace by the method described in Section 3.2.1 to deliver the model parameter $\mathbf{w}_k$ and estimate the instance probability $p(y_{ij} | \mathbf{x}_{ij})$.

In the testing procedure, for a testing sample $\mathbf{X}_t$, it also starts at the root of the learned tree, then recursively branches left or right according to the splitting rule (Eqs. (12) and (13)), and finally falls into a leaf node $L_{\mathbb{l}(\mathbf{X}_t)}$. The probability $p(y_t | \mathbf{X}_t)$ is given by

$$p(y_t | \mathbf{X}_t) = 1 - \prod_j (1 - p(y_{tj} | [\mathbf{X}_t]_j)), \tag{14}$$

where

$$p(y_{tj} | [\mathbf{X}_i]_j) = (1 + \exp(-\mathbf{w}_{\mathbb{l}(\mathbf{X}_t)}^{\mathrm{T}} [\mathbf{X}_t]_j))^{-1}. \tag{15}$$

We refer to this method as MISL, for multiple instance learning in subspaces. Although the multiple classifiers are trained separately in our method, as the output of our method is a probabilistic value, it is unnecessary to perform calibration by fitting a probability distribution as did in [30]. Once the probability of symmetry has been computed at each pixel, a standard non-maximum suppression scheme [13] is used to find the peak response of

symmetry. A typical pipeline of our symmetry detection is illustrated in Fig. 7.

4. Experimental results

In this section, we show the experimental results and give the comparisons between alternative approaches on both the SYMMAX300 dataset [43] and the WH-SYMMAX dataset. In the remainder of this section, the parameters used in our approach are set as: the number of training samples $n = 100{,}000$, the size of the rectangle used for computing the boundary strength feature $\sigma = 8$, the maximum tree depth $\mathbb{d} = 5$ and the minimum fraction size of the samples in one leaf node $\rho = 0.1$.

4.1. SYMMAX300 dataset

The SYMMAX300 dataset contains 200 training and 100 testing images with equal resolution of 481×321. As the symmetry features proposed in [43] is based on the contrasts between *l*ocal *c*ues such as brightness, color and texture, we refer to them as LC. Similarly, their *sp*ectral feature is denoted by SP. Our symmetry features combine the *b*oundary strength and the *s*elf-similarity, so we denote them by BS in the rest of the paper. In Fig. 8(a), we compare our symmetry detection method against other competing methods. We follow the evaluation protocol used in [43], under which the performances of symmetry detection approaches are

measured by their maximum *F*-measure ($\frac{2 \cdot Precision \cdot Recall}{Precision + Recall}$) as well as precision–recall curves with respect to human ground-truth symmetries. To obtain the precision–recall curves, the detected symmetry response is first thresholded into a binary map, which is then matched with each ground-truth map separately (for each image, the groundthruth data consist of 5–7 different groundtruth maps constructed by different people). The matching allows small localization errors between detected positives and groundtruths. If a detected positive is matched with at least one of the ground-truth maps, it is classified as true positive. In contrast, pixels that correspond to no ground-truth map are false positives. By assigning different thresholds to the detected symmetry response, we can obtain a sequence of precision and recall pair, which is used to plot the precision–recall curve. Perfect precision is achieved when every detected positive has correspondences to some groudtruth symmetry pixels, while perfect recall is achieved when every groudtruth symmetry pixel is matched with some detected positives. These two rates are contradictory, so *F*-measure is used as a balanced value between these two for performance validation.

The *F*-measure of the human-generated symmetry maps is just 0.73, indicating the challenge of this task. Our symmetry detector compares favorably with other leading techniques and achieves state-of-the-art results. Additionally, our symmetry detector shows both improved recall and precision at most of the precision–recall regime. Fig. 8(b) breaks down the contributions of BS and MISL to the performance of our symmetry detector. Looking at the comparisons: BS+MIL vs LC+MIL, BS+MISL vs LC+MISL, LC+MISL vs LC+MIL, BS+MISL vs BS+MIL, we observe that both BS and MISL lead to considerable performance improvements on the SYMMAX300 dataset. Especially worthy to mention that LC+MISL only produces a slightly better *F*-measure than LC+SP+MIL, however such a result is achieved without the help of the spectral feature. Spectral feature extraction is quite time and memory consuming. Besides, spectral feature is computed on the

skeleton map obtained by LC+MIL, which means to obtain the result of LC+SP+MIL, detection process has to be performed twice. This is the reason why we do not include spectral feature in our method. Due to the stochastic nature of our method, the above results are obtained by averaging the results by performing our method on the dataset five times. The standard error of mean (SEM) of BS+MISL is 1.55×10^{-4}, which is small and shows the stability of our method. Due to the randomness introduced in the training phase in our method, the SEM of BS+MISL is larger than the SEM of LC+MIL (4.46×10^{-6}), while this is acceptable. Qualitative comparisons are illustrated in Fig. 9, in which we observe that BS reduces clutter and MISL further suppresses the false responses on the clutter background qualitatively.

To demonstrate the advantage of PRPT in subspace learning, we also learn the MISL model on the subsets partitioned by *K*-means and standard RPT. In addition, to show the importance of forming the subsets only from the positive samples, we also learn MISL on the subsets obtained by applying PRPT directly to the whole training set (including positive samples and negative samples). We refer to this partition strategy as PRPTW. The performances of symmetry detectors learned by MISL via different partition strategies are listed in Table 1. We use the same parameter setting for both RPT and PRPT, and we vary the number of clusters for *K*-means to obtain the best result. Note that, the partitions obtained by RPT and PRPTW lead to just marginal improvement, and the one obtained by *K*-means even results in performance reduction.

Table 1

The comparison between different partition strategies. The Baseline is the result obtained without data partition.

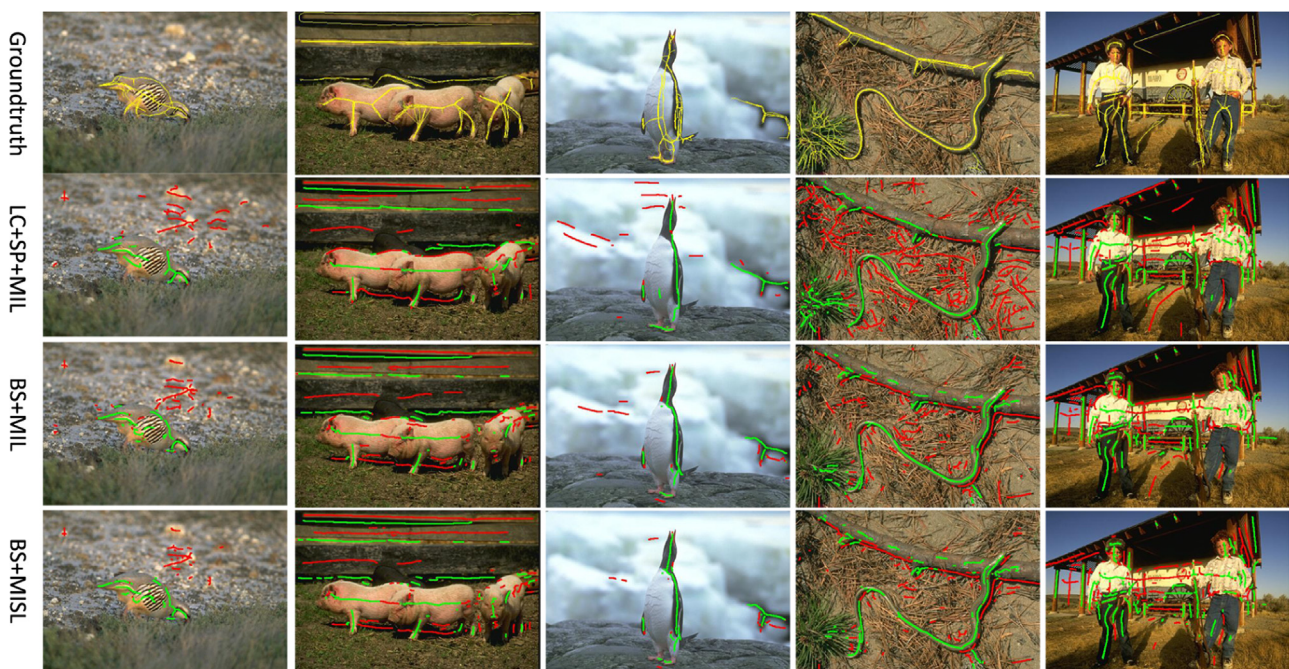| Method	Baseline	*K*-means	RPT	PRPTW	PRPT
F-measure	0.443	0.441	0.445	0.444	**0.454**

Fig. 9. Examples of symmetry detection on the SYYMAX300 [43] dataset. The groudtruths are marked in yellow in the first row. The results obtained by LC+SP+MIL [43], BS+MIL and BS+MISL are illustrated in the second, third and fourth rows respectively. All binary symmetry maps are obtained by thresholding the detector response at *Recall*=0.5. The true and false positives are marked in green and red respectively. Note when compared with LC+SP+MIL, our symmetry feature (BS+MIL) reduces false positives in all these examples. The proposed new MIL method (BS+MISL) further decreases the errors induced from the clutter, such as the textural background in the first and firth columns. (For interpretation of the references to color in this figure caption, the reader is referred to the web version of this paper.)
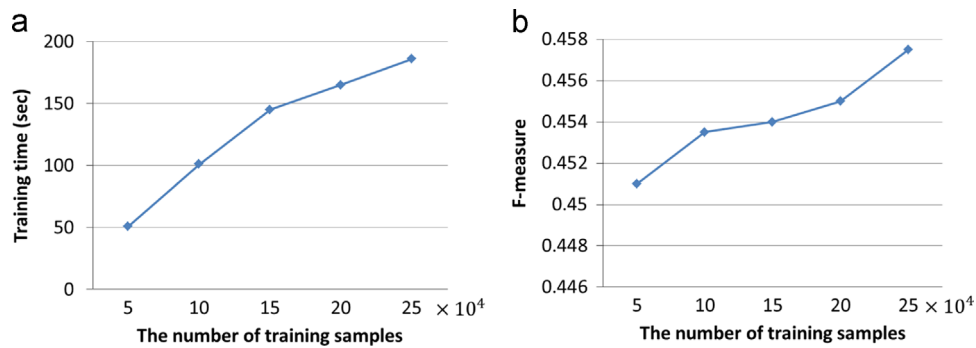
Fig. 10. The results of BS+MISL by varying the number of training samples. (a) Training time. (b) F-measure.
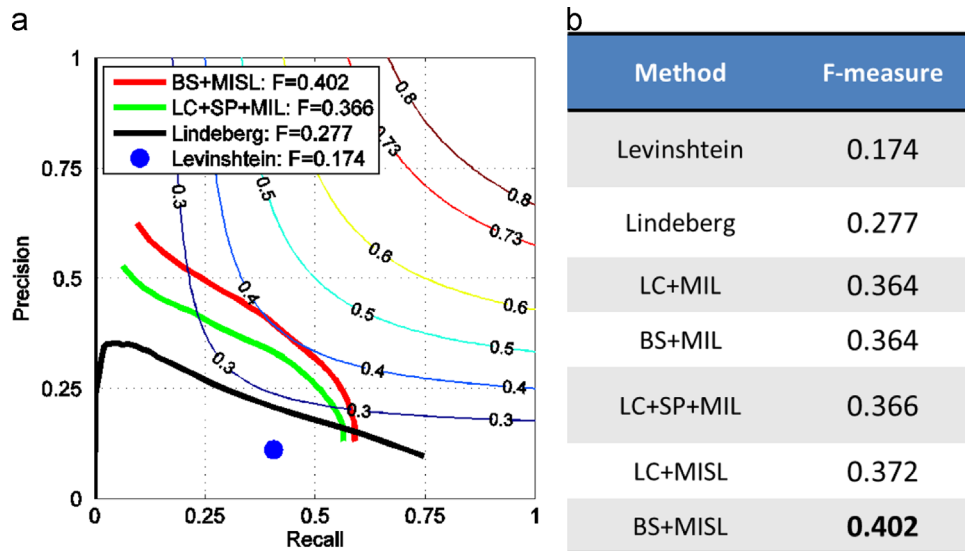

Fig. 11. Evaluation of symmetry detectors on the WH-SYMMAX dataset. (a) The precision–recall curves. (b) Leading symmetry detection approaches are ranked according to their F-measure with respect to groudtruth. Our approach (BS+MISL) performs favorably better than others (Levinshtein [24], Linderberg [27], LC+MIL [43] and LC+SP+MIL [43]). For clear visualization, we do not plot the precision–recall curves of all the approaches.
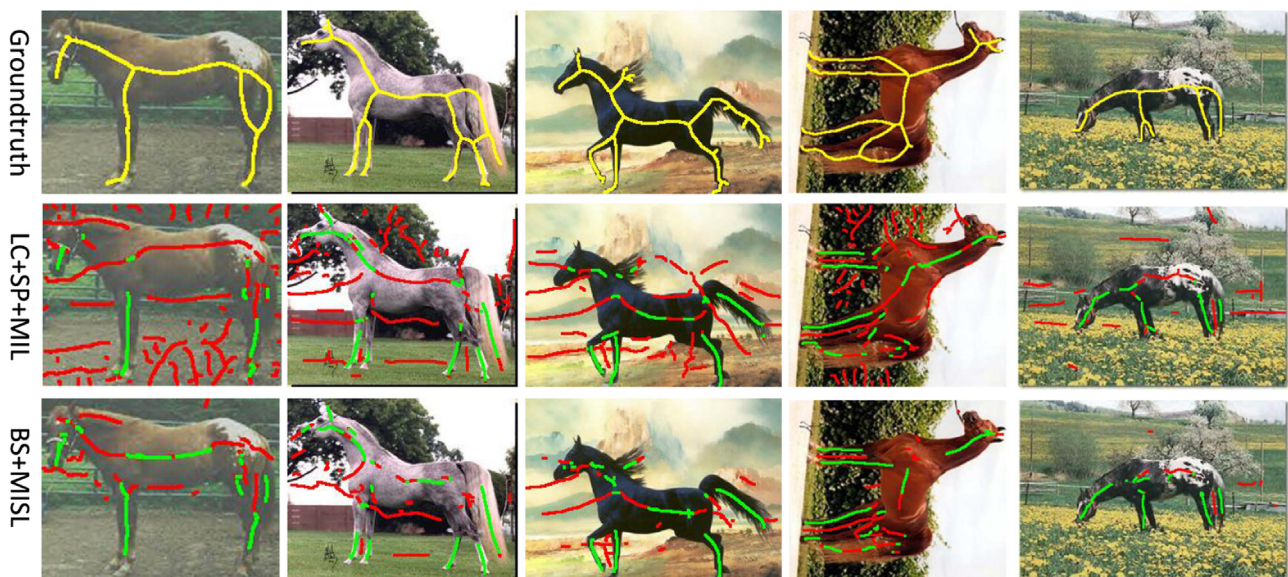

Fig. 12. Examples of symmetry detection on the WH-SYYMAX dataset. The groudtruths are marked in yellow in the first row. The results obtained by LC+SP+MIL [43] and BS+MISL are shown in the second and third rows respectively. All binary symmetry maps are obtained by thresholding the detector response at Recall=0.5. The true and false positives are marked in green and red respectively. Note when compared with LC+SP+MIL, our symmetry detector (BS+MIL) significantly suppresses false positives in all these examples.

Finally, we assess the performance change of our method (BS+MISL) by varying the number of training samples. Fig. 10(a) illustrates the relationship between the training time of our method and the number of training samples. Note that when the number of training samples becomes larger, the increase rate of training time is reduced. This may be caused by the maximum tree depth restriction used in building tree. From Fig. 10(b), we observe that the performance obtains a slight improvement, when using more training samples.

4.2. WH-SYMMAX dataset

The WH-SYMMAX dataset totally contains 328 images, among which the first 228 are used for training and the rest are used for testing. The comparison between our symmetry detection method and others is illustrated in Fig. 11, which evidences that our symmetry detector is better than competitors, as ours show both improved recall and precision at most of the precision–recall regime. Although the proposed symmetry feature BS does not lead to any performance improvement on this dataset under the standard MIL framework, the performance is significantly boosted under the proposed MISL framework, with an improvement of 0.036 in *F*-measure. This phenomenon demonstrates that MISL is able to explore the useful information embedded in the multi-scale boundary strength feature. Qualitative comparisons are illustrated in Fig. 12. It could be observed that our method significantly suppresses the false positives on the background.

5. Conclusion

We have proposed a novel multiple instance subspace learning approach for local reflection symmetry detection in nature images. The proposed MISL approach is capable of accommodating the unknown scale and orientation of the symmetry features effectively and efficiently. The encouraging experimental results on both the SYMMAX300 dataset [43] and the newly constructed dataset validate the advantages of our approach, compared with the current state-of-the-arts.

The proposed MISL learning framework is not limited to symmetry detection, but is general to some detection tasks in computer vision. Our future work is to extend this learning framework to other tasks, for example, applying it to detect objects under multiple scales and orientations.

Conflict of interest

None declared.

Acknowledgement

References

[1] N. Adluru, L.J. Latecki, R. Lakämper, T. Young, X. Bai, A.D. Gross, Contour grouping based on local symmetry, in: Proceedings of the ICCV, 2007, pp. 1–8.

[2] P. Arbelaez, M. Maire, C. Fowlkes, J. Malik, Contour detection and hierarchical image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 33 (2011) 898–916.

[3] C. Arcelli, G.S. di Baja, A one-pass two-operation process to detect the skeletal pixels on the 4-distance transform, IEEE Trans. Pattern Anal. Mach. Intell. 11 (1989) 411–414.

[4] B. Babenko, M.H. Yang, S. Belongie, Robust object tracking with online multiple instance learning, IEEE Trans. Pattern Anal. Mach. Intell. 33 (2011).

[5] X. Bai, L.J. Latecki, W. Liu, Skeleton pruning by contour partitioning with discrete curve evolution, IEEE Trans. Pattern Anal. Mach. Intell. 29 (2007) 449–462.

[6] X. Bai, X. Wang, L.J. Latecki, W. Liu, Z. Tu, Active skeleton for non-rigid object detection, in: Proceedings of the ICCV, 2009, pp. 575–582.

[7] G.S. di Baja, E. Thiel, Skeletonization algorithm running on path-based distance maps, Image Vis. Comput. 14 (1996) 47–57.

[8] J.L. Bentley, Multidimensional divide-and-conquer, Commun. ACM 23 (1980) 214–229.

[9] H. Blum, Models for the Perception of Speech and Visual Form, MIT Press, Boston, MA, USA, 1967, pp. 363–380.

[10] E. Borenstein, S. Ullman, Class-specific, top-down segmentation, in: Proceedings of the ECCV, 2002, p. 109–124.

[11] M. Brady, H. Asada, Smoothed local symmetries and their implementation, Int. J. Robot. Res. 3 (1984).

[12] J. Bruck, J. Gao, A. Jiang, Map: medial axis based geometric routing in sensor networks, Wirel. Netw. 13 (2007) 835–853.

[13] J. Canny, A computational approach to edge detection, IEEE Trans Pattern Anal. Mach. Intell. 8 (1986) 679–698.

[14] W.P. Choi, K.M. Lam, W.C. Siu, Extraction of the euclidean skeleton based on a connectivity criterion, Pattern Recognit. 36 (2003) 721–729.

[15] L.S. D. Zhang, F. Wang, T. Li, Maximum margin multiple instance clustering, in: Proceedings of the IJCAI, 2009.

[16] S. Dasgupta, Y. Freund, Random projection trees and low dimensional manifolds, in: Proceedings of the STOC, 2008, pp. 537–546.

[17] M.F. Demirci, A. Shokoufandeh, Y. Keselman, L. Bretzner, S.J. Dickinson, Object recognition as many-to-many feature matching, Int. J. Comput. Vis. 69 (2006) 203–222.

[18] P. Dollár, Z. Tu, S. Belongie, Supervised learning of edges and object boundaries, in: Proceedings of the CVPR, 2006.

[19] Y. Freund, S. Dasgupta, M. Kabra, N. Verma, Learning the structure of manifolds using random projections, in: Proceedings of the NIPS, 2007.

[20] J.H. Jang, K.S. Hong, A pseudo-distance map for the segmentation-free skeletonization of gray-scale images, in: Proceedings of the ICCV, 2001, pp. 18–25.

[21] I. Kokkinos, P. Maragos, A.L. Yuille, Bottom-up and top-down object detection using primal sketch features and graphical models, in: Proceedings of the CVPR, 2006, pp. 1893–1900.

[22] S. Lee, Y. Liu, Curved glide-reflection symmetry detection, IEEE Trans. Pattern Anal. Mach. Intell. 34 (2012) 266–278.

[23] T.S.H. Lee, S. Fidler, S.J. Dickinson, Detecting curved symmetric parts using a deformable disc model, in: Proceedings of the ICCV, 2013, pp. 1753–1760.

[24] A. Levinshtein, S.J. Dickinson, C. Sminchisescu, Multiscale symmetric part detection and grouping, in: Proceedings of the ICCV, 2009, pp. 2162–2169.

[25] J.J. Lim, C.L. Zitnick, P. Dollár, Sketch tokens: a learned mid-level representation for contour and object detection, in: Proceedings of the CVPR, 2013.

[26] T. Lindeberg, Detecting salient blob-like image structures and their scales with a scale-space primal sketch: a method for focus-of-attention, Int. J. Comput. Vis. 11 (1993) 283–318.

[27] T. Lindeberg, Edge detection and ridge detection with automatic scale selection, Int. J. Comput. Vis. 30 (1998) 117–156.

[28] Y. Liu, Computational Symmetry in Computer Vision and Computer Graphics, Now publishers Inc., Hanover, MA, USA, 2009.

[29] A.M. López, F. Lumbreras, J. Serrat, J.J. Villanueva, Evaluation of methods for ridge and valley detection, IEEE Trans. Pattern Anal. Mach. Intell. 21 (1999) 327–335.

[30] T. Malisiewicz, A. Gupta, A.A. Efros, Ensemble of exemplar-svms for object detection and beyond, in: Proceedings of the ICCV, 2011.

[31] D.R. Martin, C. Fowlkes, J. Malik, Learning to detect natural image boundaries using local brightness color and texture cues, IEEE Trans. Pattern Anal. Mach. Intell. 26 (2004) 530–549.

[32] D.R. Martin, C. Fowlkes, D. Tal, J. Malik, A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics, in: Proceedings of the ICCV, 2001, pp. 416–425.

[33] B.S. Morse, S.M. Pizer, A. Liu, Muitiscale medial analysis of medical images, Image Vision Comput. 12 (1994) 327–338.

[34] X. Ren, Multi-scale improves boundary detection in natural images, in: Proceedings of the ECCV, 2008, pp. 533–545.

[35] Y. Rubner, J. Puzicha, C. Tomasi, J.M. Buhmann, Empirical evaluation of dissimilarity measures for color and texture, Comput. Vis. Image Understand. 84 (2001) 25–43.

[36] P.K. Saha, G. Borgefors, G.S. di Baja, A survey on skeletonization algorithms and their applications, Pattern Recognit. Lett. (2015) http://dx.doi.org/10.1016/j.patrec.2015.04.006.

[37] T.B. Sebastian, P.N. Klein, B.B. Kimia, Recognition of shapes by editing their shock graphs, IEEE Trans. Pattern Anal. Mach. Intell. 26 (2004) 550–571.

[38] W. Shen, X. Bai, R. Hu, H. Wang, L.J. Latecki, Skeleton growing and pruning with bending potential ratio, Pattern Recognit. 44 (2011) 196–209.

[39] K. Siddiqi, S. Pizer, Medial Representations, Springer, New York, NY, USA, 2009.

[40] K. Siddiqi, A. Shokoufandeh, S.J. Dickinson, S.W. Zucker, Shock graphs and shape matching, Int. J. Comput. Vis. 35 (1999) 13–32.
[41] E. Sorantin, C. Halmai, B. Erdöhelyi, K. Palágyi, B. Geiger, G. Friedrich, K. Kiesel, S. Loncaric, Spiral ct based assessment of laryngotrachealstenoses with 3d image processing using a skeletonisation algorithm, IEEE Trans. Med. Imaging 21 (2002) 263–273.
[42] J.S. Stahl, S. Wang, Globally optimal grouping for symmetric closed boundaries by combining boundary and region information, IEEE Trans. Pattern Anal. Mach. Intell. 30 (2008) 395–411.
[43] S. Tsogkas, I. Kokkinos, Learning-based symmetry detection in natural images, in: Proceedings of the ECCV, 2012, pp. 41–54.
[44] P.A. Viola, J.C. Platt, C. Zhang, Multiple instance boosting for object detection, in: Proceedings of the NIPS, 2005.
[45] D.K. Wagg, M.S. Nixon, On automated model-based extraction and analysis of gait, in: Proceedings of the FG, 2004, pp. 11–16.
[46] H. Weyl, Symmetry, Princeton University Press, Princeton, NJ, USA, 1952.
[47] N. Widynski, A. Moevus, M. Mignotte, Local symmetry detection in natural images using a particle filtering approach, IEEE Trans. Image Process. 23 (2014) 5309–5322.
[48] Z. Yu, C.L. Bajaj, A segmentation-free approach for skeletonization of gray-scale images via anisotropic vector diffusion, in: Proceedings of the CVPR, 2004, pp. 415–420.
[49] M.L. Zhang, Z.H. Zhou, Multi-instance clustering with applications to multi-instance prediction, Appl. Intell. 31 (2009) 47–68.
[50] P. Zhao, L. Yang, H. Zhang, L. Quan, Per-pixel translational symmetry detection, optimization, and segmentation, in: Proceedings of the CVPR, 2012, pp. 526–533.
[51] J.Y. Zhu, J. Wu, Y. Wei, E.I.C. Chang, Z. Tu, Unsupervised object class discovery via saliency-guided multiple class learning, in: Proceedings of the CVPR, 2012, pp. 3218–3225.
[52] S.C. Zhu, A.L. Yuille, Forms: a flexible object recognition and modelling system, Int. J. Comput. Vis. 20 (1996) 187–212.

Wei Shen received his B.S. and Ph.D. degrees both in Electronics and Information Engineering from the Huazhong University of Science and Technology, Wuhan, China, in 2007 and in 2012, respectively. From April 2011 to November 2011, he worked in Microsoft Research Asia as an intern. Now he is currently an assistant Professor with the School of Communication and Information Engineering, Shanghai University, Shanghai, China. His research interests include shape analysis, edge and symmetry detection and human pose estimation.

Xiang Bai received the B.S., M.S., and Ph.D. degrees from the Huazhong University of Science and Technology (HUST), Wuhan, China, in 2003, 2005, and 2009, respectively, all in Electronics and Information Engineering. He is currently a Professor with the Department of Electronics and Information Engineering at HUST. He is also the Vice Director of the National Center of Anti-Counterfeiting Technology at HUST. His research interests include object recognition, shape analysis, scene text recognition, and intelligent systems.

Zihao Hu is currently an undergraduate student with the Department of Electronics and Information Engineering at Huazhong University of Science and Technology, Wuhan, China.

Zhijiang Zhang received his Ph.D. degree from Harbin Institute of Technology, Harbin, China, in 1999. He is currently a Professor with the School of Communication and Information Engineering, Shanghai University, Shanghai, China. His research interests include 3D reconstruction and digital holography.