

Detection and Segmentation of 2D Curved Reflection Symmetric Structures

Ching L. Teo

cteo@cs.umd.edu

Cornelia Fermüller

fer@umiacs.umd.edu

Yiannis Aloimonos

yiannis@cs.umd.edu

Computer Vision Lab, University of Maryland, College Park, MD 20742, USA

Abstract

Symmetry, as one of the key components of Gestalt theory, provides an important mid-level cue that serves as input to higher visual processes such as segmentation. In this work, we propose a complete approach that links the detection of curved reflection symmetries to produce symmetry-constrained segments of structures/regions in real images with clutter. For curved reflection symmetry detection, we leverage on patch-based symmetric features to train a Structured Random Forest classifier that detects multiscaled curved symmetries in 2D images. Next, using these curved symmetries, we modulate a novel symmetry-constrained foreground-background segmentation by their symmetry scores so that we enforce global symmetrical consistency in the final segmentation. This is achieved by imposing a pairwise symmetry prior that encourages symmetric pixels to have the same labels over a MRF-based representation of the input image edges, and the final segmentation is obtained via graph-cuts. Experimental results over four publicly available datasets containing annotated symmetric structures: 1) SYMMAX-300 [38], 2) BSD-Parts, 3) Weizmann Horse (both from [18]) and 4) NY-roads [35] demonstrate the approach's applicability to different environments with state-of-the-art performance.

1. Introduction

The world that we see abounds in several different kinds of regularities and invariants that computer vision have long attempted to capture. These invariants include: geometry [29], shape [5], scale [24] and motion [12]. The goal of capturing such invariants from low-level visual signals is to ultimately simplify the representations that feed into higher-level visual tasks: e.g. segmentation, recognition, tracking etc. In this work, we focus on extracting *symmetry*, a ubiquitous and very common shape-based regularity, from real images containing clutter (Fig. 1 (left)) and use the detected symmetries to extract *symmetry-constrained* segments/regions that support the symmetry axes (Fig. 1

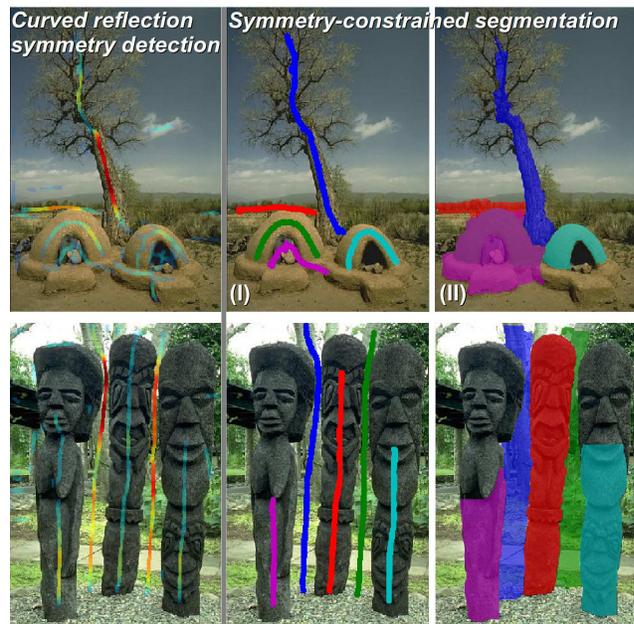


Figure 1. (Left) Detected curved reflection symmetries with red denoting stronger symmetries. (Right) Selected symmetry axes (I) and their corresponding symmetry-constrained segmentations (II).

(right)). Such a view is shared by Gestalt psychology, which considers symmetry as one of the key grouping or *mid-level* cues for explaining human visual perception. From a biological viewpoint, it is well-known that humans are extremely sensitive to reflection symmetry [39], and its detection facilitates early visual processes such as figure-ground segmentation [7]. Specifically, we detect *curved* reflection symmetries [17]. The concept is known in the literature as ridges, ribbons or centerlines [21, 35], and is related to the classical Medial Axis Transform of Blum [2]. Such approximate symmetries are much more prevalent in nature than *rigid* reflection symmetry, which many previous works [31, 25, 28] have addressed on simpler images with less clutter. Detecting and segmenting such structures is also important in the related domain of biomedical imaging, where most works have focused on the detection and segmentation of tubular structures [16, 9, 35], such as blood vessels, ax-

ons and dendrites. Detecting curved symmetries under real conditions, however, remains extremely challenging due to ambiguities of scale, noise and occlusions that limit the applicability of parametric or model-based techniques. These challenges extend to the reliable segmentation of symmetric structures as well. A few studies have previously addressed the problem of symmetry-guided segmentation [32, 36, 8], and they are mostly limited to segmenting regions exhibiting (rigid) reflection symmetries.

In contrast to previous works where either symmetry detection or segmentation are addressed separately, we describe in this paper a complete approach that tackles these two related problems in a holistic framework that consists of two main steps: 1) robust curved symmetry detection and 2) curved symmetry-constrained segmentation of such regions. For symmetry detection, we use a set of multiscale features to train a Structured Random Forest (SRF) [14] that enforces symmetry in the final predicted structure. The key advantage is that we let the SRF determine from training exemplars the optimum feature combination that predicts the best symmetry axes (location, orientation and scale) without a need to predefine a symmetry or noise model, enabling our approach to work in different environments and conditions.

In the second step, we enforce global symmetrical consistency through a foreground-background segmentation of structures/regions supporting these predicted curved symmetries via the addition of a novel pairwise symmetry prior in a Markov Random Field (MRF) representation of the input image edges. Since the symmetry prior is defined locally in the MRF clique, the optimal segmentation can be solved using graph-cuts [3], while handling even convoluted symmetry axes with multiple branches. As the predicted curved symmetries provide an initial measure of how symmetric the region should be, we modulate this prior so that the appropriate amount of symmetry is enforced in the final segmentation, a crucial requirement for natural images that mostly exhibit approximate symmetries. Additionally, because inference using SRFs and graph-cuts optimization are extremely efficient, our approach is fast: 0.1s for detection (after feature extraction which takes ≈ 1 min) and 1s for segmentation per 320×240 image. Finally, we achieve state-of-the-art accuracies in both steps over four public datasets of real images containing annotated symmetric structures.

2. Related Works

We briefly review several works in two specific areas of computer vision: 1) symmetry detection and 2) segmentation of symmetrical regions that closely relate to the proposed approach.

Symmetry detection. The detection of various types of symmetry (bilateral/reflection, rotational and translation) from 2D images has a long history in computer vision. See [23] for an up-to-date survey of past and current tech-

niques. The classic voting approach of “Generalized Symmetry Transform” of Reifeld et al. [31] is now largely surpassed by the feature-based method of Loy and Eklundh [25] that used symmetrical SIFT keypoint descriptors [24] for more robust detection of bilateral and rotational symmetries. A recent extension by Lee and Liu [17] matched these descriptors within a 3D axis parameter space to detect curved reflection symmetries from keypoints. Instead of using keypoints, Zhu [42] proposed a jump-diffusion process for detecting medial axes and junctions over a MRF representation of 2D shapes. More recently, Tsogkas and Kokkinos [38] used Multiple Instance Learning to train a curved symmetry detector, that combines multiscale patch-based feature histograms of intensity, color, texture and spectral cues to obtain state-of-the-art detection performance on a large dataset of real images with clutter. In the domain of biomedical imaging, most works had focused on detecting centerlines of 3D tubular/cylindrical structures: blood vessels, axons, dendrites and spinal columns [16, 9, 41], and in cartography symmetry was used for detecting road networks [11]. Although these works produce very good centerline predictions, their applicability is often limited to the specific imaging modality (e.g. CT, MRI or brightfield) and the expected size (scale) of the target tubular structures. Recently, Sironi et al. [35] proposed a novel regression-based technique using Regression-Trees (as opposed to classification) that showed state-of-the-art centerline detections in different applications (medical and roads). Their method, however, requires training a large number of regressors to predict the expected scale and location of the tubular structures from the input.

Segmenting symmetrical regions. Most previous approaches [15, 20] considered segmentation [27, 30] as a separate, independent step from symmetry detection. Riklin-Raviv et al. [32] embed symmetry cues dynamically into a level-set functional so that each evolution of the functional improves the symmetric properties of the current segmentation. Sun and Bhanu [36] used a region merging approach where homogeneous regions, measured in terms of color and texture, are merged while preserving reflection symmetry. The merging process, however, is sensitive to large variations of color and textures, producing oversegmented (small) regions in these areas. Along similar lines, Levinstein et al. [19] built an adjacency graph that encodes how superpixels are grouped into symmetrical parts. Lee et al. [18] extended this approach by imposing a more general deformable disc (ellipse) model that encodes affinity of superpixels in curvilinear structures better. Affinity is computed from shape similarity and differences in local color and intensity. Since superpixels are grouped in a pairwise manner via dynamic programming, this approach is limited to grouping homogeneous regions that contain a single curved symmetry (no branches). Fu et al. [8] focused on extract-

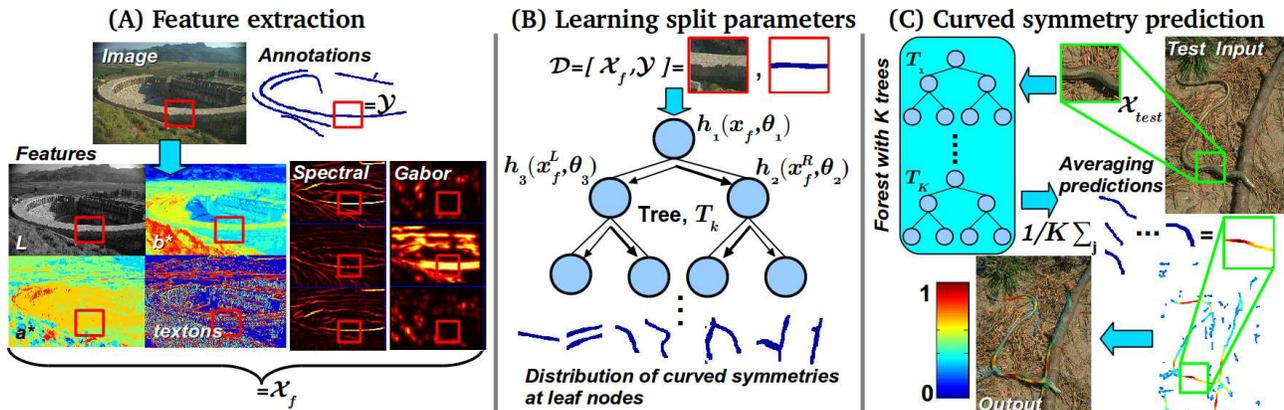


Figure 2. Training a SRF for curved symmetry detection. (A) Multiscale intensity, color, texture, spectral and oriented Gabors are used to compute a set of local symmetry responses, \mathcal{X}_f , by comparing histograms of patches. (B) By pairing patch-based features $x_f \in \mathcal{X}_f$ with their symmetry groundtruth annotations \mathcal{Y} , we determine the optimal split parameters θ associated with the split functions $h(x_f, \theta)$ that send features x_f either to the left or right child. The leaf nodes store a distribution of structured labels of symmetry axes. (C) During inference, a test patch is assigned to a leaf node within a tree that contains a prediction of the location and scale of the symmetry axis. Averaging the prediction over all K trees yields the final symmetry axes and their corresponding strengths (degree of symmetry).

ing foreground salient objects exhibiting reflective symmetry by first computing a symmetry foreground segmentation map from color-contrast cues and feature-based symmetry-induced homography, which are then set into unary and pairwise terms in a MRF-based segmentation. In the medical imaging literature, most works consider segmentation as an integral part of centerline detection, where the centerlines and their corresponding radii are solved by the same detector [16, 9, 35]. The final segmentations are therefore a combination of circles or balls located along the centerlines, with no enforcement of global curved symmetry consistency.

Our approach, described next, considers these two issues together so that we produce accurate segmentations of approximate symmetrical structures from robust curved symmetry detections in clutter. Unlike works that are limited to detecting and segmenting structures with bilateral (reflection) symmetries [20, 15, 32, 36, 8], our approach detects the more general class of curved reflective symmetry, which is more prevalent in nature. Compared to works that apply grouping and merging of superpixels or regions [19, 18] or those that iteratively improve the final symmetry segmentation [32, 36], our segmentation approach is not only faster but in addition is able to handle symmetry axes with multiple branches. Finally, our complete approach is also modality agnostic unlike [9, 41], and we show improved segmentation results when we enforce global symmetrical consistency over centerlines predicted using [35] in a dataset of road networks.

3. Approach

Our full approach for detecting and segmenting curved reflection symmetric structures consists of two steps: 1) a

SRF-based curved symmetry detector which provides input to 2) a MRF-based symmetry-constrained segmentation approach. We describe these steps in the sections that follow.

3.1. Fast curved symmetry detection via SRF

Patch-based symmetry features. In order to detect curved symmetries/centerlines in real images with clutter, a key requirement is the ability to efficiently extract robust features from the input image that are suggestive of symmetry. Our feature selection approach is motivated by two issues well-known in visual symmetry. First, similar to textures (which is a kind of translation symmetry), curved reflection symmetry is a function of image *scale*. Second, and related to the first is what features can one use to *define* symmetry in the image? In this work, we extract multiscale features based on intensity, color (from L^* , a^* and b^* channels), oriented Gabor edges and texture features [26] by comparing $N \times N$ patches with different orientations (we use 8 discrete orientations) densely in the image (Fig. 2 (A)). The reason is that such features capture different forms of symmetry information that are complementary, e.g. edge based features can suggest symmetry at textureless regions. For efficiency, we adopt the integral image implementation of [38]. For each patch, we compare the empirical distribution of feature histograms using the robust EMD-L1 distance [22] where a small value suggests a region with strong symmetry. In addition to these local features, we compute symmetric *spectral features* proposed by [38]. These are similar to the intervening contour cue of [26] except that curved symmetry responses from histogram comparisons above are used to construct the affinity matrix prior to extracting the eigenvectors using normalized-cuts [34]. The output \mathcal{X}_f is a set local symmetry responses over multiple scales (we use

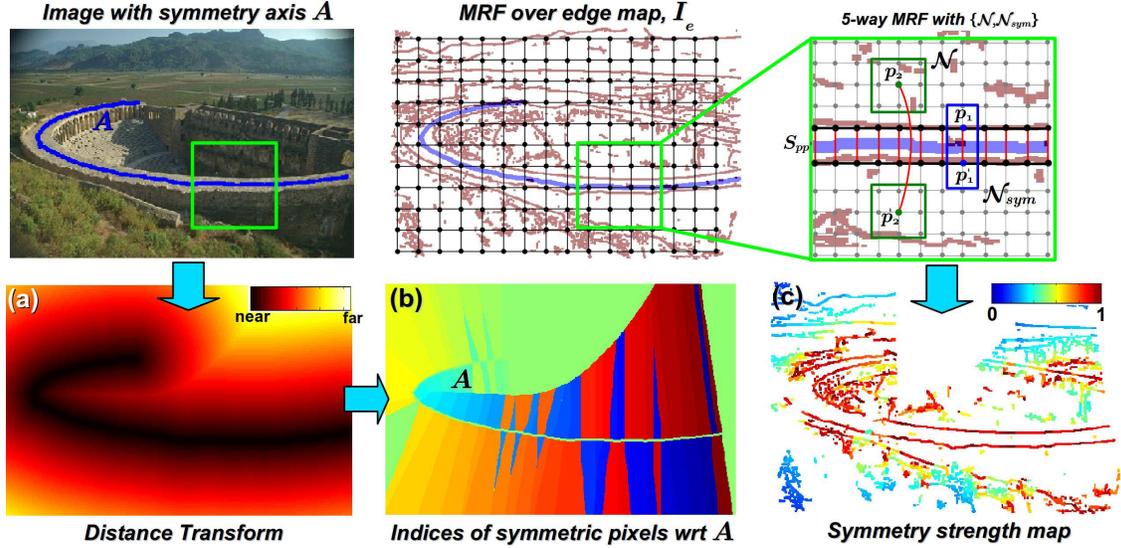


Figure 3. Curved symmetry-constrained segmentation. (Top) Constructing the 5-way MRF over I_e with cross-symmetry terms, $S_{pp'}$ given a curved symmetry axis, A . Every node in the MRF consists of \mathcal{N} (4-way, green box) and \mathcal{N}_{sym} (cross-symmetry, blue box) neighbors. We detail two symmetric neighbors $\{p_1, p_1'\}$ and $\{p_2, p_2'\}$ with their corresponding cross-symmetry terms as red links. Note that not all $S_{pp'}$ are shown for clarity. (Below) Computing the symmetry prior: (a) Given A , we compute its distance transform. (b) We then link the closest pixels along the same iso-contours on opposite sides of A to form symmetric pairs. (c) Visualization of the symmetry strength used in $eS_{pp'}$, with red denoting stronger symmetries.

4 scales here) for each of the 6 feature channels considered.

Symmetry detection via SRF. We use a variant of the Random Forest (RF) [10] classifier that predicts *structured* outputs, known as the Structured Random Forest (SRF), for predicting curved symmetries given a novel input image. A SRF, similar to a RF, is an ensemble learning technique that combines K decision trees, (T_1, \dots, T_K) , trained over random permutations of the data to prevent overfitting. A SRF, however, is able to learn mappings between inputs and outputs of arbitrary complexity such as segmentations [14] or boundary relationships [6] while retaining the fast inference inherent in RF. In this work, we train a SRF using patch-based (of size $N \times N$) symmetry responses, $x_f \in \mathcal{X}_f$ as features, and binary structured labels of groundtruth curved symmetries $\mathcal{Y} = \mathbf{1}^{N \times N}$. The goal of training the SRF is to learn, for the i^{th} internal (split) node, the optimal splitting parameters θ_i for each binary split function $h(x_f, \theta_i) \in \{0, 1\}$. If $h(\cdot) = 1$ we send x_f to the left child and to the right child otherwise (Fig. 2 (B)). $h(x_f, \theta_i)$ is an indicator function with $\theta_i = (d, \rho)$ and $h(x_f, \theta_i) = \mathbf{1}[x_f(d) < \rho]$, where d is the feature dimension of one of the input features described above. ρ is based on maximizing a standard information gain criterion M_i :

$$M_i = H(\mathcal{D}_i) - \sum_{o \in \{L, R\}} \frac{|\mathcal{D}_i^o|}{|\mathcal{D}_i|} H(\mathcal{D}_i^o) \quad (1)$$

that splits the input data $\mathcal{D}_i \subset \mathcal{X}_f \times \mathcal{Y}$ at node i into \mathcal{D}_i^L (left child) and \mathcal{D}_i^R (right child) respectively. $H(\mathcal{D}_i) =$

$\sum_y c_y(1 - c_y)$ is the Gini impurity measure with c_y denoting the proportion of symmetry features in \mathcal{D}_i with labels $y \in \mathcal{Y}$. As noted in [6], computing eq. (1) using structured \mathcal{Y} is more feasible if one imposes an intermediate mapping function $\Pi : \mathcal{Y} \mapsto \mathcal{B}$ of structured labels onto the discrete labels $b \in \mathcal{B}$. The number and type of discrete labels, $|\mathcal{B}|$, is an empirical measure of the *diversity* of structured curved symmetries that we expect to encounter. To determine Π , we first apply an Expectation-Maximization (EM)-based clustering over DAISY [37] descriptors from randomly sampled symmetry patches from \mathcal{Y} . The final clusters obtained are then used to define \mathcal{B} . The process is repeated with the remaining data \mathcal{D}^o , $o \in \{L, R\}$ at both child nodes until M_i is below a fixed threshold or a desired tree depth h_d is reached. The leaf nodes at each T_k store a distribution of curved symmetry labels encountered during training. Inference using SRF is straightforward (Fig. 2 (C)). We sample test patches densely over the image to obtain test features, $\mathcal{X}_{\text{test}}$, and pass them into the SRF to produce a structured prediction of the symmetry axes per decision tree T_k . Averaging these responses over all K trees produces the final curved symmetry predictions. We then convert these predictions (a continuous value) into a set of curved symmetry axes: $\mathcal{A} = \{A_1, A_2, \dots, A_n\}$, where each $A \in \mathcal{A}$ is defined as a contiguous single-pixel wide segment. Notably, these responses can be seen as an estimate on the symmetry strength of the test patch, denoted as $A(r)$ for a pixel $r \in A$, and we use it to modulate the amount of symmetry to enforce in the segmentation step, described next.

3.2. Curved symmetry-constrained segmentation

We embed curved symmetry constraints via a modified Markov Random Field (MRF) representation over the binary image *edge map* (Fig. 3 (top)), I_e (here we use SE [6] and retain responses >0.03). Using I_e is important here as it ensures that the segmentation results obtained are *not* influenced by color or intensity similarity but only by the detected symmetry axes, which is our goal.

Each node in the MRF is a pixel in I_e , with links (graph-edges) between nodes denoting the local relationship between connected pixels. Pixels that are directly connected with one another form a local neighborhood or clique. In addition to the unary and pairwise terms over links in a standard 4-way neighborhood clique system, we add in a link at each node that connects, based on the detected set of curved symmetry axes, \mathcal{A} , their closest *symmetrical* neighbor. To do this, we first compute its distance transform, $D_{\mathcal{A}}$. Next, pixels that lie on the same iso-contours on opposite sides of each $A \in \mathcal{A}$ are linked (Fig. 3 (below)). This additional link, called the *cross-symmetry* term, creates a new 5-way neighborhood clique system that enforces both local (4-way) and global curved symmetry constraints within a single MRF model. This ensures both global symmetrical consistency while allowing for small local deformations in the final segmentation. In addition, since this term is computed locally with respect to A , our model handles multiple axes and branching symmetries with no additional modifications. Finally, as the 5-way MRF model retains a local clique neighborhood system, the optimal labeling can be efficiently obtained using standard graph-cuts. We use the popular max-flow/min-cut toolbox of Kolmogorov and Zabih [13] in our implementation.

We detail the binary energy function E used here. Let $\mathcal{L} = \{0, 1\}$ be the labels of the background and the symmetrical region respectively. \mathcal{P} is the set of all pixels in I_e , with $\{\mathcal{N}, \mathcal{N}_{sym}\}$ denoting the 5-way neighborhood clique system consisting of the 4-way pairwise neighbors (p, q) and cross-symmetry neighbors (p, p') respectively. The energy function is thus defined as:

$$E(f) = \sum_{p \in \mathcal{P}} U_p(f_p) + \sum_{(p,q) \in \mathcal{N}} V_{pq}(f_p, f_q) + \sum_{(p,p') \in \mathcal{N}_{sym}} S_{pp'}(f_p, f_{p'}) + \sum_{(p,q) \in \mathcal{N}} B_{pq}(f_p, f_q) \quad (2)$$

where $f_p \in \mathcal{L}$ is the label assigned to pixel $p \in \mathcal{P}$ and $f = \{f_p | p \in \mathcal{P}\}$ is the labeling of all the pixels in the image. The first two terms, $\{U_p, V_{pq}\}$ of eq. (2) are the standard unary and pairwise terms that encode the foreground prior and boundary information used in the majority of MRF-based segmentation approaches [33, 4]. For the unary term, instead of a foreground model derived from color or intensity information (which we do not have), we

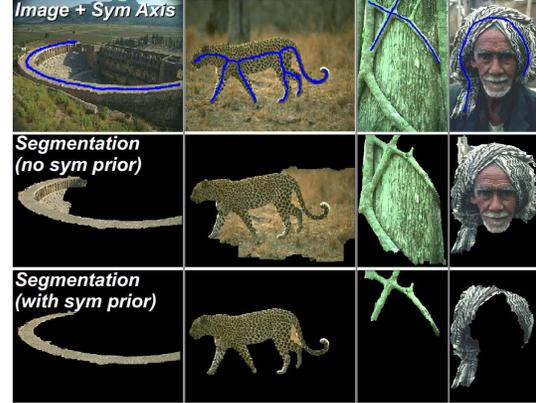


Figure 4. Example symmetry-constrained segmentations. Notice that we are able handle symmetry axes with multiple branches and produce more accurate segments with the symmetry prior term.

set pixels that overlap with any axis \mathcal{A} , $p_{\mathcal{A}}$, as foreground ($U_{p_{\mathcal{A}}}(0) = \infty$) and pixels along the image boundary, p_B , are set as background ($U_{p_B}(1) = \infty$). Similarly for V_{pq} , we replace image intensities used in [4] with their edge labels:

$$V_{pq}(f_p, f_q) = \begin{cases} \exp(-\frac{(I_e(p) - I_e(q))^2}{2\sigma^2}), & \text{if } f_p \neq f_q \\ 0, & \text{otherwise} \end{cases} \quad (3)$$

so that the final segmentation aligns with I_e .

The third term is the *symmetry prior* term. This term sums up the cost for assigning different labels to symmetrical neighbors (p, p') :

$$S_{pp'}(f_p, f_{p'}) = \begin{cases} es_{pp'}, & \text{if } f_p \neq f_{p'} \\ \beta, & \text{otherwise} \end{cases} \quad (4)$$

where $\beta < 1$ is a small positive value that provides a penalty when symmetrical neighbors are assigned similar labels. $es_{pp'}$ is a measure of symmetry strength defined as:

$$es_{pp'} = 1 + \beta - \frac{1}{Z} \log(1 + \|D_{\mathcal{A}}(p) - D_{\mathcal{A}}(p')\| + \nu_{pp'}) \quad (5)$$

where $D_{\mathcal{A}}(p) \triangleq \min_{(p_a \in \mathcal{A})} \|p - p_a\|$ is the distance between pixel p and $p_a \in \mathcal{A}$, its closest pixel along the symmetry axis obtained from the distance transform. $\nu_{pp'} = 1 - (A(p_a) + A(p'_a))/2$ is the symmetry score predicted by the SRF, where we take the mean value of the two corresponding symmetry scores along \mathcal{A} . Combining these two estimates of symmetry within $es_{pp'}$ tend to ameliorate the inherent noisy symmetric pixel correspondences caused by internal edges or textures in I_e . $Z = \max_{(p,p') \in \mathcal{P}} (\log(1 + \|D_{\mathcal{A}}(p) - D_{\mathcal{A}}(p')\| + \nu_{pp'}))$ normalizes the second term in eq. (5) to $[0, 1]$ and as a result $es_{pp'}$ is in the range $[\beta, 1 + \beta]$. Since eq. (5) sets a large $es_{pp'}$ for pixels with different labels exhibiting *strong* symmetries, this encourages symmetrical pixels to have the same labels, and as a consequence,

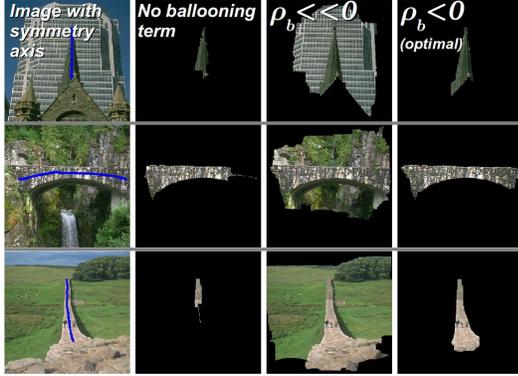


Figure 5. Effect of the ballooning term, B_{pq} . (L-R) Without B_{pq} , the final segmentation tends to be small. When $\rho_b \ll 0$, over-expansion occurs, resulting in a degenerate segmentation. Using an appropriate value for ρ_b produces an optimal segmentation.

enforces symmetry in the final segmentation. Notably, as $es_{pp'}$ is derived from D_A and the SRF predicted symmetry scores, we are able to modulate the effect of this term, allowing for symmetrical and asymmetrical configurations to occur at appropriate locations. We show some example segmentation results in Fig. 4 comparing it to the case when no symmetry prior is used (a standard 4-way MRF).

The final term B_{pq} is a “ballooning” term, introduced by Veksler [40] that encourages the final segmentation to expand, in opposite directions along both sides of A so that a reasonably-sized symmetric segment is obtained. Without this term, the final segmentation tends to be small, as symmetry strengths are usually the largest between the closest (p, p') . Assuming that pixel p is further away from pixel q with respect to A , we have:

$$B_{pq}(f_p, f_q) = \begin{cases} 0, & \text{if } f_p = f_q \\ \infty, & \text{if } f_p = 1 \text{ and } f_q = 0 \\ \rho_b, & \text{if } f_p = 0 \text{ and } f_q = 1 \end{cases} \quad (6)$$

where ρ_b is a ballooning cost that we set to control the expansion of the final segmentation. Following [40], the value of ρ_b is usually set to a small negative value. However, when $\rho_b \ll 0$, over-expansion occurs resulting in a degenerate (and undesirable) symmetrical segmentation (Fig. 5). We use $\rho_b = -0.03$ in all experiments.

A note on the submodularity of all the pairwise terms in eq. (2). Clearly, V_{pq} and B_{pq} are submodular by construction. The symmetry prior term, $S_{pp'}$, is also submodular since by eq. (5), $es_{pp'} \geq \beta$ for all values of β . From [13], E can be minimized exactly via graph-cuts.

4. Experiments

4.1. Datasets, baselines and evaluation procedures

We evaluate the performance of our approach in terms of: 1) curved symmetry prediction accuracy and 2) segmen-

tation accuracy. We detail the datasets, baselines and evaluation procedures for these two performance criteria in the following paragraphs. The supplementary material contains full details of all parameters used.

Curved symmetry prediction. We use the SYMMAX-300 dataset (200 train/ 87 test) introduced by [38], that contains curved symmetry annotations from the BSDS-300 dataset [26]. Specifically, automatically generated medial axes are presented to human annotators who select which axes best supports the groundtruth segments. We follow the same evaluation procedure as [26], where instead of boundaries, the groundtruths are human annotated curved symmetries and we report the Precision-Recall (P-R) curves and the ODS, OIS and AP metrics of [1] using the same evaluation parameters suggested by [38] where symmetry pixels close enough to the groundtruth ($<0.01\%$ of the image diagonal) are considered correctly matched. As baselines, we compare our SRF-based approach (SRFSym) with two state-of-the-art curved symmetry detectors: 1) global-symmetry (gSym) of Tsogkas and Kokkinos [38] and 2) symmetric deformable-discs (DefDiscs) of Lee et al. [18]. We also compared our approach to the recent regression-tree based (RTree) centerline prediction method of Sironi et al. [35] over published results in their “Aerial” dataset of 14 satellite images (7 train/ 7 test) of road networks from New York state (NY-roads). Following [35], we consider symmetry pixels within 2 pixels of the groundtruth axes as correct to obtain the P-R, ODS, OIS and AP metrics.

Segmentation of symmetric structures. We use three datasets (train/test splits) for our main evaluation: 1) SYMSEG-300 (200/87), 2) BSD-Parts (0/36) and 3) Weizmann Horses, WHD (20/61) (both from [18]). SYMSEG-300 is an extension of SYMMAX-300 where we extract symmetric segments based on the original BSDS-300 groundtruth segments. BSD-Parts and WHD were introduced by [18] as one dataset for evaluating the DefDiscs superpixel grouping approach. For comparisons, we applied our symmetry-constrained segmentation approach (SymSegGC) using symmetry axes predictions from: 1) SRFSym (our approach), 2) gSym and 3) DefDiscs . As an additional demonstration of the contribution of the symmetry priors, $\{S_{pp'}, B_{pq}\}$, we evaluated SRFSym and gSym without these two priors, effectively reducing the segmentation to a standard MRF-based approach (GC). We also compared the grouped superpixels segments obtained from DefDiscs (DefDiscs-SP) as an additional baseline. Following [18], we consider a segment as correct when its standard Intersect-over-Union (IoU) score with respect to the groundtruth exceeds 0.4 over all three datasets and report the resulting P-R curves and Average Precision (AP) metrics for each method. We also compared SymSegGC with the estimated centerline scales predicted by RTree (RTree-ES) over the NY-roads dataset where we used symmetry axes

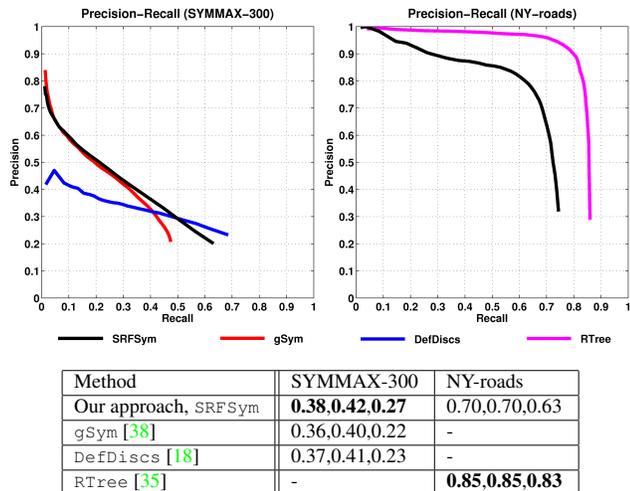


Figure 6. Symmetry prediction accuracy. (Top) Precision-recall curves: SYMMAX-300 (left) and NY-roads (right). (Below) [ODS, OIS, AP] scores [1] in each cell. Best viewed in color.

predictions from SRFSym with/without symmetry priors and similarly for RTree. The same evaluation procedure of [35] that applies an exclusion zone of 0.4% of the groundtruth radius was used to generate comparable results.

4.2. Results

Figs. 6 and 7 summarize the the evaluations performed as described in the previous section. We also show example results of curved symmetry detection and segmentation of corresponding symmetric regions using our approach compared to DefDiscs-SP [18] in Fig. 8. We briefly discuss these results and their implications.

Curved symmetry accuracy over SYMMAX-300. Our method, SRFSym, returns the most accurate curved symmetry predictions compared to gSym and DefDiscs in all accuracy metrics [ODS,OIS,AP] (Fig. 6 (top-left)). Notably, we see that beyond a recall value of 0.2, SRFSym outperforms gSym consistently at higher recalls. This shows that SRFSym’s curved symmetry predictions are more accurate across a larger range of symmetry scores. This is likely due to: 1) our complementary set of features (gSym does not use edges) which works better at non-textured regions and 2) the structured predictions over multiple scales smooth out wrong predictions across multiple decision trees.

Curved symmetry accuracy over NY-roads. In this dataset, SRFSym is unable to match the (almost) perfect curved symmetry predictions of RTree (Fig. 6 (top-right)), even with reasonably high precision (>0.8) for most recalls. The reason for the drop in precision at high recalls is that SRFSym responds to other symmetric regions (besides roads) that are not in the groundtruth. This shows that for this particular task and modality, the regression formulation proposed in [35] makes sense compared to our approach

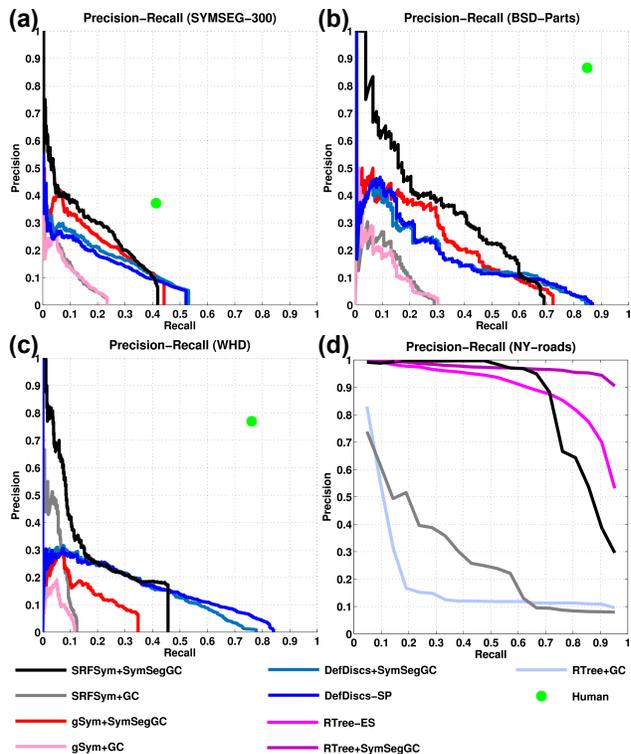


Figure 7. Symmetry-constrained segmentation accuracy. (Top) Precision-recall curves: (a) SYMSEG-300, (b) BSD-Parts, (c) WHD and (d) NY-roads. (Below) Corresponding Average precision (AP) scores per cell. Best viewed in color.

which uses more general features for detecting symmetry. Modifying our approach to take advantage of features derived from the sparse convolutional filters of RTree may also improve our performance further. Finally, it is also important to note that although SRFSym is comparatively less precise, its inference is extremely fast compared to RTree: seconds compared to the minutes/hours reported in [35].

Symmetric segmentation accuracy over SYMSEG-300, BSD-Parts and WHD. The general observation is that our full approach (SRFSym+SymSegGC) reports the best overall AP compared to other approaches as shown in Fig. 7 (a-c). Removing the symmetry prior in all approaches decreases accuracy by a significant amount, highlighting its importance. The dataset that challenges SRFSym+SymSegGC the most is WHD, where the textureless and small regions of horses (e.g. legs, tails) are better captured by the superpixels

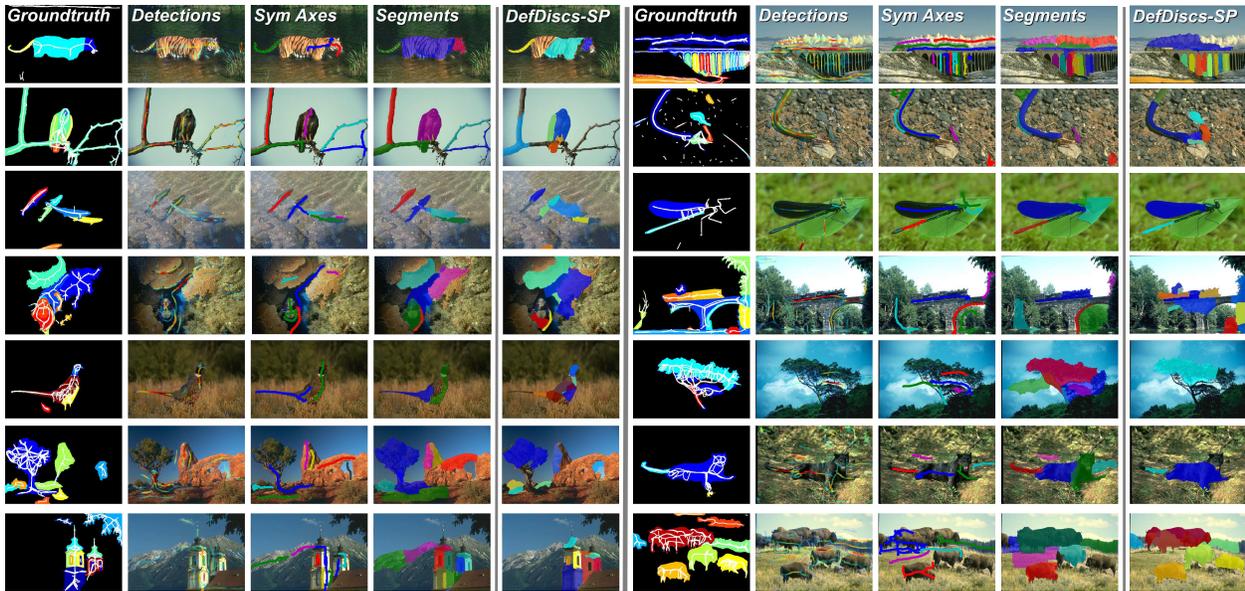


Figure 8. Example curved symmetry detection and symmetrical segmentation results, 7 results per panel: (L-R): Groundtruth symmetry axes (white) and regions, curved symmetry detections using SRFSym , extracted symmetry axes, curved symmetry-constrained segments using SymSegGC , segmentation using DefDiscs-SP [18]. More results are in the supplementary material. Best viewed in color.

computed in DefDiscs-SP . Nonetheless, our full approach is able to extract symmetric parts with better accuracy from WHD until 0.46 recall. An interesting observation is when we pair up symmetry axes predicted by DefDiscs with SymSegGC , the performance is at least on-par (or slightly better in SYMSEG-300) with DefDiscs-SP . These results highlight the *complementary* nature of both approaches: while [18] is a local (and slower) approach that groups superpixels, our proposed approach presents a faster alternative that captures longer range branched symmetries.

Symmetric segmentation accuracy over NY-roads. Although our full approach ($\text{SRFSym}+\text{SymSegGC}$) does not outperform RTree-ES in terms of overall AP, our precision is still higher than RTree-ES up to a reasonably high recall of 0.7 (Fig. 7 (d)). The rapid drop in precision after this recall is once again due to SRFSym responding to other symmetric regions in the image. Another interesting observation is the improved performance over RTree when we pair the centerlines of RTree with SymSegGC . This highlights the key advantage of enforcing a global symmetrical consistency which greatly improves the accuracy of the final segmentation.

5. Conclusions

We have presented a complete approach for detecting and segmenting curved reflection symmetric structures from real images. This is achieved by combining detections using a fast SRF-based curved symmetry detector with a novel MRF-based curved symmetry-constrained segmentation approach. Key to the approach is the use of a robust

set of multiscale patch-based symmetry features that detects local symmetry for training a SRF classifier to predict curved symmetries in real images. By modulating the cross-symmetry terms in the MRF via these predicted symmetries, we are able to produce accurate segmentations of approximate symmetrical structures, common in real images. Experimental evaluations confirm that our approach is not only more accurate than existing state-of-the-art, but is also flexible enough to improve existing segmentation approaches when paired with their curved symmetry detections.

As shown by the superior “human” performance in Fig. 7 (a-c), there is still a lot of room for improvement. We intend to investigate better local and global cues, possibly derived from superpixels to handle textureless regions or from sparse convolutional filters that have shown good performance in NY-roads. We also plan to extend this work for detecting and segmenting regions with other forms of symmetry (e.g. rotational and translational) in real images. By making our detection and segmentation code available¹, we also provide a tool to the community so that others can exploit for higher-level tasks such as recognition or scene understanding.

Acknowledgements: This work was funded by the support of the European Union under the Cognitive Systems program (project POETICON++), the National Science Foundation under INSPIRE grant SMA 1248056, and by DARPA through U.S. Army grant W911NF-14-1-0384. We thank S. Tsogkas and T. Lee for initial help with their source code and data.

¹Source code, data and more results are available at <http://www.umiacs.umd.edu/~cteo/SymmetrySegmentation/>

References

- [1] P. Arbelaez, M. Maire, C. C. Fowlkes, and J. Malik. From contours to regions: An empirical evaluation. In *CVPR*, pages 2294–2301, 2009.
- [2] H. Blum. A transformation for extracting new descriptors of shape. In *Models for the Perception of Speech and Visual Form*, pages 362–380. MIT press Cambridge, 1967.
- [3] Y. Boykov, O. Veksler, and R. Zabih. Fast approximate energy minimization via graph cuts. *PAMI*, 23(11):1222–1239, 2001.
- [4] Y. Y. Boykov and M.-P. Jolly. Interactive graph cuts for optimal boundary & region segmentation of objects in ND images. In *ICCV*, pages 105–112, 2001.
- [5] S. Dickinson and Z. Pizlo. *Shape perception in human and computer vision*. Springer, 2013.
- [6] P. Dollár and C. L. Zitnick. Fast edge detection using structured forests. *PAMI*, 37(8):1558–1570, 2015.
- [7] J. Driver and G. C. Baylis. Preserved figure-ground segregation and symmetry perception in visual neglect. *Nature*, 360(6399):73–75, 1992.
- [8] H. Fu, X. Cao, Z. Tu, and D. Lin. Symmetry constraint for foreground extraction. *IEEE Trans Cybern*, 44(5):644–654, 2014.
- [9] G. González, F. Aguet, F. Fleuret, M. Unser, and P. Fua. Steerable features for statistical 3d dendrite detection. In *MICCAI*, pages 625–632. 2009.
- [10] T. K. Ho. Random decision forests. In *ICDAR*, pages 278–282, 1995.
- [11] X. Huang and L. Zhang. Road centreline extraction from high-resolution imagery based on multiscale structural features and support vector machines. *Int’l J. Remote Sensing*, 30(8):1977–1987, 2009.
- [12] I. Junejo, E. Dexter, I. Laptev, and P. Perez. View-independent action recognition from temporal self-similarities. *PAMI*, 33(1):172–185, 2011.
- [13] V. Kolmogorov and R. Zabih. What energy functions can be minimized via graph cuts? *PAMI*, 26(2):147–159, 2004.
- [14] P. Kotschieder, S. R. Buló, H. Bischof, and M. Pelillo. Structured class-labels in random forests for semantic image labelling. In *ICCV*, pages 2190–2197, 2011.
- [15] G. Kootstra, N. Bergstrom, and D. Kragic. Using symmetry to select fixation points for segmentation. In *ICPR*, pages 3894–3897, 2010.
- [16] M. W. Law and A. C. Chung. Three dimensional curvilinear structure detection using optimally oriented flux. In *ECCV*, pages 368–382. 2008.
- [17] S. Lee and Y. Liu. Curved glide-reflection symmetry detection. *PAMI*, 34(2):266–278, 2012.
- [18] T. S. H. Lee, S. Fidler, and S. Dickinson. Detecting curved symmetric parts using a deformable disc model. In *ICCV*, pages 1753–1760, 2013.
- [19] A. Levinshstein, S. Dickinson, and C. Sminchisescu. Multi-scale symmetric part detection and grouping. In *ICCV*, pages 2162–2169, 2009.
- [20] W. H. Li, A. M. Zhang, and L. Kleeman. Real time detection and segmentation of reflectionally symmetric objects in digital images. In *IROS*, pages 4867–4873, 2006.
- [21] T. Lindeberg. Edge detection and ridge detection with automatic scale selection. *IJCV*, 30(2):117–156, 1998.
- [22] H. Ling and K. Okada. An efficient earth mover’s distance algorithm for robust histogram comparison. *PAMI*, 29(5):840–853, 2007.
- [23] Y. Liu, H. Hel-Or, C. S. Kaplan, and L. Van Gool. Computational symmetry in computer vision and computer graphics. *Foundations and Trends in Computer Graphics and Vision*, 5(12):1–195, 2009.
- [24] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004.
- [25] G. Loy and J.-O. Eklundh. Detecting symmetry and symmetric constellations of features. In *ECCV*, pages 508–521. 2006.
- [26] D. R. Martin, C. C. Fowlkes, and J. Malik. Learning to detect natural image boundaries using local brightness, color, and texture cues. *PAMI*, 26(5):530–549, 2004.
- [27] A. Mishra, Y. Aloimonos, and C. Fermüller. Active segmentation for robotics. In *IROS*, pages 3133–3139, 2009.
- [28] N. J. Mitra, L. J. Guibas, and M. Pauly. Partial and approximate symmetry detection for 3D geometry. *ACM Trans Graphics*, 25(3):560–568, 2006.
- [29] J. L. Mundy and A. E. Zisserman. *Geometric invariance in computer vision*, volume 92. MIT press Cambridge, 1992.
- [30] A. Ogale, C. Fermüller, and Y. Aloimonos. Motion segmentation using occlusions. *PAMI*, 27(6):988–992, 2005.
- [31] D. Reisfeld, H. Wolfson, and Y. Yeshurun. Context-free attentional operators: the generalized symmetry transform. *IJCV*, 14(2):119–130, 1995.
- [32] T. Riklin-Raviv, N. Kiryati, and N. Sochen. Segmentation by level sets and symmetry. In *CVPR*, pages 1015–1022, 2006.
- [33] C. Rother, V. Kolmogorov, and A. Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. *ACM Trans Graphics*, 23(3):309–314, 2004.
- [34] J. Shi and J. Malik. Normalized cuts and image segmentation. *PAMI*, 22(8):888–905, 2000.
- [35] A. Sironi, V. Lepetit, and P. Fua. Multiscale centerline detection by learning a scale-space distance transform. In *CVPR*, pages 2697–2704, 2014.
- [36] Y. Sun and B. Bhanu. Reflection symmetry-integrated image segmentation. *PAMI*, 34(9):1827–1841, 2012.
- [37] E. Tola, V. Lepetit, and P. Fua. DAISY: An Efficient Dense Descriptor Applied to Wide Baseline Stereo. *PAMI*, 32(5):815–830, 2010.
- [38] S. Tsogkas and I. Kokkinos. Learning-based symmetry detection in natural images. In *ECCV*, pages 41–54. 2012.
- [39] C. W. Tyler. *Human symmetry perception and its computational analysis*. Psychology Press, 2002.
- [40] O. Veksler. Star shape prior for graph-cut image segmentation. In *ECCV*, pages 454–467. 2008.
- [41] C. Wang, Y. Li, W. Ito, K. Shimura, and K. Abe. A machine learning approach to extract spinal column centerline from three-dimensional ct data. In *SPIE Medical Imaging*, pages 72594T–72594T, 2009.
- [42] S.-C. Zhu. Stochastic jump-diffusion process for computing medial axes in markov random fields. *PAMI*, 21(11):1158–1169, 1999.