

Beyond Planar Symmetry: Modeling human perception of reflection and rotation symmetries in the wild

Christopher Funk Yanxi Liu
School of Electrical Engineering and Computer Science.
The Pennsylvania State University. University Park, PA. 16802, USA
{funk, yanxi}@cse.psu.edu

Abstract

Humans take advantage of real world symmetries for various tasks, yet capturing their superb symmetry perception mechanism with a computational model remains elusive. Motivated by a new study demonstrating the extremely high inter-person accuracy of human perceived symmetries in the wild, we have constructed the first deep-learning neural network for reflection and rotation symmetry detection (Sym-NET), trained on photos from MS-COCO (Microsoft-Common Object in COntext) dataset with nearly 11K consistent symmetry-labels from more than 400 human observers. We employ novel methods to convert discrete human labels into symmetry heatmaps, capture symmetry densely in an image and quantitatively evaluate Sym-NET against multiple existing computer vision algorithms. On CVPR 2013 symmetry competition testsets and unseen MS-COCO photos, Sym-NET significantly outperforms all other competitors. Beyond mathematically well-defined symmetries on a plane, Sym-NET demonstrates abilities to identify viewpoint-varied 3D symmetries, partially occluded symmetrical objects, and symmetries at a semantic level.

1. Introduction

From the evolution of plants, insects and mammals, as well as the earliest pieces of art in 20,000 BCE through the modern day [20, 26, 64], perfectly symmetrical objects and scenes are rare while approximate symmetries are readily observable in both natural and man-made worlds. Perception of such *symmetries in the wild* has played an instrumental role at different levels of intelligence [10, 13, 20, 21, 52, 56, 57, 71] that function effectively in an otherwise cluttered and often uncertain world. For decades, among human vision and computer vision researchers alike, the search for computational models, analytical or biological basis, and explanations for symmetry perception [28, 39, 65, 70] has proven to be non-trivial [2, 44, 69].

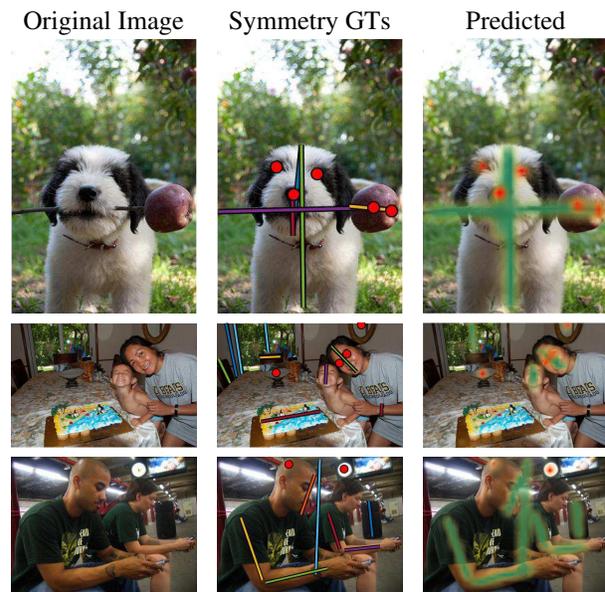


Figure 1. Sample images from the MS-COCO dataset [40]. Symmetry ground-truths (mid-column) are computed from human labels (Figure 2): line segments for reflection symmetry axes and red dots for rotation symmetry centers. Right column: heatmaps for predicted symmetries, green for reflection symmetry axes, and red for rotation symmetry centers.

The mathematical definition of a symmetry transformation g of a set S is clear and simple [9, 44, 75], $g(S) = S$. However, how to identify a symmetry in a photo remains ambiguous [42, 55]. The dilemma is: should a symmetry g in an image be determined by its mathematical definition or human perception? Human perception of symmetry can deviate grossly from 2D pixel-geometry on an image. In Figure 1: a face profile is perceived as having a reflection symmetry; a rotation center is identified for half of a round mirror, and for the highly skewed top-center of an apple; and a reflection symmetry is labeled between two sitting people playing with their phones! None of this mixture of 3D/object prior-based and semantic-level symmetries has been attempted in existing symmetry detection models.

Motivated by the strong human perceptual consistency of symmetry demonstrated in a recent study on 1,200 MS-COCO [40] images rated by 400 human raters [17], we take a first step in building a computational platform for learning to mimic human visual perception of reflection and rotation symmetries. Though multi-layer Convolutional Neural Networks (CNNs) have been trained to detect image class [30, 60, 61], semantic segments [5, 6, 19, 45], surface normals [72], face identities [62], human pose [4, 53, 74], and to be invariant to rotational symmetry [11, 31, 77], little has been reported on training CNNs for reflection and rotation symmetry detection on real images. Using state-of-the-art segmentation networks as a base [6], we transform ground truth extracted from human labels into dense 2D symmetry heatmaps (as opposed to sparse labels containing only 2D coordinates), and perform dense (per pixel) regression to those heatmaps. We compare against existing algorithms that output symmetry heatmaps with the same dimensions as the input image.

Our contributions are:

- to build the first deep, dense, and multiple symmetry detector that mimics sophisticated human symmetry perception beyond planar symmetries;
- to convert sparse symmetry labels into dense heatmaps to facilitate CNN training using human labels;
- to systematically and extensively validate and compare the predictive power of the trained CNN against existing algorithms on both mathematically well-defined and human perceived symmetries.

In short, this work advances state-of-the-art on *symmetry detection in the wild* by using CNNs to effectively mimic human perception of reflection and rotation symmetries.

2. Related Work

One can find a general review of human symmetry perception (primarily reflection) in [69], and on computational aspects of symmetry detection in [44].

2.1. Reflection Symmetry Detection

Reflection symmetry algorithms fall into two different categories depending on whether they detect sparse symmetries (straight lines or curves) [22, 34, 36, 41, 46, 76] or a dense heatmap [15, 16, 67]. The most common sparse approach to detect reflection symmetry is to match up symmetric points or contours in the image to determine midpoint and direction of the symmetry axis [3, 38, 46, 49, 50, 73]. These approaches often use a Hough transform to accumulate the axes of reflection, derived from the matched feature's midpoints and angles, and vote on the dominant symmetries. Atadjanov and Lee [1] extend the Loy and Eklundh [46] algorithm by taking the matched keypoints and then comparing the histogram of curvature around the keypoints. Wang *et al.* [73] uses local affine invariant edge

correspondences to make the algorithms more resilient to perspective distortion contours. The method does not use a Hough space to vote, opting instead to use an affine invariant pairwise (dis)similarity metric to vote for symmetries.

Pritts *et al.* [54] detect reflection, rotation and translation symmetry using SIFT and MSER features. The symmetries are found through non-linear optimization and RANSAC. Tuytelaars *et al.* [68] detect reflection through a Cascade Hough Transform. Kiryati and Gofman [27] define a Symmetry ID function implemented through Gaussian windows to find local reflection symmetry.

Lee and Liu [34, 36] have generalized the traditional straight reflection axes detection problem into finding *curved glide-reflection symmetries*. Their approach adds a translational dimension in the Hough transform so the matched features are clustered in a 3D parameter space, and the curved reflection or glide-reflection axis is found by polynomial regression between clustered features.

Tsogkas and Kokkinos [67] use a learning based approach for local reflection symmetry detection. Features are extracted using rotated integrals of patches in a Gaussian pyramid and converted into histograms. These features are spectrally clustered and multiple instance learning is used to find symmetries with multiple scales and orientations simultaneously. Teo *et al.* [63] detect curved-reflection symmetry using structured random forests (SRF) and segment the region around the curved reflection. The SRF are trained using multi-scale intensity, LAB, oriented Gabor edges, texture, and spectral features. Many trees are trained and the output of the leaves for the trees are averaged to obtain the final symmetry axes.

There have been some shallow-network reflection detection approaches (well before the current deep learning craze). Zielke *et al.* [79] use a static feed forward method to enhance the symmetric edges for detection. The max operation between the different orientations is similar to other voting systems [44]. Fukushima and Kikuchi [15, 16] present another neural network method for detecting reflection symmetry around the center of an image. They use a 4-layer network to find the symmetry axis.

Skeletonization, a related problem to reflection detection, has attracted a lot of attention recently [37, 58, 67, 76]. Shen *et al.* [59] use a deep CNN to learn symmetry at multiple scales and fuse the final output together. The network needs object skeleton ground-truth for the particular scale of the objects. The network outputs a skeleton heatmap which is thresholded to produce a binary image denoting the detected skeletons.

2.2. Rotation Symmetry Detection

Earlier work on rotation symmetry detection includes the use of autocorrelation [29, 43] and image moments [8, 47, 66]. Loy and Eklundh [46] use a variation on their

SIFT feature-based reflection symmetry approach to find rotation symmetry as well. The orientations of matched SIFT feature pairs are used to find a rotation symmetry center. The detected rotation symmetry centers emerge as maxima in the voting space. This algorithm stands out from all others since the authors have made their code publicly available, and the symmetry competition workshops in CVPR 2011/2013 have used it as the baseline algorithm for both reflection and rotation symmetry detection. Thus far, this algorithm is considered to be the best baseline algorithm for reflection and rotation symmetry detection.

Lee and Liu [32, 33, 35] have developed an algorithm to detect (1) the center of rotation, (2) the number of folds, (3) type of symmetry group (dihedral/cyclic/O(2)), and (4) the region of support. The first step of their algorithm is rotation symmetry center detection where they use a novel *frieze expansion* to transform the image at each pixel location into polar coordinates and then search for translation symmetry. The second step applies a Discrete Fourier Transform (DFT) on the frieze expansion to determine (2)-(4) listed above. In our work, for rotation symmetries we only focus on detecting rotation symmetry centers.

2.3. Dense CNN Regression

Fully Convolutional Networks [45], with CNN regressing to 2D ground-truth, have been utilized for semantic segmentation [6, 45] and pose detection [4, 53, 74]. For semantic segmentation, the output of the network is an $n \times n \times c$ matrix where n is a reduced dimension of the input image and c is the number of classes. A pixel-wise argmax operation is computed for each $n \times n$ pixel across c to classify the corresponding class. Chen *et al.* [6] uses a pyramid of upsampling atrous filters [5, 6, 24, 51] which enables more context to inform each pixel in the network output. A heatmap regression for each joint is estimated separately for human pose detection [4, 53, 74] where a Gaussian is defined over each ground-truth label to provide an easier target to regress. Without this Gaussian, the only error signaling a correct label would be from the single pixel of ground-truth output and the network would predict everything as background. The networks are trained with ℓ_2 loss. We employ the same architecture as Chen *et al.* [7] to detect symmetries while using a 2D heatmap regression similar to pose detection. Both the added context and additionally the multiple scales are relevant in detecting symmetry within the images. Similar to pose detection, we use an ℓ_2 regression where each ground-truth is defined by a Gaussian centered at the ground-truth label.

Different from recent efforts in the deep learning/CNN community where researchers are seeking networks that are rotation/reflection or affine invariant to input images [11, 18, 31, 77], our work explicitly acknowledges (near) reflection and rotation symmetries in the raw data regard-

less of the transformations applied on the input images. To the best of our knowledge, there have been no deep learning networks trained on human symmetry labels for automated reflection and rotation symmetry detections.

3. Our Approach

We propose a multi-layer, fully-convolutional neural network for reflection and rotation symmetry from real world images. We call this Sym-NET which is short for **SYM**metry detection neural **NET**work.

3.1. Data and Symmetry Heatmaps

The raw data is a collection of images from Microsoft COCO dataset [40]. The symmetry Ground-Truth (GT) labels have been collected from Amazon Mechanical Turk (Table 1). The data includes reflection axes (two points per axis) and rotation centers (one point per center). The statistics of the human labeled symmetries are shown in Figure 2. We gather human symmetry labels using the same tool as described in [17] on a superset of images reported there.

To obtain a consensus on symmetry GTs computationally, we first combine human perceived symmetry labels with an automated clustering algorithm [17]. The basic idea is to capture the exponential divergence in the nearest labeled symmetry pair distribution, use that as the minimum distance τ between neighbors and the number of required human labels as the minimum number of neighbors, and finally input both to DBSCAN [14], a method for **Density-Based Spatial Clustering of Applications with Noise** (the winner of the test-of-time award in 2014). The τ for rotation symmetry perception is 5 pixels, i.e. two symmetry labels within τ are considered to be labeling the same perceived symmetry [17]. Second, these sparse symmetry GTs

Total # of Images with GT	1,202
Total # of Images with Reflection GT	1,199
Total # of Images with Rotation GT	1,057
Mean # of GT Labelers \pm std/ Image	29.18(\pm 4.04)
Mean # of Reflection GT Labelers \pm std / Image	23.99(\pm 6.67)
Mean # of Rotation GT Labelers \pm std / Image	13.00(\pm 10.33)
Mean # GT \pm std / Image	9.14(\pm 4.74)
Mean # Reflection GT \pm std/Image	6.05(\pm 3.28)
Mean # Rotation GT \pm std / Image	3.09(\pm 2.93)
Total GT	10,982
Total Reflection GT	7,273
Total Rotation GT	3,709
Total # of Labels	107,172
Total # of Labels Used for GTs	77,756

Table 1. Statistics of labeled symmetries used in this work for training and testing Sym-NETs.

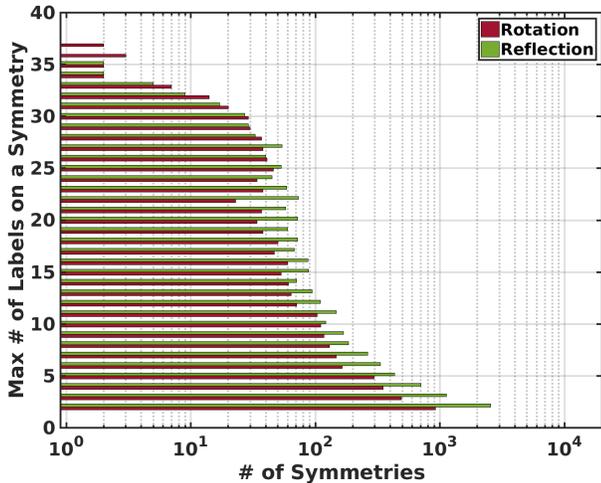


Figure 2. Histogram indicating the number of distinct human labels for each labeled reflection and rotation symmetry in the image dataset, respectively. The higher the number on the Y axis, the more consistent the raters are (agreeing with each other).

on each image are mapped into a reflection or rotation symmetry heatmap respectively [4, 53, 74]. Let GT^k be all the pixel location(s) (l) for a 1 pixel wide reflection symmetry axis or a 1 pixel rotation symmetry center and let $x_{i,j}$ be all the pixel locations for the input image. We create the dense ground truth symmetry heatmap (H) for each ground-truth symmetry k with a σ of 5 (the τ found in [17]):

$$H_{i,j,k} = \sum_{l \in GT^k} \exp\left(-\frac{\|l - x_{i,j}\|_2^2}{2\sigma^2}\right). \quad (1)$$

This is done by drawing a point for rotation center or a line for reflection axis on an image initialized with 0's and convolving with a Gaussian filter. The resulting GT heatmap is then scaled between [0,1]. The max is taken among all individual GT heatmaps in an image so that nearby labels or intersecting lines do not create artifacts in the heatmap, similar to [4]:

$$H_{i,j} = \max_k H_{i,j,k}. \quad (2)$$

This assures that the heatmap is 1.0 at each rotation center and reflection axis and decreases exponentially as it moves away. Sample heatmaps generated from human labels are shown in Figure 3. The GT images are augmented by random operations, including cropping, scaling ([.85,9,1.1,1.25]), rotating ([0°,90°,180°,270°]), and reflecting w.r.t. the vertical central axis of the image.

3.2. Network

We use two different networks, Sym-VGG is based on VGG-16 [60] and Sym-ResNet on ResNet-101 [23]. Sym-VGG uses a similar structure to VGG-16 network with a dilation of 2 pixels at the conv5 layer and then an atrous

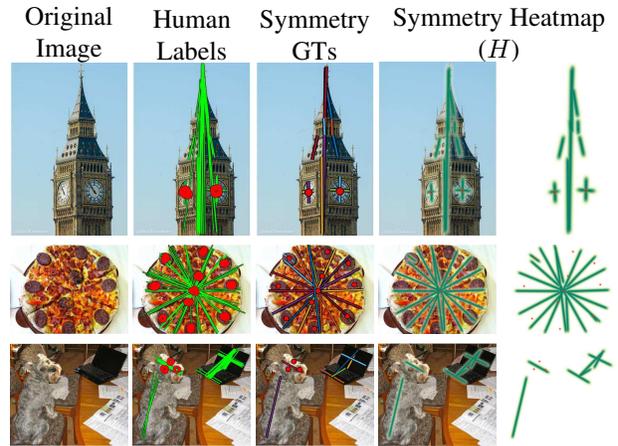


Figure 3. The progression (left to right) of converting the human labeled ground truth symmetries into symmetry heatmaps (H). The human labels are clustered to find the reflection symmetry axes and rotation symmetry centers. Reflection symmetry heatmap: green, rotation: red.

pyramid [6]. Sym-ResNet has multi-scale streams (for the entire network) of 50%, 75%, and 100% of the original image scale and dilation at the later layers. Each scale has a separate atrous convolution pyramid and is fused together using a max operation similar to [6]. The final layer of the network upsamples the output using bilinear interpolation to the actual image size (8x) and then a ℓ_2 loss is computed. This upsampling eliminates any artifacts created from downsampling the ground-truth labels and trains the network to adapt to the upsampling. We use a similar strategy of borrowing weights from networks previously trained on Imagenet and fine-tuned on MS-COCO [6, 45, 78]. This design strategy has been shown to be useful for image segmentation [5, 6, 45], and allows us to train much larger networks without the need for millions of images. Atrous convolution [5, 6, 24, 51] is useful to provide contextual information for each pixel. The context around each point proves to be crucial since symmetry detection is about finding relationships between pixels (parts).

3.3. Training

We train Sym-VGG and Sym-ResNet separately for reflection and rotation symmetry detection. We use an 80%/20% split of 1202 images from the MS-COCO dataset [40] with at least one GT for each type. The dataset includes 1199 and 1057 images with reflection and rotation symmetry ground-truth. This creates train/test datasets of 959/240 images for reflection and 846/211 for rotation. Each network is trained with an exponential learning rate multiplier of $1 - \frac{\text{batch number}^{\text{power}}}{\text{total batches}}$ similar to other recent segmentation networks [6] in the Caffe framework [25].

The reflection Sym-NETs use a learning rate of 1e-10 and 2.5e-11 and the rotation Sym-NETs use a rate of 1e-9 and 2.5e-10 for the VGG and Resnet networks respectively.

Reflection Symmetry Detection

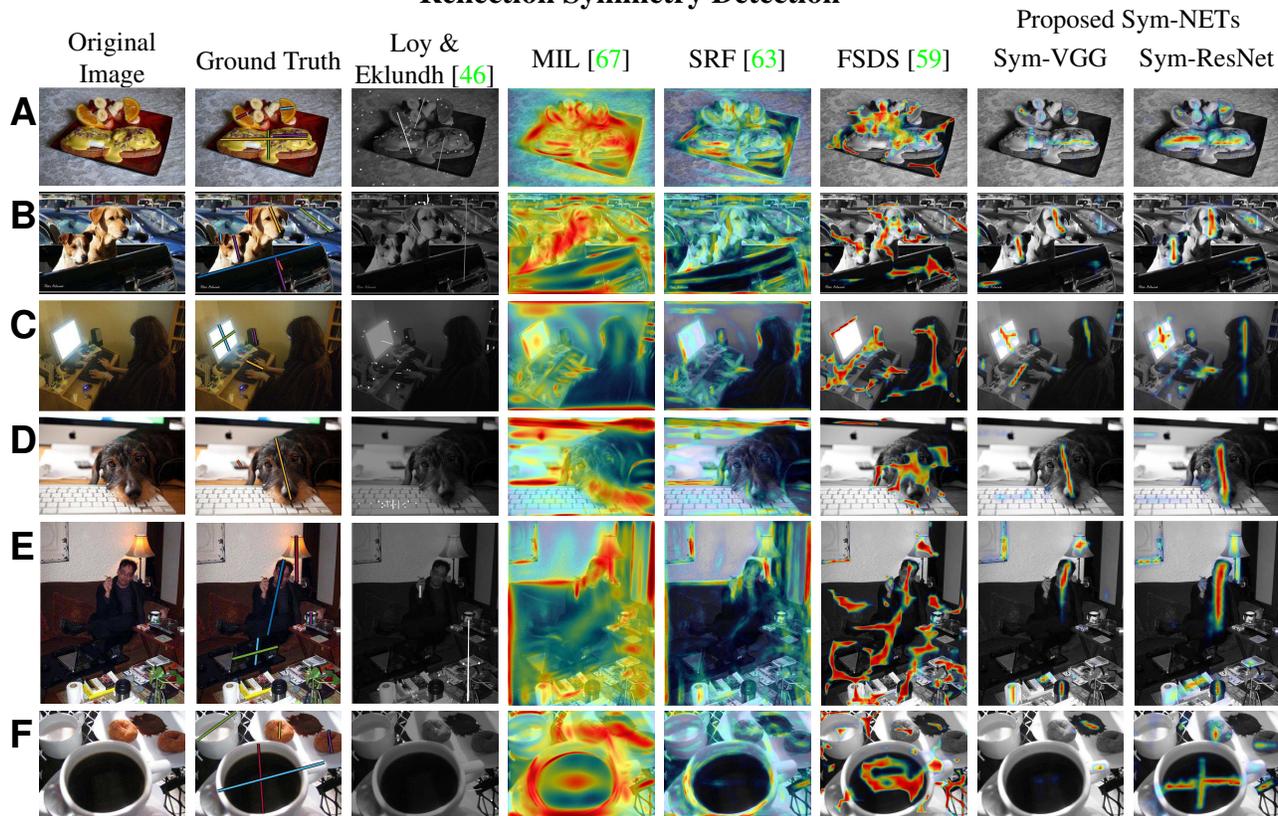


Figure 4. Examples of the original image, ground-truth, and the output for the reflection symmetry detection algorithms before thinning.

The learning rates are empirically found. The Sym-VGG takes 3 days and the Sym-ResNet takes 10 days to converge on a GeForce GTX Titan X.

3.4. Performance Evaluation

Figures 4 and 5 show sample reflection and rotation detection results. To quantitatively evaluate the performance of the networks, we compute a precision-recall curve for each symmetry detector in a similar way to [48, 67], which is generated by stepping through 100 thresholds (between [0,1]) on the networks’ heatmap output. From these scores, we also calculate the maximum F-measures ($2 \times \frac{\text{precision} + \text{recall}}{\text{precision} \times \text{recall}}$) [48, 59, 67] for each symmetry detector to obtain a single value as an indicator of its statistical strength [48, 59, 67]. For reflection, we use a one pixel-width reflection axis as the ground-truth [48, 59, 67] and use the measure defined in [48, 67]. For rotation, we use a 5-pixel radius (τ) circle around the GT symmetry [17] and calculate the explicit overlap.

3.5. Performance Comparison

Not only would we like to know which algorithm performs better on a given test set, we would also like to demonstrate whether the better performance is statistically significant. In this comparison study, we use the maxi-

um F-Measure computed from its mean precision-recall rate (Section 3.4) in order to compare all detectors at their respective optimal values. We then use a paired t-test on max F-measures between pairs of symmetry detectors and obtain the p-value indicating the significance level of their difference.

We compare the output of our symmetry detection system with both dense and sparse symmetry detection algorithms qualitatively (Figures 4 and 5) and quantitatively (Figure 6). For sparse detection, we use Loy and Eklundh’s (*LE*) [46] algorithm, a simple and fast SIFT-feature based reflection/rotation symmetry detector. The sparse output from the algorithm is transformed into dense labels by applying the same operations that create the evaluation ground truth from their sparse labels, weighted by the algorithm’s detection strength for each symmetry.

For dense detection algorithms, we include Tsogkas and Kokkinos’ Multiple Instance Learning method (*MIL*) [67], Teo *et al.*’s method (*SRF*) [63], and Deep Skeletonization network (*FSDDS*) [59] as a part of our comparison. Our goal is to determine the performance difference between the *skeletonization* and reflection symmetry detection algorithms. Even though there is a conceptual overlap on (local) symmetry, they do not detect the same types or ranges of symmetries. The same non-maximum suppression algo-

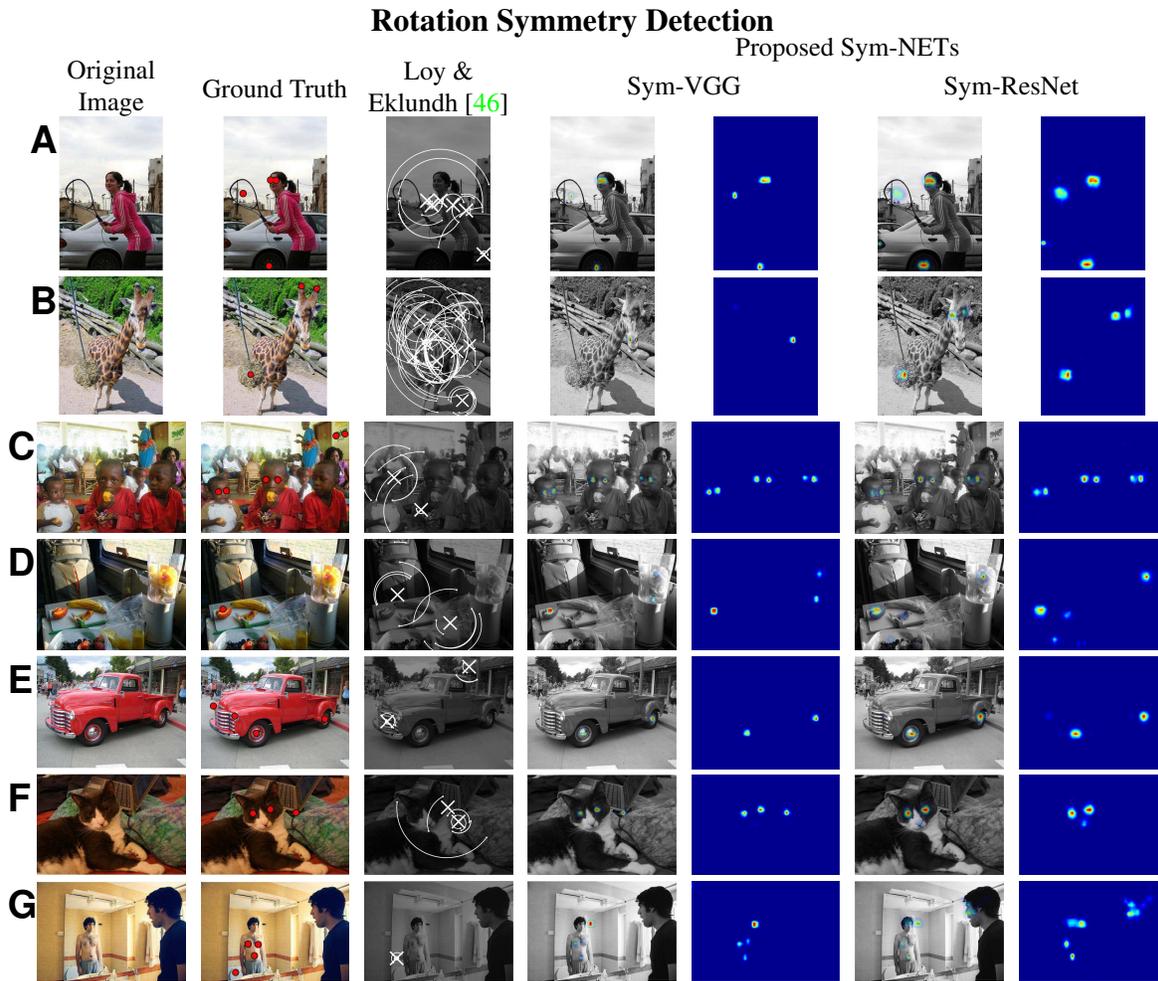


Figure 5. Examples of the original image, ground-truth, and the output for the rotation detection algorithms. Rotation symmetry heatmaps are shown for both types of Sym-NETs.

rithm [12] is applied to the output of Sym-NETs and the FS DS. All the default parameters for the algorithms are used in the comparison. On all datasets tested, at least one Sym-NET obtains significant improvement over the other detectors (Figure 6).

3.5.1 MS-COCO dataset

We test the symmetry detectors against the MS-COCO [40] dataset with symmetry labels (Section 3.3), containing 240 reflection and 211 rotation images. Both Sym-NETs significantly outperform all other detectors on the MS-COCO dataset for detecting the ground-truth symmetries derived from human labels (P -value $\ll 0.001$).

Furthermore, for the MS-COCO symmetry dataset, we take into account the number of labelers for each ground-truth symmetry. Symmetry GTs with less than 20 labels are taken out from this evaluation, creating subsets of 111 (of 240) reflection symmetry images and 73 (of 211) rotation symmetry images, representing the most prominent symme-

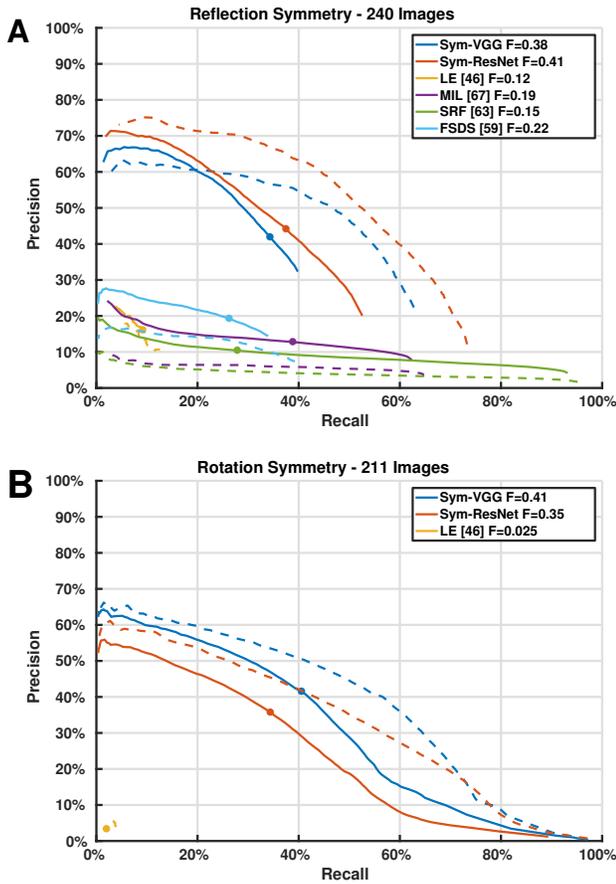
tries. The statistics of the number of human labels for each symmetry is shown in Figure 2. We observe that Sym-NETs perform better on detecting those symmetries perceived by humans as more prominent (more than 20 individual labelers for each symmetry) in the images (Figures 6A and 6B).

3.5.2 CVPR 2013 Symmetry Competition Dataset

Finally, to illustrate how Sym-NETs generalize onto other datasets, we use the test image sets from the CVPR 2013 symmetry competition [42] with 70 reflection symmetry images and 59 rotation symmetry images. Each image contains at least one labeled symmetry. During the past two CVPR symmetry detection competitions [42, 55], Loy and Eklundh’s algorithm [46] has performed most competitively. Thus we compare Sym-NET output on CVPR test image sets against those of Loy and Eklundh [46]. These visual symmetries are relatively more well-defined on the image than the MS-COCO image set.

All images and GTs of the CVPR 2013 testset are

Testset from MS-COCO [40]



Testset from CVPR'13 Symmetry Competition [42]

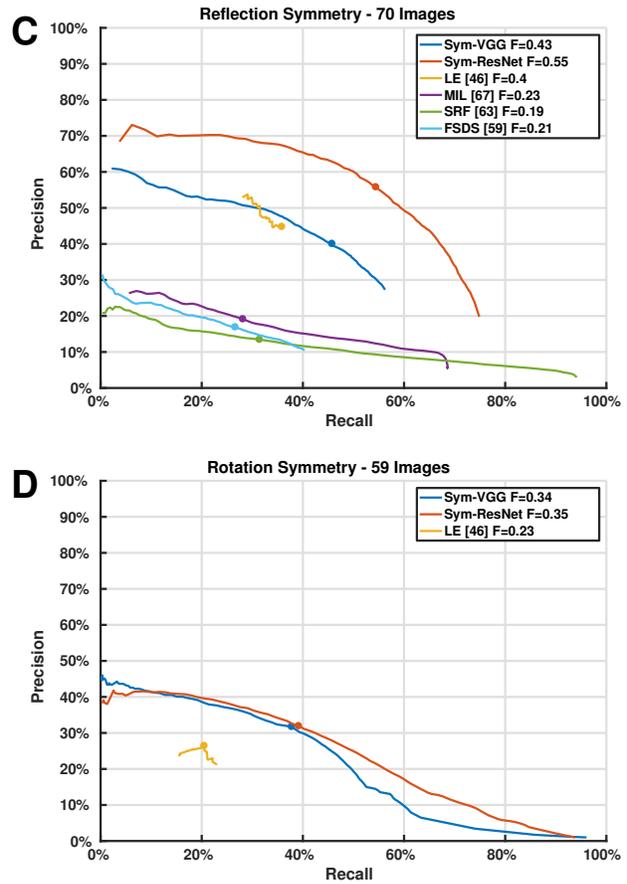


Figure 6. Comparison of the precision-recall curves for state-of-the-art symmetry detection algorithms. **A and B:** The comparison on MS-COCO images for all GT labels (solid line) and for the subset of 184 images with GT labels containing at least 20 labels (dashed line), and the maximum F-measure values (dot on the line). **C and D:** comparison on the test image set of the CVPR 2013 symmetry competition [42]. Best viewed electronically.

rescaled so the longest edge is at most 513 pixels (the maximum for our networks). The quantitative evaluations of the algorithm performance are shown in Figures 6C and 6D. For both types of symmetries, Sym-ResNet remains significantly better than all algorithms evaluated, while the F-measure of Loy and Eklundh [46] is lower but on par with Sym-VGG statistically.

4. Summary and Discussion

We have shown that our Sym-NETs, trained on human labeled data, can detect mathematically well-defined planar symmetries in images (Figures 6C, 6D CVPR 2013 symmetry detection competition testset), furthermore, it can also capture a mixture of *symmetries in the wild* that are beyond planar symmetries (Figures 4, 5, 7). The performance of the Sym-NETs is significantly superior to existing computer vision algorithms on the test images evaluated (Figure 6). Foreground as well as background symmetries are captured

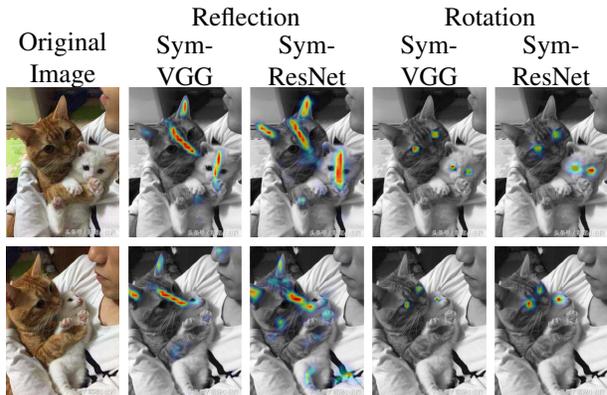


Figure 7. Two sample images to show Sym-NETs performance on out-of-plane reflection and rotation symmetry predictions. The detected reflection symmetry on human nose (bottom row) is a positive surprise, while the miss on the small white cat's face shows the limitation of Sym-NETs on detecting small, subtle symmetries.

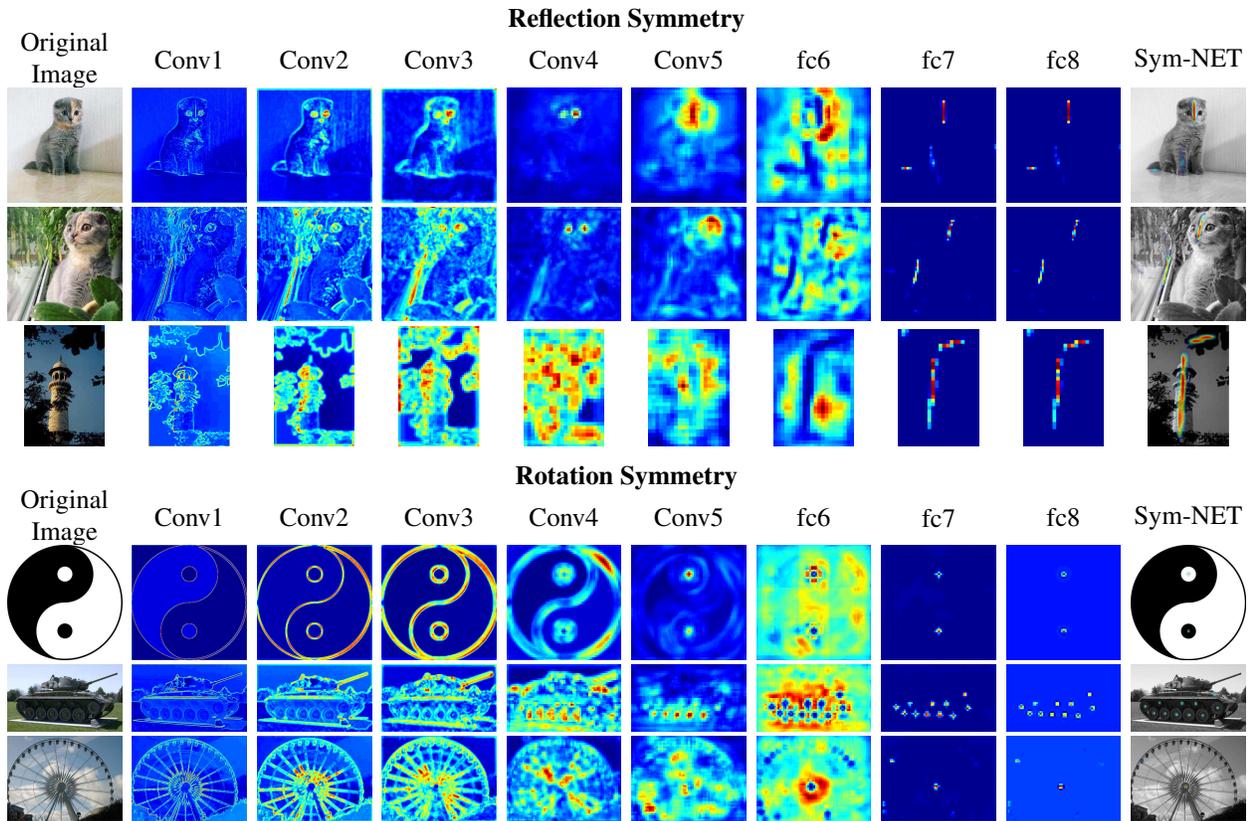


Figure 8. Example activations from Sym-VGG showing visualization of the networks. The activations shown are the sum of all the activation channels at each layer.

(Figures 4B, 4C, 5F, 5G).

Our work has provided an affirmative response to the debate on whether human perception of *symmetry in the wild* can be computationally modeled, and the deep-learning platform offers us a means to do so. However, this is only an encouraging beginning. The questions of WHAT features are learned, and HOW multiple-visual, spatial and/or semantic cues are combined to achieve the superior performance of Sym-NET remain. By peeking into the inner layers of activations in the Sym-NETs (Figure 8), we observe that for reflection symmetry, the color/shading cues fade away at deeper layers in promoting the reflection axis; for rotation symmetry, local cues seem to contribute much more to rotation centers than global (or distant) ones. Some observed human-machine discrepancies that lower the Sym-NET performance include:

- Size: Humans are better in detecting small (rotation) symmetries, e.g. the clock in Figure 1 BOTTOM, while Sym-NET fails (Figures 5C, 5E)
- Subtlety: Humans are keen at perceiving object-symmetry that is barely visible from the background, e.g. a laptop computer, Figure 4E, Sym-NETs can miss such subtleties Figure 7 BOTTOM
- Humans do not consistently label the same semantic object (such as eyes) while the networks learn to pre-

dict eyes as rotationally symmetric reliably: e.g. Figure 5C and Figure 1 TOP (dog eyes).

It has been widely accepted that symmetry perception serves as a mid-level cue that is important to human understanding of the world, ranging from how to combine shapes together into objects [52], to identify foreground from background [13], and to judge attractiveness [57]. Therefore, computer vision problems such as semantic segmentation, image understanding, scene parsing, and 3D reconstruction may benefit greatly from reliable characterizations of symmetry in the data. After many years of practice, it is about time we question the robustness of those computer vision algorithms that are solely based on first principles (i.e. mathematical definition of symmetry), and open up to a *hybridisation* of modern computing technology with classic theory. Our initial experiment with Sym-NETs has set an optimistic starting point. More examples and resources can be found on our project website: <http://vision.cse.psu.edu/research/beyondPlanarSymmetry/index.shtml>.

5. Acknowledgement

This work is supported in part by an NSF CREATIV grant (IIS-1248076).

References

- [1] I. Atadjanov and S. Lee. Bilateral symmetry detection based on scale invariant structure feature. In *Image Processing (ICIP), 2015 IEEE International Conference on*, pages 3447–3451. IEEE, 2015. [2](#)
- [2] I. Biederman. Human image understanding: Recent research and a theory. *Computer vision, graphics, and image processing*, 32:29–73, 1985. [1](#)
- [3] D. Cai, P. Li, F. Su, and Z. Zhao. An adaptive symmetry detection algorithm based on local features. In *Visual Communications and Image Processing Conference, 2014 IEEE*, pages 478–481. IEEE, 2014. [2](#)
- [4] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh. Realtime multi-person 2d pose estimation using part affinity fields. *Proceedings of the European Conference on Computer Vision (ECCV)*, 2016. [2](#), [3](#), [4](#)
- [5] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs. In *ICLR*, 2015. [2](#), [3](#), [4](#)
- [6] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *arXiv:1606.00915*, 2016. [2](#), [3](#), [4](#)
- [7] L.-C. Chen, Y. Yang, J. Wang, W. Xu, and A. L. Yuille. Attention to scale: Scale-aware semantic image segmentation. In *CVPR*, 2016. [3](#)
- [8] S.-L. Chou, J.-C. Lin, and W.-H. Tsai. Fold principal axis—a new tool for defining the orientations of rotationally symmetric shapes. *Pattern Recognition Letters*, 12(2):109–115, 1991. [2](#)
- [9] H. Coxeter. *Introduction to Geometry*. Wiley, New York, second edition, 1980. [1](#)
- [10] J. D. Delius and G. Habers. Symmetry: can pigeons conceptualize it? *Behavioral biology*, 22(3):336–342, 1978. [1](#)
- [11] S. Dieleman, J. D. Fauw, and K. Kavukcuoglu. Exploiting cyclic symmetry in convolutional neural networks. In *Proceedings of The 33rd International Conference on Machine Learning*, pages 1889–1898, 2016. [2](#), [3](#)
- [12] P. Dollár and C. L. Zitnick. Fast edge detection using structured forests. *IEEE transactions on pattern analysis and machine intelligence*, 37(8):1558–1570, 2015. [6](#)
- [13] J. Driver, G. C. Baylis, and R. D. Rafal. Preserved figure-ground segregation and symmetry perception in visual neglect. *Nature*, 1992. [1](#), [8](#)
- [14] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In *Knowledge Discovery and Data Mining (KDD)*, volume 96, pages 226–231, 1996. [3](#)
- [15] K. Fukushima. Use of non-uniform spatial blur for image comparison: symmetry axis extraction. *Neural Networks*, 18(1):23–32, 2005. [2](#)
- [16] K. Fukushima and M. Kikuchi. Symmetry axis extraction by a neural network. *Neurocomputing*, 69(16):1827–1836, 2006. [2](#)
- [17] C. Funk and Y. Liu. Symmetry reCAPTCHA. In *IEEE Computer Vision and Pattern Recognition (CVPR 2016)*, pages 5165–5174, 2016. [2](#), [3](#), [4](#), [5](#)
- [18] R. Gens and P. M. Domingos. Deep symmetry networks. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2537–2545. Curran Associates, Inc., 2014. [3](#)
- [19] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 580–587, 2014. [2](#)
- [20] M. Giurfa, B. Eichmann, and R. Menzel. Symmetry perception in an insect. *Nature*, 382(6590):458–461, Aug 1996. [1](#)
- [21] T. Hafting, M. Fyhn, S. Molden, M.-B. Moser, and E. I. Moser. Microstructure of a spatial map in the entorhinal cortex. *Nature*, 436(7052):801–806, August 2005. [1](#)
- [22] D. C. Hauagge and N. Snavely. Image matching using local symmetry features. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 206–213. IEEE, 2012. [2](#)
- [23] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. *Computer Vision and Pattern Recognition (CVPR), 2015 IEEE Conference on*, 2016. [4](#)
- [24] M. Holschneider, R. Kronland-Martinet, J. Morlet, and P. Tchamitchian. A real-time algorithm for signal analysis with the help of the wavelet transform. In *Wavelets*, pages 286–297. Springer, 1990. [3](#), [4](#)
- [25] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. In *Proceedings of the 22nd ACM international conference on Multimedia*, pages 675–678. ACM, 2014. [4](#)
- [26] O. Jones (1809-1874). *The grammar of ornament*. London: Quaritch, 1910. [1](#)
- [27] N. Kiryati and Y. Gofman. Detecting Symmetry in Grey Level Images: The Global Optimization Approach. *International Journal of Computer Vision*, 29(1):29–45, 1998. [2](#)
- [28] P. J. Kohler, A. Clarke, A. Yakovleva, Y. Liu, and A. M. Norcia. Representation of Maximally Regular Textures in Human Visual Cortex. *The Journal of Neuroscience*, 36(3):714–729, 2016. [1](#)
- [29] J. L. Krahe. Detection of symmetric and radial structures in images. In *International Conference on Pattern Recognition*, pages 947–950, 1986. [2](#)
- [30] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012. [2](#)
- [31] D. Laptev, N. Savinov, J. M. Buhmann, and M. Pollefeys. Tl-pooling: transformation-invariant pooling for feature learning in convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 289–297, 2016. [2](#), [3](#)
- [32] S. Lee. *Symmetry Group Extraction from Multidimensional Real Data*. PhD thesis, Pennsylvania State University, 2009. [3](#)

- [33] S. Lee, R. Collins, and Y. Liu. Rotation Symmetry Group Detection Via Frequency Analysis of Frieze-Expansions. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008. 3
- [34] S. Lee and Y. Liu. Curved Glide-reflection Symmetry Detection. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 1046–1053, 2009. 2
- [35] S. Lee and Y. Liu. Skewed Rotation Symmetry Group Detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI*, Accepted for publication, 2009. 3
- [36] S. Lee and Y. Liu. Curved Glide-Reflection Symmetry Detection. *IEEE transactions on pattern analysis and machine intelligence*, 34(2):266–278, 2012. 2
- [37] T. Lee, S. Fidler, A. Levinstein, C. Sminchisescu, and S. Dickinson. A Framework for Symmetric Part Detection in Cluttered Scenes. *Symmetry*, 7(3):1333–1351, 2015. 2
- [38] T. S. Levitt. Domain Independent Object Description and Decomposition. *Proceedings of American Association of Artificial Intelligence (AAAI)*, pages 207–211, 1984. 2
- [39] M. Leyton. *Symmetry, Causality, Mind*. Cambridge, Massachusetts. MIT Press, 1992. A Bradford book. 1
- [40] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft COCO: Common Objects in Context. In *European Conference on Computer Vision*, pages 740–755. Springer, 2014. 1, 2, 3, 4, 6, 7
- [41] J. Liu and Y. Liu. Curved Reflection Symmetry Detection with Self-validation. In *Asian Conference on Computer Vision*, pages 102–114. Springer, 2010. 2
- [42] J. Liu, G. Slota, G. Zheng, Z. Wu, M. Park, S. Lee, I. Rauschert, and Y. Liu. Symmetry Detection from RealWorld Images Competition 2013: Summary and Results. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2013 IEEE Conference on*, pages 200–205. IEEE, 2013. 1, 6, 7
- [43] Y. Liu, R. Collins, and Y. Tsin. A computational model for periodic pattern perception based on frieze and wallpaper groups. *IEEE Transaction on Pattern Analysis and Machine Intelligence*, 26(3):354–371, March 2004. 2
- [44] Y. Liu, H. Hel-Or, C. Kaplan, and L. Van Gool. Computational symmetry in computer vision and computer graphics: A survey. *Foundations and Trends in Computer Graphics and Vision*, 5(1-2):1–199, 2010. 1, 2
- [45] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3431–3440, 2015. 2, 3, 4
- [46] G. Loy and J.-O. Eklundh. Detecting Symmetry and Symmetric Constellations of Features. In *Computer Vision – ECCV 2006: 9th European Conference on Computer Vision, Graz, Austria, May 7-13, 2006. Proceedings, Part II*, pages 508–521. Springer Berlin Heidelberg, Berlin, Heidelberg, 2006. 2, 5, 6, 7
- [47] G. Marola. Using Symmetry for Detecting and Locating Objects in a Picture. *Computer Vision, Graphics, and Image Processing*, 46(2):179–195, 1989. 2
- [48] D. R. Martin, C. C. Fowlkes, and J. Malik. Learning to detect natural image boundaries using local brightness, color, and texture cues. *IEEE transactions on pattern analysis and machine intelligence*, 26(5):530–549, 2004. 5
- [49] M. Mignotte. Symmetry detection based on multiscale pairwise texture boundary segment interactions. *Pattern Recognition Letters*, 74:53–60, 2016. 2
- [50] H. Ogawa. Symmetry analysis of line drawings using the Hough transform. *Pattern Recognition Letters*, 12(1):9–12, 1991. 2
- [51] G. Papandreou, I. Kokkinos, and P.-A. Savalle. Modeling local and global deformations in deep learning: Epitomic convolution, multiple instance learning, and sliding window detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 390–399, 2015. 3, 4
- [52] G. Parovel and S. Vezzani. Mirror symmetry opposes splitting of chromatically homogeneous surfaces. *Perception*, 31(6):693–709, 2002. 1, 8
- [53] T. Pfister, J. Charles, and A. Zisserman. Flowing convnets for human pose estimation in videos. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1913–1921, 2015. 2, 3, 4
- [54] J. Pritts, O. Chum, and J. Matas. Detection, rectification and segmentation of coplanar repeated patterns. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2973–2980, 2014. 2
- [55] I. Rauschert, J. Liu, K. Brockelhurst, S. Kashyap, and Y. Liu. Symmetry detection competition: A summary of how the competition is carried out. In *IEEE Conference on Computer Vision and Pattern Recognition Workshop, Symmetry Detection Real World Images*, pages 1–66, 2011. 1, 6
- [56] I. Rodríguez, A. Gumbert, N. Hempel de Ibarra, J. Kunze, and M. Giurfa. Symmetry is in the eye of the ‘beeholder’: innate preference for bilateral symmetry in flower-naïve bumblebees. *Naturwissenschaften*, 91(8):374–377, Aug 2004. 1
- [57] J. E. Scheib, S. W. Gangestad, and R. Thornhill. Facial attractiveness, symmetry and cues of good genes. *Proceedings of the Royal Society of London B: Biological Sciences*, 266(1431):1913–1917, 1999. 1, 8
- [58] W. Shen, X. Bai, Z. Hu, and Z. Zhang. Multiple instance subspace learning via partial random projection tree for local reflection symmetry in natural images. *Pattern Recognition*, 52:306–316, 2016. 2
- [59] W. Shen, K. Zhao, Y. Jiang, Y. Wang, Z. Zhang, and X. Bai. Object Skeleton Extraction in Natural Images by Fusing Scale-Associated Deep Side Outputs. *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 2, 5
- [60] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 2, 4
- [61] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, and A. Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–9, 2015. 2

- [62] Y. Taigman, M. Yang, M. Ranzato, and L. Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1701–1708, 2014. 2
- [63] C. L. Teo, C. Fermüller, and Y. Aloimonos. Detection and Segmentation of 2D Curved Reflection Symmetric Structures. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1644–1652, 2015. 2, 5
- [64] D. W. Thompson. *On Growth and Form*. Cambridge University Press, 1961. 1
- [65] M. S. Treder. Behind the looking-glass: A review on human symmetry perception. *Symmetry*, 2(3):1510–1543, 2010. 1
- [66] W.-H. Tsai and S.-L. Chou. Detection of generalized principal axes in rotationally symmetric shapes. *Pattern Recognition*, 24(2):95–104, 1991. 2
- [67] S. Tsogkas and I. Kokkinos. Learning-based symmetry detection in natural images. In *European Conference on Computer Vision*, pages 41–54. Springer, 2012. 2, 5
- [68] T. Tuytelaars, A. Turina, and L. Van Gool. Non-combinatorial detection of regular repetitions under perspective skew. *IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI*, 25(4):418–432, 2003. 2
- [69] C. Tyler. *Human Symmetry Perception and Its Computational Analysis*. Lawrence Erlbaum Associates, Publishers, Mahwah, New Jersey, 2002. 1, 2
- [70] C. W. Tyler (Ed.). *Human Symmetry Perception and its Computational Analysis*. VSP, Utrecht, The Netherlands, 1996. 1
- [71] L. von Fersen, C. S. Manos, B. Goldowsky, and H. Roitblat. Dolphin detection and conceptualization of symmetry. In *Marine mammal sensory systems*, pages 753–762. Springer, 1992. 1
- [72] X. Wang, D. Fouhey, and A. Gupta. Designing deep networks for surface normal estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 539–547, 2015. 2
- [73] Z. Wang, Z. Tang, and X. Zhang. Reflection Symmetry Detection Using Locally Affine Invariant Edge Correspondence. *IEEE Transactions on Image Processing*, 24(4):1297–1301, 2015. 2
- [74] S.-E. Wei, V. Ramakrishna, T. Kanade, and Y. Sheikh. Convolutional pose machines. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4724–4732, 2016. 2, 3, 4
- [75] H. Weyl. *Symmetry*. Princeton University Press, Princeton, New Jersey, 1952. 1
- [76] N. Widynski, A. Moevus, and M. Mignotte. Local symmetry detection in natural images using a particle filtering approach. *IEEE Transactions on Image Processing*, 23(12):5309–5322, 2014. 2
- [77] D. E. Worrall, S. J. Garbin, D. Turmukhambetov, and G. J. Brostow. Harmonic networks: Deep translation and rotation equivariance. *arXiv preprint arXiv:1612.04642*, 2016. 2, 3
- [78] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson. How transferable are features in deep neural networks? In *Advances in neural information processing systems*, pages 3320–3328, 2014. 4
- [79] T. Zielke, M. Brauckmann, and W. von Seelen. Intensity and Edge-Based Symmetry Detection Applied to Car-Following. In *European Conference on Computer Vision*, pages 865–873. Springer, 1992. 2