

Bilateral symmetry of object silhouettes under perspective projection

K. Bitsakos, H. Yi, L. Yi, and C. Fermuller
Center for Automation Research, University of Maryland
{kbits, aster}@cs.umd.edu, {liyi, fer}@cfar.umd.edu

Abstract

Symmetry is an important property of objects and is exhibited in different forms e.g., bilateral, rotational, etc. This paper presents an algorithm for computing the bilateral symmetry of silhouettes of shallow objects under perspective distortion, exploiting the invariance of the cross ratio to projective transformations. The basic idea is to use the cross ratio to compute a number of midpoints of cross sections and then fit a straight line through them. The goodness-of-fit determines the likelihood of the line to be the axis of symmetry. We analytically estimate the midpoint's location as a function of the vanishing point for a given object silhouette. Hence finding the symmetry axis amounts to a 2D search in the space of vanishing points. We present experiments on two datasets as well as internet images of symmetric objects that validate our approach.

1 Introduction

An object is reflectively symmetric if it can be divided into two mirror-identical halves. Reflective symmetry is considered an important geometric property, thus a lot of research has been devoted to finding this salient structure [2, 10]. Once found, the symmetric structure of an object can be exploited in many ways e.g., in image compression [4], shape representation [17], model based matching [12], object alignment and reconstruction [14, 16].

The symmetry of an image, however, is hard to detect. One reason is that real objects often are not perfectly symmetric and a small amount of perturbation may significantly affect the detection of symmetry. Even if an object is perfectly symmetric in 3D, its 2D image under perspective projection might not be. Lighting variations only serve to aggravate the situation.

This paper focuses on computing bilateral symmetries of silhouettes of shallow objects under perspective distortion. Exploiting the invariance of the cross ratio

under perspectivities, we analytically compute a set of midpoints of the object as a function of the vanishing point. Then, we fit a straight line passing through the midpoints. The goodness-of-fit defines the likelihood of this line to be a symmetry axis. Using the proposed method, searching for the symmetry axis is reduced to searching for a vanishing point. Our approach is global in the sense that we consider the whole silhouette of the object rather than small parts of it. The results show that the method presented here is capable of finding axes of symmetry of considerably distorted perspective images.

2 Related Work

Much prior work computes symmetry in different settings with various assumptions. The generalized symmetry transform [13] provides a bilateral symmetry edge map. For real time applications, a fast radial symmetry transform [9] and a modified version of the generalized symmetry transform [7] were proposed to find both reflective and rotational symmetry. Feature based methods that use image primitives, such as corner points, lines and curves to compute pairwise symmetry, also exist [8, 11]. In [5], a frequency based method to find local symmetry on intensity images, was presented. Most of the research assumes that the input image contains affinely transformed symmetric objects, and only a few approaches [15] explicitly handle perspective distortion.

To the best of our knowledge none of the previous approaches exploits the invariance of the cross ratio [3] under perspective distortion to detect bilateral symmetries. The cross ratio allows the detection of bilateral symmetries in distorted planar objects.

3 3D symmetry under perspective projection

3.1 Problem statement and notation

Let us assume that there is a symmetric planar object O_{sym} with an axis of symmetry L_S . Our goal is to find the axis of symmetry. We denote L_i lines that are perpendicular to L_S . Both L_S and L_i are expressed in homogeneous coordinates (i.e., a column vector with 3 elements). Depending on the context, the same symbols are used for the 3D lines and their 2D projection on the image plane $I : z = 1$. In the rest of the paper, we commonly use the following symbols:

- P_V is the vanishing point, i.e., the point of intersection between any pair of lines L_i . Note that L_i are parallel by virtue of being perpendicular to L_S .
- P_A, P_B denote the intersections of a line L_i with the boundary of the object O_{sym} . P_B is the closest point to P_V , while P_A is the furthest away.
- P_M is the midpoint of P_A and P_B . By definition P_M is also the intersection of L_i with L_S , so it belongs to the symmetry axis.
- the cross ratio of four points is denoted as

$$\begin{aligned} \lambda(P_A, P_B, P_C, P_D) &= \frac{|\overrightarrow{P_C - P_A}|}{|\overrightarrow{P_C - P_B}|} : \frac{|\overrightarrow{P_D - P_A}|}{|\overrightarrow{P_D - P_B}|} \\ &= \frac{|\overrightarrow{AC}|}{|\overrightarrow{BC}|} : \frac{|\overrightarrow{AD}|}{|\overrightarrow{BD}|}, \end{aligned} \quad (1)$$

where $|\overrightarrow{P_C - P_A}| = |\overrightarrow{AC}|$ etc, is the signed distance between points C and A . Note that the order of the points in the cross ratio matters.

3.2 Computing the midpoint under perspective projection

If we consider a line L_i perpendicular to the axis of symmetry L_S , then the point P_M of the intersection $L_i \times L_S$ is between the boundary points P_A and P_B and has equal distance from both. The cross ratio of points P_A, P_B, P_M , and the vanishing point P_V is 2.

Proof. Since the cross ratio λ is preserved under any projective transformation, we consider the vanishing point to be at infinity $P_V = P_\infty$, i.e., the object is frontoparallel. In this case,

$$\lambda(P_A, P_M, P_B, P_\infty) = \frac{|\overrightarrow{P_B - P_A}| |\overrightarrow{P_\infty - P_M}|}{|\overrightarrow{P_B - P_M}| |\overrightarrow{P_\infty - P_A}|}. \quad (2)$$

If we denote the distance $|\overrightarrow{P_\infty - P_M}| = x$ and consider the limit $x \rightarrow \infty$ we have

$$\begin{aligned} \lambda(P_A, P_M, P_B, P_\infty) &= \lim_{x \rightarrow \infty} \frac{|\overrightarrow{AB}|x}{|\overrightarrow{MB}|(x + |\overrightarrow{AB}|)} \\ &= \frac{|\overrightarrow{AB}|}{|\overrightarrow{MB}|} = 2 \end{aligned} \quad (3)$$

□

In the general case the point at infinity P_∞ maps to the vanishing point P_V and the parallel lines connecting the boundary points P_A and P_B pass through the vanishing point. If we have the vanishing point we can compute the midpoint along a single line using two equations

$$\begin{aligned} L_i^T P_M &= 0 \\ \lambda(P_A, P_M, P_B, P_V) &= 2, \end{aligned} \quad (4)$$

where $L_i = P_A \times P_B$. If we denote with d_M, d_B and d_V the distance of points P_M, P_B and P_V from point P_A respectively, then

$$\lambda(P_A, P_M, P_B, P_V) = \frac{d_B(d_V - d_M)}{(d_B - d_M)d_V} \quad (5)$$

From Eqs. 4, 5 we find the distance of the midpoint from the boundary point P_A

$$d_M = \frac{d_B d_V}{2d_V - d_B}. \quad (6)$$

3.3 The Algorithm

Based on the observation of the previous section detecting the symmetry axis is reduced to finding the “best” vanishing point. In Alg. 1 we present the approach we employ to find the vanishing point (and consequently the symmetry axis). We use a least square based line fitting algorithm and the goodness-of-fit (GoF) is the average of the squared distance of the midpoints from the line. The line with the smallest value is the best candidate for the symmetry axis L_S .

4 Experiments

In the experiments we use two different datasets: the MPEG-7 dataset [6] and the Smithsonian Leaf dataset [1]. We also present results on images of symmetric objects found on the internet. In all the experiments we use 17 different radii and 360 orientations for a total of 6120 potential vanishing points. We also used 75 interior points randomly selected.

Algorithm 1 Find the bilateral symmetry of object silhouette

1. Sample the $2D$ plane for candidate vanishing point
 2. For each candidate vanishing point P_V
 - (a) Randomly select a fixed number of points P_i inside the object
 - (b) For each P_i
 - i. Find the line $L_i = P_i \times P_V$
 - ii. Find the intersection points P_A, P_B of L_i with the object boundary
 - iii. Compute the midpoint P_M^i using the cross ratio as described in Sec. 3.2
 3. Fit the least square line L_S through all the midpoints P_M^i and compute the average Euclidean distance of the points from the line
 4. Select the vanishing point corresponding to the best fitted line L_S . L_S is the axis of symmetry for the object
-

Figs. 1 and 2 display images from the MPEG-7 and the Leaf dataset, respectively, along with the axes of symmetry. Below the image we also display the goodness-of-fit value GoF (i.e. average Euclidean distance of the calculated midpoints from the symmetry axis expressed in pixels) and the distance of the vanishing point from the image center $|P_V|$ (in pixels). The smaller the distance of the vanishing point is (e.g., Fig. 1b), the more perspectively distorted the object is. Some images of the Leaf dataset (e.g., Fig. 2f) are symmetric, but the axis of symmetry is a curve and not a straight line. In these cases the proposed method is not able to provide good results. Still, it is possible to identify these cases using the GoF measure. The higher this value is, the more likely the detected symmetry axis is wrong.

Fig. 3 displays our symmetry detection results after perspectively distorting the images of the MPEG-7 dataset. To perform the perspective distortion, we rotate the images about the vertical axis and the horizontal axis and reproject all the points on the image plane.

We also performed experiments on symmetric object images collected from the internet. We used a color based segmentation to locate the objects and then applied our algorithm on the segmented image. Some of the results are displayed in Fig. 4.

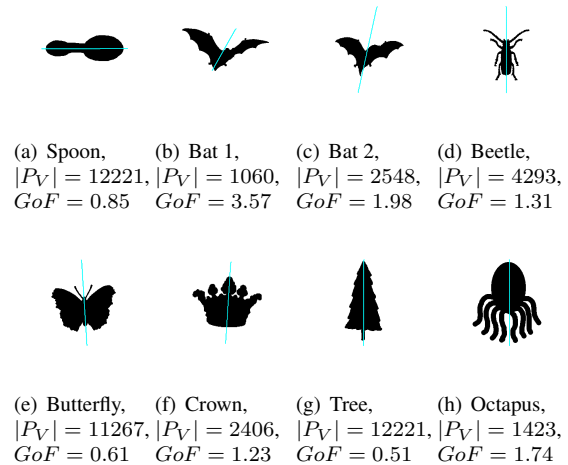


Figure 1. Symmetry detection for images of the MPEG-7 dataset.

5 Conclusions and Future Work

In this paper we present a framework for detecting bilateral symmetries in planar $3D$ objects viewed under perspective projection. The method exploits the invariance of the cross ratio under perspective transformations. Using this framework the search for $3D$ symmetries is reduced to a $2D$ search for the vanishing point of the lines perpendicular to the axis of symmetry. While $2D$ search is computationally expensive, there are ways (e.g., hierarchical search, gradient descent) to reduce the search space, which we will explore in the future.

References

- [1] G. Agarwal, P. Belhumeur, S. Feiner, D. Jacobs, W. J. Kress, R. Ramamoorthi, N. A. Bourg, N. Dixit, H. Ling, D. Mahajan, R. Russell, S. Shirdhonkar, K. Sunkavalli, and S. White. First steps toward an electronic field guide for plants. *Taxon*, 55(3):597–610, 2006.
- [2] M. Atallah. On symmetry detection. *IEEE Transactions on Computers*, pages 663–666, 1985.
- [3] R. Hartley and A. Zisserman. *Multiple View Geometry*. Cambridge Univ. Press.
- [4] M. Kirby and L. Sirovich. Application of the karhunen-loeve procedure for the characterization of human faces. In *PAMI*, volume 12, pages 103–108, 1990.
- [5] P. Kovese. Symmetry and asymmetry from local phase. In *Australian Joint Conference on Artificial Intelligence*, pages 2–4, 1997.
- [6] L. J. Latecki, R. Lakämper, and U. Eckhardt. Shape descriptors for non-rigid shapes with a single closed contour. In *CVPR*, pages 424–429, 2000.

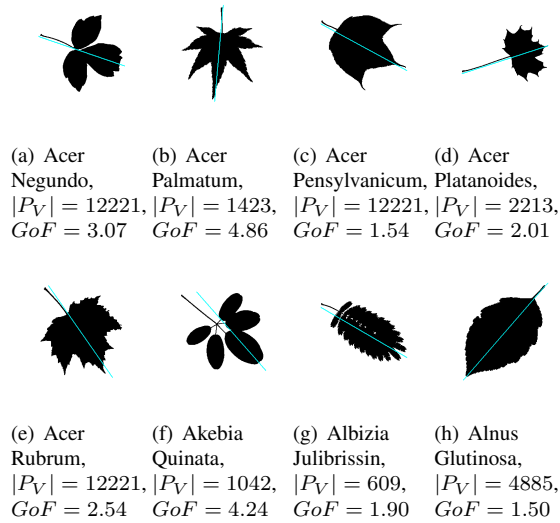


Figure 2. Symmetry detection for images of the Leaf dataset.

[7] W. Li, A. Zhang, and L. Kleeman. Real time detection and segmentation of reflectionally symmetric objects in digital images. In *IROS*, 2006.

[8] G. Loy and J. Eklundh. Detecting symmetry and symmetric constellations of features. *ECCV*, 2006.

[9] G. Loy and A. Zelinsky. Fast radial symmetry for detecting points of interest. In *PAMI*, pages 959–973, 2003.

[10] G. Marola. On the detection of the axes of symmetry of symmetric and almost symmetric planar images. In *PAMI*, volume 11, pages 104–108, 1989.

[11] D. Mukherjee, A. Zisserman, and M. Brady. Shape from symmetry detecting and exploiting symmetry in affine images. In *Philosophical Transaction of Royal Society of London*, number 351, pages 77–101, 1995.

[12] W. Oh, M. Asada, and S. Tsuji. Model-based matching using skewed symmetry information. In *ICPR*, volume 2, pages 1043–1045, 1988.

[13] D. Reisfeld, H. Wolfson, and Y. Yeshurun. Context-free attentional operators: The generalized symmetry transform. In *IJCV*, pages 119–130, 1999.

[14] D. Terzopoulos, A. Witkin, and M. Kass. Symmetry-seeking models and 3d object reconstruction. In *IJCV*, volume 1, pages 211–221, 1987.

[15] S. Utcke. Comparison of different approaches for the calculation of projective symmetry or the axis of a SHGC, bericht. fbi-hh-253/03. 2003.

[16] A. Yang, S. Rao, K. Huang, W. Hong, and Y. Ma. Symmetry-based 3-d reconstruction from perspective images. *CVIU*, 2005.

[17] H. Zabrodsky, S. Peleg, and D. Avnir. A measure of symmetry based on shape similarity. In *CVPR*, pages 703–706, 1992.

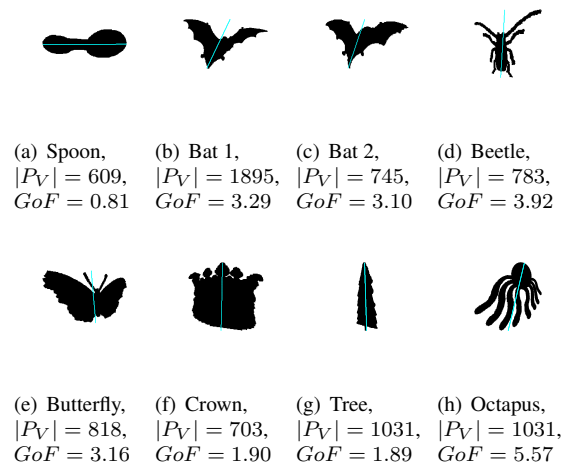


Figure 3. Symmetry detection for perspectively distorted images of the MPEG-7 dataset.

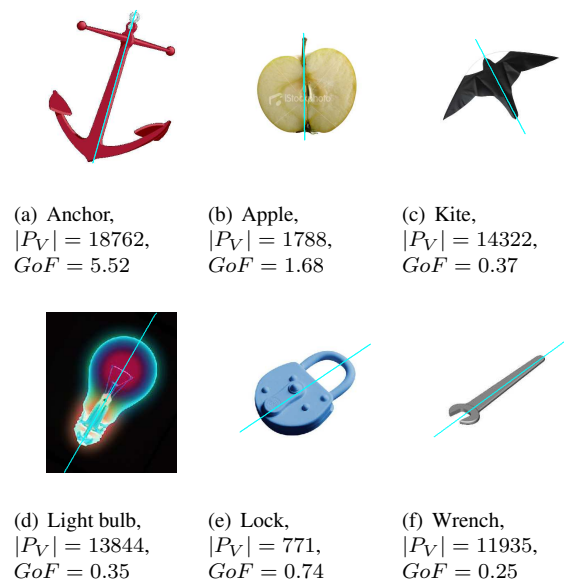


Figure 4. Symmetry detection for internet images