

Detecting Rotational Symmetries

V. Shiv Naga Prasad and Larry S. Davis
Computer Vision Lab,

University of Maryland Institute for Advanced Computer Studies (UMIACS),
University of Maryland - College Park, College Park, MD 20742, USA.

Abstract

We present an algorithm for detecting multiple rotational symmetries in natural images. Given an image, its gradient magnitude field is computed, and information from the gradients is spread using a diffusion process in the form of a Gradient Vector Flow (GVF) field. We construct a graph whose nodes correspond to pixels in the image, connecting points that are likely to be rotated versions of one another. The n -cycles present in the graph are made to vote for C_n symmetries, their votes being weighted by the errors in transformation between GVF in the neighborhood of the voting points, and the irregularity of the n -sided polygons formed by the voters. The votes are accumulated at the centroids of possible rotational symmetries, generating a confidence map for each order of symmetry. We tested the method with several natural images.

1. Introduction

Symmetry is an important element of perceptual grouping [1]. The reason is that image structures are unlikely to be arranged in such a manner as to produce spatial symmetry by chance. It is much more likely that if a group of image structures exhibits symmetry, then they are physically related, and so are perceived in relation to one another (i.e., they are grouped together perceptually). This has motivated research on the detection of symmetries in images.

There are three types of symmetries [22]:

1. Bilateral symmetry: A figure is said to possess bilateral symmetry if it is invariant under a reflection about a line - called the axis of symmetry - passing through the centroid of the figure [7, 4, 13, 6, 16].
2. Rotational symmetry: A figure is said to possess rotational symmetry of order n (C_n), if it is invariant under rotations of $\frac{2\pi}{n}$ radians about its center of mass [6, 12, 8, 22]. Central symmetry is a special case of rotational symmetry with $n = 2$ [2, 3].
3. Radial symmetry: A figure is said to possess radial symmetry if it has both bilateral and rotational symmetry [10, 21]. Radial symmetry is often used to detect points of interest in images.

Figure 1 shows examples of the three types of symmetries. When symmetric objects undergo skew transformation they produce skew symmetries [9, 15, 11].

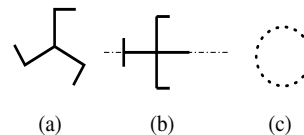


Figure 1. Example of different types of symmetries. (a) Rotational symmetry of order 3 (C_3), (b) Bilateral symmetry, and (c) Radial symmetry.

Perfect symmetries are rare in nature. Therefore, for robust detection of symmetry we must allow for deviations from the ideal; this is done by assigning continuous values to the degree of symmetry exhibited by structures in the image. Zabrodsky et.al. [22] discuss this issue in the context of rotational symmetries.

The present work addresses the problem of efficient detection of rotational symmetries in images. The following are some desirable properties of a symmetry detector:

1. Detect multiple symmetries in unsegmented natural images.
2. Computationally efficient.
3. Robust to edge breaks, occlusions etc.
4. Allow for deformations from perfect symmetry.

Zabrodsky et.al. [22] describe an approach for quantifying the degree of rotational symmetry exhibited by a figure present in an image. The figure is assumed to be specified as a sequence of points along its outline. A Symmetry Distance (SD) is defined to measure symmetry in the presence

of occlusion and deformation. Our objective is to simultaneously detect multiple symmetries present in natural images. Therefore, we would like to avoid the difficult task of generating sequences of points lying on the many contours that might be present in an image and then verifying their symmetry individually. Our algorithm uses voting to detect multiple symmetries, and trades off false detection of symmetries for efficiency. We conjecture that the foci of symmetry detected by our method can be used to obtain the outlines of the figures putatively exhibiting symmetry, which can then be analyzed using methods like [22].

There are two generic ways of characterizing structures present in images: by directly analyzing the gradient contours, and by analyzing the regions enclosed by the contours. Local edge information is sensitive to edge breaks, and often the edge segments might be too small to support inference about global structure. In contrast, region-based approaches are grounded on the observation that contours present in images “mould” the space near and around them, and it is possible to obtain information about the contour by analyzing the nature of this space. There are several ways of characterizing the space around contours, e.g. distance maps, magnetic fields [17], Gradient Vector Flow (GVF) fields [20], Hamilton-Jacobi transform [18], etc. However, region-based approaches are limited by the assumption of closed and connected contours to bound the regions. Our approach is to adopt a middle course between contours and regions by extracting local features of the space near and around contours and then collating these features to draw global inferences.

Given a color image, we compute the gradient magnitude field, and spread the information from the edges using GVF fields. The GVF in the image plane is represented using local features in the form of Taylor coefficients. Next, a graph with the pixels as nodes is constructed such that pairs of points having a high likelihood of being rotated versions of one another are connected by edges. The n -cycles present in the graph are made to vote for C_n symmetry, the votes being weighted by the errors in the transformation between the voting points. For each order of rotational symmetry, votes for rotation about a centroid are aggregated to obtain confidence maps on the image plane. Centroids of possible rotationally symmetric objects exhibit peaks in the confidence maps. Figure 2 shows an example of a test image, the corresponding GVF field, the confidence map for C_5 symmetry, and the locations of the peaks marked on the image. In our implementation, symmetries up to order C_8 are considered. The approach was tested with several natural images of flowers in frontal view obtained from [14, 19, 5]. The results are described later in the paper.

2. Characterizing the Space Around Gradients

The gradients in an image are a good source of information when searching for structures such as rotational symmetries. The reason is that in contrast to regions with uniform image features, gradients have orientation and give direct shape information. The problem with directly using gradient contour information is that discontinuities due to noise and illumination make the local orientation and shape information ambiguous. Consider the case of a line formed of dots. The orientation of such a line cannot be characterized by considering individual dots. The characteristics of the line emerge only when information from a number of dots is aggregated. Diffusion schemes such as magnetic fields [17], GVF [20], offer approaches to accomplish this. Here, GVF fields are employed for characterizing the image’s gradients.

Let f denote the gradient magnitude field for an image I , computed using the hue (I_h) and saturation (I_s) channels of the image. The hue values are cyclic in the range $[0,1]$, and the saturation values lie in the range $[0,1]$.

$$f = \sqrt{\|\Delta I_h\|^2 + \|\Delta I_s\|^2} \quad (1)$$

The GVF field \mathbf{v} at point \mathbf{x} is a vector $[u(\mathbf{x}), v(\mathbf{x})]$, where u and v are fields defined on the image plane. The reader is referred to [20, 16] for a detailed discussion of GVFs. Let \mathbf{p}_i and \mathbf{p}_j be two points on the image plane. Let $u_i = u(\mathbf{p}_i)$, $u_{ix} = \frac{\partial u}{\partial x} \Big|_{\mathbf{p}_i}$ and $u_{iy} = \frac{\partial u}{\partial y} \Big|_{\mathbf{p}_i}$. Similarly, we define v_i , u_j , v_j , v_{ix} , v_{iy} , u_{jx} , u_{jy} , v_{jx} and v_{jy} . The GVF at a point is characterized using coefficients of the Taylor series expansions of $u(\cdot)$ and $v(\cdot)$. Only Taylor coefficients up to 1st order are considered. Rotation constraints are imposed on the Taylor series expansion of the GVF in the neighborhood of \mathbf{p}_i and \mathbf{p}_j . The following are necessary conditions for the GVF field in the neighborhood of \mathbf{p}_i to be a rotation of the GVF field in the neighborhood of \mathbf{p}_j about \mathbf{p}_j by an angle θ :

$$u_i = u_j \cos \theta - v_j \sin \theta + d_1 \quad (2)$$

$$u_{iy} = u_{jy} \cos^2 \theta + (u_{jx} - v_{jy}) \sin \theta \cos \theta - v_{jx} \sin^2 \theta + d_2 \quad (3)$$

$$u_{ix} = u_{jx} \cos^2 \theta - (v_{jx} + u_{jy}) \sin \theta \cos \theta + v_{jy} \sin^2 \theta + d_3 \quad (4)$$

$$v_i = +u_j \sin \theta + v_j \cos \theta + d_4 \quad (5)$$

$$v_{iy} = u_{jx} \sin^2 \theta + (v_{jx} + u_{jy}) \sin \theta \cos \theta + v_{jy} \cos^2 \theta + d_5 \quad (6)$$

$$v_{ix} = -u_{jy} \sin^2 \theta + (u_{jx} - v_{jy}) \sin \theta \cos \theta + v_{jx} \cos^2 \theta + d_6 \quad (7)$$

where $d_1 \dots d_6$ are the error terms in the relations (2).. (7) respectively. Let $e_1(\langle i, j \rangle) = \frac{d_1^2 + d_4^2}{2}$ be the error in predict-

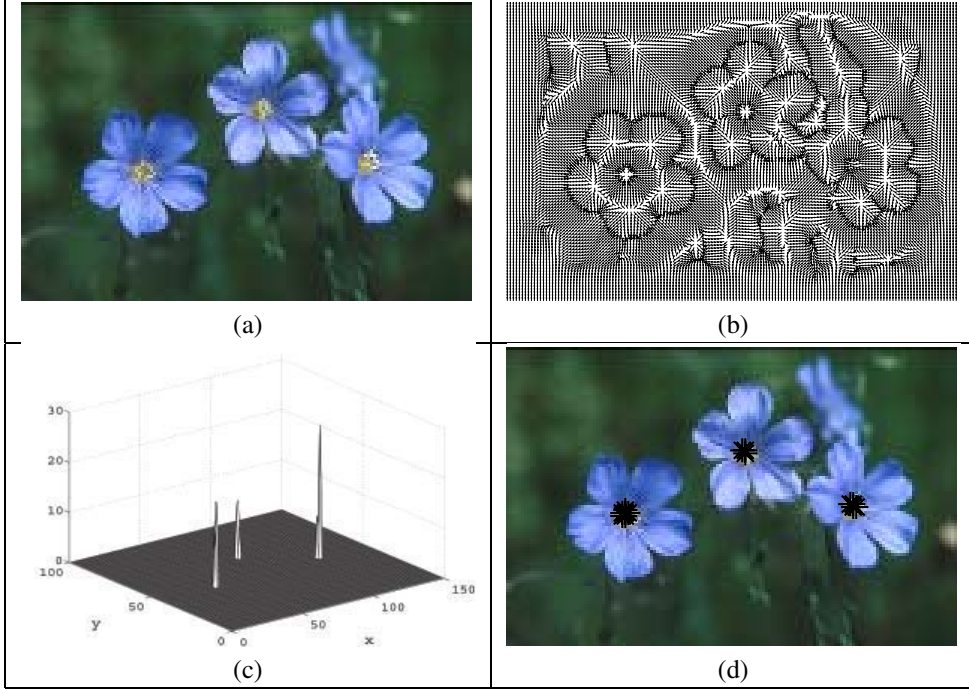


Figure 2. Example of the working system: (a) image, (b) GVF field, (c) confidence map for C_5 symmetry generated by the proposed algorithm, (d) locations of the peaks in the confidence map marked on the image.

ing u and v , and $e_2 = \frac{d_2^2 + d_3^2 + d_5^2 + d_6^2}{4}$, the error in predicting the partial derivatives. These will be used to weigh the votes for rotational symmetries.

Let $V(\mathbf{p}) = u^2(\mathbf{p}) + v^2(\mathbf{p})$, $h_1(\mathbf{p}) = u_x(\mathbf{p}) + v_y(\mathbf{p})$ and $h_2(\mathbf{p}) = u_y(\mathbf{p}) - v_x(\mathbf{p})$. Using (2)...(7), it can be shown that the following are necessary but not sufficient conditions for p_i to be a perfectly rotated version of p_j ($d_1 = d_2 = \dots = d_6 = 0$):

$$\begin{aligned} \mathcal{C}_1 : V(\mathbf{p}_i) &= V(\mathbf{p}_j) \\ \mathcal{C}_2 : h_1(\mathbf{p}_i) &= h_1(\mathbf{p}_j) \\ \mathcal{C}_3 : h_2(\mathbf{p}_i) &= h_2(\mathbf{p}_j) \end{aligned}$$

These will be used later to reduce the amount of computation during voting. Another local feature extracted is the curvature of the GVF, denoted by $C(\mathbf{p})$ [16].

$$C(\mathbf{p}_i) = \frac{1}{\|\mathbf{v}(\mathbf{p}_i)\|^3} [(v_{ix} + u_{iy})u_i v_i - u_{ix} v_i^2 - v_{iy} u_i^2] \quad (8)$$

For two points \mathbf{p}_i and \mathbf{p}_j to have GVF fields which are rotations of one another, the condition $\mathcal{C}_4 : C(\mathbf{p}_i) = C(\mathbf{p}_j)$ must hold.

3. Constructing the Voter's Graph and Collating the Votes

Having extracted local features from the GVF and defined the conditions for rotation, the next step is to collect the voters for the possible rotational symmetries. For C_n rotational symmetry, there must exist an n -tuple of points in the image such that the GVFs at the points are rotations of one another, the colors at the points are similar, and the points lie on a regular n -sided polygon. We proceed by constructing a graph \mathcal{G} with the points on the image plane as the nodes; each node q_i in \mathcal{G} would correspond to point \mathbf{p}_i in the image. In order to reduce the computational complexity of the voting, only points with GVF magnitude, $\|\mathbf{v}\|$, greater than a threshold Δ_G are considered. Two nodes q_i and q_j are connected by an edge if the following conditions are satisfied:

$$\begin{aligned} \mathcal{D} : (V(\mathbf{p}_i) - V(\mathbf{p}_j))^2 &< \Delta_V, \\ (h_1(\mathbf{p}_i) - h_1(\mathbf{p}_j))^2 &< \Delta_V, \\ (h_2(\mathbf{p}_i) - h_2(\mathbf{p}_j))^2 &< \Delta_V, \\ (C(\mathbf{p}_i) - C(\mathbf{p}_j))^2 &< \Delta_V, \text{ and} \\ h(I_h(\mathbf{p}_i), I_h(\mathbf{p}_j))^2 &< \Delta_h. \end{aligned} \quad (9)$$

where

$$h(c_1, c_2) = \begin{cases} \min(c_1 - c_2, 1 + c_2 - c_1) & : c_1 \geq c_2 \\ \min(c_2 - c_1, 1 + c_1 - c_2) & : \text{otherwise} \end{cases} \quad (10)$$

The thresholds (Δ 's) are used to reduce the number of edges in \mathcal{G} . Their magnitudes depend upon the scales of values taken by u , v and their partial derivatives - which do not vary much for natural images. Therefore, the symmetry detection is not sensitive to their values.

The votes for possible rotational symmetries are computed by searching for n -cycles in \mathcal{G} . This is accomplished for each C_n symmetry ($n = 2 \dots 8$) using an iterative distributed token passing algorithm described below.

At any given iteration t , each node q_i in the graph has a set denoted by $S_{in}^{(t)}$ which contains lists of node indices. Each list contains the indices of voters so far involved in a particular vote. At the t^{th} iteration, each node q_i propagates the lists in $S_{in}^{(t)}$ to its neighbors in the graph ($q_j \in N(q_i)$), i.e., to sets $S_{jn}^{(t+1)}$. During propagation, the destination node's index is appended to each list. A list is propagated depending upon whether $\mathbf{v}(\mathbf{p}_j)$ is a rotation by angle $\frac{2\pi}{n}$ of $\mathbf{v}(\mathbf{p}_i)$, and whether \mathbf{p}_j can form the next vertex of the n -sided polygon which will be formed by the voters present in the list. For any list $l = \langle i, b, \dots, c \rangle$ at node q_i , c is the first voter's index, and i is the last voter's index appended to l . The gradient orientation is denoted by $\psi(\mathbf{p}) = \tan^{-1} \frac{v(\mathbf{p})}{u(\mathbf{p})}$. The rotation matrix R_n , is defined as

$$R_n = \begin{bmatrix} \cos(\frac{2\pi}{n} - \pi) & -\sin(\frac{2\pi}{n} - \pi) \\ \sin(\frac{2\pi}{n} - \pi) & \cos(\frac{2\pi}{n} - \pi) \end{bmatrix}$$

For q_i to propagate list $l = \langle i, b, \dots, c \rangle$ to q_j , it must satisfy the condition

$$(\psi(\mathbf{p}_j) - \psi(\mathbf{p}_i)) \in [\frac{2\pi}{n} - \Delta_\psi, \frac{2\pi}{n} + \Delta_\psi] \wedge (\|(\mathbf{p}_j - \mathbf{p}_i) - R_n(\mathbf{p}_b - \mathbf{p}_i)\| < \Delta_d) \quad (11)$$

All lists of voters for C_n symmetry are forced to return to their initiating nodes at the end of n iterations, ensuring the completion of n -cycles. The lists are initialized as $S_{in}^{(0)} = \{\langle i \rangle\} \forall i$.

This algorithm is applied for each C_n symmetry. After its execution, the sets $S_{in}^{(n)}$ of different nodes (q_i 's) will contain lists of voters for C_n rotational symmetries. The votes are aggregated at the centroids of the n -sided polygons formed by the voters (which would be the centroids of the estimated symmetries) by analyzing the lists in $S_{in}^{(n)}$'s. Let $l = \langle a_1, a_k, a_{k-1}, \dots, a_1 \rangle$ be a list in $S_{in}^{(n)}$. The error due to the incompatibility of the GVFs at the voting points is computed as

$$e_p = \sum_{t=1}^n \exp[\frac{-1}{\sigma_p} (e_1(\langle a_{t\oplus 1}, a_t \rangle) + e_2(\langle a_{t\oplus 1}, a_t \rangle))] \min(V(\mathbf{p}_{a_{t\oplus 1}}), V(\mathbf{p}_{a_t}))$$

The operator ' \oplus ' is defined as follows: $t_1 \oplus t_2 = (t_1 + t_2) \bmod n$. The $\min(\cdot)$ factor in the summation gives more weight to pairs of points having significant magnitude of GVF field. The error due to irregularity of the k -sided polygon is computed as

$$e_d = \sum_{t=1}^n \|(\mathbf{p}_{t\oplus 2} - \mathbf{p}_{t\oplus 1}) - R_n(\mathbf{p}_{t\oplus 1} - \mathbf{p}_t)\| \quad (12)$$

The centroid of the n -sided polygon is given by $\tilde{\mathbf{p}} = \frac{1}{n} \sum_{t=1}^n \mathbf{p}_{a_t}$. The confidence map for C_n rotational symmetry is denoted by H_n . The vote from list l in set $S_{in}^{(n)}$ is accumulated to H_n as

$$H_n(\tilde{\mathbf{p}}) = H_n(\mathbf{p}) + e_p \exp(\frac{-e_d^2}{n\sigma_d}) \quad (13)$$

See Algorithms 1 and 2 for a more detailed description.

4. Experimental Results

The proposed approach was tested with natural images of flowers. They are suitable for evaluating the algorithm as they exhibit:

1. deformation of the object - bending of the petals.
2. deviations from frontal view.
3. complex illumination.
4. cluttered background.
5. broken edges, and interference from spurious gradients of the object - creases and speckles on the petals.

The values of all the thresholds are held constant for all the test cases: $\Delta_G = 0.001$, $\Delta_V = 0.01$, $\Delta_h = 0.1$, $\Delta_\psi = 0.7$ radians, and $\Delta_d = 4$. σ_p and σ_d , were kept at 0.002 and 9.0 respectively.

In order to enhance the visual evaluation of the confidence maps, peaks in the confidence maps with negligible heights were eliminated. The thresholded confidence maps, denoted by \tilde{H}_n , are computed as

$$\tilde{H}_n(\mathbf{p}) = \begin{cases} H_n(\mathbf{p}) & : H_n(\mathbf{p}) \geq \frac{1}{4} \max_{\mathbf{p}'} H_n(\mathbf{p}') \\ 0 & : \text{otherwise} \end{cases} \quad (14)$$

Figure 3 illustrates how the n -tuples of voters are collected from an image when detecting C_5 symmetry. The points which are members of the voting lists generating peaks in \tilde{H}_n are joined with line segments and then overlaid on the image; the voting points are also connected to the corresponding centroids of rotation. Notice the clutter present due to foliage and other flowers. The algorithm detects the perceptually dominant symmetry along with some spurious symmetries.

To illustrate the importance of the GVF, the approach was tested using the gradient magnitude field instead of

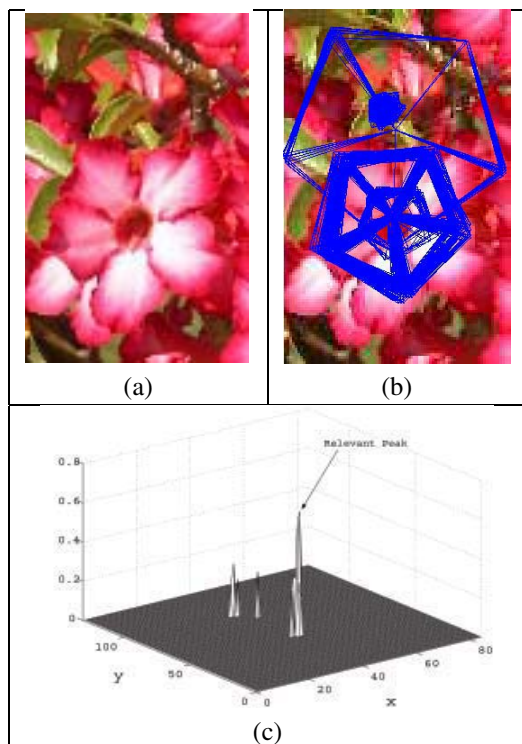


Figure 3. Example of the working system: (a) image, (b) voting points of peaks in \hat{H}_5 connected with line segments and overlaid on image, (c) confidence map, \hat{H}_5 , for C_5 symmetries.

the GVF. The performance was worse compared to when the GVF field was used to spread the information from the edges. Figure 4 shows the result obtained for the image shown in Figure 2(a). Two of the strong rotational symmetries are not detected and some spurious symmetries are detected.

Figure 5 shows some of the images used for testing the approach and the confidence maps computed by the algorithm. The location of the peaks in the confidence maps are marked on the images. The algorithm detects the significant rotational symmetries except for one of the flowers in each of cases (d) and (e). The perceptually significant symmetries are assigned high confidence values relative to other parts of the images. The placement of the peaks in the confidence maps shows that the centroids of the symmetries are detected accurately. The algorithm is able to handle the large amount of clutter due to foliage, e.g. cases (b), (e) and (j). In case (f), the symmetry is detected even though there is a pronounced skew in the petals and the illumination is complex.

As local features are considered, in some cases spurious symmetries are also detected along with the perceptually

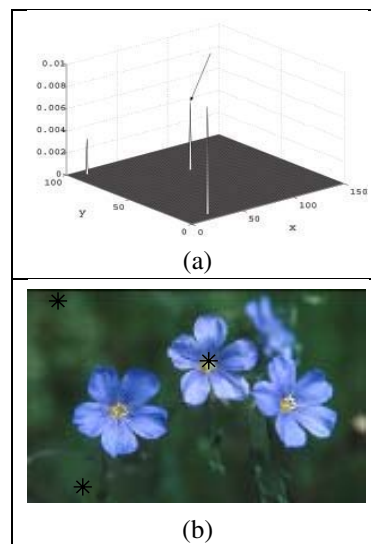


Figure 4. Example of the result obtained when the gradient magnitude is directly used instead of the GVF field, for the image shown in Figure 2. (a) Confidence map with the correct peak indicated with arrow, and (b) locations of the peaks marked on the image. The symmetry detection worsens upon using the gradient magnitudes directly.

dominant symmetries (e.g. case (g)). In case (i), the petals had a large number of speckles or dots which resulted in false detections. A more detailed analysis of the detected symmetries using contour or region based methods might help in eliminating some of the spurious detections.

Although not designed to handle skewed objects, the algorithm is able to handle some amount of skew due to its allowance for deformation. However, the method fails in cases of pronounced rotation away from frontal view (e.g. Figure 5(e) left). It might be possible to modify the conditions (2)...(7) to take into account non-frontal views of rotationally symmetric objects.

4.1. Computational Complexity

The computational complexity of the algorithm is proportional to the total number of node indices present in the lists propagated among the nodes. Ideally, the number of lists, and hence the number of indices, should be linear in the number of pixels in the image. However, in reality this is not the case as we allow for deformations in the GVF when comparing voting points, and also for irregularities in the n -sided polygons (Δ_d). For the images shown in Figure 5, the total number of edges in the graphs (\mathcal{G} 's) were supra-linear in the number of nodes/pixels (denoted by N), and

varied from $N^{1.325}$ to $N^{1.452}$. Even though the thresholds reduce the number of edges by a factor of 100, most of the perceptually significant symmetries were detected.

It is also interesting to observe the total number of node indices present in the lists as a function of the stage of the propagation, i.e. t . Let this be denoted by $M(t)$. At the first iteration, $t = 1$, $M(1)$ is proportional to the number of edges in the graph. This is large as only pairwise constraints are applied. However, in the next iteration, the lists are propagated to only those nodes which adhere to the constraints on location imposed by the n -sided regular polygons. Therefore, $M(2)$ is usually smaller than $M(1)$. Most of the propagations after the first iteration occur only in the rotationally symmetric regions in the image. However, the number of lists, and hence the number of indices ($M(t)$), increases with each iteration because of the allowance for errors in location and GVF. Consider a case of C_5 symmetry, and a node q initiating lists for it (Figure 6). Assuming the constraints for GVF and color are satisfied, q propagates lists to nodes in a circle (Circle-1) of radius at least $\frac{\Delta_d}{2}$. In the next iteration, all nodes present in this circle would propagate lists to all nodes present in Circle-2 of radius at least $\frac{\Delta_d}{2}$. Therefore, $M(t) = O([\pi(\frac{\Delta_d}{2})^2]^t)$. At the last iteration, many lists fail to propagate as they are not able to reach their initiating nodes. Therefore, for C_n symmetry, $M(n)$ is usually smaller than $M(n-1)$. In practice, the values of $M(t)$ were much smaller than those predicted by our analysis. This was mainly due to the additional constraints imposed by the GVF and the color values in the images. Figure 7 illustrates this behavior for the C_5 and C_3 symmetries present in images shown in Figure 5. The high peak at the 4th iteration in one of the test cases in Figure 7(a) is for the image in Figure 5(a).

In the actual implementation, the graph \mathcal{G} was not constructed explicitly. This is reasonable as the rate-determining step is the maintenance of propagating lists and the computation of the vote values.

5. Conclusions

An algorithm for fast detection of rotational symmetries in unsegmented natural images was presented. Given an image, its gradient magnitude field and the corresponding GVF field are computed. A graph, with the points in the image as nodes, is constructed to collect voters for possible symmetries present in images. The n -cycles present in the graph are then made to vote for C_n symmetries, the votes being weighted by the errors in transformation between the voters. Future work would include factoring in skew transformations, and using the output of the method to recover the contours of the image structures producing the detected symmetries.

Acknowledgements

This work was funded by ARDA grant DOD2004H840200000.

References

- [1] M. A. Arbib, editor. *The Handbook of Brain Theory and Neural Networks*. The MIT Press, Cambridge, Massachusetts, London, 1995.
- [2] E. M. Arkin, L. P. Chew, D. P. Huttenlocher, K. Kedem, and J. S. B. Michell. An efficiently computable metric for comparing polygonal shapes. *IEEE Trans. Pattern Anal. and Machine Intell.*, 13:209–216, 1991.
- [3] D. Ballard and C. Brown. *Computer Vision*. Prentice-Hall, Englewood Cliffs, N. J., 1982.
- [4] D. Chetverikov and A. Lerch. A multiresolution algorithm for rotation-invariant matching of planar shapes. *Pattern Recognition Letters*, 13:669–676, 1992.
- [5] D. Dennis. www.wallpaperdave.com.
- [6] H. J. A. M. Heijmans and A. V. Tuzikov. Similarity and symmetry measures for convex shapes using Minkowski addition. *IEEE Trans. Pattern Anal. and Machine Intell.*, 20(9):980–993, Sep. 1998.
- [7] D. P. Huttenlocher, G. A. Klanderman, and W. J. Rucklidge. Comparing images using the Hausdorff distance. *IEEE Trans. Pattern Anal. and Machine Intell.*, 15:850–863, 1993.
- [8] F. P. Kuhl and C. R. Giardina. Elliptic Fourier features of a closed contour. *Computer Vision, Graphics and Image Processing*, 18:236–258, 1982.
- [9] C. C. Lin and R. Chellappa. Classification of partial 2D shapes using Fourier descriptors. *IEEE Trans. Pattern Anal. and Machine Intell.*, 9(5):686–690, 1987.
- [10] G. Loy and A. Zelinsky. Fast radial symmetry for detecting points of interest. *IEEE Trans. Pattern Anal. and Machine Intell.*, 25(8):959–973, 2003.
- [11] P. S. Mark, H. Rom, and G. Medioni. B-spline contour representation and symmetry detection. *IEEE Trans. Pattern Anal. and Machine Intell.*, 15(11):1191–1197, Nov. 1993.
- [12] F. Mokhtarian and A. Mackworth. Scale-based description and recognition of planar curves and two-dimensional shapes. *IEEE Trans. Pattern Anal. and Machine Intell.*, 8:34–43, 1986.
- [13] V. S. Nalwa. Line-drawing interpretation: Bilateral symmetry. *IEEE Trans. Pattern Anal. and Machine Intell.*, 11(10):1117–1120, Oct. 1989.
- [14] A. Olmos and F. A. A. Kingdom. McGill calibrated colour image database, 2004. <http://tabby.vision.mcgill.ca>.
- [15] E. Persoon and K. S. Fu. Shape discrimination using Fourier descriptors. *IEEE Trans. Systems, Man, and Cybernetics*, 7:170–179, 1977.
- [16] V. S. N. Prasad and B. Yegnanarayana. Finding axis of symmetry from potential fields. *IEEE Trans. Image Proc.*, 13(12):1559–1566, 2004.
- [17] H. Shroff and J. Ben-Arie. Finding shape axes using magnetic fields. *IEEE Trans. Image Proc.*, 8(10):1388–1394, Oct. 1999.

- [18] K. Siddiqi, S. Bouix, A. Tannenbaum, and S. W. Zucker. The Hamilton-Jacobi transform. In *Proc. IEEE Int'l Conf. Computer Vision (ICCV-1999)*, volume 2, pages 828–834, Corfu, Greece, 1999.
- [19] C. Sprott. <http://sprott.physics.wisc.edu/fractals/collect/1999/>
- [20] C. Xu and J. L. Prince. Snakes, shapes, and gradient vector flow. *IEEE Trans. Image Proc.*, 7(3):359–369, Mar. 1998.
- [21] Q. Yang and B. Parvin. Perceptual organization of radial symmetries. *Proc. IEEE Conf. Computer Vision and Pattern Recognition (CVPR-2004)*, 1:320–325, 2004.
- [22] H. Zabrodsky, S. Peleg, and D. Avnir. Symmetry as a continuous feature. *IEEE Trans. Pattern Anal. and Machine Intell.*, 17(12):1154–1166, Dec. 1995.

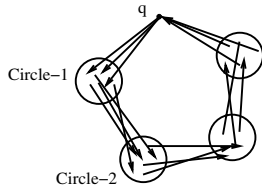
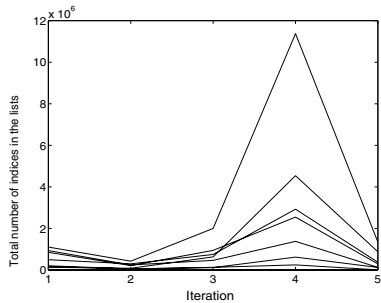
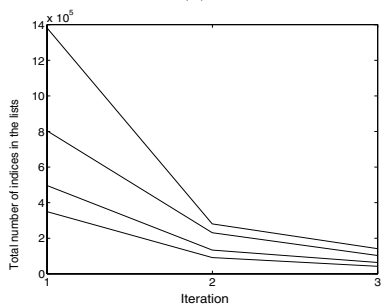


Figure 6. Allowance of irregularities in the n -sided polygons leads increase in the number of propagated lists with each iteration.



(a)



(b)

Figure 7. Plots of $M(t)$: (a) for images showing C_5 symmetries in Figure 5, and (b) for images showing C_3 symmetries.

Algorithm 1 GetVoterLists

Procedure

```

/*Initialization*/
 $\forall q_i \in \mathcal{G}, \forall n \in 2 \dots 8, S_{in}^{(0)} = \{\langle i \rangle\}, S_{in}^{(t)} = \{\} \quad \forall t > 0.$ 
/*First propagation*/
for all  $q_i, n, l \in S_{in}^{(0)}$  do
  for all  $q_j \in N(q_i)$  do
    if  $(\psi(\mathbf{p}_j) - \psi(\mathbf{p}_i)) \in [\frac{2\pi}{n} - \Delta\psi, \frac{2\pi}{n} + \Delta\psi]$  then
       $S_{jn}^{(1)} = S_{jn}^{(1)} \cup \langle j, l \rangle.$ 
    end if
  end for
end for
/*Intermediate propagations*/
for  $t = 1$  to 6 do
  for all  $q_i, t < n - 1, l \in S_{in}^{(t)}$  do
    Let  $l = \langle i, b, \dots, c \rangle$ 
    for all  $q_j \in N(q_i)$  do
      if  $((\psi(\mathbf{p}_j) - \psi(\mathbf{p}_i)) \in [\frac{2\pi}{n} - \Delta\psi, \frac{2\pi}{n} + \Delta\psi]) \wedge$ 
         $(\|(\mathbf{p}_j - \mathbf{p}_i) - R_n(\mathbf{p}_b - \mathbf{p}_i)\| < \Delta_d)$  then
           $S_{jn}^{(t+1)} = S_{jn}^{(t+1)} \cup \langle j, l \rangle.$ 
        end if
      end for
    end for
  end for
/*Final propagation: force lists to return to initiating nodes*/
for all  $q_i, n, l \in S_{in}^{(n-1)}$  do
  Let  $l = \langle i, b, \dots, c \rangle$ 
  if  $((\psi(\mathbf{p}_c) - \psi(\mathbf{p}_i)) \in [\frac{2\pi}{n} - \Delta\psi, \frac{2\pi}{n} + \Delta\psi]) \wedge$ 
     $(\|(\mathbf{p}_c - \mathbf{p}_i) - R_n(\mathbf{p}_b - \mathbf{p}_i)\| < \Delta_d)$  then
     $S_{cn}^{(n)} = S_{cn}^{(n)} \cup \langle c, l \rangle.$ 
  end if
end for

```

Algorithm 2 AggregateVotes

Procedure

```

 $\forall n, \forall \mathbf{p} \quad H_n(\mathbf{p}) = 0$  /*Initialization*/
for all  $q_i, n = 2 \dots 8, l \in S_{in}^{(n)}$  do
  Let  $l = \langle a_1, a_n, a_{n-1}, \dots, a_1 \rangle.$  /*The first and last voters are the same.*/
   $e_p = \sum_{t=1}^n \exp[\frac{-1}{\sigma_p} (e_1(\langle a_{t\oplus 1}, a_t \rangle) + e_2(\langle a_{t\oplus 1}, a_t \rangle))] \min(V(\mathbf{p}_{a_{t\oplus 1}}), V(\mathbf{p}_{a_t})).$ 
   $e_d = \sum_{t=1}^n \|(\mathbf{p}_{t\oplus 2} - \mathbf{p}_{t\oplus 1}) - R_k(\mathbf{p}_{t\oplus 1} - \mathbf{p}_t)\|.$ 
   $\tilde{\mathbf{p}} = \frac{1}{n} \sum_{t=1}^n \mathbf{p}_{a_t}$ 
   $H_n(\tilde{\mathbf{p}}) = H_n(\tilde{\mathbf{p}}) + e_p \exp(\frac{-e_d^2}{n\sigma_d}).$ 
end for

```

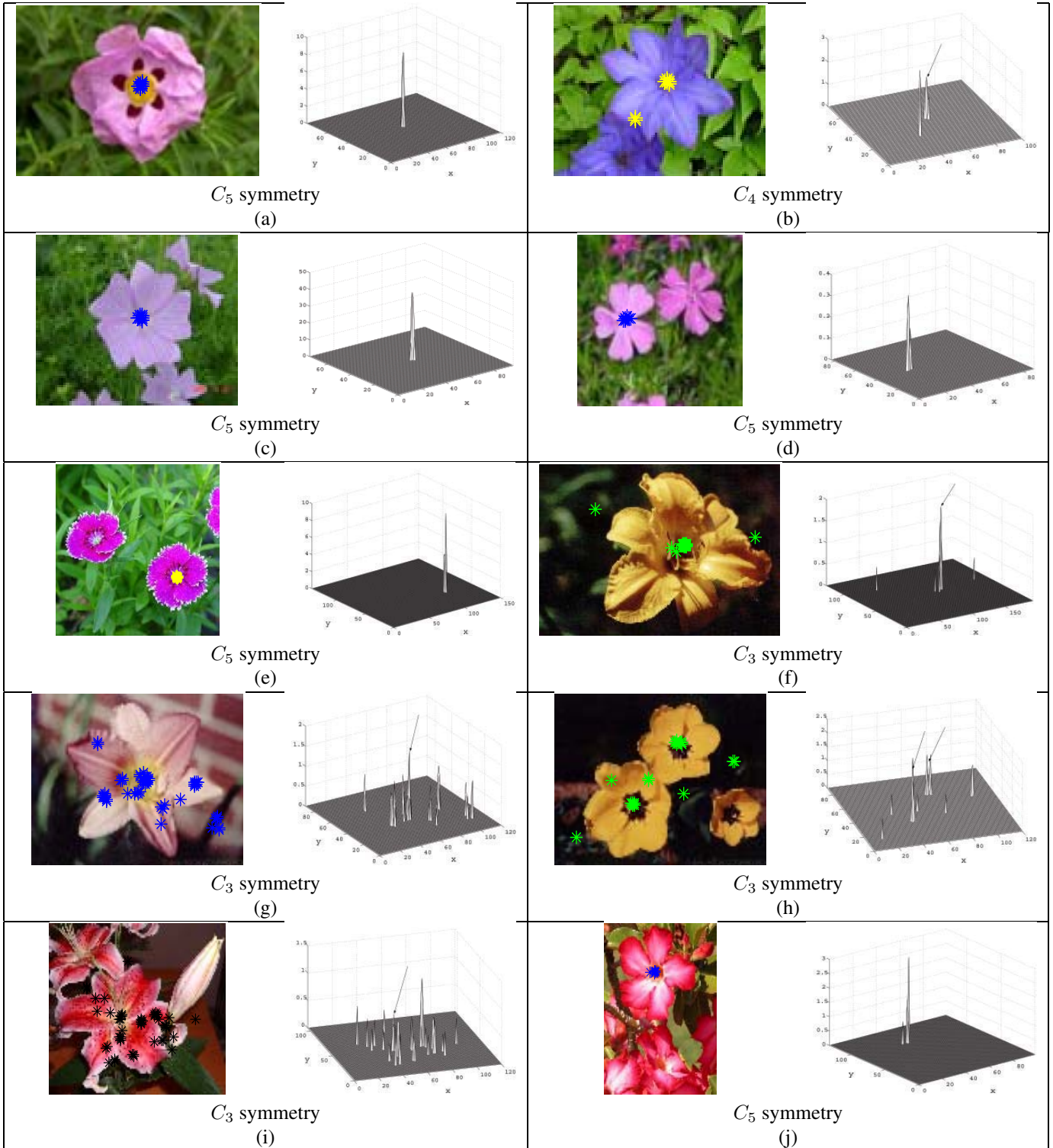


Figure 5. Test results: images with the locations of peaks in \tilde{H}_n marked on them and the confidence maps (\tilde{H}_n 's) of the indicated symmetries, computed after pruning using the thresholds (Δ 's). The results are best viewed in color.