Neural Process Lett (2015) 41:179–189
DOI 10.1007/s11063-013-9319-4

Noise-Robust Detection of Symmetric Axes by Self-Correcting Artificial Neural Network

Wonil Chang · Hyun Ah Song · Sang-Hoon Oh · Soo-Young Lee

Published online: 12 September 2013

Abstract Perception of symmetric image patterns is one of the important stages in visual information processing. However, local interference of the input image disturbs the detection of symmetry in artificial neural network based models. In this paper, we propose a noise-robust neural network model that can correct asymmetric corruptions and returns clear symmetry axes. For efficient detection of bilateral symmetry as well as asymmetry correction, our network adopts directional blurring filters. The filter responses are fed to oscillatory neurons for line extraction, which serializes the activation of multiple symmetry axes. Given an activated symmetry axis, the network estimates the difference of counterparts to generate a masking filter that covers the asymmetric parts. The network reconstructs the ideal mirror-symmetric image with complete symmetry axes by self-correction of corruptions. Through simulations on corrupted images, we verify that our network is superior to Fukushima's symmetry detection network. Our network successfully presents biologically plausible and robust symmetry perception mechanism.

W. Chang (✉)
Brain Science Research Center, KAIST,
Daejeon 305-701, Republic of Korea
e-mail: chang.wonil@gmail.com

H. A. Song · S.-Y. Lee
Department of Electrical Engineering, KAIST,
Daejeon 305-701, Republic of Korea
e-mail: hyunahsong@kaist.ac.kr

S.-H. Oh
Department of Information Communication Engineering,
Mokwon University, Daejeon 305-729, Republic of Korea
e-mail: shoh@mokwon.ac.kr

S.-Y. Lee
Department of Bio and Brain Engineering, KAIST,
Daejeon 305-701, Republic of Korea
e-mail: sylee@kaist.ac.kr

Keywords Symmetry axis detection · Asymmetry correction · Oscillator network · Directional blurring filter

1 Introduction

Symmetries are everywhere; it is easily found in both natural scenes and man-made structures. We perceive symmetric patterns more intuitively than those that are not. Scientific studies prove that mammals prefer symmetric patterns to asymmetric patterns in terms of visual perception [5,6,15,23]. Also, it is known that utilizing symmetry information in computers does help understanding of images as well [9,21]. Symmetry detection has been successfully used for practical applications such as face detection [25,27], shape matching [22], and urban scene figure groupings [18], as well as biological modeling of visual selective attention [8].

There are mainly two approaches in bilateral symmetry axes detection. One is to take mathematical algorithms and the other is to adopt biological mechanisms. Many of methods in the algorithmic approach use local feature descriptors [13,16] to detect symmetric patterns in images [3,14]. Compared to pixel-wise comparison method [1], local descriptor-based methods substantially speed up the detection process by reducing the number of symmetry axis candidates. In addition, recent works advance to detect symmetric axes from curved [2,12], skewed [10,11], or three-dimensional patterns [26]. This approach concentrates on the aspect of practical usage.

On the other hand, the second approach considers biological plausibility of the model; the whole process of symmetry detection are expressed in terms of parallelized neural computation in artificial neural networks(ANN). The scope of our work also lies within this approach. Spatial filter responses [4,17] are commonly used for feature extraction. Poirier and Wilson [20] presented symmetry perception model based on their previous model of shape perception in visual cortex [19]. Fukushima [7] successfully demonstrated a fast and reliable symmetry axis detection network.

Although various symmetry axis detection methods have been studied, they do not work well when asymmetric corruptions exist. Correction of asymmetries is significantly important for robust symmetry axes detection. In order to solve this problem, we propose an ANN model that provides robust symmetry axes detection and is able to correct asymmetries by itself. We first detect symmetry axes by comparing filter responses of counter parts. By utilizing directional filters, we are informed of in which direction there exists strong asymmetries. Oscillator alternatively activates one symmetry axis by one, and corrects asymmetries by masking them with back projection of asymmetric measure provided in filter responses. As a result, image with perfect symmetry is returned with clear symmetry axes.

Rest of the paper is organized as follows. In Sect. 2, we describe our neural network model for symmetry detection. In Sect. 3, experimental results are presented. We close our paper with conclusion in Sect. 4.

2 Method

The proposed network consists of two parts. In Part 1, symmetry axes are detected, and in Part 2, asymmetry is self-corrected. The overall structure of proposed network is described in Fig. 1. There are mainly three contributions in our network. First, we propose directional blurring filters for symmetry axis detection and asymmetry correction. Second, we adopt oscillatory network to serialize symmetry axis activation for back projection of asymmetries.

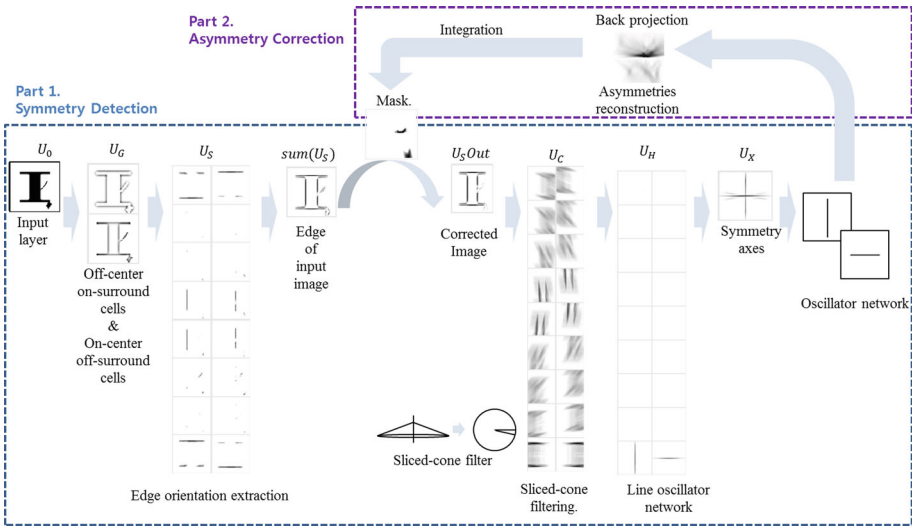


Fig. 1 The architecture of self-correcting symmetry detection network

Third, we extend symmetry axis detection network to asymmetry correction with aid of oscillatory network.

2.1 Part 1 - Symmetry Axis Detection

2.1.1 Edge Extraction (U_G and U_S layer)

Part 1 resembles much of the process in [7]. Input image is introduced to layer of photoreceptors, U_0. It proceeds to layer of lateral geniculate nucleus cells (LGNs), U_G, for extracting positive and negative contrast by off-center-on-surround cells and on-center-off-surround cells, respectively. U_G layer returns $U_G^{(k)}(m,n)$, where k is the index for off-center ($k=1$) and on-center ($k=2$) cell-plane, and (m,n) is the image coordinate. From $U_G^{(k)}(m,n)$, we detect directional edges in U_S layer, which returns $U_S^{(k)}(m,n)$. In contrast with [7], we have an additional layer, $U_S^{sum}(m,n) = \sum_{k=1}^{K} U_S^{(k)}(m,n)$, which sum up the edge information altogether.

2.1.2 Sliced-Cone Filtering (U_C Layer)

For symmetry axis detection, our proposed network uses directional blurring filters, named as 'sliced-cone filters.' While [7] applies cone filters to blurred edges of each direction, our sliced-cone filter itself possesses directional property. Therefore, we do not need edge blurring in each direction in U_S. Instead, we use summation of orientational edges, $U_S^{sum}(m,n)$.

The sliced-cone filter of kth direction $a_k = \frac{2\pi k}{K}$ is defined by

$$F_{SC}^{(k)}(m,n) = F_S^{(k)}(m,n) \times F_C(m,n), \tag{1}$$

$$F_S^{(k)}(m,n) = \begin{cases} 1 & if\,|\text{atan2}(n,m) - a_k| \le \frac{\pi}{K}\ and\ 0 < \sqrt{m^2+n^2} < L \\ 0 & otherwise \end{cases} \tag{2}$$

$$F_C(m,n) = \frac{\varphi(L - \sqrt{m^2+n^2})}{L} \tag{3}$$

Here, (2) is slice filter of direction a_k, and (3) is cone filter of radius L, where $\varphi(\alpha) = \max(\alpha, 0)$. Sliced-cone filter is generated by diving a cone into K radial slices. Each slice is assigned a directional angle for detecting edges in corresponding directions. Sliced-cone filter returns filter response $U_C^{(k)}(m,n)$ as defined below.

$$U_C^{(k)}(m,n) = \sum_{n'=-L}^{L} \sum_{m'=-L}^{L} F_{SC}^{(k)}(m',n') \times U_S^{sum}(m+m', n+n'). \tag{4}$$

The set of sliced-cone filters is rotation-invariant that it can be used to detect any direction of symmetry axis. The shape of long and thin slice is advantageous to detect and re-locate asymmetric parts. And it covers the entire local region of all directions, unlike simple line-shaped filters.

2.1.3 Symmetry Axis Detection (U_H and U_X Layer)

To determine symmetry axis, we compare left and right side of an axis. We take the same approach here as in [7]. Given a symmetry axis candidate at position (m,n) and orientation a_k, we compare sliced-cone filter responses of direction a_{k+i} and a_{k-i} for $i = 1, 2, \ldots, K/2$. If all pairs of filter responses $U_C^{(k+i)}(m,n)$ and $U_C^{(k-i)}(m,n)$ are equal, the candidate is regarded as symmetry axis. This process is described in Fig. 2.

In U_H layer, we calculate symmetry measure of axis orientation a_k at point (m,n) of image as,

$$U_H^{(k)}(m,n) = \varphi\left(\sum_{i=1}^{K/2}\left[\gamma_i \underbrace{\left(U_C^{(k+i)}(m,n) + U_C^{(k-i)}(m,n)\right)}_{\text{summation term}} - \delta \underbrace{\left|U_C^{(k+i)}(m,n) - U_C^{(k-i)}(m,n)\right|}_{\text{difference term}}\right]\right). \tag{5}$$

Here, $\gamma_i = \Gamma \min\left(\frac{4i}{K}, 2 - \frac{4i}{K}\right)$ and two variables δ and Γ are constant.

Symmetry axis is detected by acquiring the common parts of filter responses $U_C^{(k+i)}(m,n)$ and $U_C^{(k-i)}(m,n)$. We subtract the difference term from the summation term to leave the

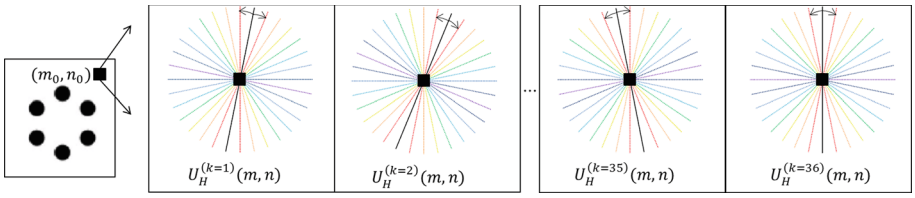
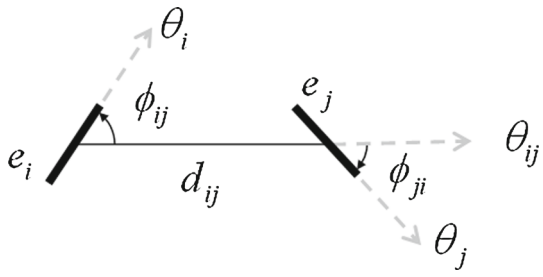


Fig. 2 Symmetry axis detection process. *Black lines* indicate given symmetry axis candidates with angles of a_k ($k = 1, 2, ..., 36$). Colored pairs of lines with same *colors* indicate pairs of directions of sliced-cone filters with angles of a_{k+i} and a_{k-i}, to verify the symmetry of given axis (*black line*). (Color figure online)

Fig. 3 Two edge pixels



commons of the two filter responses. To reduce false positives of symmetry axis for small i, γ_i is maximized when a_{k+i} and a_{k-i} are perpendicular to a_k and minimized when they are parallel to a_k.

The output of the Part 1 $U_X(m,n) = \max_k U_H^{(k)}(m,n)$ displays detected symmetry axes. Individual neurons may generate noise. However, activated neurons on a long line reliably represent local mirror-symmetry.

2.1.4 Oscillatory Network

To correct asymmetries, we generate masks by back projection of asymmetric filter response on the symmetry axis. We use locally-excitatory globally-inhibitory oscillator networks (LEGION) [24] and mutual excitatory connectivity between co-linear edge pixels for sequential activation of symmetry axes.

To understand how we detect and activate an axis, imagine two edge pixels e_i and e_j as in Fig. 3. Here, θ_i, θ_j, and θ_{ij} denote the orientation of edge pixels e_i and e_j, and line $\overline{e_i e_j}$, respectively. The distance between two edge pixels is denoted as d_{ij}. The angle between each edge pixel and line $\overline{e_i e_j}$ is defined as $\phi_{ij} = D(\theta_i - \theta_{ij})$ and $\phi_{ji} = D(\theta_j - \theta_{ij})$. We use a function $D(\theta) = \theta - \lfloor \frac{\theta}{\pi} + 0.5 \rfloor \pi$ to convert angle into the range of $[-\frac{\pi}{2}, \frac{\pi}{2})$, where $\lfloor x \rfloor$ is the round down value of x.

We detect straight lines based on the closeness and co-linearity between two edge pixels, using excitatory connection $w_{ij} = f_d(d_{ij}) f_l(\phi_{ij}, \phi_{ji})$ and inhibitory connection $v_{ij} = g_{inh} f_d(d_{ij})$. The term $f_d(d_{ij})$ and $f_l(\phi_{ij}, \phi_{ji})$ denote distance factor and linearity factor, respectively. They are computed as

$$f_d(d_{ij}) = \begin{cases} \left(\frac{\sigma_d}{d_{ij}}\right)^2 & \text{if } d_{ij} > \sigma_d, \\ 1 & \text{otherwise} \end{cases}, \text{ and} \tag{6}$$

$$f_l(\phi_{ij}, \phi_{ji}) = \begin{cases} \cos b\left(|\phi_{ij}| + |\phi_{ji}|\right) & \text{if } b\left(|\phi_{ij}| + |\phi_{ji}|\right) \le \frac{\pi}{2}, \\ 0 & \text{otherwise,} \end{cases} \tag{7}$$

where b is a constant. The neurons that are close together and appear on a straight line score high value for excitatory connection w_{ij}.

The total connection from neuron i to neuron j is $w_{ij} - v_{ij} = f_d(d_{ij})(f_l(\phi_{ij}, \phi_{ji}) - g_{inh})$, where g_{inh} is a threshold parameter that controls neurons' activation.

Oscillator $X^{(k)}(m,n)$ represents the activity of the symmetry axis with orientation a_k and position (m,n). It receives external input $U_H^{(k)}(m,n)$. The network generates alternate activation of symmetry axes. (See [24] for the details).

2.2 Part 2 - Asymmetry Reconstruction and Correction

Sequentially activated symmetry axis is fed to Part 2 for asymmetry reconstruction and correction.

2.2.1 Asymmetry Reconstruction

Given one activated symmetry axis, we calculate back projection coefficient for each of neurons on the symmetry axis. It is computed from the difference term in (5). If oscillator $X^{(k')}(m,n)$ is activated, we compare filter response $U_C^{(k)}(m,n)$ and its counter part $U_C^{(k'')}(m,n)$, which satisfy $k'' = mod(2k' - k - 1, K) + 1$; they correspond to $U_C^{(k+i)}(m,n)$ and $U_C^{(k-i)}(m,n)$ in (5), respectively. The bilateral difference accumulates to the back projection coefficient $B_C^{(k)}(m,n)$.

$$B_C^{(k)}(m,n) = \sum_{k'=1}^{K} A^{(k')}(m,n) \times \varphi\left(U_C^{(k)}(m,n) - U_C^{(k'')}(m,n)\right). \tag{8}$$

In (8),

$$A^{(k)}(m,n) = \begin{cases} 1 \; if \; X^{(k)}(m,n) \geq \Theta_x, \\ 0 \; otherwise \end{cases} \tag{9}$$

is the discretized neural activity $X^{(k)}(m,n)$ with threshold Θ_x.

Sliced-cone filters $F_{SC}^{(k)}(m,n)$ project back $B_C^{(k)}(m,n)$ and reconstruct asymmetric parts. Sliced-cone filters $F_{SC}^{(k)}(m,n)$ reconstruct asymmetric parts by back projection with coefficient of $B_C^{(k)}(m,n)$. The process is described in Fig. 4b. The back projection result $B_P(m,n)$(see Fig. 4c) is

$$B_P(m,n) = \sum_{k=1}^{K} \sum_{n'=-L}^{L} \sum_{m'=-L}^{L} F_{SC}^{(k)}(m',n') \times B_C^{(k)}(m+m', n+n'). \tag{10}$$

2.2.2 Asymmetry Correction

To correct asymmetries, we generate a mask $M(m,n)$ (see Fig. 4d) by accumulating back projection coefficient $B_P(m,n)$ with lateral inhibition $B_I(m,n)$.

$$M(m,n) \leftarrow \psi\left[M(m,n) + \mu\left(B_P(m,n) - \max\left(\eta B_I(m,n), \varepsilon\right)\right)\right]. \tag{11}$$

In this equation, $\psi[\alpha] = \max(\min(\alpha, 1), 0)$ is a clipping function. The lateral inhibition is defined as

$$B_I(m,n) = \sum_{k=1}^{K} \sum_{n'=-L}^{L} \sum_{m'=-L}^{L} \left(F_C(m',n')\right)^2 \times A^{(k)}(m-m', n-n'). \tag{12}$$

As an asymmetric image pixel gets closer to the symmetry axis, it receives stronger back-projection (Compare the left and the right side in Fig. 4c). The inhibition term $B_I(m,n)$ compensates for the spatial irregularity and eliminates the local blurring artifacts of back projection.

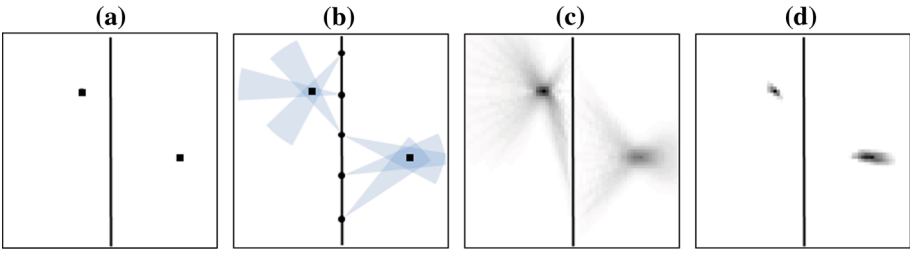


Fig. 4 **a** Asymmetry part with respect to given symmetry axis, **b** back projection of $B_C^{(k)}(m,n)$ by *sliced-cone filter* $F_{SC}^{(k)}(m,n)$, **c** accumulated back projection, $B_P(m,n)$, **d** generated mask, $M(m,n)$

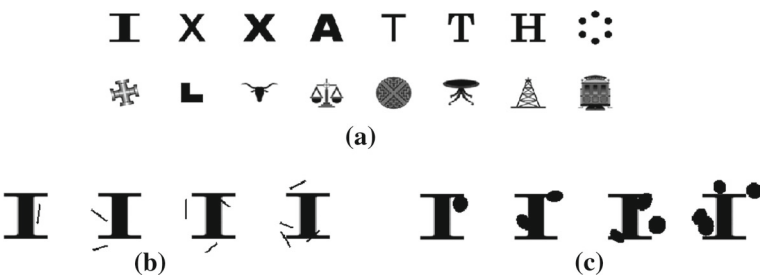


Fig. 5 **a** Input images, **b** example of an image with *line* noises, and **c** ellipse noises

The mask is applied to $U_S^{sum}(m,n)$ by

$$U_S^{sum}(m,n) \leftarrow (1 - M(m,n))U_S^{sum}(m,n). \tag{13}$$

$U_S^{sum}(m,n)$ is recurrently updated in a way that asymmetries are removed.

3 Experimental Results

To verify noise-robust symmetry axis detection and asymmetry correction ability of our proposed network, we generated some corruptions or noises in the images provided by Fukushima [7]. We used 16 input images of binary alphabetical characters or gray scale patterns as shown in Fig. 5a. The size of each image is 79×79. We combined filters of two different scales $L = 25$ and 50 for more accurate symmetry detection. We set the number of orientation $K = 36$. For calculation of $U_H^{(k)}(m,n)$ in Eq. (5), we used $\gamma_1 = \frac{1.2}{\pi}$, and $\delta = 5$. For mask generation in Eq. (11), we used $\mu = 0.003$, $\eta = 0.6$, and $\varepsilon = 1.0$. For fast simulation of oscillatory network, we implemented simplified algorithm of LEGION in [24]. We used two types of noises: lines and ellipses. We limited the length of line and the major / minor of ellipse noises to be in range of 10 %–20 % of input image diagonal. To observe symmetry axis detection performance change with respect to the degree of corruptions, we increased the number of noises from 1 to 4. For more reliable analysis, we generated 10 random noise samples for each cases. An example of image with line and ellipse noises of 1 to 4 is shown in Fig. 5b, c respectively.

An example of the experimental data with noise is displayed in Fig. 6. Fukushima's network fails to detect all of the symmetry axes as shown in (g). However, in (b), our proposed network detects weakly activated symmetry axes at first, and then by back projection,

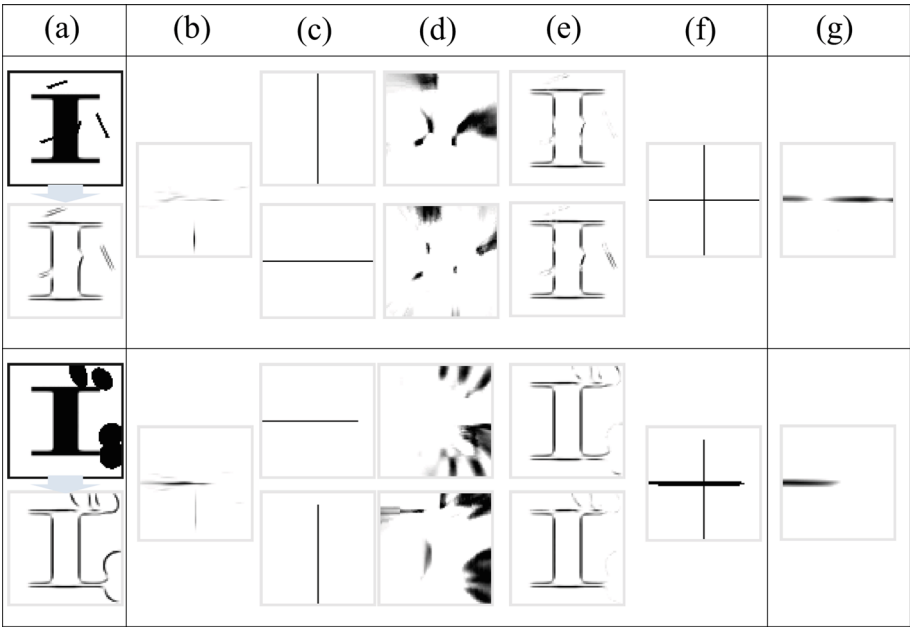


Fig. 6 **a** Input image corrupted with 4 noises of line and ellipse noises, and $U_S^{sum}(m,n)$, **b** symmetry axes detected before oscillatory network, $U_X(m,n)$, **c** serially activated symmetry axis by oscillatory network, **d** image mask and **e** self-corrected image with respect to activated symmetry axis, **f** accumulation of neural activity in oscillator network (returned symmetry axes after oscillator), and **g** symmetry axes detected by Fukushima's network [7]

asymmetries are corrected successfully by mask generated in (d). Proposed network returns corrected image along with clear symmetry axes in (e) and (f), respectively.

3.1 Evaluation of Symmetry Axis Detection

For quantitative analysis of symmetry axis detection performance, we evaluated the result with F-measure: $F = 2 \times \frac{precision \times recall}{precision + recall}$, where $precision = \frac{TP}{TP+FP}$, and $recall = \frac{TP}{TP+FN}$, TP = true positive (number of correctly detected pixels), FP = false positive (number of falsely detected pixels), and FN = false negative (number of missing pixels). When the result of symmetry axis detection is not binary as in Fukushima's model [7], we calculated F-value with respect to increasing thresholds and selected the maximum value. We compared the average F-measure of our result before correction (initial $U_X(m,n)$), our result after correction (accumulated activity of neural oscillator), and Fukushima's network result in Fig. 6b, f, and g, respectively. As noise increases, proposed network shows less performance drop compared to Fukushima's. This shows that proposed network performs more robust against corruption (see Fig. 7).

3.2 Evaluation of Asymmetry Correction

In this section, we analyze asymmetry correction performance of our proposed network in quantitative manner. An example of asymmetry correction results is shown in Fig. 8. Figure 8f is the ideal mirror symmetry image that can be obtained by optimal masking of noisy U_S^{sum}. Figure 8e resembles much of the ideal symmetry image in (f). Provided that the network

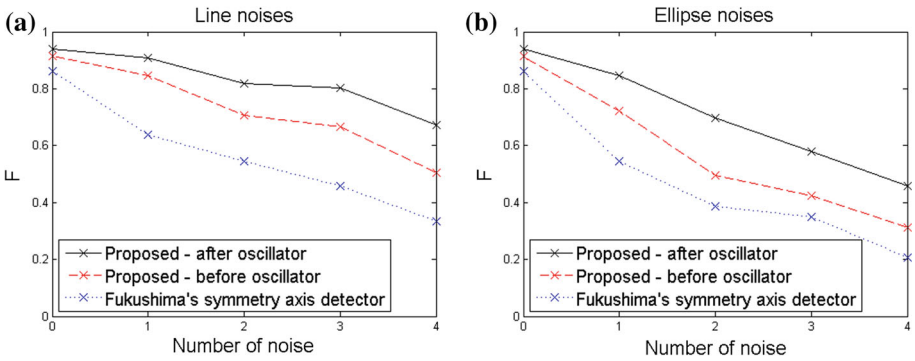


Fig. 7 Averaged F values of 16 images with 10 samples for each, for **a** line noises, and **b** ellipse noises

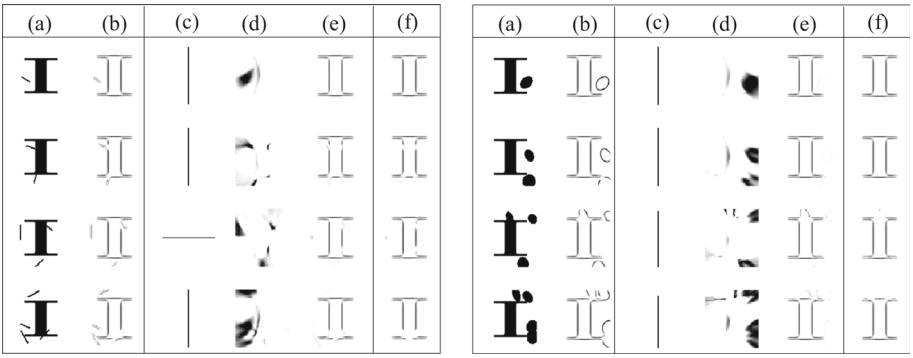


Fig. 8 Self-correction examples. (**a**) Corrupted input image, (**b**) $U_S^{sum}(m,n)$, (**c**) detected symmetry axis, (**d**) mask and (**e**) corrected image with respect to symmetry axis in (*c*), and (**f**) mirror-symmetry image with respect to ground-truth symmetry axis of same direction as (c)

detects appropriate symmetry axis in (c), the proposed network successfully reconstructs asymmetry masks in (d) and returns symmetry images.

We introduce two measures for the asymmetry correction performance: Asymmetry Correction Rate (ACR) and Symmetry Distortion Rate (SDR). Assume an input image $I_i(m,n)$, its ideal symmetric correction $I_G(m,n)$, and self-correction of input $I_c(m,n)$. ACR is defined as $1 - \frac{\sum_{m,n}\phi(I_c(m,n)-I_G(m,n))}{\sum_{m,n}(I_i(m,n)-I_G(m,n))}$. SDR is the proportion of symmetry part that has been accidentally removed by the network out of perfect symmetries, which is calculated by $SDR = \frac{\sum_{m,n}\phi(I_G(m,n)-I_c(m,n))}{\sum_{m,n} I_G(m,n)}$.

For easier comparison of $I_G(m,n)$ and $I_c(m,n)$, we selected images that have either vertical or horizontal or both as ground-truth symmetry axes (Image 1-7, 11-12, and 14-16 of Fig. 5a).

To observe pure asymmetry correction ability regardless of symmetry axis detection performance, we only considered the reconstructed outputs from successfully detected symmetry axes that shares more than 50 % similarities with the ground-truth axis.

The experimental result is summarized in Table 1. Proposed network corrects asymmetries induced by ellipse noises better than line noises, while maintaining SDR as low as less than 1 %. The proposed network successfully corrects asymmetric parts while retaining symmetric parts intact.

Table 1 ACR and SDR of line/ellipse noises

Number of noises	Line noises		Ellipse noises	
	ACR	SDR	ACR	SDR
1	0.5774	0.0112	0.6782	0.0116
2	0.5344	0.0107	0.6253	0.0130
3	0.5700	0.0093	0.6175	0.0119
4	0.5184	0.0118	0.6599	0.0156

3.3 Discussion

Our network is mainly composed of symmetry axis detection and asymmetry correction. The performance of the whole process depends on the initial detection of symmetry axes. If the network fails to detect even a weak activation of symmetry axes, the proceeding stages can neither correct asymmetries nor return clarified axes. This emphasizes the important role of directional filtering in the initial stage.

While directional blurring filters enable reconstruction and correction of asymmetry, they are sensitive to slight deviation of symmetry axes. When the angle of symmetry axis lies in-between the angles of directional filters, network often failed to detect them. We might reduce the sensitivity by increasing the number of directional filters or adopting circular blurring filters as in Fukushima's network [7].

The performance of asymmetry correction is not always stable and accurate. It sometimes leaves artifacts on the image and masks empty spaces. Further optimization of directional filter shape and mask generation method remains as a further work.

4 Concluding Remarks

In this paper, we proposed a biologically plausible symmetry detection network that can correct asymmetries by itself. We used directional blurring filters for symmetry axis detection and asymmetry correction. Oscillatory neurons activate one symmetry axis at a time, and the network corrects asymmetries by back projection of asymmetry measure with respect to activated axis. Through experiments with corrupted images of various cases, we verified that our proposed network successfully detects symmetry axis against strong asymmetric corruptions. In addition, proposed network corrects asymmetric parts and returns mirror-symmetry image with clearly reconstructed symmetry axes. The extended function of asymmetry correction opens more chances for practical applications in real environments.

Acknowledgments This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education, Science and Technology (2011-0029816). Wonil Chang and Hyun Ah Song are co-first authors, who equally contributed to this paper. We thank professor Fukushima for providing code for symmetry detection network [7].

References

1. Beck J, Hope B, Rosenfeld A (1983) (U.S.), N.S.F.: Human and machine vision. Academic Press
2. Chertok M, Keller Y (2010) Spectral symmetry analysis. IEEE Trans Pattern Anal Mach Intell 32(7):1227–1238
3. Cho M, Lee KM (2009) Bilateral symmetry detection via symmetry-growing. In: Proceedings of the British machine vision conference

4. Dakin S, Watt R (1994) Detection of bilateral symmetry using spatial filters. Spat Vis 8(4):393–413
5. Enquist M, Arak A (1994) Symmetry, beauty and evolution. Nature 372(6502):169–172
6. Frintrop S, Rome E, Christensen HI (2010) Computational visual attention systems and their cognitive foundations: A survey. ACM Trans Appl Percept 7(1):6
7. Fukushima K (2005) Use of non-uniform spatial blur for image comparison: symmetry axis extraction. Neural Netw 18(1):23–32
8. Jeong S, Ban SW, Lee M (2008) Stereo saliency map considering affective factors and selective motion analysis in a dynamic environment. Neural netw 21(10):1420–1430
9. Kootstra G, Nederveen A, De Boer B (2008) Paying attention to symmetry. In: Proceedings of the British Machine Vision Conference (BMVC2008), pp 1115–1125. The British Machine Vision Association and Society for Pattern Recognition
10. Lazebnik S, Schmid C, Ponce J., et al. (2004) Semi-local affine parts for object recognition. In: British Machine Vision Conference (BMVC'04), pp 779–788
11. Lee S, Liu Y (2010) Skewed rotation symmetry group detection. IEEE Trans Pattern Anal Mach Intell 32(9):1659–1672
12. Lee S, Liu Y (2012) Curved glide-reflection symmetry detection. IEEE Trans Pattern Anal Mach Intell 34(2):266–278
13. Lowe DG (2004) Distinctive image features from scale-invariant keypoints. Int J Comput Vis 60(2):91–110
14. Loy G, Eklundh JO (2006) Detecting symmetry and symmetric constellations of features. In: European Conference on Computer Vision (ECCV2006), pp 508–521. Springer
15. Machilsen B, Pauwels M, Wagemans J (2009) The role of vertical mirror symmetry in visual shape detection. J Vis 9(12):1–11
16. Mikolajczyk K, Schmid C (2004) Scale & affine invariant interest point detectors. Int J Comput Vis 60(1):63–86
17. Osorio D (1996) Symmetry detection by categorization of spatial phase, a model. Proceedings of the Royal Society of London. Series B: Biological Sciences 263(1366):105–110
18. Park M, Brocklehurst K, Collins RT, Liu Y (2010) Translation-symmetry-based perceptual grouping with applications to urban scenes. In: Asian Conference on Computer Vision (ACCV2010), pp 329–342. Springer
19. Poirier FJ, Wilson HR (2006) A biologically plausible model of human radial frequency perception. Vision Res 46(15):2443–2455
20. Poirier FJ, Wilson HR (2010) A biologically plausible model of human shape symmetry perception. J Vis 10(1):1–16
21. Reisfeld D, Wolfson H, Yeshurun Y (1995) Context-free attentional operators: the generalized symmetry transform. Int J Comput Vis 14(2):119–130
22. Temlyakov A, Munsell BC, Waggoner JW, Wang S (2010) Two perceptually motivated strategies for shape classification. In: Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, pp 2289–2296. IEEE
23. Treder MS (2010) Behind the looking-glass: a review on human symmetry perception. Symmetry 2(3):1510–1543
24. Wang D, Terman D (1997) Image segmentation based on oscillatory correlation. Neural Comput 9(4):805–836
25. Wang W, Gao Y, Hui SC, Leung K (2002) A fast and robust algorithm for face detection and localization. In: Internatonal Conference on Neural Information Processing (ICONIP2002), vol. 4, pp 2118–2121. IEEE
26. Wang Y, Xu K, Li J, Zhang H, Shamir A, Liu L, Cheng Z, Xiong Y (2011) Symmetry hierarchy of man-made objects. In: Computer Graphics Forum, vol. 30, p 287–296. Wiley Online Library
27. Won WJ, Jang YM, Ban SW, Lee M (2007) Biologically motivated face selective attention model. In: Internatonal Conference on Neural Information Processing (ICONIP2007), pp 953–962. Springer