InnerSpec: Technical Report

Fabrizio Guerrini, Alessandro Gnutti, Riccardo Leonardi
Department of Information Engineering, University of Brescia
Via Branze 38, 25123 Brescia, Italy
{fabrizio.guerrini, a.gnutti006, riccardo.leonardi}@unibs.it

Abstract

In this report we describe "InnerSpec", an approach for symmetric object detection that is based both on the computation of a symmetry measure for each pixel and on gradient information analysis. The symmetry value is obtained as the energy balance of the even-odd decomposition of an oriented square patch with respect to its central axis. Such an operation is akin to the computation of a row-wise convolution in the midpoint. The candidate symmetry axes are then identified through the localization of peaks along the direction perpendicular to each considered angle. These axes are finally evaluated by computing the image gradient in their neighborhood, in particular checking whether the gradient information displays specular characteristics.

1. Introduction and Related Work

In this paper a novel reflection symmetry detection algorithm for natural images ("InnerSpec") is described. The core of the method presented here relies on how symmetric is a patch around its central point by computing its auto-convolution in the direction orthogonal to a given reflection axis, with varying angle. When the patch (*i.e.* the whole image) is rotated in such a way that the reflection axis becomes vertical, the auto-convolution is computed row-wise and thus it is still a combination of 1D computations in the end. It is important to note, though, that the symmetry computation on a single row can be affected by "noise", *i.e.* the possible presence of non-symmetric and highly energetic content out of the bounds of the symmetric object. However, the fact that adjacent rows are used as well in the 2D patch helps to smooth out possible noisy results in such rows, so the true symmetry axis can be correctly identified.

The core computation, that is the auto-convolution of a patch, has been proposed in a different form in [8] as a Planar Reflective Symmetry Transform (PRST). A 1D version of the patch auto-convolution was also analogously described in [3], that was employed in a very different context to find hierarchies of symmetries in 1D discrete sequences. As it turns out, the patch auto-convolution is used here with some key differences w.r.t. what was done in both [8] and [3]. Such differences will be pointed out in the following Sections. Moreover a similar procedure, though acting on extracted interest points, entered the 2013 CVPR competition as well [4]. According to the survey paper for the 2013 workshop [5], that procedure was clearly outperformed by two other methods participating to the workshop [6][7]. However, the results were deemed to be not much satisfactory, so that the authors in [2] proposed searching for symmetry axes in natural images as an effective Turing test to replace character-based queries. Recently, another method has been proposed that further improves performance [1]. It extracts multi-scale 4D Appearance of Structure descriptors which are based on the detection of important edges. This method, although highly effective for images with sharp edges, incurs into some difficulties when dealing with natural images with background noise or too small objects.

As a matter of fact, using just a mathematical tool such as the patch auto-convolution to find data symmetries as a method for detecting symmetric objects has twofold limitations: first, what we perceive as a symmetric object may not correspond to a significant symmetry in the actual data because of distortions of various types (illumination changes, partial occlusions, etc.) that the human brain is able to filter out when recognizing the object. On the other hand, data with a high level of symmetry may be instead classified as unimportant in the brain, and therefore the symmetry therein goes largely unnoticed: for example, this can happen for background areas presenting symmetric patterns. So, the gradient information can help to improve the results obtained by just analyzing raw data symmetries.

Therefore, the symmetric object detection system that is proposed for natural images exploits spatial correlation properties in the data and at the same time uses relevant image gradient information to recognize and properly place the symmetry. In this paper it is shown how a relatively simple processing of the gradient image can dramatically aid the symmetry structure measure based on the data symmetry to improve the detection of symmetric objects.

2. Reflection symmetry detection principles for natural images

We begin discussing in Subsection 2.1 a theoretical framework for detecting whether a given 1D discrete sequence possesses reflection symmetry properties to some extent. The approach is then extended to the problem of detecting symmetric objects in natural images.

2.1. Joining Symmetry Detection and Symmetry Perception for Natural Images

First, let us consider the problem of finding strong local reflection symmetries in a 1D sequence $x[n]$. The inner product between $x[n]$ and its mirrored version around m, *i.e.* $x[2m-n]$, can be computed, and the particular m that maximizes it identifies the best global reflection symmetry point [3]. If a local symmetry around a given point n_1 is sought for, instead of the best global one of the entire sequence, the inner product should be computed on a window W of size n_p centered around n_1, because even if a strong local symmetry is indeed present, the rest of the sequence in faraway positions would hurt the computation. In addition, it is best to normalize said inner product by the energy of the sequence in the window, so that the values computed at different locations can be properly compared with one another.

Of course, the window size n_p is the key parameter in the whole process. In general, it would be ideal for the window to be limited exactly by the symmetric content extent. Being conservative by choosing a small n_p may hurt the symmetry measure and drown its value among false positives that would be abundant for small window sizes. On the other hand, overshooting the symmetry support would introduce non symmetric content into the computation, resulting in noisy values for the symmetry measure.

Thus, the best local symmetry is found by sliding the window across the sequence and computing a symmetry measure $S(n_1)$. For a window W of size n_p centered around n_1, such measure is computed as follows:

$$S(n_1) = \frac{\sum_{n=-n_p}^{n_p} x[n_1+n] \cdot x[n_1-n]}{\sum_{n=-n_p}^{n_p} |x[n_1-n]|^2} \qquad (1)$$

Next, our objective is to extend the basic mathematical tool we just derived to solve the problem of symmetric object detection in 2D images. To start, let us consider the case of the search for a vertical symmetry axis, *i.e.* there is a horizontally symmetric object in the image, clearly perceived by the human eye. To search for symmetry axes in different directions, it may turn convenient to rotate the image first in such a way that the sought symmetry axis becomes vertical. A batch of 1D convolutions is computed all at once for a number of rows, in effect performing the row-wise, windowed and normalized auto-convolution over a 2D square patch. This way, we both correct for noisy placement of symmetry due to image noise and exploit the 2D correlation information present in the image.

Consequently a symmetry measure S_2 can be associated to each center of a 2D patch P as follows:

$$S_2(P) = \frac{\sum_{m=-n_p}^{n_p} \sum_{n=-n_p}^{n_p} x[m,n] \cdot x[m,-n]}{\sum_{m=-n_p}^{n_p} \sum_{n=-n_p}^{n_p} |x[m,n]|^2} \qquad (2)$$

where n_p has been fixed to 60 as showed in Fig. 2. For the assessment of the PR curve, our algorithm has been tested on a subset of the training set of the present competition and we referred to the evaluation metric used in [5] which is also mentioned in Subsection 3.3.

The symmetry measure in Eq. (2), as we mentioned, is similar to the PRST described in [8]. It was computed as the inner product between the considered function (*i.e.* a 3D model) and its mirrored version through a given plane γ. Local symmetries are then found by thresholding such measure and considered maxima as candidate symmetry points. To reduce the complexity, the symmetry measure is sampled at regular intervals and the exact position of the symmetry is then refined through an optimization procedure. There are however two important differences w.r.t. the proposed algorithm: the most obvious one is the fact that here the symmetry is computed just on a local support, as on the other hand has been already done for the PRST as well [9]. The other one concerns how a candidate symmetry axis is not found through a global thresholding on the symmetry measure, but instead searching for local maxima along the direction orthogonal to said axis. In addition, such maxima on different rows are connected in the direction of the candidate axis to reinforce the belief in the symmetry presence.

So, Eq. (2) permits to detect symmetries in 2D data based on the correlation between a patch and its reflected version. However, using 2D contextual information to detect symmetries is not enough. We also employ a simple strategy to choose among all the candidate axes those that pertain to an interesting object as opposed to *e.g.* symmetric background. This process is based on the information given by the gradient image to identify the object edges and thus segment it out of the background or non-symmetric content. In the end, the axis of a symmetric object should lie between specular gradient information, especially when limiting the analysis to the most significant gradient magnitude. Thus, we propose to compare the gradient in a 2D patch around the candidate symmetry axis in the direction orthogonal to it as a crude but effective way to detect the presence of a symmetric object, and just implementing this approximate process leads to a much more precise matching between perceived symmetry and automatically detected symmetry.

3. Symmetry Axis Detection in Natural Images

In this Section, an algorithm for the identification of the main local symmetry axes in a given image I is described.

The discussion is targeted to the identification of symmetry axes pertaining to foreground objects, using a pair of features to sort out the most significant ones.

We subdivided the whole process into three main stages as illustrated in Fig. 1. The image I first enters the symmetry computation stage, where the objective is to compute a symmetry value for each position of I and for each direction, using the core computation outlined in Subsection 2.1. That information is stored into a 3D stack that is fed into the next stage, where the symmetry axes identification takes place. Such a process outputs a list of candidate symmetry axes. Finally, these candidate symmetry axes (segments) are further processed to produce a sorted list, most significant symmetry coming first.

3.1. Symmetry Computation

The objective of the first part of the algorithm is to attribute a symmetry value to each pixel of the image I and store that information in a 3D stack. Such a process is composed of three main tasks. In the first one, we rotate I by n different angles α_i that are referred to as I_{α_i}. The symmetry value is calculated separately for each angle, so that all possible symmetries can be captured independently of their tilt. I_{α_i} will return symmetries in the α_i direction ($0 \leq \alpha_i < 180$, with $\alpha_i = 180° i/n$). In our experiments we set $n = 90$, giving a 2° slope resolution.

As described in Section 2, the auto-convolution operation applied to a sequence allows to locate the position of the optimal symmetry axis. Since such a position is determined with a half pixel accuracy, in order to keep the patch size consistent between integer and half-integer positions, a columns interpolation by a factor of 2 is performed on I_{α_i}. So, n images are formed, that we refer to as $\tilde{I}_{\alpha_i}$, representing rotated and interpolated versions of the original image I. The interpolation process may be skipped if sub-pixel precision for the symmetry axis placement is not pursued.

Then, the level of symmetry of each pixel p of $\tilde{I}_{\alpha_i}$ is computed. First, a 2D patch of fixed size $2n_p + 1$ and centered in p is extracted and then the symmetry measure $S_2(P)$ according to Eq. (2) is computed and associated to p. By construction $S_2(P)$ can assume values in $[-1, 1]$. Values close to 1 (-1) are associated to even (odd) symmetries.

This process is performed for all images $\tilde{I}_{\alpha_i}$, so that n maps M_{α_i} are generated. The borders are discarded since the patches fall outside of the image boundaries. Last, a 3D stack of n maps, in which the i-th slice is the map M_{α_i}, is constructed. We refer to such a stack as "symmetry stack" since it represents the local (pixel) information about the level of symmetry. Of course the stack "wraps around" since 0° and 179° are actually neighboring directions.

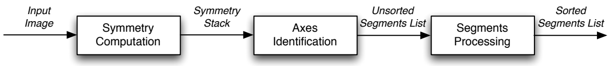

Figure 1: The three main stages of "InnerSpec".

3.2. Symmetry Axes Identification

At this point all symmetry axes present in the image I can be identified by processing the symmetry stack. In the end, all candidate symmetry axes are extracted and mapped back to the original image domain. The process is as follows: first, a straightforward (half-wave) rectification is performed, by zeroing all negative values of the stack. Indeed, in the previous Subsection we have mentioned that negative coefficients may correspond to an odd symmetry. In case such odd symmetries would be also of interest, the absolute value of the symmetry stack should be adopted, corresponding in effect to a full-wave rectification.

Following the discussion we have carried so far, one could assume that symmetry measures close to 1 identify stronger symmetries and should be thus sufficient to identify symmetric objects. On the contrary, we observed experimentally that this is not necessarily true. For example, large values can be associated to uniform background regions that correspond to highly symmetric structures. Moreover, objects that are perceived by humans as clearly symmetric could have lower values due to shadows, illumination changes, low resolution, etc. The use of 2D patches may only alleviate the problem.

Instead of just taking the symmetry value, a key observation is that searching for the peaks (local maxima) of the symmetry map is more relevant. So, the absolute value of the coefficient is not as important as its relationship w.r.t. its neighbors. All local maxima along the rows of every symmetry map can be associated to a specific direction α_i. A connectivity analysis between such maxima can be performed by means of a flood-fill algorithm. Consequently, all possible reflection symmetries define a series of segments linking connected local maxima existing in each row. The validation of the consistency of the local maxima through the connectivity analysis carried out along each expected orientation of symmetry is a crucial factor. This not only reinforces the belief that a symmetry is present, but also allows to define the extent of each symmetry segment.

In order to project back the symmetry information onto the original image coordinate system, each map M_{α_i} is first horizontally scaled down by a factor 2 to be consistent with the size of the input image I and then rotated by $-\alpha_i$.

3.3. Symmetry Segments Processing

The aim of the last part of the algorithm is to remove redundancy in the raw output of the axis identification process and to assign a confidence measure to the surviving segments.

It turns out that among the list of unsorted axes there is a significant number of symmetries that can be deemed of little interest for two possible reasons. First, a symmetry axis can be associated to unperceived, mild symmetries or to symmetries confined in small regions. On the other hand, a symmetry axis can be detected in the same approximate location on many slices of the stack pertaining to directions close to one another, thus offering redundant information for a same symmetry. These issues are managed in sequence by two processing blocks.

For each axis, we extract a feature f_1 that is the product of its average values S_2 in the symmetry stack associated to a given symmetric segment and the square root of the segment length l, as follows:

$$f_1 = \frac{\sum_k S_2(k)}{l} \cdot \sqrt{l} = \frac{\sum_k S_2(k)}{\sqrt{l}} \qquad (3)$$

Such a feature embodies a good compromise between the symmetry measure through the average of S_2 and the length of the symmetry segment l. To make f_1 comparable across images with different sizes, l may be normalized *e.g.* w.r.t. the image diagonal size. Taking the square root of l is another compromise, trying to penalize both excessively long segments probably corresponding to extended uniform backgrounds and symmetries too short to be relevant.

The overlapping segments removal process handles deletion of segments corresponding to the same symmetry. We adopted a processing similar to the one involved in the evaluation of true positives in [5]: redundant segments are identified whenever their centers are closer than 20% of the least of their length and at the same time the angle between them is less than $10°$. Of the redundant set, only the segment with the higher f_1 value is kept. A sorted symmetric segment list can be constructed, according to the associated f_1 value.

Even with the expedients described so far to improve the detection accuracy of data symmetries, such as the search for peaks in the orthogonal direction, the candidate symmetry axes still may correspond to a lot of false positives. To correct this, the idea is to favor those axes whose edge information around them is as specular as possible, hinting at the fact that the axis is running through the center of a symmetric object. To favor it, first we compute the gradient image using a standard Sobel operator. The gradient magnitude is normalized w.r.t. to its maximum value and then a mask $G[m,n]$ is obtained using a threshold $M = 0.3$ as the retained most significant gradient magnitude values percentage, whereas the least significant are set to 0.

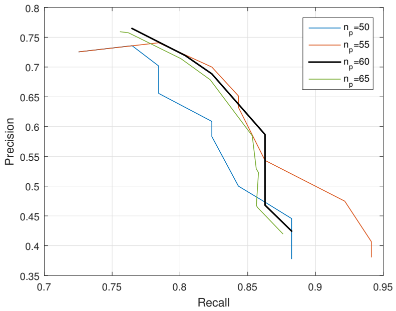

Figure 2: PR curves for different n_p, each of them obtained varying the number of detected axes for each image. The comparison proves the robustness of the method in relation to the window size (the PR values are similar for different n_p). $n_p = 60$ returns a slightly better result.

For each axis, a patch P_A is extracted. Its starting width is the same as the patches P used for the symmetry measure computation (namely n_p), and it is extended vertically to cover all the axis locations (recall that for each axis the image is rotated so as to have the considered axis in the vertical direction). The width is then extended to a larger n_g in case not enough significant edges are covered by the patch, in particular at least half of the rows must have at least a non-zero mask value. The width n_g cannot in any case exceed one quarter of the gradient image width.

The even-odd decomposition is performed on the gradient magnitude $G[m,n]$ in P_A, as follows:

$$M_G(P_A) = \frac{\sum_{m=m_1}^{m_2} \sum_{n=-n_g}^{n_g} G[m,n_1+n] \cdot G[m,n_1-n]}{\sum_{m=m_1}^{m_2} \sum_{n=-n_g}^{n_g} (G[m,n_1+n])^2} \qquad (4)$$

where the axis is detected in the n_1-th column and in the $[m_1, m_2]$ rows interval. If the gradient magnitude is approximately specular around the axis, $M_G(P_A)$ takes on values close to 1, otherwise its value is around 0.

Next, the aim is to ensure that the gradient vectors with large magnitude, as individuated by the mask $M_G(P_A)$, in mirrored position w.r.t. to the symmetry axis possess matching directions, namely, the absolute value of the sum of their directions should be π. Therefore, another mask $M_{D1}(P_A)$ is computed on the gradient directions $D[m,n]$, again through a normalized inner product, as follows:

$$M_{D1}(P_A) = \sum_{m=m_1}^{m_2} \sum_{n=-n_g}^{n_g} D[m,n_1+n] \cdot D[m,n_1-n] \qquad (5)$$

and then the directions mask $M_D(P_A)$ is obtained as:

$$M_D(P_A) = \begin{cases} 0 & \text{if } |M_{D1}(P_A) - \pi| > \frac{\pi}{2} \\ 1 - \frac{2|M_{D1}(P_A) - \pi|}{\pi} & \text{if } |M_{D1}(P_A) - \pi| < \frac{\pi}{2} \end{cases} \qquad (6)$$

Finally, the masks are combined to evaluate the second feature f_2 based on the gradient information:

$$f_2 = \frac{\sum_k M_G(k) \cdot M_D(k)}{l} \cdot \sqrt{l} \qquad (7)$$

where the normalization by $\sqrt{l}$ has the same function as that in Eq. (3). So, the candidate axes found by the f_1 feature are finally sorted according to their f_2 values.

References

[1] I. R. Atadjanov and S. Lee. Reflection symmetry detection via appearance of structure descriptor. In *European Conference on Computer Vision*, pages 3–18. Springer, 2016. 1

[2] C. Funk and Y. Liu. Symmetry recaptcha. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5165–5174, 2016. 1

[3] A. Gnutti, F. Guerrini, and R. Leonardi. Representation of signals by local symmetry decomposition. In *Proc. of the 23rd European Signal Processing Conference (EUSIPCO '15)*, pages 983–987, 2015. 1, 2

[4] S. Kondra, A. Petrosino, and S. Iodice. Multi-scale kernel operators for reflection and rotation symmetry: further achievements. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 217–222, 2013. 1

[5] J. Liu, G. Slota, G. Zheng, Z. Wu, M. Park, S. Lee, I. Rauschert, and Y. Liu. Symmetry detection from real-world images competition 2013: Summary and results. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pages 200–205, 2013. 1, 2, 4

[6] G. Loy and J.-O. Eklundh. Detecting symmetry and symmetric constellations of features. In A. Leonardis, H. Bischof, and A. Pinz, editors, *ECCV (2)*, volume 3952 of *Lecture Notes in Computer Science*, pages 508–521. Springer, 2006. 1

[7] V. Patraucean, R. Grompone von Gioi, and M. Ovsjanikov. Detection of mirror-symmetric image patches. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 211–216, 2013. 1

[8] J. Podolak, P. Shilane, A. Golovinskiy, S. Rusinkiewicz, and T. Funkhouser. A planar-reflective symmetry transform for 3d shapes. *ACM Transactions on Graphics (TOG)*, 25(3):549–559, 2006. 1, 2

[9] P. Speciale, M. R. Oswald, A. Cohen, and M. Pollefeys. A symmetry prior for convex variational 3d reconstruction. In *European Conference on Computer Vision*, pages 313–328. Springer, 2016. 2