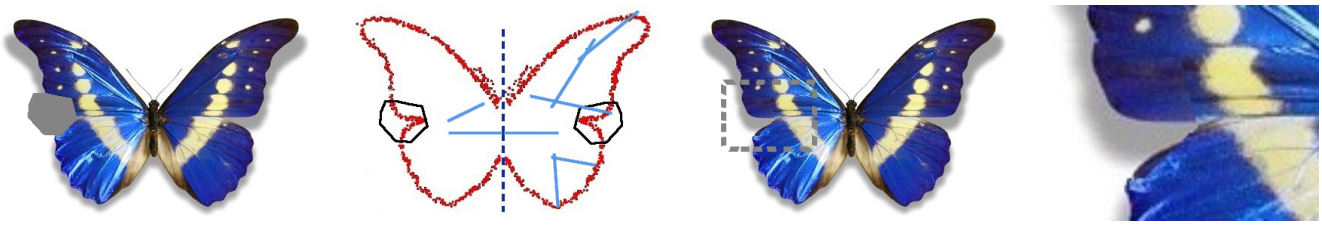
Symmetry-based completion

THIAGO PEREIRA[1], RENATO PAES LEME[1], LUIZ VELHO[1] AND THOMAS LEWINER[2]

[1] Visgraf Laboratory — IMPA — Rio de Janeiro — Brazil
[2] Department of Matematics — Pontifícia Universidade Católica — Rio de Janeiro — Brazil
{tpereira,renatoppl,lvelho}@impa.br. www.mat.puc--rio.br/~tomlew.

Abstract. Acquired images often present missing, degraded or occluded parts. Inpainting techniques try to infer lacking information, usually from valid information nearby. This work introduces a new method to complete missing parts from an image using structural information of the image. Since natural and human-made objects present several symmetries, the image structure is described in terms of axial symmetries, and extrapolating the symmetries of the valid parts completes the missing ones. In particular, this allows inferring both the edges and the textures.

Keywords: ***Image completion. Inpainting. Symmetry detection. Structural image processing.***



(a) Hole. (b) Axis. (c) Completed image. (d) Completion detail.

Figure 1: *Completion of a butterfly image: the marked missing region (a), in gray, is identified in the global structure of the image through axial symmetries (b). It can be completed with texture from its symmetric part (c),(d).*

1 Introduction

Acquired images are generally incomplete, either due to the degradation of the media, like old paintings, pictures or films, due to occlusion of scene parts from undesired objects or due to channel losses in digital image transmission [18]. To overcome those issues, inpainting techniques try to complete missing regions of an image. Since the ground truth is unknown in real applications, the inferred content must be consistent with the image as a whole.

This implies two step in the inpainting pipeline: analysis and synthesis. The analysis step determines the characteristics of the image relevant to completion. The synthesis step then uses the gathered knowledge to extend the valid region. Local methods analyze the boundary of the invalid region and the synthesis is usually performed by diffusion-like processes to propagate the boundary's color. However, the diffusion step may blur the inpainted region, harming the texture coherency. Other methods segment the image in texture-coherent regions and synthesize a new texture to fill the hole, based on the closest match with the boundary texture. Although this solves the blur problem, it may not respect the global structure of the image. In particular, completing very curved shapes or big holes remains an issue.

In this work, we propose to exploit the global structure of the image for inpainting (see Figure 1). More precisely, we estimate the image's symmetries and complete the missing part by their valid symmetric match. Since symmetry is an important coherency criterion both for natural and human-made objects, its analysis reveals much of the relevant image structure. The present paper restricts to axial symmetries of the image's edges, and may be easily extended to entail more general transformations and features. However, nice results, including textures, can already be obtained with these restrictions.

2 Related Work

Image restoration. Inpainting methods can be categorized according to the extent of the region the analysis and synthesis operations work on. Early approaches use local analysis to extend the valid image from a small neighborhood around the missing region. In particular, Bertalmio and Sapiro [3] propagate image lines (isophotes) into the missing part using partial differential equations, interleaving propagation steps with anisotropic diffusion [16]. This extends the smooth regions while still respecting image

Preprint MAT. 13/08, communicated on May 19th, 2008 to the Department of Mathematics, Pontifícia Universidade Católica — Rio de Janeiro, Brazil. The corresponding work was published in the proceedings of GRAPP 2009, INSTICC 2009.

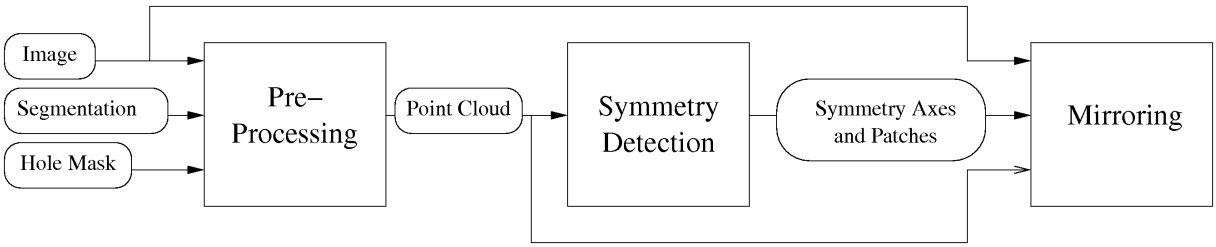


Figure 2: *The input image is pre-processed to extract a sampling of the image's edge. The symmetry detection extracts the structure of this point cloud, and these symmetries are used for completing the missing parts by mirroring adequate valid parts.*

edges. Petronetto [17] created an inpainting algorithm inspired by heat diffusion to improve propagation. Barcelos and Batista [2] combined the original inpainting with a variational approach to also extend level lines. Their method avoids the diffusion step, as such it reduces blur and improves speed. These local methods work very well for small holes but introduce blur when dealing with large regions, which harms the quality of results on regions with high frequencies and texture. Global analysis try to locate relevant regions in the entire image, or even in a large image database [10] to handle very large missing regions if similar objects are present in the database.

On the synthesis side, several approaches consider completion as a texture synthesis problem: instead of completing at a pixel level, these methods identify small regions of the hole to be filled first and search for a best match throughout the image. The matched region is copied and blended with the surroundings. In particular, Efros and Freeman [7] create new textures by putting together small patches from the current image. Drori *et al.* [6] and Criminisi *et al.* [5] complete the holes by propagating texture and contours. These methods preserve local structure of the image, but may fail to propagate global structure of the image like bending curves. In this work, we propose a technique that identifies the object structure and boundaries and incorporate this information in the completion process. We argue that structure from object symmetry can be used for inpainting in more complex examples.

Symmetry detection. Early works in symmetry detection deal with global and exact symmetries in point sets like [1, 23] based on pattern-matching algorithms. This restricts their applicability to image processing since most symmetries found in nature or human-made are not exact or might be slightly corrupted by noise. Zabrodsky *et al.* [25, 26] measure the symmetry of a shape by point-wise distance to the closest perfectly symmetric shape. The level of symmetry can also be measured by matching invariant shape descriptors [13], such as the histogram of the gradient directions [19], correlation of the Gaussian images [20] or spherical functions [11]. Such symmetry measures work well for detecting approximate symmetry, although they are designed for global symmetry detection.

Recently, Loy and Eklundh [12] used the Hough Transform to identify partial symmetries, i.e., symmetries of just one part of the object [21]. Such partial symmetries can also be obtained by partial matching of the local geometry [8, 14]. In particular, Mitra *et al.* [14] accumulate evidences of larger symmetries using a spatial clustering technique in the symmetry's space. The technique used in this paper is close to Mitra *et al.* [14]. However, we focus on incomplete symmetries due to occlusion in images, and thus adapted their symmetry detection.

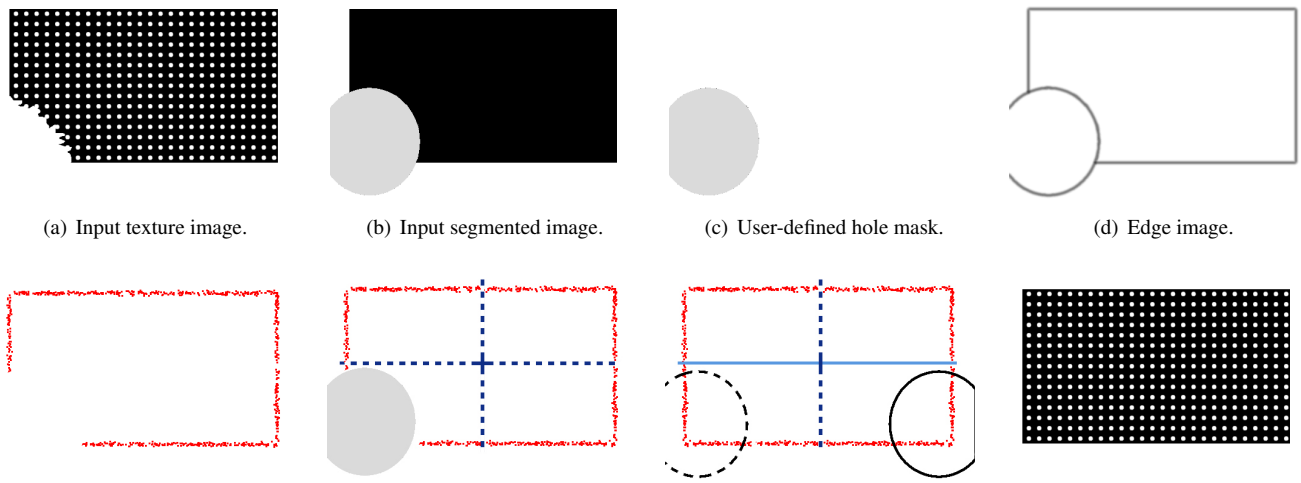
Symmetry-based completion. Symmetries have been used to complete shapes in different contexts. For example, Thrun *et al.* [22] detect symmetries in 3D range image to complete based on a search in the symmetry space, and complete the whole model by a global reflection. Zabrodsky *et al.* [24] use rectify shapes by symmetrization, even with occluded parts. Mitra *et al.* [15] achieve similar results for 3D shape. However, these techniques do not handle partial symmetries or affect parts of a 2D image that are not missing.

3 Method Overview

The proposed method is composed of two main steps: symmetry detection, corresponding to the image analysis, and mirroring for synthesis of lacking information. A simple pre-processing is applied to the image to extract a sampling of the image edges. The interactions between these steps is schematized in Figure 2 and illustrated in Figure 3. The input image contains a user-defined mask around the invalid region. A simple pre-processing segments the image and extracts a sampling of its edges (see Figure 3(a)-(d)). Then, the symmetry detection step identifies the many symmetry axes present in the object, as seen in Figure 3(f). Finally, the completion step chooses the symmetry axis that best fits the missing region and mirrors the texture and edges of the valid parts into the hole (Figure 3(g)-(h)). These steps are detailed below.

(a) Pre-processing

Object identification is a well studied problem. Many algorithms have been proposed to segment images. While extremely relevant to our method, segmentation is not the focus of this work. As such, we assume receiving a segmented image as input. Symmetry extraction should ideally take into account the border as well as the interior of the image's ob-



(a) Input texture image. (b) Input segmented image. (c) User-defined hole mask. (d) Edge image.

(e) Edge's point cloud. (f) Symmetry axes and patches. (g) Mirrored edges. (h) Completed image.

Figure 3: *Illustration of the pipeline of Figure 2.*

jects. We use here only the border (edge) information for the sake of simplicity. Moreover, we represent those edges by points. Although it may loose some connectivity information, it permits a versatile representation and fits better for adapting geometric modeling techniques for symmetry detection. Therefore, we perform an edge detection on the input image through a difference of Gaussians implemented in the GIMP package [9], and remove the artificial edges generated from the user-defined hole mask. We then perform a stochastic sampling of the edges taking the gray values of the edge image as probability. This generates a point set representation of the edges. Finally, we compute the normal and the local curvature at each point of the point set.

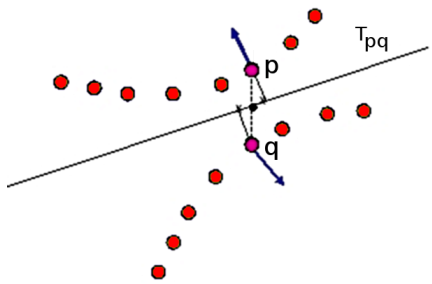
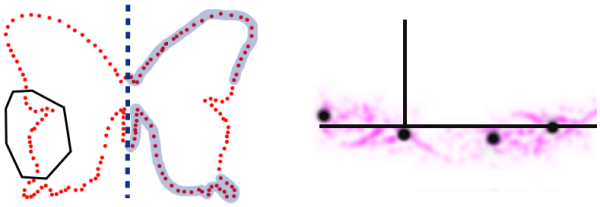


Figure 4: *The symmetry axis are robustly defined from the normals.*

(b) Analysis: symmetry detection

We are interested in approximate and partial symmetries, since the image has incomplete information in the hidden regions and since the image content may present several inexact symmetries. Therefore, our approach is largely based on the method proposed by Mitra *et al.* [14]. However, in this paper, we will restrict the space of symmetries to axial symmetries. Using the point set representation described



(a) Symmetry axis (b) Axis clusters

Figure 5: *Clusterization of candidate symmetry axis.*

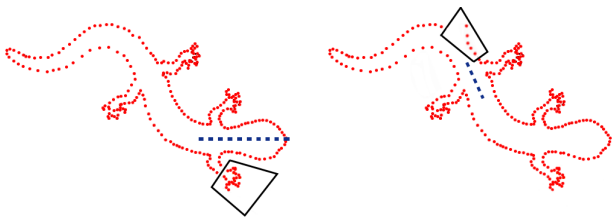
above, valid partial symmetries should map a substantial subset of the points to another one. In its basic form, the symmetry detector stores for each pair of points their bisector as a candidate symmetry axis (see Figure 4). Then it returns the clusters of candidates with their associated matching regions (see Figure 5). The clustering allows detecting only approximate symmetries.

To improve robustness and efficiency of this basic scheme, we enhanced this basic scheme as follows. On the one hand, we can observe that the sampling of the edges does not guarantee that a point p of the set is the exact symmetric of another sample point q. However, their normals should be mapped even with random sampling. Therefore, we define for each pair pq the candidate reflection axis T_{pq} as the line passing through the midpoint of pq and parallel to the bisector of the normals at p and q (see Figure 4). The normals are then symmetric by T_{pq}, although the points p, q may not. On the other hand, reducing the number of candidate axis would accelerate the clustering. Therefore, we reject a pair p, q if their respective absolute curvatures are too different, since the curvature is covariant with reflections. We also reject a candidate axis T_{pq} defined above if points p and the reflection of q are too far away.

The filtered candidate axes are then represented by their distance to the origin and their angle $\phi \in [0, \pi[$ with respect to the horizontal line. Clustering is performed in this two-dimensional parameter space using Mean-Shift Clustering [4], taking into account the inversion at $\phi = \pi$. Given a candidate axis T_{pq} at the center of a cluster, its matching regions are computed by propagating from the initial set $S = \{p, q\}$: a neighbor r of a point $s \in S$ is added if its reflection through T_{pq} is either close to a point of S or inside the hole mask of the image. This last condition allows detecting incomplete symmetries, which are crucial for completion.

(c) Synthesis: mirroring

The completion process first identifies from the image edge structure which of the detected symmetries to use, and then reflects the image's texture from the valid parts to the missing one. The ideal situation for our structural completion occurs when a symmetry structure traverses the hole. In that case, the sampled edges around the missing region clearly define which valid part of the image is to be reflected. More precisely, the border of the valid part must match the border of the hole. We thus choose among all detected symmetries the one that best fits the edges around the hole with edges of the valid part.



(a) Successful reconstruction (b) Failed reconstruction

Figure 6: *In the ideal case, a single reflection achieves continuity on both sides of the hole. In real case, several symmetries must be involved.*

However, in many real cases, in particular those with large missing parts, no single symmetry axis would map valid edges to the edges near the hole (see Figure 6). To overcome this issue, we complete the boundary from the border inwards. To do so, we look for the largest structure that maps with one of edges neighboring the hole and mirror that structure (see Figure 7). The hole is diminished from the synthesized region, and the process repeats until no structure matches the reduced hole border.

Once the axes have been defined and the valid structures have been mapped to the hole, we proceed to the image-based completion. For each pixel i of the hole, we look for the closest point p of the synthesized structure. This point p has been reflected from a valid structure by a symmetry T which is used to find the symmetric pixel j of i. The color of j is simply copied into i. This approach is very simple and may be enhanced in future works by more advanced texture synthesis and insertion.

(a) Input image (b) Half completed image (c) Ground truth

Figure 7: *Fish example: a single axis may not ensure boundary coherency on both sides.*

Model	# Points	# Symmetries	Timing
Butterfly	506	12	44
Eagle	765	9	83
Turtle	575	8	46
Lizard hand	271	10	87
Lizard body	294	10	122

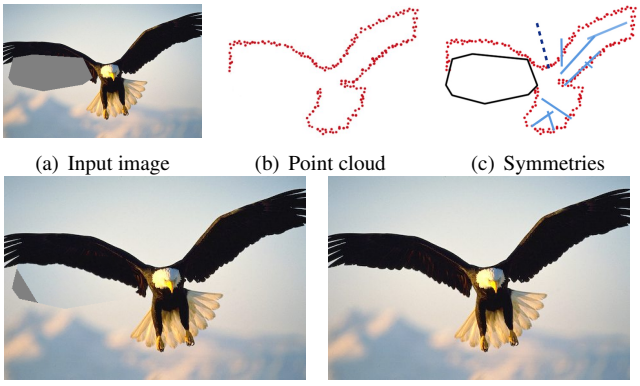
Table 1: *Characteristics of the symmetry-based completion for the illustration of this paper. The timings (in seconds) were obtained on a 2.8 GHz Pentium D.*

4 Results

Implementation details. The method described at the previous section can be implemented with different algorithmic optimizations. During many steps of our algorithm, proximity queries were required. Therefore, we build a Delaunay triangulation at pre-processing in order to support k-nearest-neighbors queries. Among other already mentioned uses, these queries serve the normal and curvature approximations by a local second degree polynomial Monge form. In order to choose efficiently the best axis that maps a valid structure to the hole's edge, we build a proximity graph. The vertices of this graph are the valid structure points that are mirrored into the hole. A link between vertices is created when they have a common symmetry axis T and when their reflection by T are close-by. The longest path in that graph determines the best symmetry axis T.

Experiments. We experimented our technique in different contexts. 1 presents the execution times including the entire pipeline. The symmetry detection step accounts for 85% of total time. The butterfly image of Figure 1 has symmetric structures and background with the same axes. The eagle image of Figure 8 has symmetric structures for the main shape, but the background has a different symmetry. On the contrary, the turtle image of Figure 9 has a symmetric background but the animal's symmetry is artificial, although very coherent with the image. The lizard structure of Figure 6 was tested in two opposite configuration: perfect symmetric and lack of symmetry.

Discussion. We achieve good results even by considering only axial symmetries and simply copying the image texture in the unknown region. When the symmetry structures com-



(a) Input image (b) Point cloud (c) Symmetries

(d) Completed image (e) Ground truth

Figure 8: *Eagle example: although the foreground is well completed, the background texture needs further or separate processing.*

port the holes, the completion of the foreground is neat (see Figures 1 and 9). The quality obtained in Figure 1 is a consequence of symmetry being present in the background also. Only in a detailed inspection (see Figure 1(d)), seams can be detected between visible and the reconstructed region. These seams can only be noted in the texture region and not in the background.

Our method completes images based on symmetries from the image's edges, and supposes that the object's texture is likely to follow the same transformation. However, this may not be the case. For example in Figure 8, the missing wing of the eagle was well reconstructed from the visible one, although the synthesized background differs in the tone of blue from the original one. A simple blending may solve this case.

When no symmetry structure comports the missing part, the method may fail in completing the hole. This is the case of the lizard body of Figure 6(b) and of the background in Figure 8: most of the missing pixels reflect out of image bounds. They could be completed using local inpainting methods [3, 17, 2]. A very interesting case is the turtle image of Figure 9, where the original symmetry structure does not cross the hole, although it can be extrapolated there. The completion does not match the ground truth since an extra limb was created for the turtle, but this is not easily noticed since the model's symmetry is strong. Such result would be delicate to obtain with texture-based inpainting due to the complex curved shapes involved.

5 Conclusions and Future Works

In this work, we propose to incorporate global structural information of an image into inpainting techniques. In particular, we present a method for inpainting images that deals with large unknown regions by using symmetries of the picture to complete it. This scheme is fully automated requiring from user only the specification of the hole. The current technique restricts itself to the analysis of axial symmetries of the image's edges, focusing on structure rather than texture. On the one hand, the transformation space can be easily

(a) Input image (b) Completed image (c) Ground truth

Figure 9: *Turtle example: although the completion differs from the original model, it is very coherent.*

(a) Completed image (b) Ground truth

Figure 10: *Flower example: texture elements are not yet considered in the analysis.*

extended using the same framework, incorporating translation, rotations and eventually projective transformations. On the other hand, texture descriptors could be used to improve both the symmetry detection and the image synthesis (see Figure 10). Moreover, the insertion of the synthesized parts into the image can be improved by existing inpainting techniques.

Acknowledgments

The authors would like to thank the CNPq for the grants "Projeto Universal" and "Bolsa de produtividade em pesquisa".

References

[1] M. Atallah. On symmetry detection. *Transactions on Computers*, 34(7):663–666, 1985.

[2] C. Barcelos, M. Batista, A. Martins and A. Nogueira. Level lines continuation based digital inpainting. In *Sibgrapi*, pages 50–57. IEEE, 2004.

[3] M. Bertalmio, G. Sapiro, V. Caselles and C. Ballester. Image inpainting. In *Siggraph*, pages 417–424. ACM, 2000.

[4] D. Comaniciu and P. Meer. Mean shift: a robust approach toward feature space analysis. *Pattern Analalysis and Machine Intelligence*, 24(5):603–619, 2002.

[5] A. Criminisi, P. Pérez and K. Toyama. Object removal by examplar-based inpainting. In *Computer Vision and Pattern Recognition*, volume 2, pages 721–728, 2003.

[6] I. Drori, D. Cohen-Or and H. Yeshurun. Fragment-based image completion. *Transactions on Graphics*, 22(3):303–312, 2003.

[7] A. A. Efros and W. T. Freeman. Image quilting for texture synthesis and transfer. In *Siggraph*, pages 341–346. ACM, 2001.

[8] R. Gal and D. Cohen-Or. Salient geometric features for partial shape matching and similarity. *Transactions on Graphics*, 25(1):130–150, 2006.

[9] P. Mattis and S. Kimball. Gimp, the GNU Image Manipulation Program, 2008.

[10] J. Hays and A. Efros. Scene completion using millions of photographs. In *Siggraph*, page 4. ACM, 2007.

[11] M. Kazhdan, T. Funkhouser and S. Rusinkiewicz. Symmetry descriptors and 3d shape matching. In *Symposium on Geometry Processing*, pages 115–123. ACM/Eurographics, 2004.

[12] G. Loy and J.-O. Eklundh. Detecting symmetry and symmetric constellations of features. In *European Conference on Computer Vision*, pages 508–521, 2006.

[13] G. Marola. On the detection of the axes of symmetry of symmetric and almost symmetric planar images. *Pattern Analysis and Machine Intelligence*, 11(1):104–108, 1989.

[14] N. Mitra, L. Guibas and M. Pauly. Partial and approximate symmetry detection for 3d geometry. *Transactions on Graphics*, 25(3):560–568, 2006.

[15] N. Mitra, L. Guibas and M. Pauly. Symmetrization. *Transactions on Graphics*, 26(3):63, 2007.

[16] P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. *Pattern Analysis and Machine Intelligence*, 12(7):629–639, 1990.

[17] F. Petronetto. Retoque digital. Master's thesis, *IMPA*, 2004. oriented by Luiz-Henrique de Figueiredo.

[18] J. Shen. Inpainting and the fundamental problem of image processing. *SIAM News*, 36(5), 2003.

[19] C. Sun. Symmetry detection using gradient information. *Pattern Recognition Letters*, 16(9):987–996, 1995.

[20] C. Sun and J. Sherrah. 3d symmetry detection using the extended gaussian image. *Pattern Analysis and Machine Intelligence*, 19(2):164–168, 1997.

[21] S. Tate. Symmetry and shape analysis for assembly-oriented cad. PhD thesis, *Cranfield University*, 2000. oriented by Graham Jared.

[22] S. Thrun and B. Wegbreit. Shape from symmetry. In *International Conference on Computer Vision*, pages 1824–1831. IEEE, 2005.

[23] J. Wolter, T. Woo and R. Volz. Optimal algorithms for symmetry detection in two and three dimensions. *The visual computer*, 1(1):37–48, 1985.

[24] H. Zabrodsky, S. Peleg and D. Avnir. Completion of occluded shapes using symmetry. In *Computer Vision and Pattern Recognition*, pages 678–679. IEEE, 1993.

[25] H. Zabrodsky, S. Peleg and D. Avnir. Symmetry as a continuous feature. *Pattern Analysis and Machine Intelligence*, 17(12):1154–1166, 1995.

[26] H. Zabrodsky and D. Weinshall. Using bilateral symmetry to improve 3d reconstruction from image sequences. *Computer Vision and Image Understanding*, 67(1):48–57, 1997.

The corresponding work was published in the proceedings of GRAPP 2009, INSTICC 2009.