

Chantal DING

Chloé MACUR

December 2013

SYMMETRY DETECTION

A Planar-Reflective Symmetry Transform for 3D Shapes

Project report

INF555 – Digital Representation and Analysis of Shapes



Contents

Introduction	3
1 A Planar-Reflective Symmetry Transform for 3D Shapes	3
2 Our implementation	3
2.1 Adjustments for 2D	3
2.2 Choice of the $1/N$ factor in the calculus of D	3
3 Results	3
4 Possible extensions	4
Conclusion	4
References	4

Introduction

A lot of 3D shapes, whether natural or man-made, present some kind of symmetry that can be really useful in computer vision and 3D geometry. Indeed symmetries allow certain economy, especially in digital representation, but they are also involved in pattern recognition or geometry completion. Thus, numerous methods are used to detect symmetries, partly because of the diversity of datas (point clouds, polygon meshes, NURBS, patches, etc.).

between existing methods to gain a better understanding of a fundamental problem in digital geometry processing and shape understanding in general. We discuss a variety of applications in computer graphics and geometry that benefit from symmetry information for more effective processing. An analysis of the strengths and limitations of existing algorithms highlights the plenitude of opportunities for future research both in terms of theory and applications.

1 A Planar-Reflective Symmetry Transform for 3D Shapes

· 1 page description of the method you're implementing. (*A Planar-Reflective Symmetry Transform for 3D Shapes paper*) It goes without saying that you shouldn't copy text from the paper, but rather describe in your own words what the method is trying to do and how it does it.

We chose to try to implement the planar reflective symmetry transform (PRST) described by Podolak et al. [2].

Monte Carlo bla bla évite brute force, sélection intelligente selon l'énergie de la fonction

2 Our implementation

· 2 page description of your implementation. Here you should describe the main building blocks of your implementation. We are especially interested in: whether you had any problems, whether there were things not mentioned in the paper that you had to discover or derive yourself (be very explicit about your own work!), whether you used any external libraries, etc. What we're not interested in: what are the names of your classes and variables, what operating system you were using, if you had to change some header files, etc. Whenever possible (which is most of the time), please try to use images instead of text to explain concepts.

Your project should include some amount of independent work, either by implementing a technique and showing its performance on some examples not included by the authors, or by doing some independent theoretical analysis.

2.1 Adjustments for 2D

the polar issue, formulas different, etc...

2.2 Choice of the $1/N$ factor in the calculus of D

explain that it comes out of the blue ...

3 Results

· 1.5-2 pages of results. Show (especially in, graphs, screenshots, etc.) the results that you have obtained. Comment on discrepancies (if any) with the results shown in the paper. Comment on whether you had to tweak parameters to get good results and, if so, how you picked them.

4 Possible extensions

· 0.5-1 pages of possible extensions. Can you suggest how the method can be improved? Can you suggest other application domains for your method?

Conclusion

References

- [1] Niloy J. Mitra, Mark Pauly, Michael Wand, and Duygu Ceylan. Symmetry in 3d geometry: Extraction and applications. In *EUROGRAPHICS State-of-the-art Report*, 2012.
- [2] Joshua Podolak, Philip Shilane, Aleksey Golovinskiy, Szymon Rusinkiewicz, and Thomas Funkhouser. A planar-reflective symmetry transform for 3D shapes. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, 25(3), July 2006.