Computational symmetry

Yanxi Liu
The Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213, U.S.A.

Introduction

Symmetry is pervasive in both natural and man-made environments [1–7]. Humans have an innate ability to perceive and take advantage of symmetry [8] in everyday life, but it is not obvious how to automate this powerful insight. The introduction of computers poses challenging tasks for machine representation and reasoning about symmetry and group theory. I make continuous efforts to develop computational tools for dealing with symmetry in various applications using computers [9–11,21,34,35].

This chapter gives a sampler of an emerging area of research and applications, namely *computational symmetry*. Computational symmetry refers to the practice of representing, detecting and reasoning about symmetries on computers. The reasons to care about computational symmetry in computer science are many-fold: (i) symmetry *exists* everywhere; (ii) symmetry is intellectually *stimulating*; (iii) symmetry implies a structure that can be either *helpful* or *harmful* in applications; (iv) machine computation of symmetry is *challenging*, as it has to connect abstract mathematics with the noisy, imperfect, real world; and (v) few computational tools exist for dealing with real-world symmetries.

I demonstrate, through three concrete applications, the power, the difficulties and the feasibility of using symmetry and group theory on computers. These applications are a robot assembly planner, an intelligent neuroradiology image database and a computational model for periodic pattern perception. A computational framework is proposed to study symmetry in a multi-dimensional, continuous space.

A group-theoretical formalization of surface contact

One basic question in robotics automation is how do you describe contacts between solids to a robot? For example, how would you ask a robot to put a cube in a corner? This seemingly simple task requires 24 equivalent, but different, sets of task specifications if you wish to enumerate all the geometric possibilities. It is a non-trivial task to communicate the full range of spatial relationships between locally symmetrical objects with a robot that does not understand symmetry. Such task specifications are forced to be either tedious and redundant, or suffering from incompleteness. Current engineering practice is still limited to a finite set of case-based scenarios. Computing the relative positions of solids that are in contact is a fundamental problem in many fields, including robotics, computer graphics, computer-aided design and manufacturing, and computer vision. It is the focus of

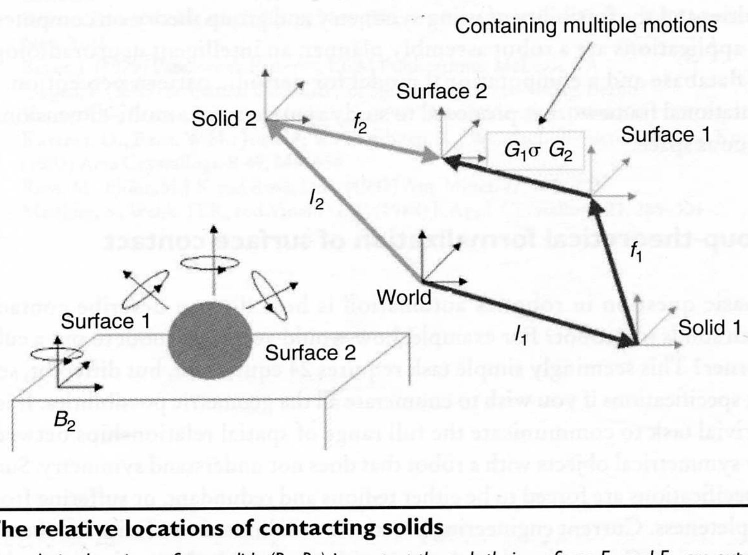
E-mail yanxi@cs.cmu.edu

this work to formalize solid contact based on local symmetry, to construct a computational framework using group theory, and to demonstrate the effectiveness of applications of computational group theory in robotics [9,10]. It can be shown (Figure 1) that there is a direct relationship between the relative locations of two solids in contact and the symmetry groups of their contacting surfaces. Furthermore, it can be proven [11] that the most basic group operation is symmetry group intersection. The computational challenge is to find out, on computers, (i) how to denote symmetry groups, which can be finite, infinite, discrete or continuous subgroups of the proper Euclidean group $\epsilon+$; and (ii) how to compute these subgroup intersections in Euclidean space under different locations and orientations efficiently in non-exponential time.

We have employed a geometric approach to denote and intersect an important family of subgroups of the Euclidean group $E+$. They are called TR groups, defined as a semi-direct product $G=TR$, where T and R are translation and rotation subgroups of $E+$ respectively. By mapping a TR group to a pair of translation and rotation characteristic invariants, the intersection of two subgroups can be done geometrically. We have developed and implemented a group-intersection algorithm that has been proven to be correct and efficient (Figure 2).

As an application platform for our group-theoretical formalization of surface contact among solids, a Kinematic Assembly planning system KA3 (Figures 3 and 4) [12–16] has been implemented. A designed assembly is input to KA3 as a set of computer-aided-design models (boundary files from solid modelling systems ACIS and PADL2) of individual parts and symbolic relationships. KA3 generates a partial ordered precedence graph with symmetry

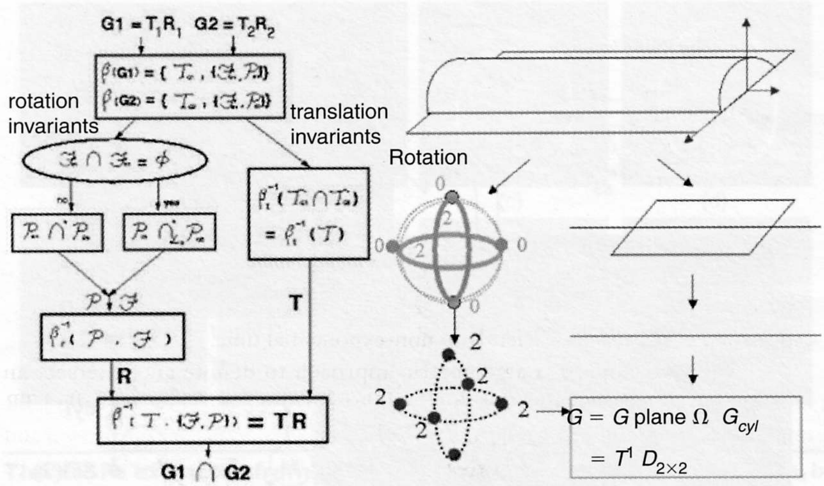
Figure 1



The relative locations of contacting solids

The relative locations of two solids (B_1, B_2) in contact through their surfaces F_1 and F_2, are expressed in terms of their respective symmetry groups G_1 and G_2: $l_1^{-1} l_2 \in f_1 G_1 \sigma G_2 f_2^{-1}$, where l_1 and l_2 specify the locations of solids B_1 and B_2 in the world co-ordinate system and f_1 and f_2 specify the locations of F_1 and F_2 in their respective body co-ordinates. σ is a transformation bringing the two co-ordinates together.

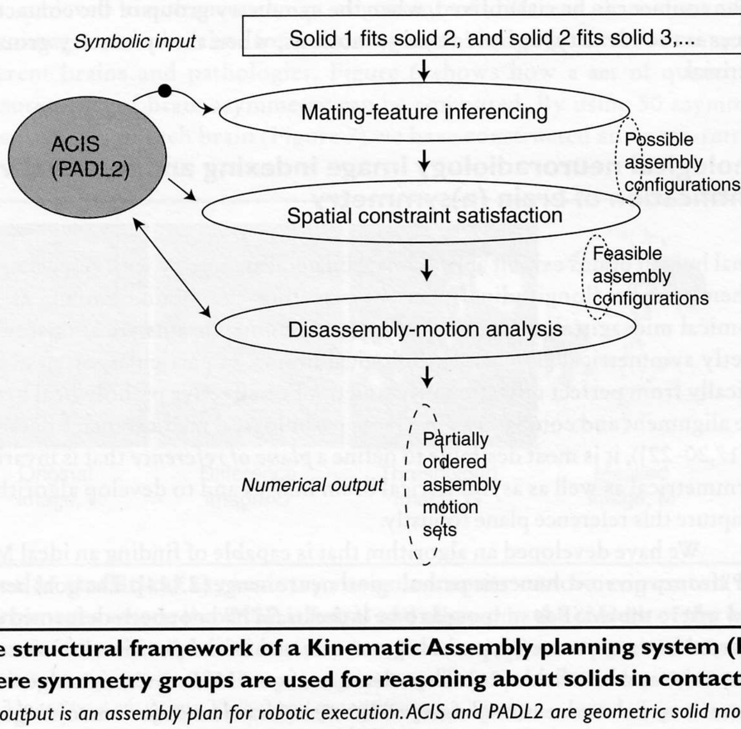
Figure 2



Group intersection

Left: TR group-intersection algorithm. Right: an example of the intersection of two TR groups, symmetry groups of a plane (G_{plane}) and a cylinder (G_{cyl}) [9,11]. This is an $O(n^2)$ algorithm, where n is the number of countable poles.

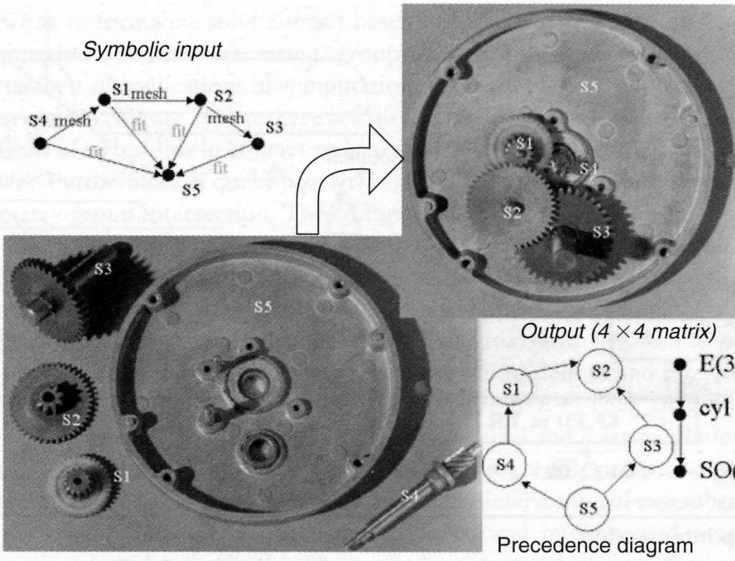
Figure 3



The structural framework of a Kinematic Assembly planning system (KA3) where symmetry groups are used for reasoning about solids in contacts

The output is an assembly plan for robotic execution. ACIS and PADL2 are geometric solid modelling systems, providing boundary files of individual assembly parts for KA3.

Figure 4



KA3 analyses the spatial and kinematic relations of a gearbox

Where E(3) is a three-dimensional Euclidean group, cyl is the symmetry group of a cylindrical surface and SO(2) is a special orthogonal group (see [9,11] for details).

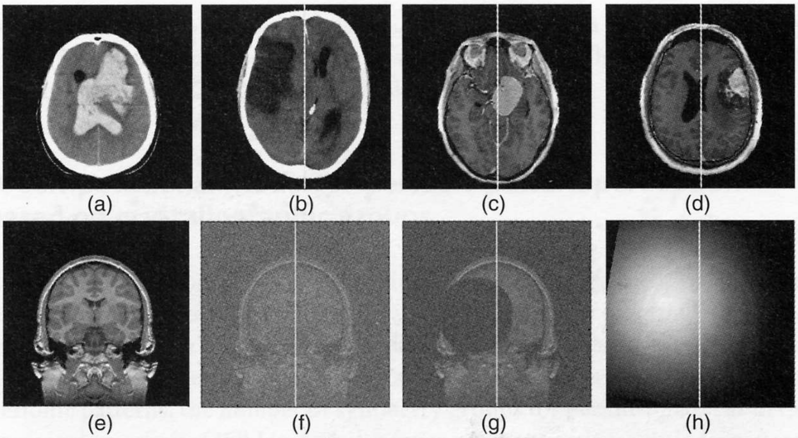
groups and homogeneous transformation matrices attached to each contact. Note that the contact can be either fixed, when the symmetry group of the contacting surfaces is the identity, or have relative motions, when the symmetry group is non-trivial.

Pathological neuroradiology image indexing and retrieval via quantification of brain (a)symmetry

Normal human brains exhibit approximate bilateral symmetry with respect to the interhemispheric (longitudinal) fissure bisecting the brain, known as the anatomical midsagittal plane (MSP). However, human brains are almost never perfectly symmetrical [17–19]. Pathological brains, in particular, often depart drastically from perfect reflectional symmetry. For effective pathological brain-image alignment and comparison in a large pathological medical image database (e.g. [17,20–22]), it is most desirable to define a *plane of reference* that is invariant for symmetrical as well as asymmetrical brain images and to develop algorithms that capture this reference plane robustly.

We have developed an algorithm that is capable of finding an ideal MSP (iMSP) from a given volumetric pathological neuroimage [23,24]. The goal here is to find where the iMSP is supposed to be if the brain had not been deformed due to internal brain asymmetry, pathology or external initial position/orientation offsets, noise and bias fields [24]. The tolerance of our iMSP-extraction algorithm to these internal and external factors in various volumetric neuroimages is demonstrated in Figure 5.

Figure 5

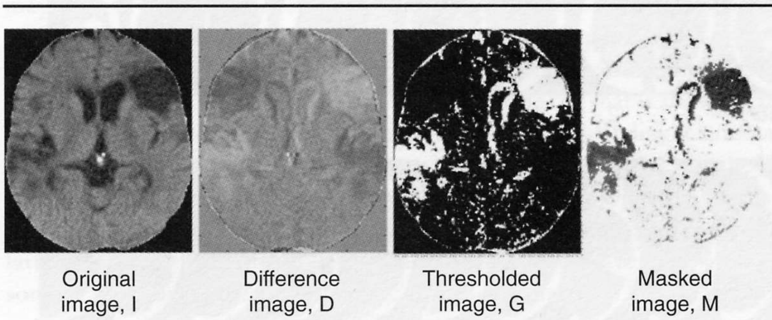


The iMSPs extracted from different clinical three-dimensional CT (a, b) and MR images (c, d)

*The two-dimensional line is the intersection of the iMSP and the given two-dimensional brain slice. (**e**) One MR brain slice without noise. (**f**) On a dataset with added noise; the signal-to-noise ratio of breaking point is −10.84 dB. (**g**) On a dataset with an artificial lesion plus noise; the signal-to-noise ratio of breaking point (the point where the iMSP extraction algorithm fails to find the correct plane) is −4.82 dB. (**h**) On a dataset with an added bias field of G= 10, from which our algorithm still finds the iMSP correctly [24]. CT, computed tomography; MR, magnetic resonance.*

After the iMSP is identified for each three-dimensional brain image, we have achieved simultaneously an alignment of different three-dimensional neuroimages and a baseline for extracting useful image features for comparing different brains and pathologies. Figure 6 shows how a set of quantitative measurements of brain asymmetry can be computed. By using 50 asymmetry measurements of each brain (Figure 7) we have constructed an image-retrieval

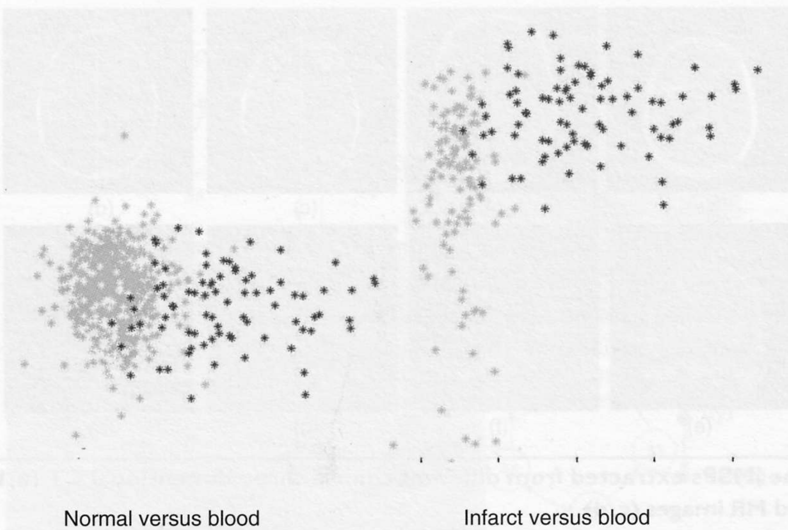
Figure 6



A set of statistical asymmetry measurements can be computed from neuroimages where the iMSP is found and aligned in the middle of the image

Features include: multi-scaled statistical properties (mean ± S.D.); x- and y-gradients of grey-level intensity of image I, D, G and M. Image I, original image with centred iMSP; image D, the intensity difference between image I and vertically reflected image I; image G, thresholded image D to a binary image; image M, the product of images I and G.

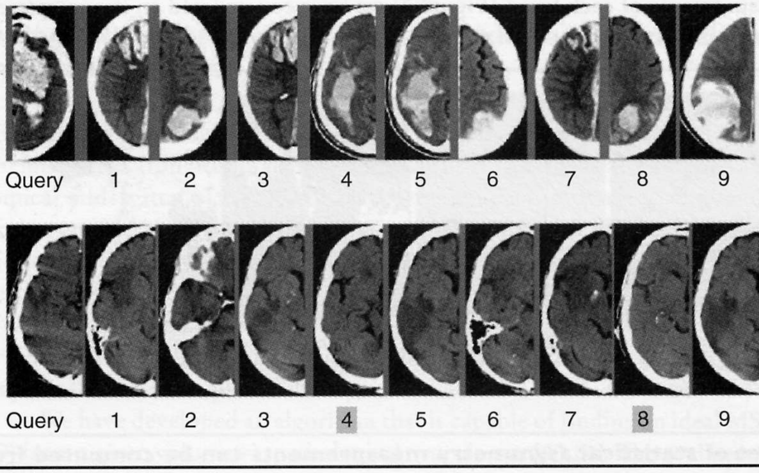
Figure 7



The 50 dimensional asymmetry measurements of each brain in the database are projected on to a plane

Separations can be observed between the distributions of asymmetry measurements of normal- and blood-, infarct- and blood-typed brains.

Figure 8



Classification-driven semantic-based image retrieval results: the top nine most similar images to the query image drawn from a database of 1200 images

The query image on the top row is an acute blood case and the one on the bottom row is an infarct case. Shaded labels indicate misclassified images.

system to find the most-similar images in the database for a given queried image. Figure 8 displays two sample retrieval results. The system achieves around an 80% average true positive rate during retrieval [20,21].

A computational model for periodic pattern perception based on crystallographic groups

A mature mathematical theory for periodic patterns has been known for over a century [25–27], namely the crystallographic groups. These are groups composed of symmetries of periodic patterns in n dimensional Euclidean space. The amazing result is that regardless of the value of n and the fact that there are infinite possible periodic patterns, the number of symmetry groups for periodic patterns in that space is always finite [25]! In particular, for monochrome planar periodic patterns, there are seven *frieze groups* [28,29] for two-dimensional patterns repeated along one dimension (strip patterns), 17 *wallpaper groups* [30] describing patterns extended by two linearly independent translational generators (wallpaper patterns) and 230 *space groups* [31,32] extended by three linearly independent translations (regular crystal patterns).

It is the goal of this research to construct a computational model for periodic pattern perception and analysis based on the theory of crystallographic groups. Given the digital form of a periodic pattern, a computer can discover its underlying lattice, its symmetry group, its motifs *and* what other symmetry groups it can be associated with when the pattern undergoes affine deformations [33,34].

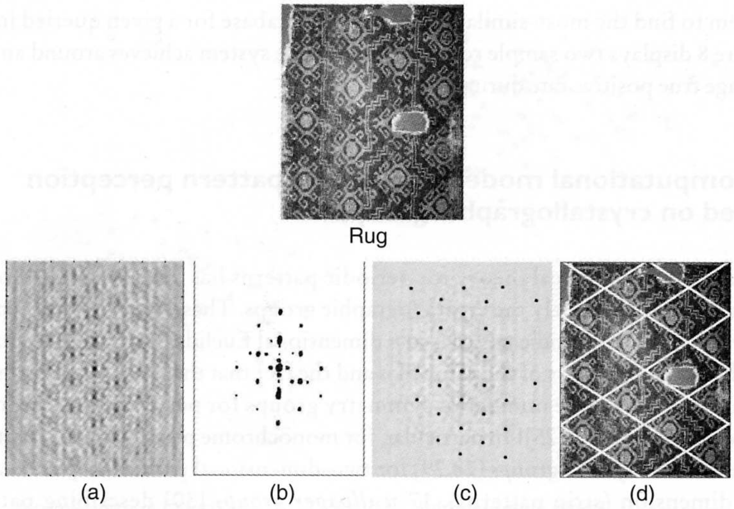
Automatic lattice extraction

Autocorrelation of a given periodic pattern, which may only contain two or three cycles and lots of noise, is used to detect the underlying lattice structure. Even noise-free computer-generated patterns can cause problems for lattice-detection algorithms. Halfway between actual lattice translations, the large sub-patterns may partially match smaller sub-patterns interspersed between them, causing spurious peaks to form. Furthermore, these spurious peaks can have a higher value than actual peaks located at the periphery of the autocorrelation image. Figure 9(a) shows an autocorrelation surface for the rug that is shown at the top. Although the grid of peaks is apparent to the human eye, finding it automatically is very difficult. Simple approaches such as setting a global threshold yield spurious results (Figure 9b). We used a novel peak-detection algorithm based on "regions of dominance" [33] to automatically detect the underlying translational lattice. The trouble is that many legitimate grid peaks have a lower value than some of the spurious peaks. Figure 9(c) represents the first 32 peaks found by our peak detection algorithm. Figure 9(d) shows the formed lattice.

Symmetry-group classification

Table 1 lists the eight symmetries checked in the classification algorithm. The determination of a specific rotation or reflection or glide-reflection symmetry is performed with respect to the unit lattice orientations found by applying the symmetry to be tested to the entire pattern, then checking the similarity between the original and transformed images.

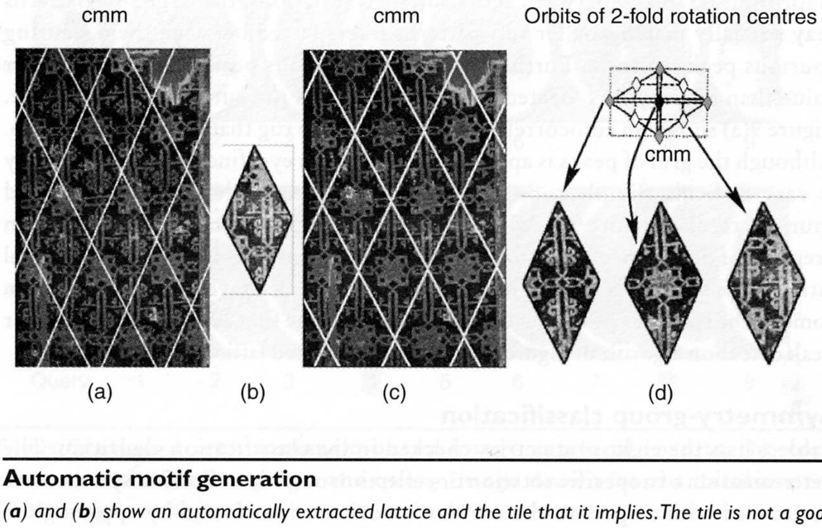
Figure 9



Automatic lattice extraction

*An oriental rug and (**a**) its autocorrelation surface, (**b**) peaks found using a global threshold, (**c**) the 32 most-dominant peaks found using our approach described in the text and (**d**) the detected lattice.*

Figure 10



Automatic motif generation

*(**a**) and (**b**) show an automatically extracted lattice and the tile that it implies. The tile is not a good representation of the pattern motif. (**c**) shows the lattice positioned in one of the three orbits of 2-fold rotation centres in symmetry group cmm, and (**d**) displays the three most symmetric motifs found.*

Table 1

	p1	*p2*	*pm*	*pg*	*cm*	*pmm*	*pmg*	*pgg*	*cmm*	*p4*	*p4m*	*p4g*	*p3*	*p3m1*	*p31m*	*p6*	*p6m*
2		Y				Y	Y	Y	Y	Y	Y	Y					
3													Y	Y	Y	Y	Y
4										Y	Y	Y					
6																Y	Y
T1			Y	Y(g)		Y	Y(g)	Y(g)			Y	Y(g)			Y		Y
T2						Y	Y	Y(g)			Y	Y(g)			Y		Y
D1					Y				Y		Y	Y		Y	Y		Y
D2									Y		Y	Y					Y

Wallpaper-group classification

Numbers 2, 3, 4 or 6 denote n-fold rotational symmetry, and Tx or Dx denote reflectional symmetry about one of the translation or diagonal vectors of the unit lattice. Y means the symmetry exists for that symmetry group; an empty space means it does not. Y(g) denotes the existence of a glide reflection.

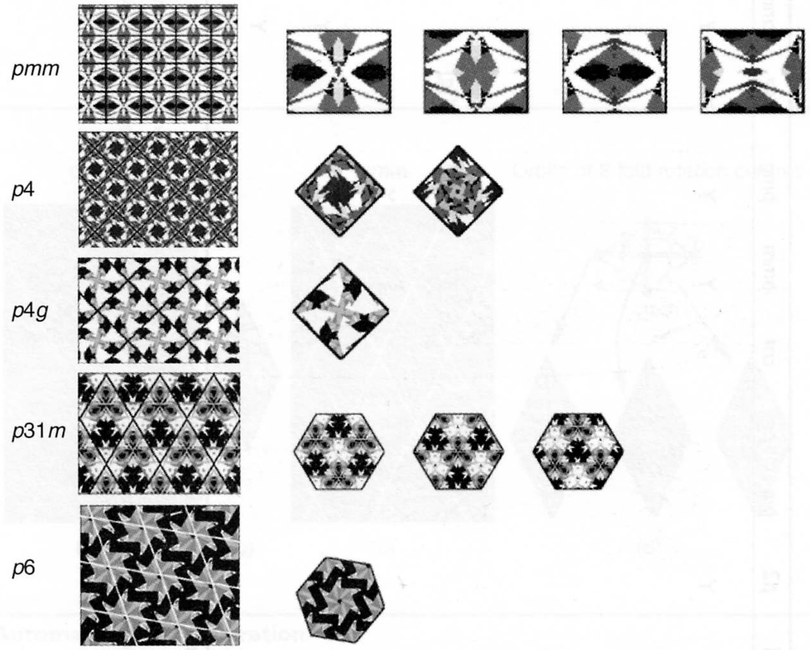
It is interesting to notice the difference between the symmetry-group classification flow charts for humans [6,30] and the computer's classification algorithm. The computer has first to find the underlying lattice structure (but not necessarily where it is anchored), whereas humans do that implicitly, so the first question for them is: What is the smallest rotation?

Automatic motif generation

Choosing a good motif should help one see, from a single tile, what the pattern looks like (Figure 10). From work in perceptual grouping, it is known that the human perceptual system often has a preference for symmetric figures [8].

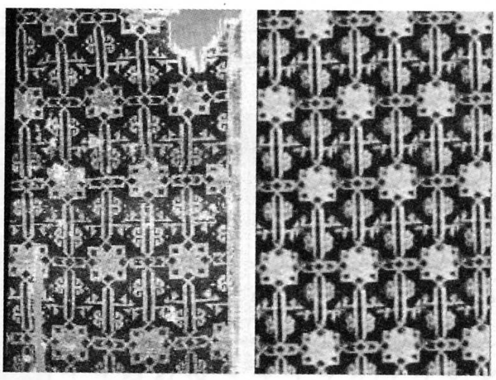
If we entertain the idea that the most representative motif is the one that is the most symmetrical, one plausible strategy for generating motifs is to align the motif centre with the centre of the highest-order rotation in the pattern. Candidate motifs can then be determined systematically by enumerating each distinct centre point of the highest-order rotation. Two rotation centres are distinct if they lie in different *orbits* of the symmetry group [33]. Figure 11 shows a set of symmetrical motifs from periodic patterns of various symmetry groups. *Approximate symmetries* in a pattern are used to fix the unconstrained lattice structure for symmetry groups like *pm*, *pg* and *cm* or *p*1 that do not have rotation centres [33]. Aside from motif selection, knowledge of the lattice structure of a repeated pattern allows us to determine which pixels in

Figure 11



Automatically detected lattices and motifs for some of the 17 wallpaper groups

Figure 12



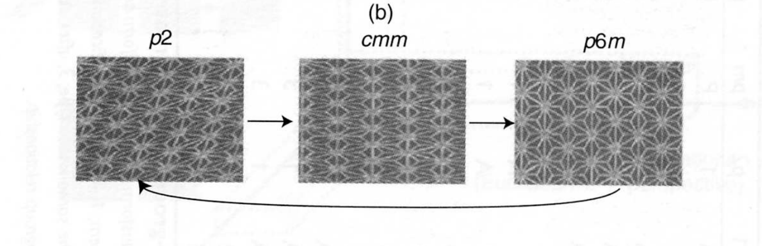
Real oriental rug and a perfectly symmetric virtual rug formed by translating the median tile

an image should look the same. Taking the median of corresponding pixels across the multiple tiles of the rug image, for example, creates a 'median tile' with noise and irregularities filtered out. Figure 12 compares the original worn rug with a virtual rug generated from the median tile.

Skewed symmetry groups

Table 2 is a transition matrix where each entry indicates whether the row group can be affinely transformed into a column group. It was discovered that symmetry groups of periodic patterns form small orbits (between two and four members) when they are affinely skewed (Figure 13). A classification algorithm has been

Figure 13



While the pattern is deformed by affine transformations its symmetry group migrates to different groups within its orbit (see also Table 2)
*(**a**) p2→pmm→cmm→p4m; (**b**) p2→cmm→p6m.*

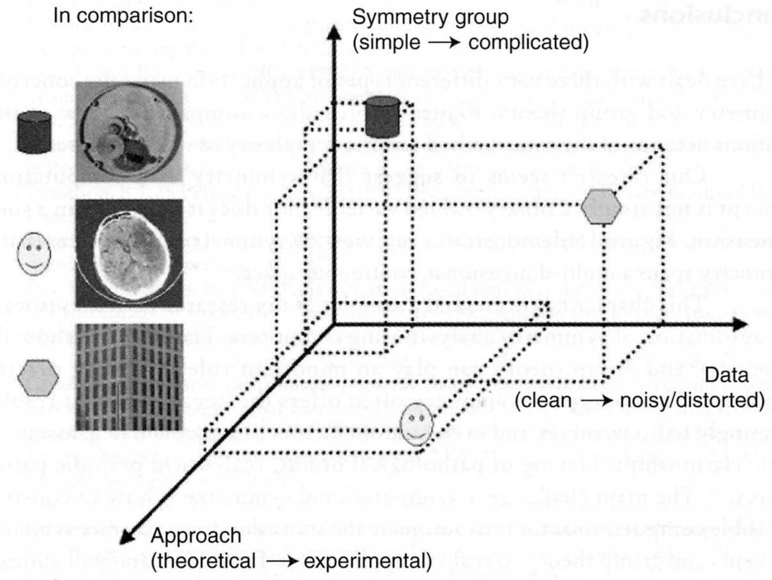
Table 2

	p1	*p2*	*pm*	*pg*	*cm*	*pmm*	*pmg*	*pgg*	*cmm*	*p4*	*p4m*	*p4g*	*p3*	*p3m1*	*p31m*	*p6*	*p6m*
p1	A	1	P	P	P	1	1	1	1	1	1	1	P	P	P	1	1
p2	1	A	1	1	1	P	P	P	P	P	P	P	1	1	1	P	P
pm	A	1	N	2	3	1	1	1	1	1	1	1	4	3	3	1	1
pg	A	1	2	N	3	1	1	1	1	1	1	1	4	3	3	1	1
cm	A	1	3	3	N	1	1	1	1	1	1	1	4	P	P	1	1
pmm	1	A	1	1	1	N	2	2	P	3	P	3	1	1	1	4	3
pmg	1	A	1	1	1	2	N	2	3	3	3	3	1	1	1	4	3
pgg	1	A	1	1	1	2	2	N	P	3	3	P	1	1	1	4	3
cmm	1	A	1	1	1	P	3	P	N	3	P	P	1	1	1	3	P
p4	1	A	1	1	1	3	3	3	3	S	2	2	1	1	1	4	3
p4m	1	A	1	1	1	N	3	3	N	2	S	2	1	1	1	3	3
p4g	1	A	1	1	1	3	3	N	N	2	2	S	1	1	1	3	4
p3	A	1	4	4	4	1	1	1	1	1	1	1	S	2	2	1	1
p3m1	A	1	3	3	N	1	1	1	1	1	1	1	2	S	2	1	1
p31m	A	1	3	3	N	1	1	1	1	1	1	1	2	2	S	1	1
p6	1	A	1	1	1	4	4	4	3	4	3	3	1	1	1	S	2
p6m	1	A	1	1	1	3	3	3	N	3	3	4	1	1	1	2	S

Wallpaper-group transition matrix under affine transformations

S, similarity transformation; N, non-uniform scaling perpendicular or parallel to all reflection axes in the group to the left; A, general affine transformation other than S or N; P, depending on the particular pattern; 1–4, a labelled justification of why no affine deformation can possibly transform the row group to the column group: 1, G_1 has a 2-fold rotation symmetry but G_2 does not; 2, G_1 and G_2 have the same lattice type; 3, after deforming the lattice type of G_1 to that of G_2, at least one remaining symmetry of the deformed G_1 differs from all symmetries of G_2; 4, G_1 and G_2 do not have a subgroup relationship.

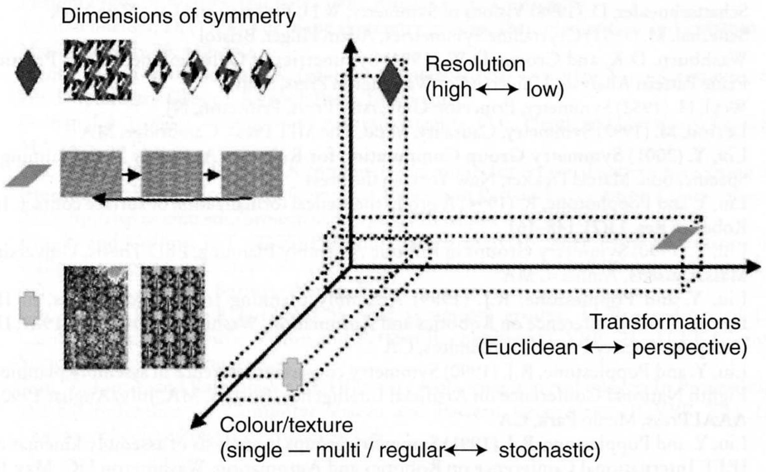
Figure 14



A perpective view of the three examples reported

developed to evaluate the potential skewed symmetry groups of a given pattern [33,34]. The practical value of this result in computer vision includes a new principled measure for potential symmetry, indexing and retrieval of regular patterns, estimation of shape and orientation from texture, and replacement of regular patterns from real images [35,36].

Figure 15



Symmetry spans a continuous, multi-dimensional space

Conclusions

We have dealt with three very different types of application using the concept of symmetry and group theory. Figure 14 provides a comparative view of their relations in terms of the data, method and the complexity of symmetry treated.

Our research seems to suggest that symmetry as a computational concept is not simply a binary-valued variable, nor does it only vary in a single dimension. Figure 15 demonstrates my view of symmetry in practice: that is, symmetry spans a multi-dimensional, continuous space.

This chapter has presented a sampler of my research activities towards the automation of symmetry analysis using computers. These results show that symmetry and group theory can play an important role in solving practical problems. The concept of symmetry often offers the key insight that resolves seemingly tedious, messy and even random factors in a problem (e.g. assembly-part relationships, idexing of pathological brains, real-world periodic pattern analysis). The main challenge in computational symmetry is how to construct plausible computational tools to automate the transition from abstract symmetry concepts and group theory to realistic applications. It has been and will continue to be a challenging yet rewarding process.

I thank all my collaborators and express my immense gratitude towards those who have provided advice, inspiration and support during my pursuit of computational symmetry, including (but not limited to): Professor M. Senechal, Professor D. Schattschneider, Professor M. Leyton, Professor D. Crowe, Professor D. Washburn, Professor T. Kanade and Professor H.S.M. Coxeter.

References

1. Feynman, R.P. (1998) Six Not-So-Easy Pieces: Einstein's Relativity, Symmetry, and Space-Time, Perseus Press, Cambridge, MA
2. Hargittai, I. and Hargittai, M. (1994) Symmetry: a Unifying Concept, Shelter Publications, Bolinas, CA
3. Hargittai, I. and Hargittai, M. (2000) In Our Own Image: Personal Symmetry in Discovery, Kluwer Academic/Plenum Publishers, New York
4. Schattschneider, D. (1998) Visions of Symmetry, W.H. Freeman and Co., New York
5. Senechal, M. (1991) Crystalline Symmetries, Adam Hilger, Bristol
6. Washburn, D.K. and Crowe, D.W. (1991) Symmetries of Culture: Theory and Practice of Plane Pattern Analysis, University of Washington Press, Seattle
7. Weyl, H. (1952) Symmetry, Princeton University Press, Princeton, NJ
8. Leyton, M. (1992) Symmetry, Causality, Mind, The MIT Press, Cambridge, MA
9. Liu, Y. (2001) Symmetry Group Computation for Robotics Assembly Task Planning and Specification, Marcel Dekker, New York, in the press
10. Liu, Y. and Popplestone, R. (1994) A group theoretical formalization of surface contact. Int. J. Robotics Res. **13(2)**, 148–161
11. Liu, Y. (1990) Symmetry Groups in Robotic Assembly Planning, PhD Thesis, University of Massachusetts, Amherst, MA
12. Liu, Y. and Popplestone, R.J. (1989) Assembly planning from solid models, in IEEE International Conference on Robotics and Automation, Washington DC, May 1989, IEEE Computer Society Press, Los Alamitos, CA
13. Liu, Y. and Popplestone, R.J. (1990) Symmetry constraint inference in assembly planning, in Eighth National Conference on Artificial Intelligence, Boston, MA, July/August 1990 The AAAI Press, Menlo Park, CA
14. Liu, Y. and Popplestone, R.J. (1991) Symmetry groups in analysis of assembly kinematics, in IEEE International Conference on Robotics and Automation, Washington DC, May 1991, IEEE Computer Society Press, Los Alamitos, CA

15. Liu, Y. and Popplestone, R.J. (1992) From symmetry groups to stiffness matrices, in IEEE International Conference on Robotics and Automation, Washington DC, May 1992, IEEE Computer Society Press, Los Alamitos, CA
16. Liu, Y. and Popplestone, R.J. (1992) Symmetry groups in solid model-based assembly planning, in Artificial Intelligence Applications in Manufacturing (Famili, A., Nan, D. and Kin, S.H., eds), pp. 103–131, AAAI Press and The MIT Press, Cambridge, MA
17. Crow, T.J. (1993) Schizophrenia as an anomaly of cerebral asymmetry, in Imaging of the Brain in Psychiatry and Related Fields (Maurer, K., ed.), Springer-Verlag, Berlin
18. Davidson, R.J. and Hugdahl, K. (eds) (1996) Brain Asymmetry, MIT Press/Bradford Books, Cambridge, MA
19. Geschwind, N. and Levitsky, W. (1968) Human brain: left-right asymmetries in temporal speech region. Science **161**, 186–187
20. Liu, Y. and Dellaert, F. (1998) A classification-based similarity metric for 3D image retrieval, in Proceedings of Computer Vision and Pattern Recognition Conference, June 1998, pp. 800–805, IEEE Computer Society Press, Los Alamitos, CA
21. Liu, Y., Dellaert, F., Rothfus, W., Schneider, J., Moore, A. and Kanade, T. (2001) Classification-driven neuroimage retrieval using statistical asymmetry measures. In Proceedings of International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI 2001), 13-17 October 2001, Utrecht, The Netherlands, Lecture Notes in Computer Science series, vol. 2208 (Niessen, W. and Viergever, M. eds), pp. 655–665, Springer-Verlag, Berlin
22. Liu, Y., Rothfus, W.E. and Kanade, T. (1998) Content-based 3d neuroradiologic image retrieval: Preliminary results. IEEE workshop on Content-based access of Image and Video Databases in conjunction with International Conference of Computer Vision (ICCV'98), January 1998, IEEE Computer Society Press, Los Alamitos, CA
23. Liu, Y., Collins, R.T. and Rothfus, W.E. (2000) Evaluation of a robust midsagittal plane extraction algorithm for coarse, pathological 3d images, in Medical Imaging Computing and Comptuer Assisted Intervention (MICCAI 2000), October 2000 (Delp, S.L., DiGioia, T. and Jaramaz, G., eds), pp. 81–94, Springer-Verlag, Berlin
24. Liu, Y., Collins, R. and Rothfus, W. (2001) Automatic extraction of ideal midsagittal plane from normal and pathological neuroradiology images. IEEE Trans. Med. Imaging **20(3)**, 175–192
25. Bieberbach, L. (1910) Über die Bewegungsgruppen der n-dimensional en Euklidischen Räume mit einem endlichen Fundamental bereich. Göttinger Nachrichten 75–84
26. Fedorov, E.S. (1891) Symmetry in the plane [in Russian]. Zapiski Imperatorskogo S. Peterburgskogo Mineralogicheskogo Obshchestva [Proc. S. Peterb. Mineral. Soc.] **(2)28**, 345–390
27. Grünbaum, B. and Shephard, G.C. (1987) Tilings and Patterns, W.H. Freeman and Co., New York
28. Jones, O. (1972) The Grammar of Ornament, Van Nostrand Reinhold, New York; first published in 1856
29. Shubnikov, A.V. and Koptsik, V.A. (1974) Symmetry in Science and Art (Harker, D., ed.), Plenum Press, New York; translated from Russian by G.D. Archard
30. Schattschneider, D. (1978) The plane symmetry groups: their recognition and notation. Am. Math. Monthly **85**, 439–450
31. Henry, N.F.M. and Lonsdale, K. (eds) (1969) International Tables for X-ray Crystallography, vol. 1, Symmetry Groups, Kynoch Press, Birmingham
32. Gallian, J.A. (1986) Contemporary Abstract Algebra, D.C. Heath and Company, Lexington, MA
33. Liu, Y. and Collins, R.T. (2000) A computational model for repeated pattern perception using frieze and wallpaper groups, in Computer Vision and Pattern Recognition Conference, June 2000; ftp://ftp.cs.cmu.edu/project/nist/cvpr2000final.ppt
34. Liu, Y. and Collins, R.T. (2000) Periodic pattern analysis under affine distortions using wallpaper groups, in Lecture Notes in Computer Science, vol. 1888 (Sommer, G. and Zeevi, Y.Y., eds), pp. 241–250, Springer-Verlag, Berlin
35. Liu, Y. and Collins, R.T. (2001) Skewed symmetry groups. In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR '01), IEEE Computer Society Press, Los Alamitos, CA
36. Tsin, Y., Liu, Y. and Ramesh, V. (2001) Texture replacement in real images. In Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR '01), IEEE Computer Society Press, Los Alamitos, CA