Detecting Symmetry in Cellular Automata Generated Patterns Using Swarm Intelligence

Mohammad Ali Javaheri Javid, Mohammad Majid al-Rifaie,
and Robert Zimmer

Department of Computing
Goldsmiths, University of London
London SE14 6NW, UK
{m.javaheri,m.majid,r.zimmer}@gold.ac.uk

Abstract. Since the introduction of cellular automata in the late 1940's they have been used to address various types of problems in computer science and other multidisciplinary fields. Their generative capabilities have been used for simulating and modelling various natural, physical and chemical phenomena. Besides these applications, the lattice grid of cellular automata has been providing a by-product interface to generate graphical patterns for digital art creation. One important aspect of cellular automata is symmetry, detecting of which is often a difficult task and computationally expensive. In this paper a swarm intelligence algorithm – Stochastic Diffusion Search – is proposed as a tool to identify axes of symmetry in the cellular automata generated patterns.

Keywords: Cellular automata, swarm intelligence, symmetry, aesthetics.

1 Introduction

Creating aesthetically pleasing images has been investigated by many researches in the context of evolutionary computing, including the Bimorphs of Dawkins [9], Mutator of Latham [30], and Virtual Creatures of Sims [29]. Although some impressive results have been achieved, there still remain problems with the aesthetic selection. According to [18], first, the subjective comparison process, even for a small number of phenotypes, is slow and forms a bottleneck in the evolutionary process. Human users would take hours to evaluate many successive generations that in an automated system could be performed in a matter of seconds. Secondly, genotype-phenotype mappings are often not linear or uniform. That is, a minor change in genotype may produce a radical change in phenotype. Such non-uniformities are particularly common in tree or graph based genotype representations such as in evolutionary programming, where changes to nodes can have a radical effect on the resultant phenotype. In this study we approach the problem in the framework of dynamical systems and define a criterion for aesthetic selection in terms of its association with symmetry. The association of aesthetics and symmetry has been investigated from different points of view.

In this work, a brief account on cellular automata is presented, followed by a section on symmetry and its significance in aesthetics. Then a swarm intelligence algorithm – Stochastic Diffusion Search – is explained, highlighting its main features. Afterwards, the application of the algorithm in detecting symmetry along various axes of symmetry is detailed, illustrating the performance of the method proposed.

2   Cellular Automata

Definition 1: *A Cellular Automaton (CA) is a lattice of regularly arranged homogeneous finite state automaton as unit cells in Euclidean space.* It can be represented as a quadruple of $A = \{\, S, N, d, f \,\}$ where:

1. S is a finite set of integer numbers as *states*,
2. N is a finite set of integer numbers as *neighbourhood*,
3. d is a finite set of integer numbers as the *dimension* of the space,
4. $f : S^n \mapsto S$ is the *transition function*.

In a *discrete two-dimensional (d = 2) finite lattice* with a *periodic boundary* the state of each cell (automaton) at time (t) is determined by the states of immediate surrounding neighbourhood cells at time $t - 1$.

The 9-cell mapping is also known as a *Moor* neighbourhood and a mapping that satisfies the following condition is called a *quiescent state* ($S = 0$).

$$f(0, 0, 0, 0, 0, 0, 0, 0, 0) = 0 \tag{1}$$

The behaviour of CA at a certain point of time emergences from a *synchronous* iterative application of transition function (local rule) over the initial configuration at time t_0. There are some distinctive characteristics in CA which can make them particularly attractive to digital artists and suitable for image and pattern generation purposes (each automaton acting as picture element). Furthermore, the significance of CA for computer art comes from the fact that simple rules can generate observationally unpredictable complex behaviours and there is a vast universe of behaviours which can be explored. Generally the behaviour of a particular cellular automaton is constrained by its initial configuration, transaction function and number of states. A two-dimensional multi-state cellular automaton with periodic boundary provides an endless environment for the growth of patterns and the observation of emergent complex behaviour over the time of evolution. For some rules the periodic generation of patterns creates an animated sequence of pattern formations. This opens up possibility of generating animations based on the development of pattern formation where both symmetries and the element of surprise coexist. This capability was observed in [27] where CA are described as "self-generating computer graphics movies". This is a new way of generating imagery which has no precedent in human culture [26]. The role of symmetry in art, architecture and its association with aesthetic preferences is a well known concept [21]. The iterative application of transition function
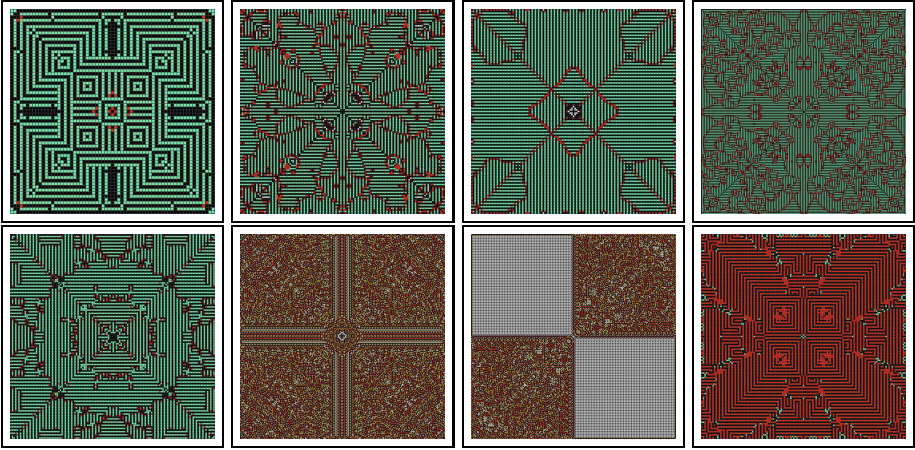
Fig. 1. Sample CA generated symmetrical patters

over initial configuration, especially in multi-state CA, can generate complex symmetrical patterns [12,22] which are extremely challenging to construct using conventional mathematical methods. Figs. 1 and 2 show experimental patterns generated by the authors to demonstrate the generative capabilities of CA in creating symmetrical patterns.

3   Symmetry and Aesthetic

Symmetry, having proportionality and balance is an important element of aesthetics. The association of aesthetics and symmetry has been investigated extensively in literature. A study to investigate the effect of symmetry on interface judgements, and relationship between a higher symmetry value and aesthetic appeal for the basic imagery, showed that subjects preferred symmetric over non-symmetric images [4]. Further studies found that if symmetry is present in the face or the body, an individual is judged as being relatively more attractive and if the body is asymmetric the face is rated unattractive, even if the person doing the rating never sees the body [25,10]. Symmetry plays a crucial role in theories of perception and is even considered a fundamental structuring principle of cognition [15]. In the Gestalt school of psychology things [objects] are affected by where they are and by what surrounds them... so that things [objects] are better described as more than the sum of their parts [5]. The Gestalt principles emphasise the holistic nature of perception where recognition is inferred, during visual perception, more by the properties of an image as a whole, rather than its individual parts [13]. Thus, during the recognition process elements in an image are grouped from parts to whole based on Gestalt principles of perception such as proximity, parallelism, closure, symmetry, and continuation [23]. In particular, symmetric objects are more readily perceived [8]. It is not surprising that we humans find sensory delight in symmetry, given the world in which
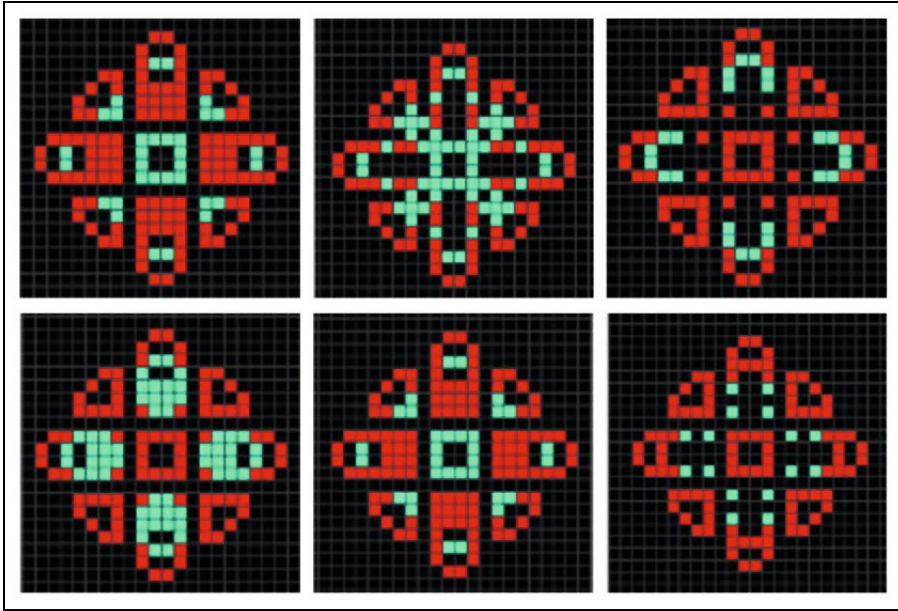
Fig. 2. Sample CA generated symmetrical patters

we evolved. In our world the animals that have interested us and our ancestors
(as prey, menace, or mate) are overwhelming symmetric along at least one axis
[24]. Evolutionary psychologists examine physical appearances like as symmetry,
and perceived level of aesthetics as an indirect measure in mate selection [21,20].
In this view symmetrical faces are examined as more attractive faces. In other
words symmetry is positively linked with both psychological and physiological
health indicators [28]. In geometry symmetrical shapes are produced by apply-
ing four operations of translations, rotations, reflections, and glide reflections.
However developing computational methods which generate symmetrical pat-
terns is still a challenge since it has to connect abstract mathematics with the
noisy, imperfect, real world; and few computational tools exist for dealing with
real-world symmetries [16]. Applying evolutionary algorithms to produce sym-
metrical forms leaves the formulation of fitness functions, which generate and
select symmetrical phenotypes, to be addressed . Lewis describes two strategies
in evolutionary algorithms approach for generating and selecting symmetrical
forms: " A common approach is to hope for properties like symmetry to grad-
ually emerge by selecting for them. Another strategy is to build in symmetry
functions which sometimes activate, appearing suddenly. However this leads to
a lack of control, as offspring resulting from slight mutations (i.e., small steps in
the solution space) bear little resemblance to their ancestors [14]".

The next section explains the swarm intelligence algorithm which will be used
in detecting symmetrical patterns.

4   Swarm Intelligence Algorithm

The swarm intelligence algorithm used in this work is Stochastic Diffusion Search (SDS) [6,1] which is a probabilistic approach for solving best-fit pattern recognition and matching problems. SDS, as a multi-agent population-based global search and optimisation algorithm, is a distributed mode of computation utilising interaction between simple agents [19]. Its computational roots stem from Geoff Hinton's interest 3D object classification and mapping. See [11,17] for Hinton's work and [6,7] for the connection between Hinton mapping and SDS. SDS algorithm has been used in various fields including optimisation and generative arts (e.g. [2,3]).

In order to introduce SDS, a social metaphor, *the Mining Game*, is introduced.

4.1   The Mining Game

The mining game provides a simple metaphor outlining the high-level behaviour of agents in SDS:

> A group of friends (miners) learn that there is gold to be found on the hills of a mountain range but have no information regarding its distribution. On their maps the mountain range is divided into a set of discrete hills and each hill contains a discrete set of seams to mine. Over time, on any day the probability of finding gold at a seam is proportional to its net wealth.
>
> To maximise their collective wealth, the miners need to identify the hill with the richest seams of gold so that the maximum number of miners can dig there (this information is not available a-priori). In order to solve this problem, the miners decide to employ a simple Stochastic Diffusion Search.
>
> – At the start of the mining process each miner is randomly allocated a hill to mine (his hill hypothesis, h).
> – Every day each miner is allocated a randomly selected seam on his hill to mine.
> – At the end of each day, the probability that a miner is happy is proportional to the amount of gold he has found.
> – At the end of the day the miners congregate and over the evening each miner who is unhappy selects another miner at random to talk to. If the chosen miner is happy, he happily tells his colleague the identity of the hill he is mining (that is, he communicates his hill hypothesis, h, which thus both now maintain). Conversely, if the chosen miner is unhappy he says nothing and the original miner is once more reduced to selecting a new hypothesis - identifying the hill he is to mine the next day - at random.

In the context of SDS, agents take the role of miners; active agents being 'happy miners', inactive agents being 'unhappy miners and the agent's hypothesis being the miner's 'hill-hypothesis'. It can be shown that this process is

isomorphic to SDS, and thus that the miners will naturally self-organise and rapidly congregate over hill(s) on the mountain range with a high concentration of gold.

4.2   SDS Architecture

The SDS algorithm commences a search or optimisation by initialising its population (e.g. miners, in the mining game metaphor). In any SDS search, each agent maintains a hypothesis, h, defining a possible problem solution. In the mining game analogy, agent hypothesis identifies a hill. After initialisation two phases are followed (for high-level SDS description see Algorithm 1):

- Test Phase (e.g. testing gold availability)
- Diffusion Phase (e.g. congregation and exchanging of information)

In the test phase, SDS checks whether the agent hypothesis is successful or not by performing a partial hypothesis evaluation and returning a domain independent boolean value. Later in the iteration, contingent on the strategy employed, successful hypotheses diffuse across the population and in this way information on potentially good solutions spreads throughout the entire population of agents.

In the Test phase, each agent performs *partial function evaluation, pFE*, which is some function of the agent's hypothesis; $pFE = f(h)$. In the mining game the partial function evaluation entails mining a random selected region on the hill, which is defined by the agent's hypothesis (instead of mining all regions on that hill).

In the Diffusion phase, each agent recruits another agent for interaction and potential communication of hypothesis. In the mining game metaphor, diffusion is performed by communicating a hill hypothesis.

Algorithm 1. SDS Algorithm

```
01: Initialising agents()
02: While (stopping condition is not met)
03:     Testing hypotheses()
04:         Determining agents' activities (active/inactive)
05:     Diffusing hypotheses()
06:         Exchanging of information
07: End While
```

The next section details how SDS is instructed to detect symmetry in CA generated patterns.

5   Experiments and Results

In this work Stochastic Diffusion Search is tasked to identify various types of symmetry. The input to the system are some sample patterns to show the functionality of the method and later some real world cellular automata generated patterns are fed in the system to evaluate the overall performance of the algorithm in detecting symmetry.
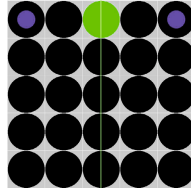
Fig. 3. Figure showing the search space (5×5); hypothesis in green; and the micro-features in blue

5.1   Applying SDS Algorithm

In order to adopt SDS to use for identifying symmetries, the following are considered:

- the search space comprises of the entire cells on the canvas (see Fig. 3 where the search space size is 5×5)
- SDS hypothesis is a cell index along one of the axes of a symmetry. See Fig. 3 where the hypothesis is highlighted in green (i.e. index $= 0$)
- the cells on either side of each axes of symmetry are considered micro-features[1] of the hypotheses (see Fig.3 where sample micro-features are highlighted in blue)

   As shown in Fig. 4, there are four axes of symmetry in four-fold symmetrical patterns. Fig. 5 shows each of these axes separately.
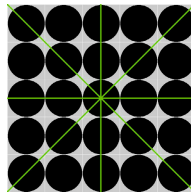

Fig. 4. Four axes of symmetry in a four-fold symmetrical pattern

   The process through which the test and diffusion phases of SDS algorithm iterates is explained below:

Initialisation Phase. During the initialisation phase one of the symmetrical axes is chosen and is set as a model (to be used for comparing the cells on either sides). Then each agent is associated to a cell index which is between 0 and the length of the side of the screen (i.e. width or height). In other words each agent's hypothesis is set to one of the cells along the axis of symmetry.

---

[1] Micro-features are used in the test phase of SDS to determine the status of the agent (i.e. active or inactive).
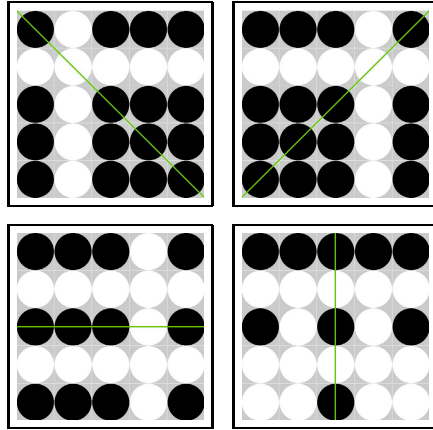
Fig. 5. Individual axes of symmetry

Test Phase. During the test phase, each agent, which is already allocated to an index on the axis of symmetry, picks a cell (micro-feature) from either side of the axis and checks if the mirror of the cell has the same value. If the difference between the two corresponding micro-features is zero, the agent becomes active, otherwise inactive.

Diffusion Phase. The process in the diffusion phase is the same as the one detailed in the algorithm description: each inactive agent picks an agent randomly from the population; if the randomly selected agent is active, the inactive agent adopts the hypothesis of the active agent (i.e. the cell index on the axis of symmetry), otherwise the inactive agent picks a random cell index between 0 and the length of the side of the canvas.

After n number of iterations all agents converge on the points of symmetry.

One of the main features of SDS is partial function evaluation which here manifests itself in: each time checking one cell on one side of the symmetrical axis to its corresponding cell on the other side. Therefore even when an agent is active, in the next iteration it picks another micro-feature and checks the point from "a different perspective" to ensure that the symmetry holds.

Fig. 6 shows few iterations in which the cells are marked in blue (the current micro-features) are checked. This feature is also useful in dynamically changing environments where cells change their characteristics over time and as such might lose their symmetrical patterns.

The process can be repeated to test other axis of symmetry. Fig. 7 shows a pattern that is symmetrical on two points on one of the symmetrical axis. Therefore, the agents converge to the 'optimal' points. Fig. 7(1) shows that all agents except the one at the bottom are active (green); the micro-features selected for the inactive agents are clearly not identical (black on the left hand side and white on the right hand side). Therefore, now that the agent is inactive,
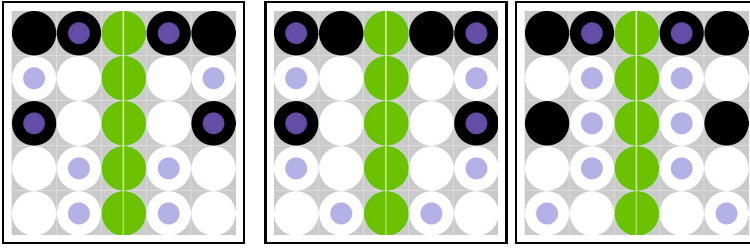
Fig. 6. Regular change in picking mico-features in the test phase

it selects another cell (during the diffusion phase) along the symmetrical line. It selects the one on top as shown on Fig. 7(2) where there are two micro-features on each side been tested by the algorithm (during the test phase). This process is repeated until all agents converge to the points where symmetry or partial symmetry is detected in the pattern (see Fig. 7(4)).
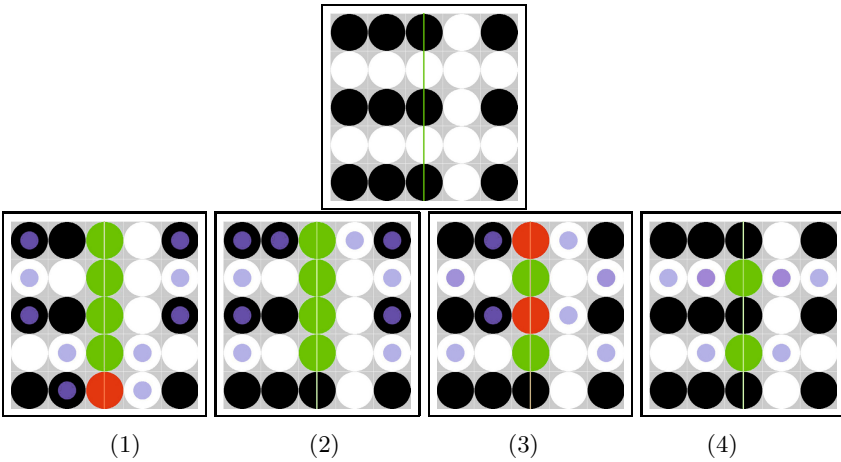

(1)   (2)   (3)   (4)

Fig. 7. Identifying partial symmetry

Using this approach, the algorithm allocate its resources "wisely" and repeatedly tests the already maintained points of interest against any asymmetrical discovery.

Fig. 8 shows a larger pattern which is fed into the system and the algorithm confirms its symmetrical nature against all four axes of symmetry.
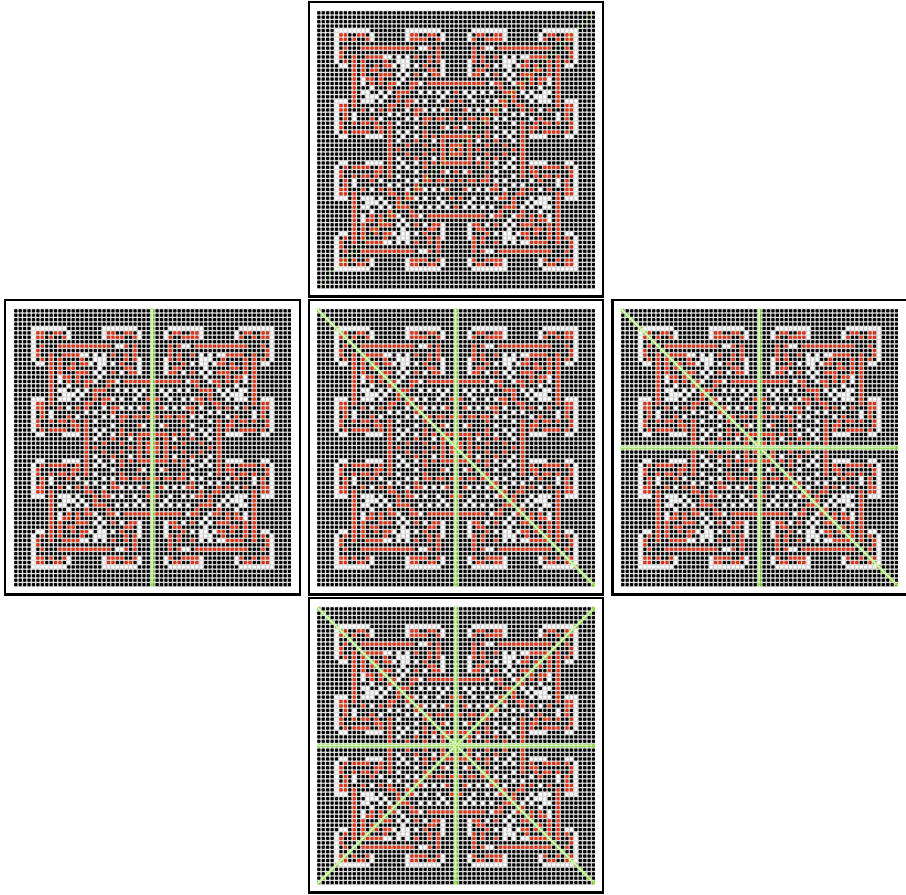
Fig. 8. Detecting four-fold symmetry in a cellular automata generated pattern

6   Conclusion

CA provide perspective and powerful tools in generating computer graphics. The multi-state CA rule space is a vast set of possible rules which can generate interesting patterns with high aesthetic qualities. The interaction of CA rules at local level generates emergent global behaviour, that can sometimes demonstrate attractive complexity. Some characteristics of CA, such as the regularity and complexity of the rules that are employed locally, suggest that they could be well suited to generating computer graphics.

This paper demonstrates the capability of a swarm intelligence algorithm – Stochastic Diffusion Search – in detecting symmetrical patterns along various axes of symmetry. Evaluating the symmetry of cellular automata generated patterns is often a difficult task partly due the the large size of the search space or canvas, and partly due to the constantly changing, dynamic environment in

which the cellular automata patterns are generated. These factors contribute to making the detection of symmetrical patterns computationally expensive. One of the main features of Stochastic Diffusion Search is partial function evaluation which is particularly useful when dealing with large problems with high dimensions. The performance of this algorithm is explained in the paper and the results are accordingly reported. In addition to identifying symmetry along one or all axes of symmetry (four-fold symmetry), the algorithm demonstrates its ability in identifying partial symmetry.

Following the introduction of this novel technique, among the future research topics are: conducting a comparison with other evolutionary and non-evolutionary techniques, computing the correlation between the size of search space and the computational complexity of the process, and applying this method on dynamically evolving cellular automata generated patterns.

References

1. al-Rifaie, M.M., Bishop, M.: Stochastic diffusion search review. Paladyn, Journal of Behavioral Robotics 4(3), 155–173 (2013)
2. al-Rifaie, M.M., Bishop, M., Blackwell, T.: Information sharing impact of stochastic diffusion search on differential evolution algorithm. J. Memetic Computing 4(4), 327–338 (2012)
3. al-Rifaie, M.M., Bishop, M., Caines, S.: Creativity and autonomy in swarm intelligence systems. J. Cognitive Computation 4(3), 320–331 (2012)
4. Bauerly, M., Liu, Y.: Computational modeling and experimental investigation of effects of compositional elements on interface and design aesthetics. International Journal of Man-Machine Studies 64(8), 670–682 (2006)
5. Behrens, R.R.: Design in the visual arts. Prentice-Hall (1984)
6. Bishop, J.: Stochastic searching networks. In: Proc. 1st IEE Conf. on Artificial Neural Networks, pp. 329–331. IET, London (1989)
7. Bishop, J., Torr, P.: The stochastic search network. In: Neural Networks for Images, Speech and Natural Language, pp. 370–387. Chapman & Hall, New York (1992)
8. Carroll, M.J. (ed.): HCI Models, Theories, and Frameworks: Toward a multidisciplinary science. Morgan Kaufmann Publishers, San Francisco (2003)
9. Dawkins, R.: The blind watchmaker. New York: Norton & Company, Inc. (1986)
10. Gangestad, S.W., Thornhill, R., Yeo, R.A.: Facial attractiveness, developmental stability, and fluctuating asymmetry. Ethology and Sociobiology 15(2), 73–85 (1994)
11. Hinton, G.F.: A parallel computation that assigns canonical object-based frames of reference. In: Proceedings of the 7th International Joint Conference on Artificial Intelligence, vol. 2, pp. 683–685. Morgan Kaufmann Publishers Inc. (1981)
12. Javid, M.A.J., te Boekhorst, R.: Cell Dormancy in Cellular Automata. In: Alexandrov, V.N., van Albada, G.D., Sloot, P.M.A., Dongarra, J. (eds.) ICCS 2006. LNCS, vol. 3993, pp. 367–374. Springer, Heidelberg (2006)
13. Jiang, H., Ngo, C.W., Tan, H.K.: Gestalt-based feature similarity measure in trademark database. Pattern Recognition 39(5), 988–1001 (2006)
14. Lewis, M.: Evolutionary visual art and design. In: Romero, J., Machado, P. (eds.) The Art of Artificial Evolution. Natural Computing Series, pp. 3–37. Springer (2008)

15. Leyton, M.: Symmetry, causality, mind. Bradford Books, MIT Press (1992)
16. Liu, Y.: Computational symmetry. In: CMU Robotics Institute (2000)
17. McClelland, J.L., Rumelhart, D.E., Group, P.R., et al.: Parallel distributed processing. Explorations in the Microstructure of Cognition 2 (1986)
18. McCormack, J.: Interactive evolution of l-system grammars for computer graphics modelling. Complex Systems: from biology to computation, 118–130 (1993)
19. de-Meyer, K., Bishop, J.M., Nasuto, S.J.: Stochastic diffusion: Using recruitment for search. In: McOwan, P., Dautenhahn, K., Nehaniv, C.L. (eds.) Evolvability and interaction: evolutionary substrates of communication, signalling, and perception in the dynamics of social complexity, Technical Report 393, vol. 393, pp. 60–65 (2003)
20. Møller, A.P., Cuervo, J.J.: Asymmetry, size and sexual selection: meta-analysis, publication bias and factors affecting variation in relationships, p. 1. Oxford University Press (1999)
21. Møller, A.P., Thornhill, R.: Bilateral symmetry and sexual selection: a meta-analysis. Am. Nat. 151(2), 174–192 (1998)
22. Nowak, M.A.: Evolutionary dynamics: exploring the equations of life. Harvard University Press (2006)
23. Park, I.K., Lee, K.M., Lee, S.U.: Perceptual grouping of line features in 3-D space: A model-based framework. Pattern Recognition 37(1), 145–159 (2004)
24. Railton, P.: Aesthetic Value, Moral Value and the Ambitions of Naturalism. In: Aesthetics and Ethics, vol. 3, University of Maryland (2001)
25. Randy, T., Steven, G.: Human facial beauty. Human Nature 4, 237–269 (1993)
26. Roth, T.O., Deutsch, A.: Universal synthesizer and window: Cellular automata as a new kind of cybernetic image. In: Imagery in the 21st Century, pp. 269–288. The MIT Press (2011)
27. Rucker, R.: Seek!: Selected Nonfiction. Running Press Book Publishers (1999)
28. Shackelford, T.K., Larsen, R.J.: Facial symmetry as an indicator of psychological emotional and physiological distress. Journal of Personality and Social Psychology 72 (1997)
29. Sims, K.: Evolving virtual creatures. In: Proceedings of the 21st Annual Conference on Computer Graphics and Interactive Techniques, pp. 15–22. ACM (1994)
30. Todd, S., Latham, W., Hughes, P.: Computer sculpture design and animation. The Journal of Visualization and Computer Animation 2(3), 98–105 (1991)