Genet Program Evolvable Mach (2013) 14:339–367
DOI 10.1007/s10710-013-9187-8

Aesthetic 3D model evolution

Steve Bergen · Brian J. Ross

Received: 20 July 2012 / Revised: 8 March 2013 / Published online: 18 April 2013

Abstract A new research frontier for evolutionary 2D image generation is the use of mathematical models of aesthetics, with the goal of automatically evolving aesthetically pleasing images. This paper investigates the application of similar models of aesthetics towards the evolution of 3-dimensional structures. We extend existing models of aesthetics used for image evaluation to the 3D realm, by considering quantifiable properties of surface geometry. Analyses used include entropy, complexity, deviation from normality, 1/f noise, and symmetry. A new 3D L-system implementation promotes accurate analyses of surface features, as well as productive rule sets when used with genetic programming. Multi-objective evaluation reconciles multiple aesthetic criteria. Experiments resulted in the generation of many models that satisfied multiple criteria. A human survey was conducted, and survey takers showed a statistically significant preference for high-fitness highly-evolved models over low-fitness unevolved ones. This research shows that aesthetic evolution of 3D structures is a promising new research area for evolutionary design.

Keywords Aesthetics · L-systems · 3D models · Genetic programming · Multi-objective evaluation

1 Introduction

Recent research in evolutionary 2D image synthesis (evo-art) is exploring the use of mathematical models of aesthetics [2, 6, 16, 34, 35]. The intention is to give the fitness evaluator a means for evaluating aspects of the aesthetic quality of generated images. By doing so, one goal is to remove the human as a mandatory participant in

S. Bergen · B. J. Ross (✉)
Department of Computer Science, Brock University, 500 Glenridge Ave., St. Catharines, ON L2S 3A1, Canada
e-mail: bross@brocku.ca

the evolutionary process. This has been the case for most evo-art systems in the past, which require interactive fitness evaluation by a human user. Although the mathematical modelling of aesthetics is not new, its use in artificial intelligence topics such as evolutionary design is a recent development. The aesthetic models used in evolutionary 2D bitmap generation are necessarily simple and efficient to compute, as evolutionary algorithms are by nature computationally expensive. Furthermore, these models are also experimental and somewhat controversial. They should be treated as heuristics for measuring specific technical aspects of images, and should not be considered to be comprehensive absolute measurements of the aesthetic quality of fine art. These theories do not account for all the profound complexities and subtleties involved when considering aesthetics. It is still an open question regarding the extent to which deeper aspects of the aesthetics of images *can* be captured by formal mathematical models. Deeper insight into this topic will undoubtedly be multidisciplinary, with contributions from psychology [55], neuroscience [23], and other branches of biology and cognitive science [28]. Further research is necessary, especially if advanced applications of artificial intelligence in arts and design are to be fully realized in the future.

The mathematical modeling of aesthetics is not new. For over 2 millenia, the Golden ratio has been posited to be a measure of proportion having inherent pleasing qualities [54]. The Rule of Thirds is a well-known rule, used by artists and photographers for positioning the horizon and objects of interest in a scene [56]. An attempt at a general mathematical theory of aesthetics is that of Birkhoff [7]. He proposed that beauty can be characterized by having a high aesthetic order (or simplicity), and low complexity. This idea has inspired others in evo-art [31] (discussed below).

Recently, research in mathematical modeling of aesthetics has been gaining more attention. A review in [15] surveys a number of mathematical models of art. Factors such as amplitude spectra, sparseness, luminance, and color can be modeled by statistics and 1/f distributions. Of importance here is that researchers are finding connections between the mathematical models and human biology. Hence the mathematical models may be justified as formal models of the human visual-neural sensory responses to art.

There are many examples of using aesthetic models for analyzing and generating images with evolutionary algorithms. Svangard and Nordin propose an approach for measuring aesthetic quality in evo-art applications, based on a similarity metric [46]. Rigau et al. [40] perform a thorough mathematical study of artworks, using Shannon entropy, Kolgomorov complexity, and others. Greenfield [16] evolves images in a multi-objective genetic programming (GP) environment. His aesthetic model controls different aspects of colour segmentation throughout the image. Using ideas similar to Birkhoff's aesthetic theory [7], Machado and Cardoso [31] posit that an image's aesthetic appeal is related to a relationship between the complexity of an image, and the ease in processing it. Also, based on Spehar et al.'s [44] study of user preferences of fractal dimensions, Machado found that a fractal dimension of 1.35 was suitable as an "ease of processing" target. Ralph proposed that many fine art works exhibit a normal distribution of colour gradient [39]. This was used as a fitness measure to evolve images [42] and filters [34]. In [18], den

Heijer and Eiben compare the aesthetic models in [31, 42, 44], to see how well aesthetic theories "appreciate" each other's results. They discovered some correlation in scores between the aesthetic models studied.

Motivated by developments in research in aesthetic image evolution listed above, this paper investigates the automatic synthesis of 3D models via evolutionary computation. We use aesthetic fitness evaluation with GP, in order to evolve aesthetically-pleasing 3D models. A goal of this research is to investigate whether models of aesthetics used to evaluate 2D images might have analogues with respect to 3-dimensional structures. For example, image aesthetics have considered colour distributions, gradients, luminosity, simplicity, complexity, and other factors. Similarly, there are many ways in which a 3D model might be analyzed with respect to volume, surface area, surface changes, and other geometric and topological properties. If parallels between 2D and 3D aesthetic models can be derived, an obvious goal is to see whether they are effective as fitness functions during evolution. That being said, it is outside of the scope of this research to perform a comparative evaluation of these aesthetic theories in 3D model evolution (such as is done in [18] for aesthetic models for 2D image evolution). Therefore, we will not rank or compare these aesthetic models with each other in this paper.

Another goal is to determine whether evolved models that satisfy the mathematical aesthetic criteria have any aesthetic merit whatsoever to humans. The results from analogous 2D evo-art research suggest that results may be too subjective to expect rigidly strong correlations between aesthetic models and human preferences. Evolved 3D models are often very abstract, and it could be the case that aesthetic modelling could be too tenuous and subjective to be effective in this domain. As mentioned above, it is outside the scope of this paper to do a comprehensive empirical investigation of aesthetic theories. Such a study would need to perform a comparative study between the aesthetic models, as well as a comparison to human-created images and models. Our goal in this paper is much more modest: we wish to see whether evolved models, with more optimized aesthetic scores, are preferable to random, unevolved (non-trivial) models that score poorly. An affirmative answer—that human viewers prefer the evolved, high-scoring models—would give evidence that the aesthetic theories do indeed have merit, and thus 3D models can be automatically evolved to have moderate attractiveness to the human eye. We are not implying, however, that these models would necessarily be human-competitive, compared to sculptures and models created by artists.

There are many applications for the automatic synthesis of aesthetic-motivated 3D structures. Manual 3D modelling is a difficult and time-consuming task, which requires considerable technical and artistic skill. Automatic and semi-automatic model synthesis can be used by high-level modelling tools. Such tools can be used as "exploratory aids", to generate forms that can be inspirational to an architect, artist, or designer, and can be used for further alteration and refinement as desired. Forms can also be used directly in computer games and computer animations, possibly generated on the fly as needed. 3D printing technology is also becoming commonplace, which makes it feasible to cheaply render models as physical objects.

We see the mathematical aesthetics as a contribution towards the evolution of more human-meaningful outcomes in evolutionary art and design. The use of mathematical models of aesthetics is still nascent in the field of evolutionary design, and until now has been studied exclusively in 2D image generation. We do not believe that models as overviewed in [18] for 2D images, and those that used in this paper for 3D model generation, are in any way complete, comprehensive formalisms of aesthetics capable of generating human-competitive fine art. On the contrary, they are fairly simple measurements that are nevertheless useful for guiding evolution towards broad, stylistic areas of the search space. When successful, the results obtained have general stylistic characteristics. Experience with 2D image evolution also suggests that these aesthetic heuristics certainly help automate the evolutionary process—which until recently has always required a human for interactive evaluation. However, final human appreciation and judgement of results is still a requirement, since images with high fitness can still be unpleasing and unremarkable, depending on the eye of the beholder. The nature of the final factors that come into play in such situations is a fascinating question. Research in formal modeling of aesthetics has much to discover.

The use of evolution for 3D model design is well established [2]. A selection of examples include its use for curvi-linear surfaces [20], trusses [50], plants [21], buildings [10, 36], and artificial life [43, 47]. Many such applications rely on interactive fitness evaluation by the user. McCormack was one of the first to use mutation-guided evolution to evolve an L-system grammar which produced 3D models, through the use of user-based fitness evaluation [32]. As far as we know, the explicit use of aesthetic evaluation functions has not yet been used in 3D model evolution.

The aesthetic models and other structural analyses used in this research are presented in Sect. 2. Section 2.1 introduces the feature tests used for fitness evaluation, and presents a study that examines effect of the feature tests on a variety of pre-made 3D models. A new L-system implementation is discussed in Sect. 3. Multi-objective evaluation and other GP system details are reviewed in Sect. 4. Some example experimental results using multi-objective GP are presented in Sect. 5. Results from a human survey of evolved results are presented in Sect. 6. Some comparisons to related work are made in Sect. 7. Section 8 concludes the paper with a summary discussion, and directions for future research. See [4, 5] for further details of this research.

2 Aesthetic and structural evaluation of 3D models

2.1 Mathematical modelling

A library of 11 evaluation functions are defined as possible fitness criteria. They fall into two categories: model constraint functions and aesthetic (distribution-based) functions. Subsets of these tests are normally selected for simultaneous use in runs.

Model constraint functions measure basic geometric properties of a model. They are useful for establishing practical constraints for the size of generated models. In

many cases, these functions are not considered as important as aesthetic functions, and so they are typically weighted lower during fitness evaluation. The model constraint functions available are:

1. *Volume* A voxel defines a unit volume. The volume is calculated as the sum of all active voxels. Since the marching cubes algorithm gives an approximation of a surface over a voxel surface, the volume fitness function is an approximation of the actual volume. Gravity wells (see Sect. 3) further reduce the accuracy of this measurement, as they apply transformations to voxel geometries after the marching cube algorithm.
2. *Dimension* This function measures the difference in the width, height and depth of a model's bounding box from a bounding box specified by the user. This is used to encourage models to occupy a desired volume and is therefore primarily used to constrain the size of the model produced, though it is meant to be weighted much lower than the other functions.
3. *Surface area* This is measured as the sum of all face areas on the model, and is useful for constraining a model's surface complexity. The employ of a target surface area in a run introduces a personal aesthetic preference with respect to surface complexity. When paired with a target volume or dimension, a large surface area will often lead the evolutionary process in producing very complex forms, as the algorithm attempts to fill the available target space with the model in order to satisfy all fitness targets simultaneously. The marching cubes algorithm (see Sect. 3) guarantees that there are no surface polygons embedded within a model, as the surface generated from the algorithm is simply an approximation to a voxel volume, and hence the area computation is fairly accurate [29]. Some minor inaccuracies can arise should gravity wells "meld" faces together through vertex modification, however.
4. *Unique surface normals* Every face on a polygon's surface defines a normal representing its forward direction perpendicular to the surface. This calculation tallies the number of unique surface normals in the model, where a large number of unique normals typically indicates a smoother, more rounded model. Fluid, organic shapes typically have a very high number of unique normals, and can be directly inspirational when choosing an appropriate target for this function.

The aesthetic or distribution-based functions measure various distributions of a specified feature found over an entire model. There are many features of a 3D model that can be potentially analyzed. We choose between two particular features, due to their similarities with pixel intensities as used in aesthetic models of images [18, 31, 34, 52]: (a) A measure of the signed difference between adjacent face normals, ranging from 0° (no change) to ±180°. (b) A measure of the signed difference between adjacent face areas. These measurements are computed for the entire surface of a model. The resulting frequency distribution histogram is then calculated, and the appropriate calculation is applied to the histogram according to the distribution test being used. The last two functions—symmetry and complexity—are an exception, as they do not directly use the data in an ordered manner. Instead of a histogram, these two functions focus more on the distribution

of the model's physical properties, such as its distribution of visible segments across a production string—as seen in the complexity measure—and cumulative distribution of vertices across a chosen axis—as seen with symmetry. The distribution functions used are:

1. *Mean* (μ) *and standard deviation* (σ) These statistics are computed for the frequency histogram. If they are used as fitness criteria, then the user supplies desired target values. The error between the computed and target is used as a fitness objective.
2. *Deviation from normality (DFN)* The DFN model is a heuristic that favours a balance of uniformity and complexity to measured entities, be they 2D images or 3D models. Originally proposed in [39], Ralph found that many examples of fine art paintings exhibited a normal distribution of such gradient frequencies. The DFN model measures the conformity between the gradient frequency histogram and an estimated normal distribution histogram (Bell curve distribution). It has since been used in evolutionary image generation [42] and image filtering [34].
   To compute the DFN, the absolute difference between these histograms is first calculated. The normal distribution is estimated by first calculating μ and σ of the frequency histogram, and then calculating the expected probabilities for each histogram bin value x. The probabilities are calculated using:

$$pDensity(x) = \frac{1}{\sqrt{2\pi}} * e^{\frac{x^2}{-2}} \quad \text{and} \quad curve(x) = \frac{1}{\sigma * pDensity((x-\mu)/\sigma)}$$

With these probabilities, a second histogram is created from the expected frequency of each bin value:

$$expected(x) = curve(x) * inc * total$$

where *inc* is the increment between bin values, and *total* is the size of the input data set. The absolute difference between bin values of this histogram and the original frequency histogram are then computed at each bin x, summed, and scaled to the range between 0 and 10. A DFN of 0 is a perfect match to the normal curve. Figure 1 shows two example fits using the DFN calculation.

3. *1/f noise* Also known as *pink noise*, this is a well-studied distribution found in many diverse areas of nature and human endeavour, for example, physics, electronics, and music [33]. It was famously shown by Voss and Clarke [49] that music exhibits a 1/f distribution of frequency changes, or intervals between successive notes. Random music generated with a 1/f noise generator sounds more musically structured to the human ear, than that generated by white (uniform) noise generator.
   To compute the conformity to a 1/f distribution, a histogram is computed as described above for the DFN calculation. A line of best fit is constructed from the data, which was plotted using the logarithm of each bin in the histogram as the y-values. The absolute difference between the slope of this line and the generated 1/f curve—which is constructed using the data supplied for this fitness target—is then calculated. Note that the vertical displacement is not

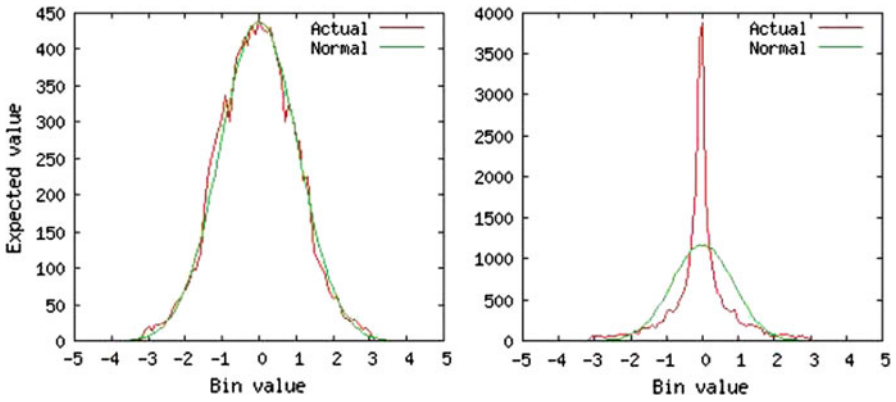

Fig. 1 Examples of normal curves and respective actual curves. The actual curves (*red*) are the curves plotted using the real model data for a particular measurement, such as difference between adjacent surface normals. The normal curves (*green*), are the expected normal curves generated from the real model data of the same model. The calculated difference between the two curves is the DFN, used in model fitness evaluation (Color figure online)

considered when comparing the two lines. The rationale for this calculation is that if the curve plotted by the model data exhibits similarities to the 1/f curve—as was seen in many musical examples—the model may potentially exhibit features associated with aesthetic forms. Figure 2 illustrates this measurement.

4. *Entropy* Shannon's entropy measures the level of uncertainty associated with the data distribution [40]. At one extreme, a non-changing measurement has a low entropy; at the other extreme, a purely random measurement has a high entropy.
   As with DFN and 1/f, the data is sorted into histogram bins, and the bin values are used to compute probabilities $p(x)$ for each x. Then,

$$Entropy(X) = -\sum_{x \in X} p(x) log\, p(x)$$

A lower entropy indicates a low level of uncertainty, which is typical of simpler models. A higher entropy is more common in large, complex models.

5. *L-system complexity* This measurement is related in varying degrees to Birkhoff's aesthetic measure [7], Komolgorov complexity [26], the Box-counting method [38], and Machado's aesthetic model [31]. Although separate concepts, they all incorporate ideas of the interrelationship between order and complexity.
   We consider L-system complexity to be a measure of the capacity for growth of an L-system's derivation string over many iterations. Because large strings may not necessarily generate large models, the size of the string is factored with the surface area of the resulting model—in that a large string that generates a small model is not considered inherently complex. This phenomena occurs most prominently when a production string generates a loop in upon itself. String growth is measured as:

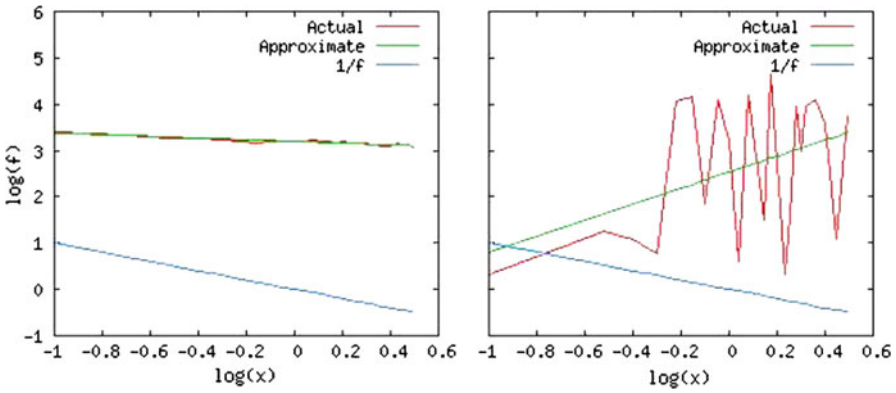

Fig. 2 Examples of 1/f curves and respective actual curves. The *red* curves indicate the actual plotted real model data, while the *green* curves are the line-of-best-fit for the data. The calculated difference between this curve and the generated 1/f curve (*blue*) for the raw model data gives our 1/f value used in model fitness evaluation (Color figure online)

$$growth = \frac{\sum_{i=0}^{n-1} v(\omega^{i+1}) - v(\omega^{i})}{n}$$

where ω^i is the string generated at iteration i, $v(\omega^i)$ is the number of drawing variables present in ω^i, and n is the maximum iteration. Complexity is then:

$$complexity = log(growth) + log(surface\ area)$$

Therefore, the complexity of a model is greatest when both its surface area and L-system string growth are large. This will discourage degenerate cases that do not measure L-system complexity on their own: (a) a large derivation string by itself (which can still result in a simple model); (b) a large surface area by itself (perhaps a large cube, as generated by a simple derivation string).

6. *Symmetry* Symmetry is a much-studied fundamental concept in science and mathematics [11, 13, 41]. The human eye is naturally attracted to symmetry in art and design. There are many ways of defining and measuring symmetry. Some are also more complex to measure than others.

   Our symmetry measurement is chosen for its simplicity, and is inspired by similar ones for 2D and 3D model analysis [17, 24, 27]. Our approach is shown in Fig. 3, where symmetry is measured for a 2D image (far left). When division boundaries are introduced, the presence of colour is determined on each mirrored side of a boundary (axis of symmetry). The middle figure shows the first division of the image with X and Y axis. This yields a small symmetry score, since only the green square shows partial symmetry on its dividing line. A further division (far right) results in a high symmetry score, since all three shapes produce symmetry on division boundaries. 3D symmetry is computed in a similar way, in which an estimate of the similarity in number of vertices found on mirrored subdivison volumes in 3-space are determined. The space is

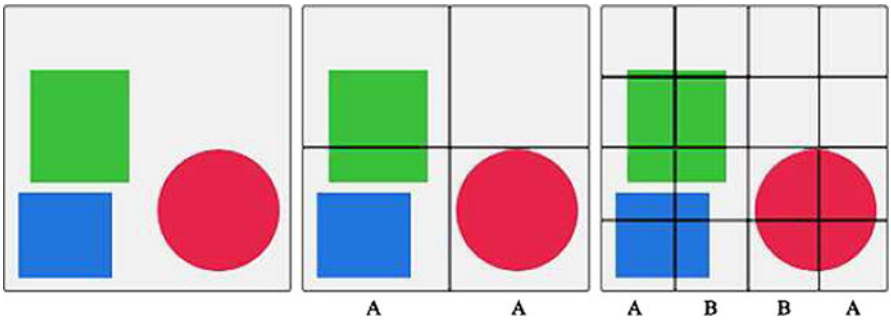

Fig. 3 Examples of symmetry testing on a 2D image

iteratively subdivided into cubic volumes, and the corresponding distribution of vertices of the model along mirrored cubic volumes is calculated. The separate symmetries of division are recorded during iterative subdivision, and the highest (most symmetric) value is returned as the result. Note that this is necessarily a heuristic approximation of 3D symmetry. 3D models can be arbitrarily orientated in space, and determining the actual boundaries of symmetry (if they exist at all) is difficult to determine. Hence this symmetry measurement, although necessarily inaccurate, is very fast to calculate.

2.2 Preliminary evaluation of aesthetic functions on 3D models

Before proceeding with GP experiments, we wished to gain some insight into the aesthetic distribution functions. In particular, we were curious to see the nature of aesthetic scores on various 3D models. Are there clear correlations between aesthetic feature scores and model shapes and forms? Are there correlations between the aesthetic models themselves, for example, does a low DFN means a high entropy? Another goal is to locate potential *sweet spots* in each aesthetic distribution—fitness target areas which many of the models had in common. Answers to any of these issues might provide insight into potential fitness targets for GP experiments.

Approximately 200 pre-made 3D models were selected from Archive 3D [1], a repository of public domain models. An attempt was made to select a set of models with a variety of geometric shapes and complexities. Each model was assigned to one of four categories—object, human, plant and polygon. The aesthetic fitness functions used were DFN, entropy, symmetry and 1/f noise. Complexity was omitted, because it requires models to have corresponding L-systems, which were not available for the downloaded models. The geometric feature measured by the aesthetic tests is the difference of adjacent surface normals.

The distribution of the results can be seen in Figs. 4 and 5. From these histograms, the following score ranges are common for the models analysed: mean (0.0–0.2), standard deviation (0.2–0.8). DFN (2.0–4.0), 1/f (2.0–3.0), entropy

(2.0–3.0), and symmetry (0.5–1.0). These are suggestive evidence of "typical" score ranges for models—at least for the selection of models that we examined. Some examples of models and their scores are shown in Fig. 6.

Next, the Pearson correlation between the fitnesses of all models was calculated for the chosen fitness functions [48]. The results are in Table 1. Correlation coefficients closer to 1.0 or −1.0 indicate a high correlation between two samples. From this table, three notable correlations were found: standard deviation and DFN ($r = 0.478$), standard deviation and entropy ($r = 0.424$), and entropy and DFN ($r = -0.380$). The first two imply that lower standard deviation is required for low DFN and entropy. The entropy-DFN relationship initially surprised us. We later

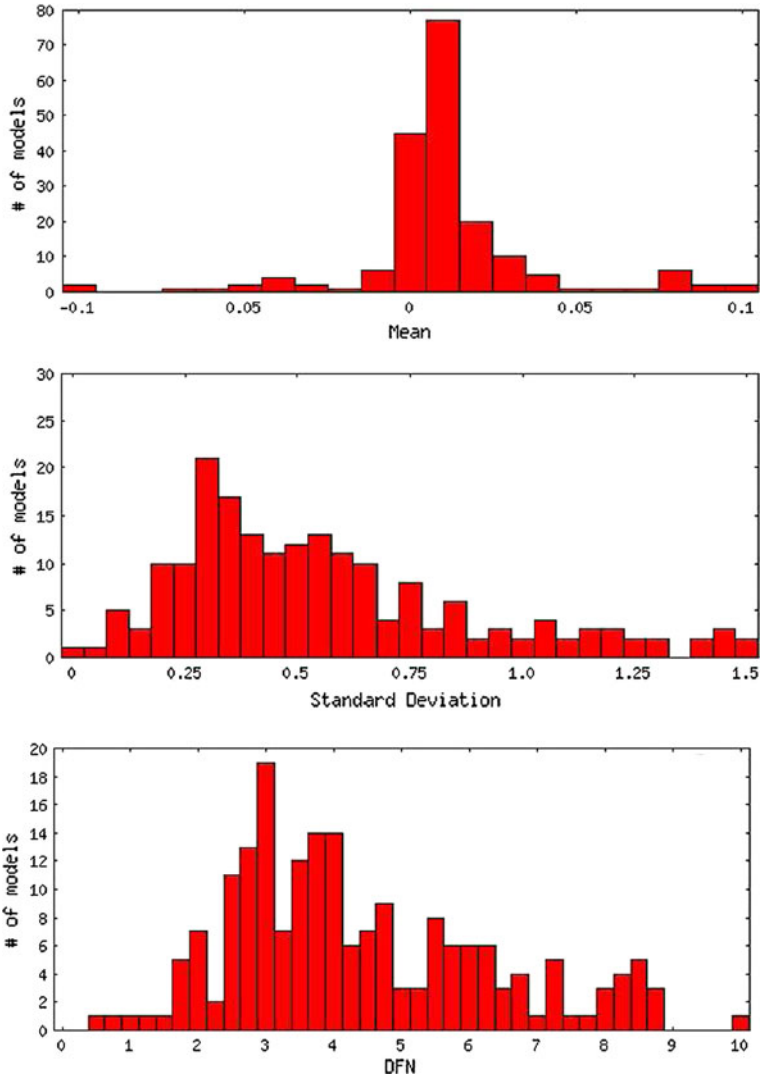

Fig. 4 Distributions of mean, standard deviation, DFN scores

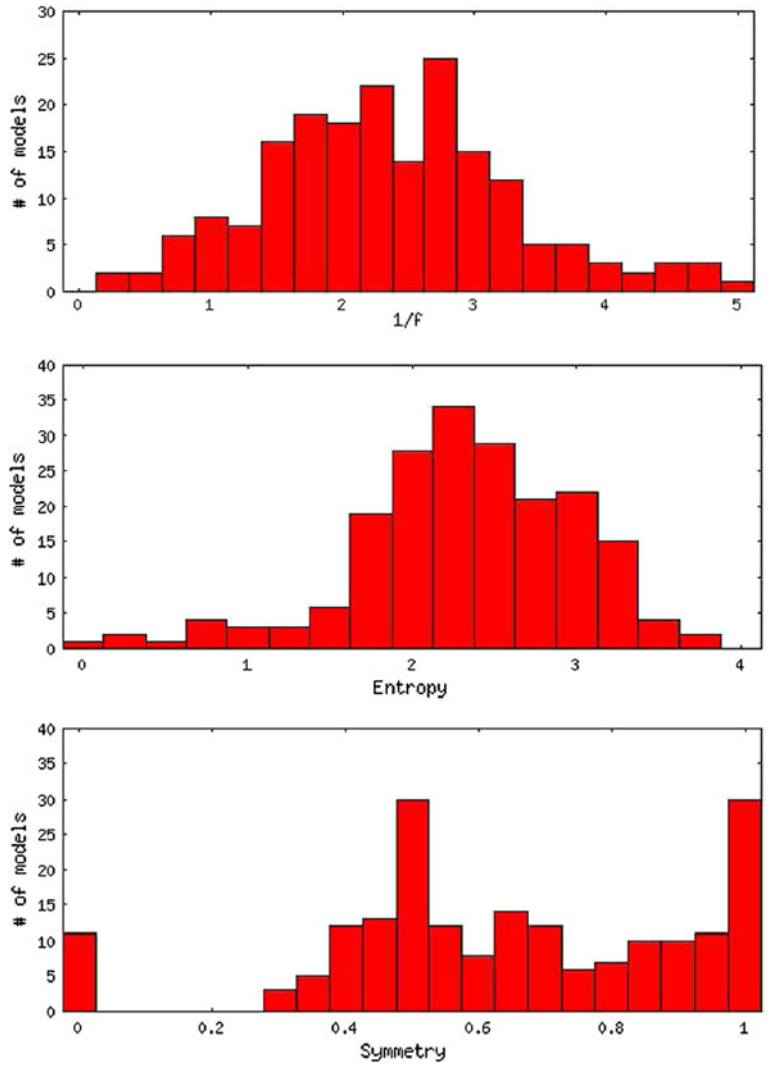

Fig. 5 Distributions of 1/f, entropy, symmetry scores

discovered that a common formula for normality (not used by us here) uses entropy as a sub-calculation. Hence they are naturally related.

The above analysis of pre-made models produced mixed results. One positive result is that we gained some insight into typical score ranges for the various aesthetic measurements. We also learned that general predictions of some score values are possible for 3D models. This was useful for setting reasonable target values for future GP experiments. Some measurements (entropy, symmetry) are clearer to understand in their effects, while others (DFN, 1/f) are more difficult, though not impossible, to intuit. On the other hand, we found that we could not

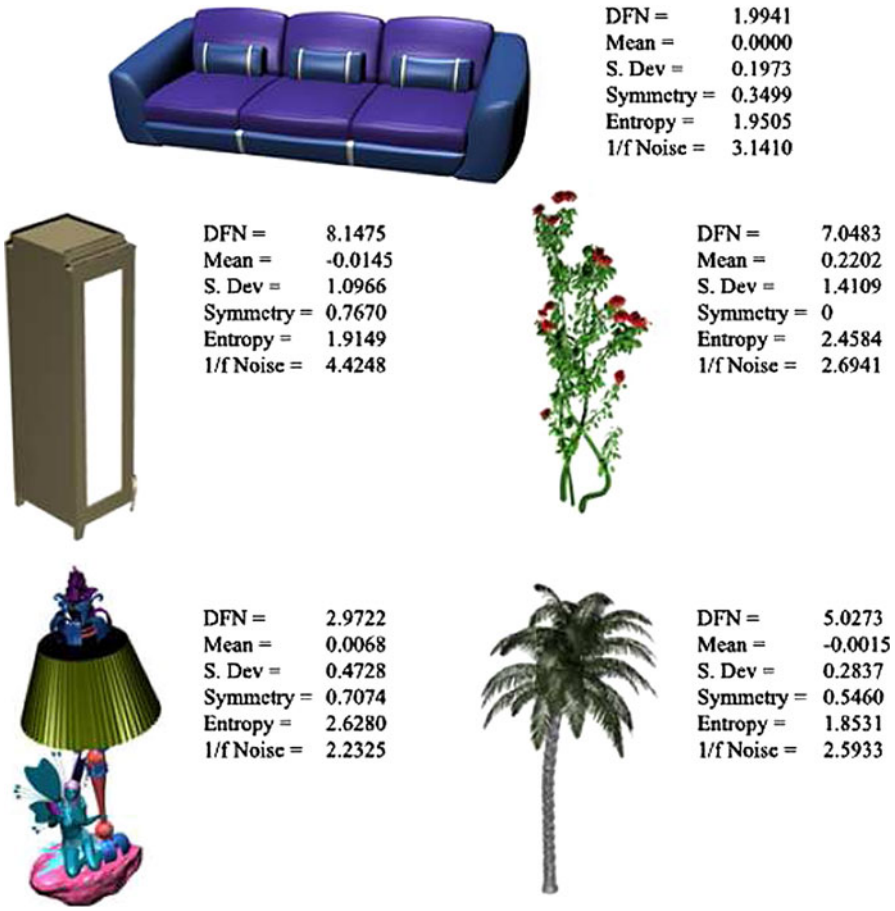

Fig. 6 Examples of models and their scores

Table 1 Pearson correlation scores between aesthetic functions

	DFN	Mean	Std. Dev.	Symmetry	Entropy
Mean	0.081644	1			
Std. Dev.	0.478077	0.249283	1		
Symmetry	−0.04697	−0.03974	−0.16465	1	
Entropy	−0.38027	−0.00526	0.424442	−0.10574	1
1/f	−0.08336	−0.07625	−0.12265	0.076309	−0.18523

make precise predictions of what numerical scores a tested model would generate. One reason for this is that scores are highly sensitive to the low-level structure and complexity of mesh geometries. For example, a sphere model generated a DFN of 8.1 for a low polygon model, to 2.9 for a higher polygon one. The nature of meshes is determined by many factors, such as the size and complexity of model, the means

by which it was created (manual 3D modelling program, L-System, 3D scanner, ...), and the characteristics of substructures composing the model. Since the majority of the models contained hundreds of thousands of vertices comprising many heterogeneous substructures, and which were created manually with 3D modelling programs, the resulting distributions are unlikely to closely match any tested distribution. Finding a pre-made model of a desklamp will unlikely show an exceptionally low DFN or very close match to a 1/f distribution. Similar effects were observed for the symmetry calculations. Since our simple symmetry measurement depends on the positional placement of vertices, many models that clearly exhibit symmetric properties had low symmetry scores, because their vertex symmetries did not conform to the partitioning done by our test. This erroneous calculation is due to many factors, including an imbalance of vertices across all three major axis, irregular polygon face sizes, and model rotation. For example, a cube, when rotated a few degrees on any axis, is less symmetrical according to our measure than if it had been oriented parallel to any one axis. More sophisticated symmetry analyses would indeed correct this.

The choice to use this specific set of measures for this experiment was due, in part, to the variety of visual properties required to produce the ranges for each measure. These measures each use different means of utilizing a model's descriptive data in assessing its aesthetic value. Although they may not be any more relevant than other methods—such as those used in the past—a model's given value can be predicted for many of these measures within a reasonable error threshold. A low DFN, for example, is easily recognizable in many fluid and organic shapes, as well as a large number of unique normals in these same shapes. It is the fine balance between the predictability and unpredictability of the outputs of these measures that make them of interest and useful as fitness functions for our EC trials.

3 A voxel-based L-system language

A new L-system language was implemented for this research. The L-system is inspired by the D0L-system of Jacob et al. [22], and shares many of their language's primitives. A D0L-system is the simplest form of an L-system, and is characterized as a context-free language having exactly one production rule per symbol (as opposed to a non-deterministic or probabilistic language with multiple productions per symbol). Like us, Jacob et al.'s language was used with GP to evolve a variety of structures. Our motivation was to implement an L-system to help promote more effective 3D structure generation in a GP environment. Space limitations prevent an adequate description of this language here, and the reader is referred to [4] for a complete description.

The approach taken by most L-systems for 3D model generation is the direct production and manipulation of polygonal models in 3-space. The resulting model geometries, however, can often be difficult to analyze accurately. For example, an L-system might embed a complex tree within a large cube, which is effectively hidden from the viewer. Analyzing the surface features of such models is challenging, because it is not obvious which portions of the model are visible or

hidden. The determination of visible surfaces can be computationally involved, and should be avoided if possible to reduce processing time.

To promote the accuracy of model evaluation, our L-system generates models in voxel space [53]. Voxels are the 3D equivalent of pixels. We define a discrete 128 by 128 by 128 resolution voxel space. The rendering of an L-system string—which has been produced from the GP tree encoding—involves turning on voxels within this discrete world. After a model has been rendered, the *marching cubes algorithm* is applied to extract an approximation of a continuous iso-surface from the voxel space [29]. This replaces the coarse, bumpy voxel surfaces with a smoother polygonal "wrapping" over the model. The advantage of a voxelized L-system model is that it is more accurately analyzed, because visible surfaces are exactly identifiable, and embedded surfaces are removed. A disadvantage, however, is that rendered models are of a lower resolution than non-voxel L-system models. This is an acceptable trade-off for this research, given the importance we place on model analysis. Modifications to the marching cubes algorithms allow a minor increase in the resolution, however, and many of the L-system primitives further increase the visual complexity of generated surfaces.

Features of our L-system implementation and GP tree encoding include:

1. *Drawing variables* A differentiation is made between L-system primitives—variables and non-variables. Variables are primitives which result in the direct production of voxels in 3D space. They can also be non-drawing variables which act purely as replacement elements within the L-system. For example, *C* is an L-system primitive which draws a cube of voxels at the current position in 3D space. The variable *X* does not effectively produce any change in the 3D model, but instead is only used as a variable in the LHS of a production rule. Non-variables are those which manipulate the current model or drawing position, but have no drawing capabilities of their own.
2. *Validity* The system tries to maximize the validity of evolved L-system grammars. Here, validity means that encoded rules are active during L-system interpretation. On the other hand, an invalid system would include rules that are not contributing to the generative process, and therefore represent bloat code. Validity is promoted in the implementation by requiring the joint use of L-system alphabet variables in the LHF and RHS of production rules, which ensure that all rules have an increased—if not guaranteed— chance to be used at least once.
3. *Complexity* The system evolves L-system productions in which complexity of structures increases during repeated iterations of rule application. Otherwise, systems will arise in which structures do not change or grow during iterative refinement. This is done both by the validity method above, and by requiring at least two variables in the RHS of all production rules.
4. *Gravity wells and fields* Gravity wells are vertex-modifiers in the L-system alphabet—denoted with 'G' and 'g'—and are inspired by the attractors and repulsor's of Hemberg and O'Reilly's *GENR8* system [19]. They were introduced to the alphabet as a way of introducing new surface normals to the surface generated from the Marching Cubes algorithm, as well as to aid in

producing more fluid and natural forms. There are two types; gravity wells (g)—which pull vertices inwards by a force F_g—and repulsion fields (G)—which push vertices away by a force F_G. In the alphabet, gravity wells are placed at the current coordinate of the pen in fractional space, as opposed to being placed at a specified voxel. Each gravity well's effect on a vertex is proportional to its distance, and wells that are stacked can accumulate their effects, sometimes cancelling one another out or amplifying their magnitude substantially.

5. *Surface-based model system (SMS)* This option lets L-system models generate upwards from a ground base, stacking voxels on a 2D grid structure. It is useful for generating variant styles of models, for example, cities and plants.

Examples of evolved L-system expressions are in Fig. 9. A selection of the functions seen in these expressions include: *C* (move forward, draw a cube), *D* (increase global depth), [and] (push and pop state), @ (rotate position on Z-axis), and *G* (create gravity well). See [4] for more details.

4 Multi-objective evaluation and other system details

We use multi-objective GP. A multi-objective optimization problem (MOP) requires the optimization of multiple features simultaneously [9]. GP evolution will evaluate models based on combinations of the evaluations from Sect. 2. Structural and aesthetic scores may affect one other in complex, non-linear ways.

We use *rank sum* (or *average rank*) MOP evaluation. Rank sum was originally intended for use in high-dimensional MOP [3, 12]. It is also effective for low-dimension problems [6], with the added benefit that outlier solutions (good scores in a minority of objectives) are discouraged. Outlier solutions are detrimental in evolutionary design applications, because having a solution that is adequate in only one or two objectives is usually unacceptable on the whole. For example, if we wish to evolve a model with DFN, mean, standard deviation and entropy fitness objectives, then a solution model having only an acceptable mean will be next to useless. Consider a search problem in which each population member i has a feature vector $\overrightarrow{V}^i = (f_1^i, \ldots, f_k^i)$, where each $f_j^i (j = 1, \ldots, k)$ is one of the different objectives or feature scores. For each feature j, all the individuals in the population are assigned an integer rank for each of its features. For example, the individual with the best score for one of the objectives gets assigned a rank 1 for that objective, the next best a rank 2, and so on. After all individuals are ranked over all objectives, each individual i is assigned a rank vector $\overrightarrow{R}^i = (r_1^i, \ldots, r_k^i)$, where $1 \leq r_j \leq N$, for a population of size N. The maximum rank can be less than N, due to tied scores. $\overrightarrow{R}$ is then normalized, by dividing each r_i by the maximum rank found for that objective. The sums of the normalized ranks are used as fitness values (low values are preferred).

We use the ECJ GP system [30]. Rank sum evaluation is introduced into ECJ for MO evaluation. Various GP and L-system parameters are given in Table 2. The GP

parameters are common in the literature [25], and the L-system parameters are discussed in Sect. 3 and [4]. A population diversity strategy is used. After the population is evaluated and fitness scores are assigned, the population is parsed for duplicate derivation strings. Individuals with duplicate strings are penalized by increasing their rank scores.

5 Evolution of 3D models using genetic programming

5.1 Single-objective runs

Initially, single-objective GP runs were done in order to determine the feasibility of the feature tests in Sect. 2.1 as evolvable metrics for model evaluation. Experiments would optimize each feature test independently, using the following target goals: volume = 4,500, dimension = (40,120,80), surface area = 52,500, unique normals = max, symmetry = 1, mean = 0.025, standard deviation = 0.25, DFN = 0, 1/f noise = 0, entropy = max, complexity = max, symmetry = 1. The parameters from Table 2 were used, except that a maximum of 30 generations was sufficient for some features. A total of 10 runs were done per feature.

Results of the runs showed that each objective is a suitable goal for optimization by GP. In other words, evolution is capable of navigating the search space posed by each individual objective, and returning solutions appropriate for the requirements of that objective. Two example fitness plots (averaged over 10 runs) are shown in Fig. 7. The DFN plot (minimization goal) is an example of a feature with stable progress over the duration of evolution, while that for unique normals (maximization goal) shows a higher deviation and less stability.

Figure 8 gives a selection of solutions for different objectives. In some cases, it is clear to understand why the evolved models corresponds to the objective used. For example, the surface area model's total elements have an area of 52,501—which almost perfectly matches the goal area of 52,500. The symmetry model is clearly symmetrical. The unique normals model is spheroid, which has a surface that will

Table 2 Typical GP and L-system parameters

GP parameter	Value	L-system parameter	Value
Generations	60	Rule RHS min/max	2/∼10
Population size	500	Iterations min/max	3/6
Crossover rate	90 %	ω min/max length	2/∼10
Max crossover depth	10	Alphabet	C, A, B, +, -, *,
Mutation rate	10 %		/, @, &, w, W,
Max mutation depth	17		d, D, h, H, [,]
Prob. terminal selection during reprod.	10 %	Variables	C, A, B
Tournament size	3	ω_i parsing time-out	30 s
Tree grow min/max	5/5		
Tree full min/max	5/12		

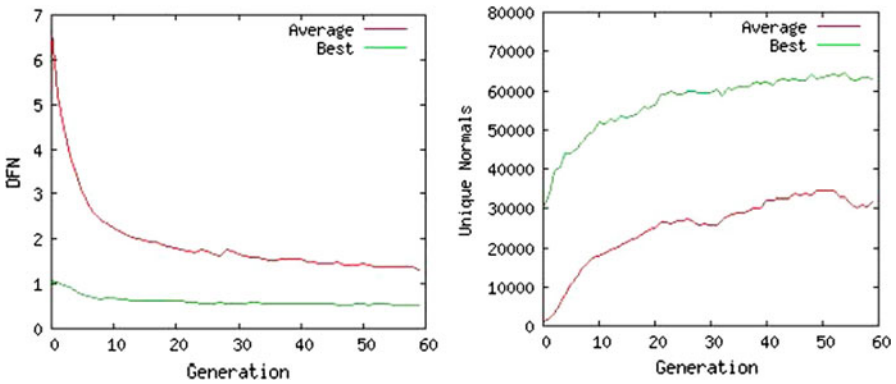

Fig. 7 Fitness plots for DFN and unique normals (avg. 10 runs). Minimize DFN, and maximize unique normals

maximize unique normals (as would the standard deviation model). Other models are less intuitive, such as the 1/f and DFN models.

5.2 Multi-objective results

The previous section shows that individual features are easily evolvable. The resulting models, however, are often of little interest. More intriguing models arise, however, when multiple features are used together. This section illustrates an assortment of results from multi-objective runs (see [4, 5] for more examples).

All the results in this section use distribution measurements of the difference of neighbouring face normals (Sect. 2), and the parameters from Table 2.

Figure 9 shows results taken from some multi-objective runs. The three objectives used are: DFN = 0.0, symmetry = 1.0, and maximize complexity. The number of iterations used to generate each model is indicated. Note that all models have a perfect symmetry.

Figure 10 shows a result obtained with 3 objectives: target DFN = 3.5, symmetry, low dimension constraint. This target DFN is within the range seen when examining pre-made models in Sect. 2.2. The raw result (left image) is a torus or ring. It has a DFN = 3.23, and symmetry = 1.0. When the same L-system is interpreted in SMS mode, it generates a bowl or chalice (DFN = 5.33, symmetry = 1.0). The right image shows the same models rendered in Blender [8], with textures and polygonal smoothing.

Figure 11 (left) is another solution from a run with the same three objectives as above. It has a DFN = 3.35, and symmetry = 1.0. Figure 11 (right) is a result from an experiment intended to find organic forms. The four objectives were: DFN = 0.0, volume = 50,000, area = 1,000,000, maximize unique normals. Low DFN's are usually seen in fluid, organic shapes. The large difference in surface area and volume is expected to result in a large, thin, complex form. The unique normals requirement increases the complexity of the geometry. The result has the form of an octopus or sea anemone.

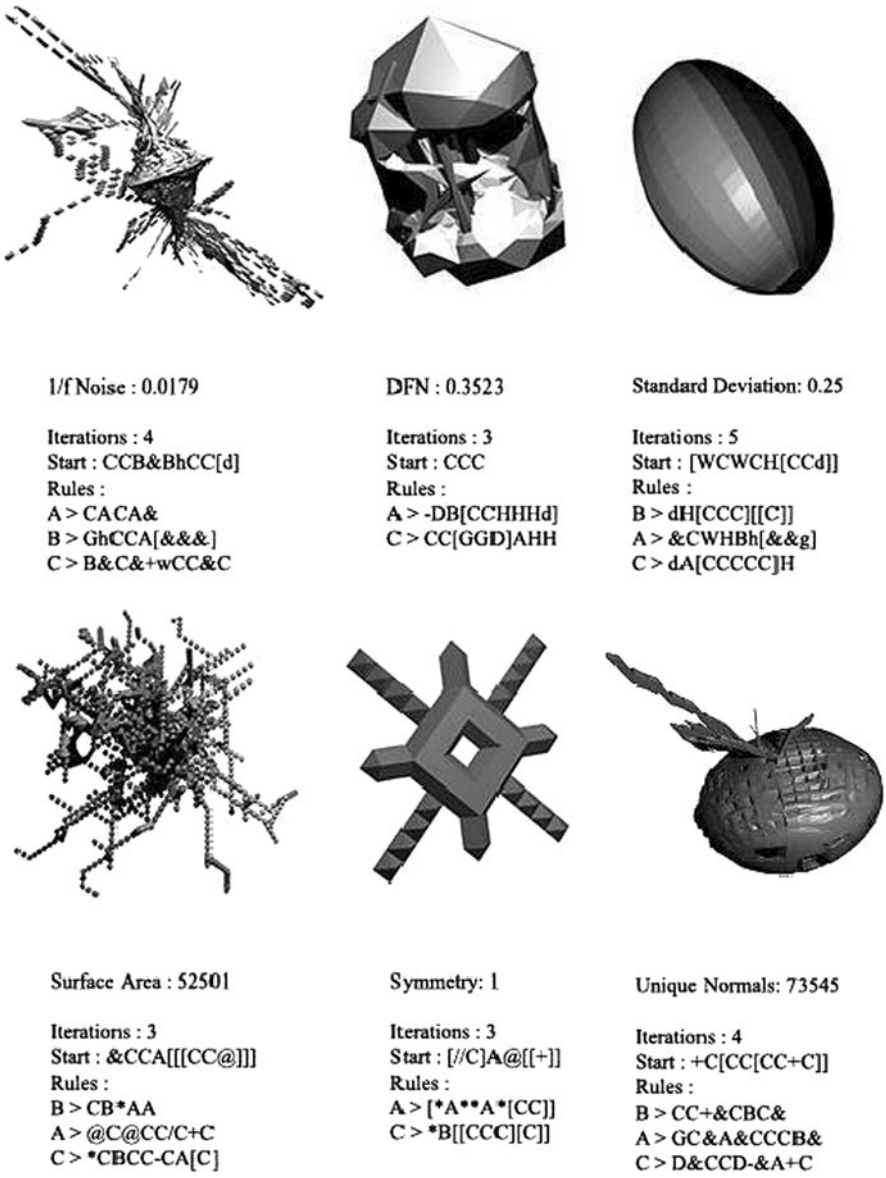

Fig. 8 Selected results for different single-objective runs

Figure 12 shows an SMS L-system result, that represents a city. It used three objectives: dimension, target surface area, and an entropy target of 5.0. Gravity wells were not used in the L-system. This Blender rendering has textures, but no polygonal shading.

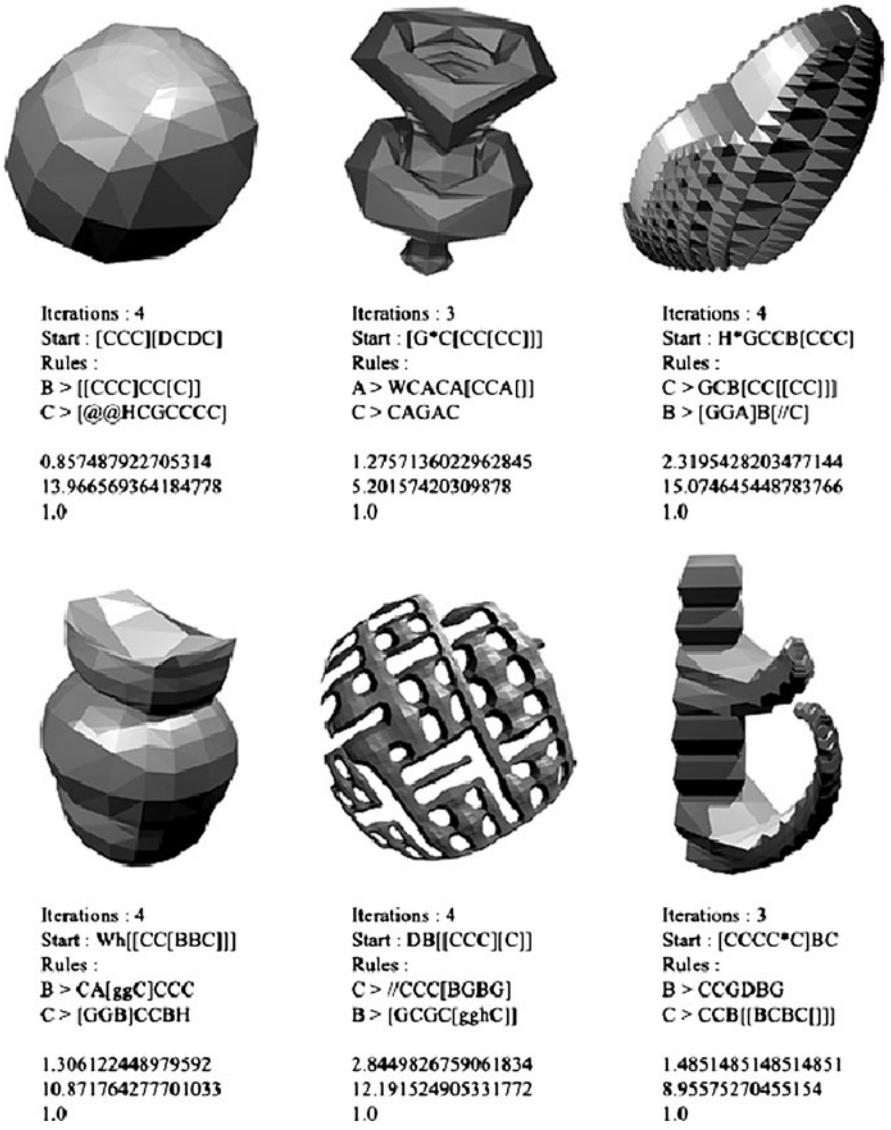

Fig. 9 Sample models chosen from the top ten individuals of various MO runs, each with the same target DFN (0.0), complexity (MAX) and symmetry (1.0) fitness values, and different starting seeds. Shown with L-system and model fitnesses below each (DFN, complexity and symmetry)

5.3 MOP evolution with conflicting aesthetic goals

Multi-objective problems often have conflicting objectives, where an improvement in one objective may be detrimental to another. To see how this might affect aesthetic model evolution, some experiments were performed that involved cases of cooperating and conflicting objectives. The features selected were DFN and

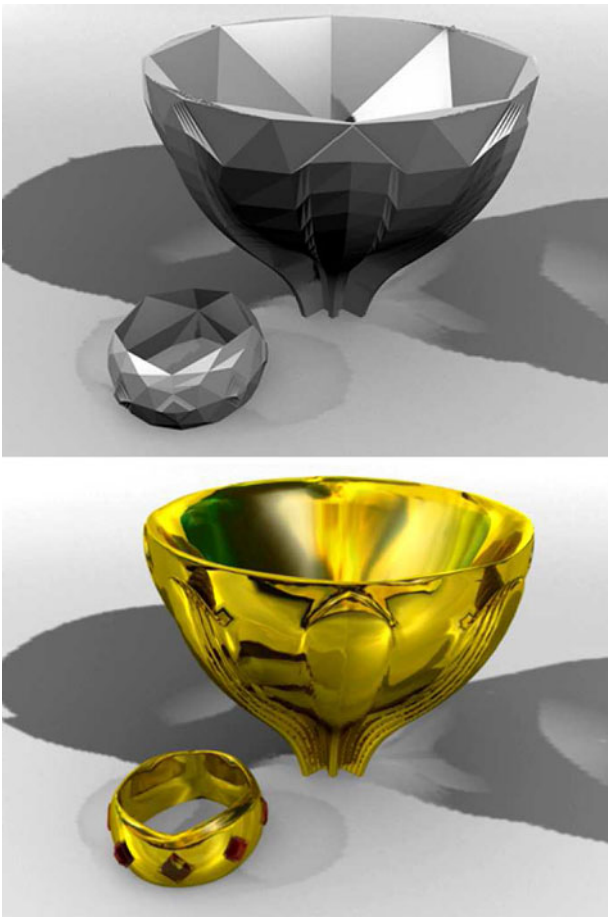

Fig. 10 Solution from a 3-objective run, using DFN, symmetry and complexity as fitness targets. Result L-system from run used in both different 'generative' L-system parsers after run completion to produce completely different models (*left* using default modeling and *right* using SMS). Raw models (*top*), and with polygonal smoothing and textures (*bottom*)

entropy. It can be seen in Table 1 that these features have a measurable negative correlation: a lower DFN score correlates with a higher entropy score, and vice versa. As mentioned in Sect. 2.2, entropy is actually a sub-calculation of normality, and so these calculations are naturally related to each other.

Multiple experiments were performed, in which both low and high DFN, and low and high entropy, are used as target objectives. The experiments using low DFN/high entropy, and high DFN/low entropy, are instances of cooperative objectives, while low DFN/low entropy, and high DFN/high entropy, are conflicting objectives. A total of 10 runs were done for each of the four cases.

The results of the runs were as expected: cooperative objectives resulted in higher-quality scores, while conflicting objectives result in less optimized scores. The performance plots in Fig. 13 are for the cases involving low DFN/high entropy,

Fig. 11 Results (textures and polygonal smoothing) hand-picked from *top* results of separate runs. Model at *left* is *vase-shaped*, while model at *right* is an organic form. Objectives were: DFN = 3.5, symmetry, dimension constraints

Fig. 12 A model with city-like characteristics, generated using SMS L-system. A relatively high target entropy of 5.0 was set as a fitness objective, along with dimension constraints and surface area objectives. The SMS grammar encouraged the model to spread and grow upwards from the center

and low DFN/low entropy. The population's average DFN is around 2 by generation 30 in the cooperative case, while it is at 6 in the conflicting experiment. Similarly, the average best DFN is around 1 in the cooperating case, and 2 in the other. The solution scatter plots for these two experiments are given in Fig. 14. Solutions having similar scores were found in both cases. However, in the cooperative case (a), there is a much higher density of solutions converging to the desired target area of high-entropy and low-DFN (the top-left portion of scatter). This convergence is not apparent in the conflicting plot, where there are also very few solutions below a DFN of 2. A greater density of solutions implies that it is more likely a given run will generate one or more solutions, compared to a search space in which solutions are less likely to be discovered. Also, a greater density of solutions means the user

has a greater number of possibilities to select from. This is particularly important in evolutionary design applications, in which many other subjective factors will likely be involved when judging the ultimate suitability of different results.

Figure 15 shows some representative sample solutions from the four experiments. Again, the top two rows in the figure are the cases of conflicting objectives. As is clear from the scatter plots, a variety of models clearly arise from each experiment, thus showing the diversity of results possible. The simplistic solutions in the low/low (A, B) models are mostly due to the low entropy of the models. Similarly, the simple models in the high/low row are a result of the low entropies and very high DFN's. These feature targets naturally prefer very simple 3D structures.

Situations often arise in multi-objective runs in which there are indeed differing conflicting objectives, as shown above. It can also be the case that some objectives are more critical than others, perhaps because they denote cost or functionality factors that cannot be ignored. Our summed rank scoring strategy has some advantages in this regard. Firstly, as with Pareto ranking, the initial ranking of objectives replace the raw objective measurements, thus preventing direct reconciliation of different objective measurements with one another. Weights can then be introduced to the normalized ranks, to increase the priority of critical objectives. Although we did not do this in these experiments, other research has found it to be a very useful strategy [14].

6 Human survey of results

Although many high-scoring models were evolved over many runs, it is difficult to say whether people would find them appealing. A model with a low DFN or good 1/f score does not guarantee that the object has aesthetic value. Also, evolved 3D models are often abstract and non-representational.

To verify whether there is evidence of aesthetic merit of evolved models, a human survey was conducted. One priority was to conduct a simple survey with modest goals; a comprehensive investigation of different aesthetic models would be premature at this stage of our research. The basic question we wanted to pose is: *Do*

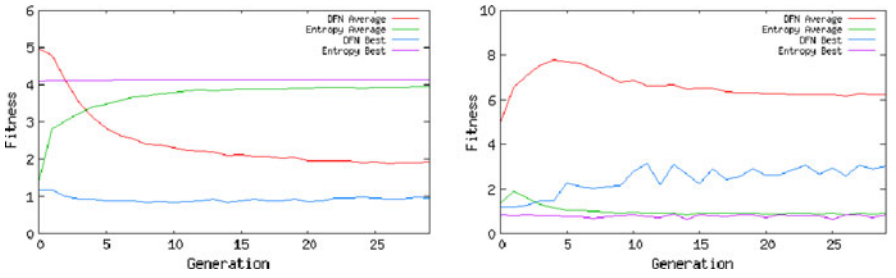

Fig. 13 Performance plots for cooperative and conflicting objectives. Cooperative—low DFN/high entropy *(left)*. Conflicting—low DFN/low entropy (*right*)

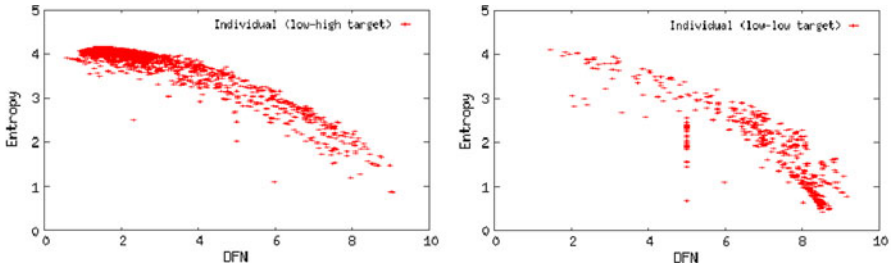

Fig. 14 Solution scatter plots for cooperative (*left*) and conflicting (*right*) objectives

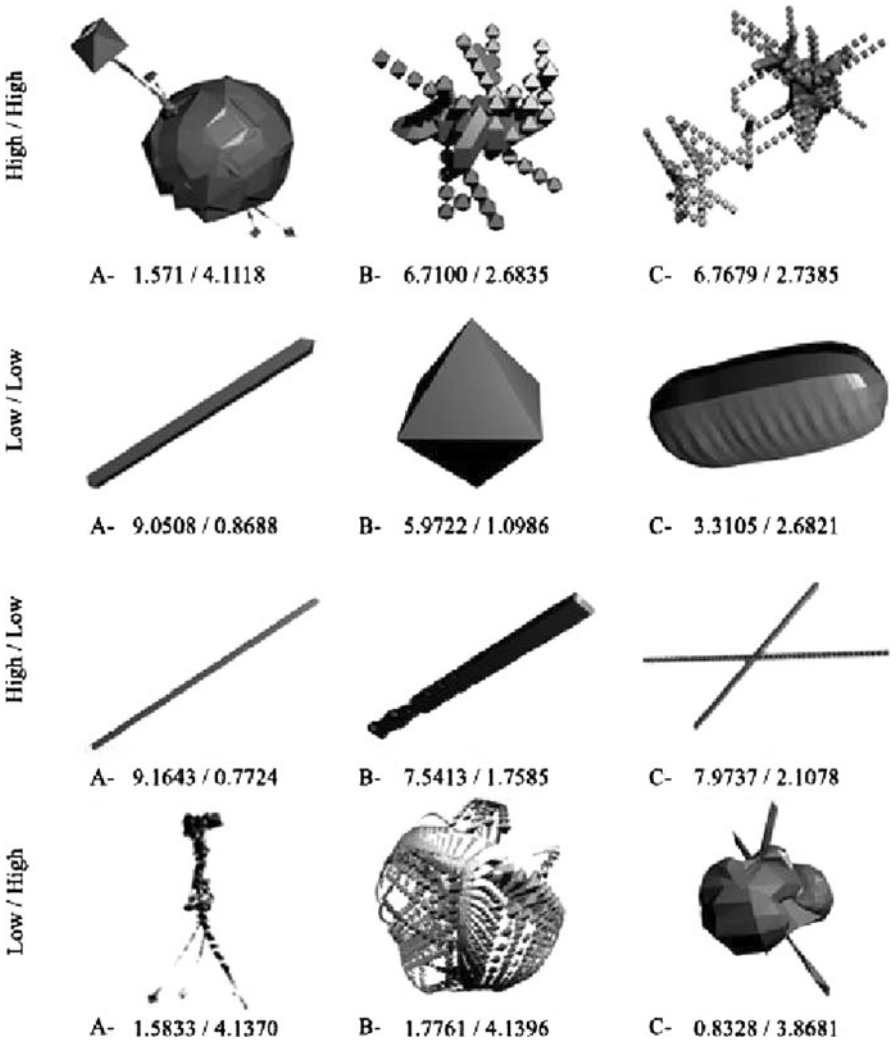

Fig. 15 Samples of solutions from the four experiments (DFN/entropy)

people prefer highly evolved models with good aesthetic scores, over unevolved models with low aesthetic scores? The survey taker will be presented with a choice between two models: one model from the end of a run, which is highly evolved with optimized aesthetic scores; and one model from generation 0, with low scores. In posing the above question, we do not consider the specific aesthetic models used, nor will we compare aesthetic models with each other. Our goal is to therefore examine whether people show a preference of evolved models over unevolved ones in general. An affirmative result would give evidence that the aesthetic criteria are may be affecting the nature of models in an aesthetically pleasing way. In summary, the survey will give evidence whether the aesthetic fitness criteria collectively contribute any aesthetic qualities to generated 3D models.

The survey was constructed as follows, and is similar to one in [52]. A diverse set of runs using different sets of fitness measures were considered. DFN was a common objective in all these runs, in combination with other selections of aesthetic objectives. 20 top-ranked solutions from the final generation of separate runs were randomly chosen as evolved "high-quality" models, with optimized fitness scores. Next, 20 models were selected from the same runs, by randomly selecting models from the middle-ranked individuals from generation 0. These were designated unevolved "low-quality" models. Although the middle rank individuals had low aesthetic scores, they were still non-trivial and often very complex. (The lowest-ranked individuals were not selected, because these were inevitably simple primitives such as cubes, which would bias the user towards assigning low scores of merit.) The high- and low-quality models were randomly paired with each other. A web site was constructed, in which a viewer is asked to view each pair of unlabelled models. A Java 3D viewer displays the raw polygonal models using flat shading, and permits them to be interactively rotated and examined. The survey taker is instructed, "You will be presented with one pair of models at a time, and are asked to choose which one is most interesting—or appealing—to you. When choosing, try to imagine you're in a situation where you must choose between spending money on one model or the other—which one would you want sitting on your desk for an extended period of time?" (See Fig. 16). The subject indicates which of the two models is preferable. This is repeated for all 20 pairs. All survey takers are presented with the same pairs of models, in the same order. The survey requires all 20 pairs to be evaluated in order to be considered. The survey was advertised to computer science students at Brock University. They were encouraged to invite others to participate.

A total of 34 completed surveys were submitted. Figure 17 shows a tally of the survey results. A non-parametric sign test was computed for the results [51], to see whether users expressed preference for the high-quality models in the presented pairs. The sign test confirmed that participants preferred high-fitness models, with a 95 % certainty. Approximately 71 % of survey takers chose high quality models over low quality ones. It was noted that in seven pairs of models the low quality model was not Pareto dominated by its high quality counterpart. This is to be expected with MO evaluation, and with rank sum scoring. When these cases were excluded, it was found that 80 % of survey takers preferred the higher-fitness models (see right-side histogram in Fig. 17).

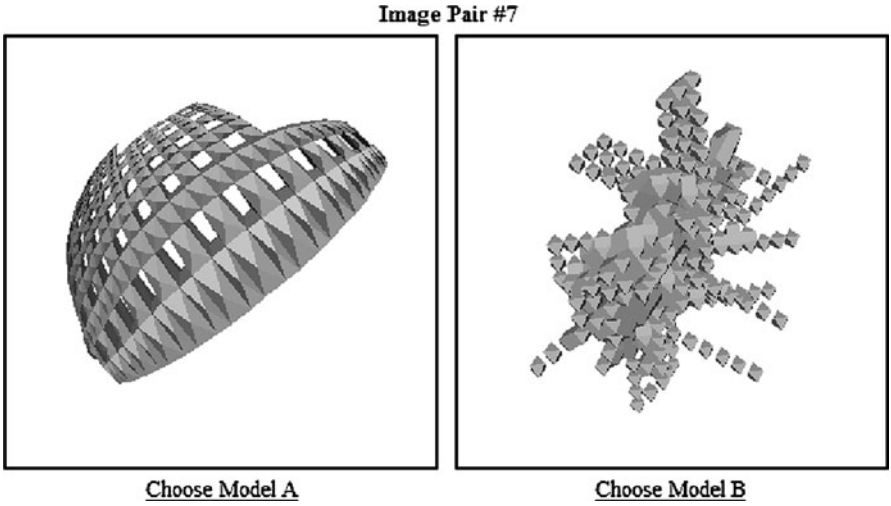

Fig. 16 Example of model selection in user survey

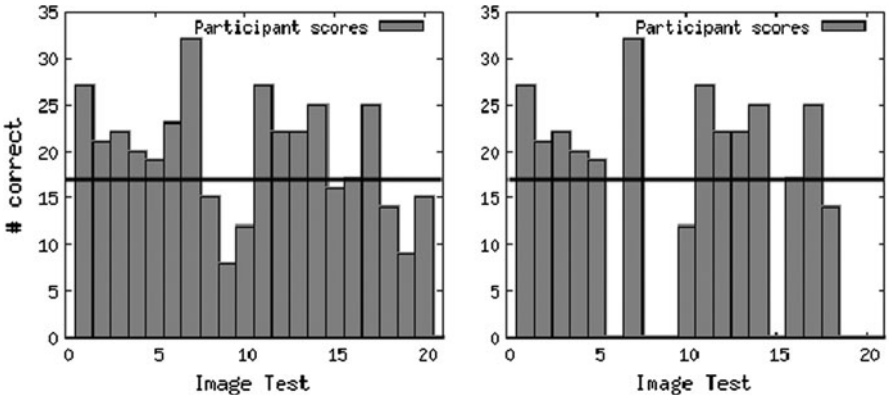

Fig. 17 *Left* Histogram displaying number of times the correct image (image with hypothesized-best DFN) was chosen by participants, out of 34, for all 20 tests. *Horizontal line* at 17 shows the expected average number of correct choices for a uniform distribution. *Right* Same as left, except with non-dominating model pairs removed

The results of the survey are encouraging, as they show a user preference for models with higher aesthetic fitness. This gives some evidence that the aesthetic evaluation of 3D models may be justified. However, there are many new questions arising. For example, it might be the case that other fitness objectives besides the ones we used, that impose a degree of order on the surface structure of models, might be preferable over the unevolved models from generation 0. In other words, our aesthetic models might belong to a large family of order-inducing functions, which might also produce aesthetically viable models. Furthermore, it is still

unknown which of the fitness criteria we used might be preferable to human viewers. More research is required.

7 Comparisons to related work

This research is closely related to other work in evolutionary design of 3D models, shape grammars and L-systems. 3D building models are evolved with GP and sum of ranks in [10]. Rather than evolving an L-system, a shape grammar is evolved in order to generate complex polygonal structures. As with out work, the fitness evaluation is automatic and uses many geometric measurements that are introduced in this paper (Sect. 2.1). Hemberg et al. [20] evolve curved surfaces in a semi-automated GP environment, which directly inspired the idea of gravity wells and repulsion fields from their use of repulsion and attraction forces used to transform model geometry. Jacob et al.'s [22] L-system encoding is an early inspiration for ours, and was his function set was extended greatly to incorporate the necessary system constraints introduced earlier. Jacob later explored problems such as plant evolution using his system [21].

O'Neill et al. uses evolution to evolve 2D L-systems [35], 2D shape grammars [37], and 3D shape grammars [36]. All of these papers use basic geometric or functional (problem requirement) constraints, and no aesthetic modelling. The concept of using EC as a tool for more practical problems is more common than for use in aesthetic form generation, which serves to outline the growing interest in the field. A clear direction for future research is to combine aesthetic models with functional constraints, in order to evolve aesthetically-pleasing models that satisfy functional requirements for a problem. This combination has a wide variety of more specific applications, such as the generation of structures (interior and exterior), virtual landscapes, and 2D and 3D art. It is also more common for such systems to use guided evolution, whereas our research focuses on automatic evaluation.

8 Conclusion

This research shows an application of aesthetic evaluation to 3D model evolution. By selecting thoughtful combinations of fitness criteria, various classes of models were evolved. The aesthetic distribution functions act as generalized "model filters", which often promote models having particular characteristics: organic or synthetic, complex or simple, symmetrical, and others. The approach is similar in spirit to other evo-art applications, in which evolution is used as a tool for design exploration and discovery. Our human survey gives evidence that the aesthetic criteria may have merit. However, more research in this area is required.

There are many directions for future research. More work is needed in studying aesthetic models of 3D structures. Our analyses of pre-made models in Sect. 2.2 should be done in more detail, by selecting models in a more methodical manner. The effect of surface features and model size and complexity should be considered, as well as the effect of the modelling environment used to generate the model. It is

clear that an aesthetic score of an object's 3D model is highly sensitive to the nature of its surface mesh, and this mesh will vary considerably if it is generated by a simple L-System, a voxel-space L-System (as used in this paper), a polygonal CAD 3D modeller, a 3D scanner, or a set of 3D splines defining a continuous surface model. Many more aesthetic measurements are possible, for example, fractal dimensions and more advanced symmetry measurements. Our numeric scores are tied to our particular L-system implementation, with its voxelization and marching cubes surface generation. We expect, however, that the results will generalize appropriately to other modelling paradigms. Future work should verify this, by considering other modelling languages such as shape grammars [45] and curvilinear surfaces [20].

Acknowledgments Thanks to Beatrice Ombuki-Berman, Sheridan Houghten, Bill Ralph, and Cale Fairchild for their advice and assistance, as well as anonymous referees for their constructive comments. This research is supported by an OGSST award and NSERC Discovery Grant 138467.

References

1. A3D, Archive 3D. In: http://archive3d.net/ (2012)
2. P. Bentley, D. Corne, *Creative Evolutionary Systems*. (Morgan Kaufmann, USA, 2002)
3. P. Bentley, J. Wakefield, in *Soft Computing in Engineering Design and Manufacturing*. Finding acceptable solutions in the pareto-optimal range using multiobjective genetic algorithms (Springer, Berlin, 1997)
4. S. Bergen, Automatic structure Generation using genetic programming and fractal geometry. Master's thesis, (Department of Computer Science, Brock University, 2011)
5. S. Bergen, Aesthetic 3D model evolution gallery. http://www.cosc.brocku.ca/∼bross/Aesth3Dmodels/ (2012)
6. S. Bergen, B. Ross, in *Genetic Programming—Theory and Practice VIII*. Evolutionary art using summed multi-objective ranks. (Springer, Berlin, 2010), pp. 227–244
7. G.D. Birkhoff, *Aesthetic Measure*. (Harvard University Press, Cambridge, 1933)
8. Blender, http://www.blender.org/. Last Accessed 4 Dec 2011
9. C.C. Coello, G. Lamont, D.V. Veldhuizen, *Evolutionary Algorithms for Solving Multi-Objective Problems*. 2nd edn. (Kluwer, Dordrecht, 2007)
10. C. Coia, B. Ross, in *Proceedings of the CEC 2011, IEEE*. Automatic evolution of conceptual building architectures (2011)
11. J. Conway, H. Burgiel, C. Goodman-Strauss, *The Symmetries of Things*. (CRC Press, Boca Raton, 2008)
12. D. Corne, J. Knowles, in *Proceedings of the GECCO 2007*. Techniques for highly multiobjective optimisation: some nondominated points are better than others. (ACM Press, New York, 2007), pp. 773–780
13. M. Field, M. Golubitsky, *Symmetry in chaos*. (SIAM, Philadelphia, 2009)
14. R. Flack, Evolution of architectural floor plans. Master's thesis, (Department of Computer Science, Brock University, Canada, 2010)
15. D. Graham, C. Redies, Statistical regularities in art: relations with visual coding and perception. Vision. Res. **50**, 1503–1509 (2010)
16. G. Greenfield, in *Proceedings of the CEC 2003*. Evolving aesthetic images using multiobjective optimization (2003), pp. 1903–1909
17. G. Gunlu, H. Bilge, in *ICSCCW*. Symmetry analysis for 2D images by using DCT coefficients (2009), pp. 1–4
18. E. den Heijer, A. Eiben, in *Proceedings of the EvoMusArt, LNCS 6025*. Comparing aesthetic measures for evolutionary art, vol. 2. (Springer, Berlin, 2010), pp. 311–320
19. M. Hemberg, U.M. O'Reilly, in *GECCO 2002: Proceedings of the Bird of a Feather Workshops*, ed. by A. Barry. GENR8—using grammatical evolution in a surface design tool. (AAAI, New York, 2002), pp. 120–123

20. M. Hemberg, U.M. O'Reilly, A. Menges, K. Jones, M. da Costa Goncalves, S.R. Fuchs, in *The Art of Artificial Evolution*. Genr8: architects' experience with an emergent design tool. (Springer, Berlin, 2008)
21. C. Jacob, *Illustrating Evolutionary Computation with Mathematica*. (Morgan Kaufmann, USA, 2001)
22. C. Jacob, A. Lindenmayer, G. Rozenberg, in *Parallel Problem Solving from Nature III, Lecture Notes in Computer Science Genetic l-system Programming*. Genetic l-system programming. (Springer, Berlin, 1994), pp. 334–343
23. H. Kawabata, S. Zeki, Neural correlates of beauty. Neurophysiology **91**, 1699–1705 (2004)
24. M. Kazhdan, B. Chazelle, D. Dobkin, T. Funkhouser, S. Rusinkiewicz, A reflective symmetry descriptor for 3D models. Algorithmica **38**(1), 201–225 (2004)
25. J. Koza, *Genetic Programming: On the Programming of Computers by Means of Natural Selection*. (MIT Press, Cambridge, 1992)
26. M. Li, P. Vitanyi, *An Introduction to Kolmogorov Complexity and its Applications: Preface to the First Edition*. (Springer, New York, 1997)
27. H. Lipson, W. Cochran, *The Determination of Crystal Structures—3rd Revised and Enlarged ed*. (Cornell University Press, Ithaca, 1966)
28. M. Livingstone, *Vision and Art: The Biology of Seeing*. (Abrams, New York, 2002)
29. W.E. Lorensen, H.E. Cline, Marching cubes: A high resolution 3D surface construction algorithm. SIGGRAPH 87 **21**, 163–169 (1987)
30. S. Luke, Ecj. http://cs.gmu.edu/eclab/projects/ecj/. Last Accessed 3 Dec 2011
31. P. Machado, A. Cardoso, in *Proceedings of the XIVth Brazilian Symposium on AI*. Computing aesthetics. (Springer, Berlin, 1998), pp. 239–249
32. J. McCormack, in *Complex Systems: From Biology to Computation*. Interactive evolution of l-system grammars for computer graphics modelling. (ISO Press, Amsterdam, 1993), pp. 118–130
33. E. Milotti, 1/f noise: a pedagogical review. Arxiv preprint, physics/0204033 http://arxiv.org/abs/physics/0204033 (2002)
34. C. Neufeld, B. Ross, W. Ralph, The evolution of artistic filters. In: J. Romero, P. Machado (eds) *The Art of Artificial Evolution*, (Springer, Berlin, 2008)
35. M. O'Neill, A. Brabazon, in *Evolutionary Computation*. Evolving a logo design using lindenmayer systems. (2008), pp. 3788–3794
36. M. O'Neill, J. McDermott, J. Swafford, J. Byrne, E. Hemberg, A. Brabazon, Evolutionary design using grammatical evolution and shape grammars: designing a shelter. Intl. J. Des. Eng. **3**, 4–24 (2010)
37. M. O'Neill, J. Swafford, J. McDermott, J. Byrne, A. Brabazon, E. Shotton, C. McNally, M. Hemberg, in *Proceedings of the GECCO '09*. Shape grammars and grammatical evolution for evolutionary design. (ACM, New York, 2009), pp. 1035–1042
38. W. Pang, K. Hui, Interactive evolutionary 3D fractal modeling. Vis. Comput. **26**, 1467–1483 (2010)
39. W. Ralph, Painting the bell curve: the occurrence of the normal distribution in fine art. (2006, in preparation)
40. J. Rigau, M. Feixas, M. Sbert, in *Proceedings of the Eurographics Workshop Computational Aesthetics in Graphics, Visualization and Imaging*. Conceptualizing Birkhoff's aesthetic measure using Shannon entropy and kolmogorov complexity. (2007), pp. 105–112
41. J. Rosen, *Symmetry Discovered: Concepts and Applications in Nature and Science*. (Dover, New York, 1998)
42. B. Ross, W. Ralph, H. Zong, in *CEC 2006*. Evolutionary image synthesis using a model of aesthetics (2006)
43. K. Sims, Evolving Virtual Creatures. In: *SIGGRAPH 94*, pp. 15–22 (1994)
44. B. Spehar, C. Clifford, B. Newell, R. Taylor, Universal aesthetic of fractals. Comput. Graph. **27**, 813–820 (2003)
45. G. Stiny, Introduction to shape and shape grammars. Environ. Plan. B **7**, 343–351 (1980)
46. N. Svangard, P. Nordin, in *EvoWorkshops 2004, LNCS 3005*. Automated aesthetic selection of evolutionary art by distance based classification of genomes and phenomes using the Universal similarity metric. (Springer, Berlin, 2004), pp. 447–456
47. S. Todd, W. Latham, *Evolutionary Art and Computers*. (Academic Press, London, 1992)
48. M. Triola, *Essentials of Statistics*. (Pearson Education, New Jersey, 2010)
49. R. Voss, J. Clarke, 1/f noise in music: music from 1/f noise. J. Acoust. Soc. Am. **63**(1), 258–263 (1978)

50. von P. Buelow, *Genetically Engineered Architecture—Design Exploration with Evolutionary Computation*. (VDM, Saarbrücken, 2007)
51. D.D. Wackerly, W.M. III, R.L. Scheaffer, *Mathematical Statistics with Applications*, 6th edn. (Duxbury Advanced Series, CA, 2002)
52. P. Walsh, P. Gade, in *IEEE Congress on Evolutionary Computation*. The use of an aesthetic measure for the evolution of fractal landscapes. (IEEE, New York, 2011), pp. 1613–1619
53. A. Watt, F. Policarpo, *The Computer Image*. (Addison-Wesley, Reading, MA, 1998)
54. Wikipedia: golden ratio. http://en.wikipedia.org/wiki/Golden_ratio (2012). Last Accessed 22 Nov 2012
55. Wikipedia: psychology of art. http://en.wikipedia.org/wiki/Psychology_of_art (2012). Last Accessed 22 Nov 2012
56. Wikipedia: rule of thirds. http://en.wikipedia.org/wiki/Rule_of_thirds (2012). Last Accessed 22 Nov 2012