Using DNA to Generate 3D Organic Art Forms

William Latham
Goldsmiths
University of London
w.latham@gold.ac.uk

Miki Shaw
Goldsmiths
University of London
miki@gold.ac.uk

Stephen Todd and
Frederic F. Leymarie
Goldsmiths
University of London
ffl@gold.ac.uk

Categories and Subject Descriptors:
J.5 [**Computer Applications**]:Arts and Humanities

General Terms: Design, Experimentation

Keywords: 3D Shape, DNA, Growth, Visualization

1. OVERVIEW

We present a novel "biological" approach to define and evolve 3D art forms. The work combines a re-implementation of the *FormGrow* system of Todd and Latham [1] with an external source to define the shapes: DNA sequences. *FormGrow* is a "virtual machine" producing 3D computer art forms or designs. It embodies the particular "organic" aesthetics favored by Latham together with a "shape grammar" made of primitives, *e.g.*, horn-like structures, transforms or assembly rules, and a number of parameters encoding, *e.g.*, color, scale or texture. We have re-visited the *FormGrow* system of Latham and Todd and brought it back to life in a modern implementation taking advantage of standard graphics libraries and portable coding. The main emphasis here however is on how we are bringing this system closer to the realm of biology.

Real DNA data, in the form of nucleotide sequences, is transformed via a series of tables we have empirically designed to become readable by *FormGrow*. These tables process nucleotides as "codon" triplets of data as would ribosomes in a live cell. Notions of "start," "stop," and "junk" DNA code are also embedded in our system. We explore the application of our novel method to generate 3D organic art forms in the visualisation of particular genetic defects, *e.g.*, sickle cell anaemia mutation. Using appropriately designed transformation tables, this subtle mutation can be "visualised" in a striking manner.

Our motivation for re-visiting Latham and Todd's work is that it is a powerful system which offers the possibility of generating organic-like shapes and which from its origins was meant as a metaphor to nature's way of evolving forms. In re-visiting this work, on the one hand we bring up-to-date the technology developed in [1] in the context of recent advances in graphics and computational geometry, and on the other hand we bring it much closer to biology via the recent advances made in understanding the working of nature in the fields of genomics and proteomics.

2. USE OF DNA IN FORMGROW

The DNA, a very long string of nucleotides, of a given organism can be said to fundamentally specify its unique shape. On a lower level, DNA encodes proteins, and it is these proteins that constitute the body's key builders and building blocks. A protein is also a string of simpler molecules: the amino acids. There are twenty types of amino acid and only four types of nucleotide, and the DNA translation mechanism looks at these nucleotides in groups of three, triplets called "codons." Every codon translates to a single amino acid, and working down the chain of DNA generates the corresponding chain of amino acids, so yielding a protein.

Following this model, we created an analogous translation system to convert DNA sequences into *FormGrow* code. At a coarse level, *FormGrow* code can be viewed as a series of function calls, with each function requiring a small number of arguments (this number varies from 0 to 3 depending on the particular function). So we created 2 translation tables: the "transform table," which translates from codons to transformational functions; and the "number table," which translates from codons to numerical arguments (integers in the range 0 to 63). Given our input sequence, we translate the first codon into a function using the transform table, and then generate numerical arguments for that function by translating the following codons into numbers, using the number table. Once we have sufficient arguments, we return to the transform table to generate our next function, and so the cycle continues. Finally we render the generated *FormGrow* code to produce a 3D shape.

It is interesting to note some similarities between nature's translation method and ours. In the original translation table there is a "start" codon which signals that a new protein is being specified. Likewise, in our transform table, the "add horn" transform flags the beginning of a new shape. The "stop" codon is also mirrored in our system. A side effect of adopting the "start" and "stop" mechanism is that we end up with large sections of "junk code," *i.e.*, code which generates no proteins or shapes because it lies in a non-coding section of the sequence. By changing the layout of the transform table we could affect the proportion of junk code produced. We experimented with producing a few different iterations of the transform table in order to get a balance of functions that would produce an interesting variety of shapes.

Could this methodology have more direct and short-term scientific applications as well? While our shapes bear no resemblance to the proteins that the genes encode, they are still being driven by the same initial DNA sequences. So it is possible that we could use our system as a visualisation tool. Additionally, our system is deterministic; thus, given a sequence and transform table set, the same shape will result every time. However, we have not yet implemented metrics to relate large (nucleotide) changes to cause proportionally large changes in the resulting shapes. This is one of our next avenues of exploration.

REFERENCE:

[1] S. Todd and W. Latham. *Evolutionary Art and Computers*, Academic Press, 1992.

GECCO'07, July 7–11, 2007, London, England, United Kingdom.
ACM 978-1-59593-697-4/07/0007.