In Proceedings of the Computer Human Interaction Conference (CHI 2008). New York, NY: ACM, 2008

Picbreeder: Evolving Pictures Collaboratively Online

Jimmy Secretan, Nicholas Beato, David B. D'Ambrosio, Adelein Rodriguez, Adam Campbell, and Kenneth O. Stanley
School of Electrical Engineering and Computer Science
University of Central Florida, Orlando, FL 32816-2362
{jsecreta,nbeato,ddambro,adeleinr,acampbel,kstanley}@eecs.ucf.edu

ABSTRACT

Picbreeder is an online service that allows users to collaboratively evolve images. Like in other Interactive Evolutionary Computation (IEC) programs, users evolve images in Picbreeder by selecting ones that appeal to them to produce a new generation. However, Picbreeder also offers an online community in which to share these images, and most importantly, the ability to *continue* evolving others' images. Through this process of branching from other images, and through continually increasing image complexity made possible by the NeuroEvolution of Augmenting Topologies (NEAT) algorithm, evolved images proliferate unlike in any other current IEC systems. Picbreeder enables all users, regardless of talent, to participate in a creative, exploratory process. This paper details how Picbreeder encourages innovation, featuring images that were collaboratively evolved.

Author Keywords

Interactive Evolutionary Computation, Collaborative Interactive Evolution

ACM Classification Keywords

H.5.3 [Information Interface]: Group and Organization Interfaces — Collaborative computing, Computer-supported cooperative work, I.2.6 [Artificial Intelligence]: Learning — Connectionism and neural nets, J.5 [Arts and Humanities]

INTRODUCTION

Interactive Evolutionary Computation (IEC), i.e. artificial evolution guided through human direction, can potentially create significant digital artifacts without explicit design. IEC applications generate a random population of individuals from which the user selects those that are most appealing. Selected individuals then become the parents of the next generation. As this process iterates, the individuals evolve to satisfy the user. IEC is well-suited to domains in which success and failure are subjective and difficult to formalize. For example, traditional evolutionary algorithms would struggle to determine whether an image is "attractive" or not, yet humans can easily perform such evaluations. IEC can thus generate a variety of digital artifacts including images [23, 25, 26], music [8], particle systems [17], and dancing avatars [6].

Collaborative Interactive Evolution (CIE) augments IEC to involve multiple users, adding a social dimension that increases the variety and number of solutions that can be evolved. Yet effectively combining the opinions of multiple users is nontrivial because the preferences and goals of multiple users are often in conflict. Picbreeder [3], an online service where Internet users collaborate to evolve pictures, introduces an effective new approach to this challenge by allowing each user to guide a branch of evolution on its own unique path.

Picbreeder users can begin evolving in one of two ways: In the traditional option, users start from a random population of images and select those that they like, which spawn a new generation. When the user is satisfied with an image, he or she *publishes* the image, making it visible to others. The key idea in Picbreeder is that other users can alternatively begin evolving from an *already* published image instead of from scratch by *branching* the image, thereby continuing its evolution. Through the compounding effect of branching, and the ability of the underlying NeuroEvolution of Augmenting Topologies (NEAT) algorithm to increase the images' complexity, users collaboratively search for images.

Picbreeder contributes a novel way to generate and maintain a large catalog of user-created content by enabling collaborative search through vast design spaces by multiple users. It empowers users of all experience levels to enjoy being recognized for their creative contributions. Users thereby experience a new kind of creative social recreation through playful collaborative exploration. While Picbreeder focuses on generating images, it embodies a general framework that can harness the power of a large group to search together.

This paper explains how Picbreeder allows users to collaboratively evolve images through its web-based portal and demonstrates its potential through evolved images. A principled analysis demonstrates how publishing and branching, which are unique to Picbreeder within IEC, address key challenges inherent in many CIE systems.

BACKGROUND

This section reviews several technologies from which Picbreeder draws its inspiration, beginning with groupware and then moving to evolutionary approaches.

Groupware

Groupware, which implements Computer Supported Cooperative Work (CSCW) [14], is software that coordinates users interacting across a network for a shared purpose. For example, Sourceforge [4] hosts open source projects and allows software and documentation to be authored by loosely associated groups across the Internet. In the Sourceforge community, there is an implicit branching of projects in which many influence spin-off projects, or enable others.

Another example, Wikipedia [5], is a popular online encyclopedia that allows numerous users across the Internet to edit and add subjects about which they have some expertise. Similarly, in IBM's *Many Eyes* [9], users can create and share visualizations of data, from which the Internet community can draw and share their interpretations. While such services encourage collaborative work, they also require specific talents or expertise, which limits participation.

An early collaborative system based explicitly on branching was introduced by Jorda to help musicians collectively produce compositions [19]. In Jorda's *Faust* system, users can branch from a previously saved song and edit it to create a variant that is then saved back into the tree. Faust produced a collection of appealing songs, providing precedent for the approach in Picbreeder. However, Faust did not employ an evolutionary algorithm; rather, users had to directly manipulate the notes of the song and therefore required some basic musical knowledge. In contrast, Picbreeder enables all users, including non-experts, to contribute creatively.

Interactive Evolutionary Computation

In single-user interactive evolution [34], the user is presented with a set of alternatives generated by the system. This initial *population* is then evolved over generations through a process similar to animal breeding: In each generation, the user *selects* the most promising designs, which are then mated and mutated to create the next generation. In effect, IEC assists the user in exploring a potentially vast design space in which he or she may have little knowledge or expertise.

IEC is often applied to images in what is commonly called *genetic art* [24,27,35]. Genetic art programs follow the original Blind Watchmaker idea from Dawkins [10], in which simple genetically-encoded patterns are evolved through an interactive interface. IEC principles have also influenced practical digital graphics tools, such as Kai's Power Tools Texture Explorer [20] and Apophysis [2], which interactively evolve image textures and digital flame designs, respectively.

While IEC is a powerful approach to helping users generate digital artifacts, results are often limited by human fatigue [34]. According to Takagi [34], a normal IEC process should only require 10 to 20 generations from the user. However, it is challenging to produce notable images within this limit.

Collaborative Interactive Evolution

CIE [33] systems involve multiple users in one IEC application, hoping to create products with broader appeal and greater significance. CIE systems can be physical installations or online services; each has unique methods through which to merge input from different users.

Among the first CIE installations were two museum exhibits by Sims [25,26]. The Genetic Images [25] exhibit let visitors stand in front of several displays to select pictures to produce the next generation. Other users could select individuals at the same time, or continue the evolutionary run where prior users left off. In the Galapagos [26] exhibit, visitors evolved virtual three-dimensional creatures with a similar interface. While these exhibits were innovative, the museum environment does not encourage users to frequently return to the installation, and users cannot begin evolution from a point other than where their immediate predecessors leaves off.

Pfeiffer [21] is another pioneering CIE system that allows users to endorse candidates for further evolution in an online, multiuser environment aimed exclusively at evolving snowflakes. Even with this limited scope, Pfeiffer processed over 68,000 user inputs from every continent since 2001. While Pfeiffer demonstrates that users are willing to participate in CIE, it raises the question whether a broader scope of images would be possible to evolve collaboratively.

Szumlanski et al. [33] introduced a different CIE approach based on conflict resolution. Users log in to vote on a particular individual selected by the system. To overcome user fatigue, the system combines these inputs to form a *fitness function*, i.e. a measure of quality, for a traditional genetic algorithm. The genetic algorithm then evolves an individual to meet the combined user requirements. This approach evolved characters for an interactive story. While the system effectively circumvents user fatigue, it does not encourage a proliferation of content because a large collection of user input is combined to reach only a single objective.

Another system, Imagebreeder [16] also offers an online community coupled with an IEC client for evolving images. Users can save their creations to a general pool to be viewed by the larger community. However, Imagebreeder does not include the ability to continue evolving others' images, which means that the complexity of images evolved is limited to what a single user can evolve before fatiguing.

These systems highlight the intriguing potential for evolution to benefit from the collective input of users across the world. However, they also signal a potential drawback: Users' preferences are often in conflict, resulting in mediocre results that cannot satisfy divided opinions. Furthermore, genetic representations often limit genuine creativity by constraining search to a predetermined set of possibilities (e.g. only snowflakes [21]). To evolve a broad class of images, an open-ended representation is needed that can potentially represent anything. Because of its ability to complexify, the NEAT evolutionary algorithm satisfies this requirement, as explained in the next section.

NeuroEvolution of Augmenting Topologies (NEAT)

The creative process can be constrained for novices by controlling the method through which the space of images is searched. In Picbreeder, this constraint is provided by an evolutionary algorithm called NeuroEvolution of Augmenting Topologies (NEAT [30, 32]), which addresses several fundamental challenges in evolving complex structures. Although NEAT was originally introduced as a method for evolving artificial neural networks (ANNs), a major appeal of NEAT is its ability to evolve increasingly complex structures of any type, so that evolutionary search is not limited to a fixed space of possibilities.

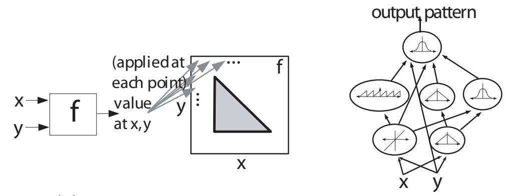
A significant obstacle to evolving complex structures is that heuristically determining the appropriate number of genes, i.e. the number of dimensions in the solution space, is difficult for challenging problems. For example, how many nodes and connections are necessary for an ANN that draws a picture of a bicycle? The answers to such questions cannot be based on empirical experience or analytic methods, because little is known about the solutions. To address this problem, instead of starting evolution in the space of the final solution, NEAT begins with a population of small, simple genomes and elaborates on them over generations by adding new genes. Each new gene expands the search space, adding a new dimension of variation that previously did not exist. That way, evolution begins searching in a small, easily-optimized space, and adds new dimensions as necessary. This approach is more likely to discover highly complex phenotypes than an approach that begins searching directly in the intractably large space of complete solutions. The process of *complexification*, i.e. incrementally adding new genes over generations, also occurs in nature, leading to increasing phenotypic complexity [22]. By starting minimally and gradually complexifying over the course of evolution, NEAT was able to solve several difficult control problems [30, 32].

Although it was originally introduced to evolve ANNs, NEAT is sufficiently general to evolve any variable-length genotype. Thus complexification is now a general tool for evolutionary computation. The next section introduces NEAT-based IEC for art, which is the basis for Picbreeder.

NEAT-based Genetic Art

Independent researchers have released several NEAT-based genetic art programs, beginning with Mattias Fagerlund's DelphiNEAT-based Genetic Art (DNGA) in 2003 [11, 12]. DNGA was followed by Holger Ferstl's SharpNEAT-based Genetic Art (SNGA) in 2006 [13]. While these applications evolve realistic-looking objects (see Stanley [29] for examples), they still can require a considerable number of generations to do so, and therefore are prone to user fatigue.

There also currently exists an online collaborative NEAT-based genetic art system called the *Living Image Project*, which was introduced on the world-wide-web in September, 2006 [1]. Living Image displays a population of color images generated by a NEAT-based genetic art program on a web page where users can *vote* for their favorite candidate. Each user is allowed to cast at most 25 votes in one day. After about 300 votes, the next generation replaces the current generation and the process begins again. The idea is to integrate the preferences of a broad user population and thereby



(a) CPPN to Image (b) Graph Representation

Figure 1. How CPPNs Work. The diagrams show how a CPPN is converted to an image and how CPPNs are internally represented. (a) The CPPN f takes arguments x and y, which are coordinates in a two-dimensional space. When all the coordinates are drawn with an intensity corresponding to the output of f, the result is an image. In this example, f produces a triangular image. (b) The CPPN network graph determines which functions connect to which. The connections are weighted such that the output of a function is multiplied by the weight of its outgoing connection. If multiple connections feed into the same function then the downstream function takes the sum of their weighted outputs. Note that the topology is unconstrained and can represent any relationships. The depicted functions exemplify the CPPN's ability to compose functions chosen from a canonical set. This structure allows the CPPN to represent a large and diverse space of patterns, biased by the set of canonical functions.

choose parent images more intelligently. Unfortunately, evolution is slow in such a system because many users must contribute to a single selection event, and at the time of this writing the project is only on its 24th generation of images after one year. Furthermore, it is difficult for the system to evolve toward a recognizable form because users have no way to coordinate their potentially conflicting choices.

The main idea in DNGA, SNGA, and Living Image is to enable NEAT to evolve a special kind of network that represents images. These evolved networks are called *Compositional Pattern Producing Networks* (CPPNs) because they produce patterns by composing functions [28, 29]. The next section explains how CPPNs represent arbitrary images.

Compositional Pattern Producing Networks (CPPNs)

A CPPN is a function of n Cartesian dimensions that outputs a pattern in space. For example, a two-input CPPN produces a two-dimensional image. For each (x, y) coordinate in that space, its level of expression is output by the CPPN that encodes the image. Figure 1a shows how a two-dimensional image is generated by a function of two parameters.

Internally, a CPPN is represented as a connected-graph (i.e. a network) of functions chosen from a canonical set. The structure of the graph represents how the functions are composed to process each coordinate. Figure 1b shows an example CPPN structure.

While CPPNs are similar to ANNs, they differ in their set of activation functions and how they are applied. While ANNs often contain only sigmoid functions (and sometime Gaussian functions), CPPNs can include both types of functions and many others. The choice of functions for the canonical set creates a *bias* toward specific types of patterns and regularities. Thus, the architect of a CPPN-based genetic art system can bias the types of patterns it generates by deciding the set of canonical functions to include.

Furthermore, unlike typical ANNs, CPPNs are applied across the entire space of possible inputs so that they can represent a complete image. Because they are compositions of functions, CPPNs in effect encode images at infinite resolution and can be sampled for a particular display at whatever resolution is optimal. The next section details several ways that CPPNs benefit image representation.

CPPN Image Representation

Representation is crucial in both evolutionary computation and artificial intelligence. A good representation can both efficiently encode complex information and also organize it effectively for search. Particularly in evolutionary computation, significant research in recent years has sought to clarify how complex structures can be encoded most efficiently for evolutionary search [7, 18, 31, 32]. It is now widely recognized that a good encoding allows information in the genotype to be *reused* in producing the phenotype. Encodings with this property are called *indirect encodings* [31].

It turns out that CPPNs are an indirect encoding with several powerful representational properties that make them particularly suited to encoding and searching for spatial patterns. In particular, they are designed to efficiently encode repetition, repetition with variation, symmetry, and elaboration.

Repetition is essential to many common forms from fish scales to window tilings and is naturally encoded in CPPNs that include periodic functions, such as sine and cosine. These functions produce a repetition of parts without the need to duplicate the information that encodes each part. *Repetition with variation* is another fundamental motif evident in e.g. fingers and leaves. Repetition with variation means that a pattern is repeated while varying each repeated element a small amount. It is accomplished in CPPNs by combining periodic functions with other functions (for instance, sines and Gaussians). *Symmetry*, which is fundamental to faces, animals and vehicles, allows the same information to encode both sides of an object. Symmetry is produced in CPPNs by symmetric functions, such as Gaussian. Finally, the ability to gracefully elaborate is essential to image evolution. Elaboration encourages increasing complexity by making each image a springboard to the next level of complexity. The NEAT algorithm adds functions and connections to CPPNs as they evolve, thereby elaborating the images they encode. A full review of the capabilities of CPPN image representation can be found in [29].

CPPNs in Picbreeder include cosine, sine, Gaussian, identity, and sigmoid functions to represent the images. These functions are chosen to capture regularities that appear frequently in nature (e.g. symmetry, repetition, repetition with variation) without intentional aesthetic bias.

THE PICBREEDER APPROACH

Picbreeder aims to address the primary challenges facing CIE:

1. **Empowering groups, regardless of skill, to collaboratively search a design space.** While groupware often coordinates users by sharing expertise [4, 5], few such projects empower users who may lack such expertise.
2. **Overcoming user fatigue.** In existing single-user IEC applications and CIE services, most users succumb to fatigue before generating significant products (e.g. as in [1, 12, 13, 16]).
3. **Proliferating content.** Most CIE systems do not encourage a proliferation of content, but instead concentrate the efforts of many users on single decisions [1].
4. **Collaborating without diluting individual contribution.** While existing CIE systems aim to produce more meaningful output by involving many users, they frequently average the contributions of many users to generate an image that is not necessarily pleasing to any [1, 25, 26].
5. **Encouraging participation.** CIE systems need to encourage participation through recognizing user achievements and through a flexible interface, which most do not [1, 25, 26].

This section describes the innovations that allow Picbreeder to overcome these difficulties.

Empowering Groups to Collaboratively Search Designs

Users who participate in online communities are a largely untapped resource for creativity. Specifically, even users without special expertise can perform tasks that are too difficult or subjective for a computer. For example, computers still struggle to visually parse scenes, understand speech, and, significantly for Picbreeder, appreciate art.

By allowing the user to direct what types of artifacts should be proliferated, Picbreeder empowers users, regardless of talent, to search a vast design space. Users simply select which generated images they find more compelling. Through the evolutionary process, the images are mutated to produce a new generation. Because of the CPPN-NEAT algorithm, the images will gradually become more complex and reflect the selections of the user. Because of these mechanisms, users can evolve complex digital content irrespective of their level of experience, unlike in most groupware systems.

The Picbreeder IEC client program that supports evolving images is shown in figure 2. The user breeds images by selecting one or more images from the 15 displayed. The user then presses *Spawn* to produce the next generation of images from the current selections. In case the user does not find any images that are worth selecting, the *Redo* button respawns the current generation. The *Back* button allows the user to return to a previous generation and restart the evolutionary progression from there. The user can also navigate back up to the current generation with the *Forward* button, much like a web browser. The *Save* button stores the evolutionary progression of an image in the user's account as an *unpublished image*, so that the user can continue it later. When the user decides that the image in the image panel is suitable, he or she selects that single image and presses *Publish*. A publishing interface then allows the user to share it with the community through the Picbreeder website. This simple design aims to appeal to the broad Internet community so that everyone's input can be harnessed to create evolved art.

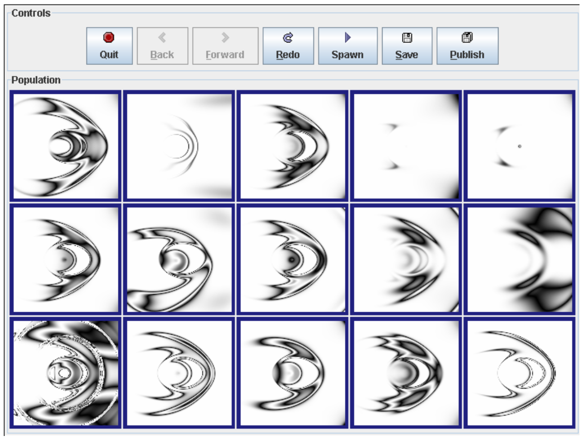


Figure 2. The Picbreeder client program. The interface is designed to be simple. It can spawn or re-spawn a new generation, move back and forth through the generations, and either publish the image or save it for later editing.

While some users may aim to evolve a particular image (e.g. a snake), others may simply explore by selecting images that happen to appear compelling. A novel image appearing in one generation may inspire the user to continue in a different direction than initially expected. Even if the user has no concrete goal and is not familiar with the IEC process, the images should become more compelling to the user through the mechanics of the evolutionary process. In this way, the client supports the user's creative exploration of a design space, allowing any user to continually guide the computer in directions of his or her own interest.

Thus, in contrast to expertise-based groupware (Sourceforge [4] and Wikipedia [5]), users without specific expertise can contribute images to Picbreeder through the simple IEC client. Picbreeder's easy access (requiring only a Java-enabled web-browser) encourages wide participation and the online format makes it possible for Picbreeder to scale to larger communities than could be supported by a physical installation.

Overcoming User Fatigue

It likely takes many generations to find interesting designs within a vast search space. Thus the chance is high that within the typical 10 to 20 generations of IEC [34], the user does not see anything significant, hence losing interest in exploring further. Even if the user retains interest throughout many more generations, searching over days can be too much, even if it is spread over several sessions. Without a means to accumulate many generations of evolution, it is difficult for images to evolve into anything significant. User fatigue is thus a fundamental problem in IEC [34] that single user IEC systems do not explicitly address [12, 13].

Picbreeder addresses user fatigue through a mechanism called *branching*. If the user finds an interesting image on the Picbreeder website, he or she can then choose to branch it, which means continue its evolution. As branches accumulate upon branches, it becomes easy for the complexity of an image to compound for hundreds of generations without fatiguing any single user. Because the user is likely to branch from images that interest him or her and because the IEC process steers images closer to the user's preferences, conflict over evolving of a single image is eliminated. The originating image and the results of its new branches are all stored separately on the Picbreeder website, allowing continued access to all of them.

A typical user session in Picbreeder begins with viewing published images (as seen in figure 3), which can be filtered by different criteria such as *highest rated* and *newest*. Users can choose to branch any image they see, thereby entering the IEC client program (figure 2), which loads a copy of the root image's CPPN. The user then continues the image's evolution through the IEC process, and publishes the branch when satisfied with its appearance.

Figure 3. Example image view from the Picbreeder website. Any of these images can be branched for further evolution.

When the user branches, Picbreeder follows the process illustrated in figure 4. The collection of genomes evolved throughout the generations of a single evolution session, along with their associated images, is a *series*. When a series is published, the last individual selected is its *representative*. While Picbreeder retains every image in each series for future analysis, users browsing the site only see representative images. When branched, a representative's genome spawns the first generation of the new branched series. This design accommodates branching while keeping individual series in the chain intact, thereby allowing long chains of content to grow while minimizing the work of each individual user.

Because NEAT lets images complexify throughout evolution, images evolved through a chain from many other users may have already gained significant complexity. Therefore, users can immediately begin with complex structures through branching. Figure 5 illustrates the benefit of branching from an already-complex image. In 5a the user required 40 generations to evolve from a random initial starting point to a compelling image. In contrast, it took only 14 generations in 5b to evolve the final image as a branch from a prior image because the new image borrows significant structure from its parent (i.e. a shoreline, a skyline, and the general layout of the scene). In this way, user fatigue is overcome.

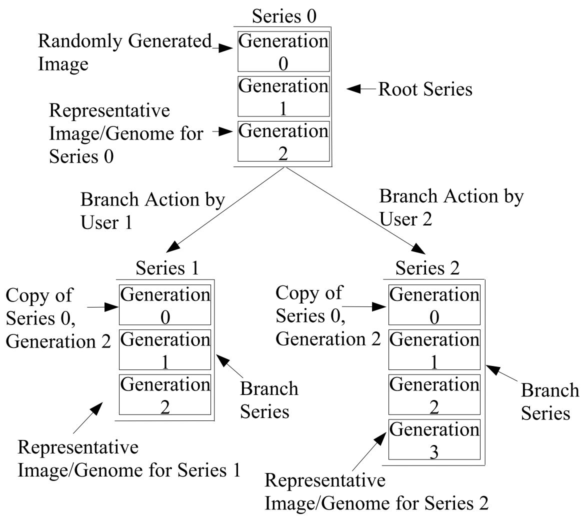
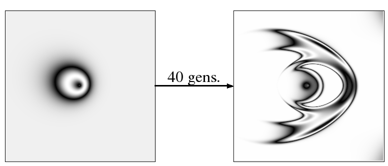
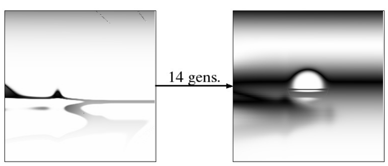


Figure 4. How branching works in Picbreeder. The server saves each set of generations in a *series*. When new series are branched from series 0, its representative genome spawns their initial generations.



(a) Evolved from random



(b) Branched

Figure 5. Overcoming user fatigue through branching. The example in (a) was evolved from scratch (i.e. from a random initial image) in 40 generations, beginning with the individual on the left and ending on the right. In (b), the user branched from the individual on the left to yield the individual on the right (another landscape) in only 14 generations by reusing some of the existing image's structure.

Content Proliferation

Some CIE systems [1, 33] combine user input to generate only a few products, which means that the amount of content generated per person is less than in single-user IEC. In contrast, branching in Picbreeder creates a new image with every branch. Importantly, an image may be branched multiple times, and all images are preserved indefinitely. Thus, instead of needing many users to generate few images [1], Picbreeder allows even a few users to generate many.

Furthermore, content is only displayed from images that users found *worthwhile to publish*. Thus, what results is a proliferation of *meaningful* content through ever-expanding branches. Interestingly, although all branches can be ultimately traced back to an initial series that started from nothing but a completely random population, a surprisingly diverse and meaningful set of distinct styles and themes nevertheless proliferates.

Collaboration Without Diluting Individual Contribution

Although systems like The Living Image Project [1] are seminal in promoting the idea of CIE, their focus is to merge the artistic sensibilities of several individuals, which can obfuscate the contributions of the individual user. It is possible for users to cancel out each others' contributions by pulling in opposite artistic directions. Furthermore, in most CIE systems [1,25,26,33], it is difficult to determine what contributions each user made to the evolution of a particular image. If users are not recognized for their contributions, they may lack motivation to participate in the system.

Picbreeder's branching also addresses this problem. Each lineage is tracked such that although a branched image is linked to its parent, the user can nevertheless continue evolution in any way desired. Each chain of branches is influenced by every contributor in the chain, yet each individual user takes it on a path chosen solely by that user.

Picbreeder provides a simple interface for browsing the images and users that have contributed to a lineage tree. Recall that each image displayed on the site is a representative of a series that began with its parent series' representative. The most proximate series in a image's lineage can be inspected in a *detailed view* panel (figure 6). The user can browse the lineage tree in either direction by clicking on the parent or children representatives.

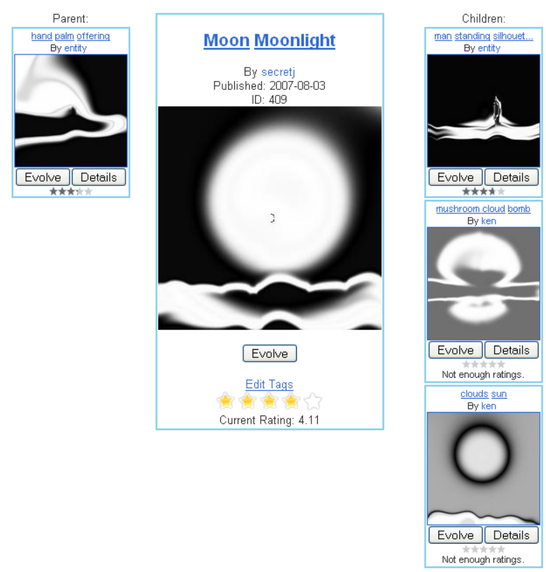


Figure 6. Navigating the tree of life. This view shows both the children and the parent of an image. Users can explore the tree by clicking on any image.

In this way, branching creates content that embodies the tastes and ideas of many different users, all while maintaining individual ownership and creative autonomy.

Encouraging Participation

Without an active user base, images cannot branch and complexify. A major challenge for any groupware is to assemble a critical mass of users to become productive [15]. An effective method to entice users is to highlight the most appealing content from which to branch. Picbreeder therefore also motivates participation through image ratings and user rankings, and enables users to find interesting images through tagging, browsing and searching mechanisms.

Users can rate interesting images and thereby credit other users for their creations. The average rating is shown under the image. The front page shows a group of *all time top rated* images, which sorts the images in descending order by average rating. Furthermore, the Picbreeder front page shows the *most branched*, which are images ordered by the number of times they have been branched to evolve new images. These views aim to maximize participation by immediately showing users images that have generated the most interest. They also encourage users to publish interesting images, so that they too can be featured on the front page. In addition, Picbreeder assesses overall rankings for each user, based on the number of unique users who have branched off one of their images. This ranking encourages users to contribute images that others would want to branch.

Picbreeder further helps users find interesting images from which to branch through *tagging*, *browsing*, and *searching*. Tags associated with an image during publishing let users find appealing images. Users can search for tags with a search-engine style interface. In addition, tags are automatically grouped into browseable categories and subcategories. The text boxes in which tags are entered during publishing suggest tags as the user types to reduce redundancy. The most frequent tags form the top-level categories. Images with these tags are queried for their other associated tags, which provide the next level of the categorical hierarchy. This approach creates browseable hierarchies without administrative overhead (figure 7). In these ways, Picbreeder makes participation easy and fun, which is essential for CIE to succeed.

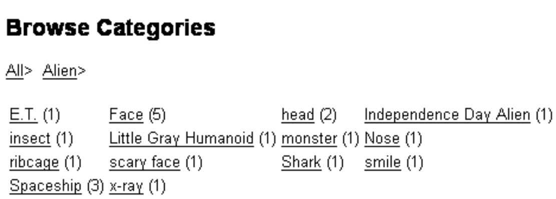


Figure 7. Example browseable categories from the Picbreeder website. The categories shown are subcategories of the "Alien" tag that were automatically discerned.

SYSTEM ARCHITECTURE

Figure 8 illustrates the architecture of Picbreeder, which is a database-driven website. The database stores meta-data about the images including lineage, authorship, ratings, and tags. The images and the CPPNs that generate them are stored on the standard file system.

The Java-based IEC client communicates to the server through web service calls. The client performs the IEC process, including image rendering, on the user's local machine, thereby reducing the load on the server. When the user saves his or her image, the generating CPPN is transmitted to the server in XML form, and saved in the user's account. The user

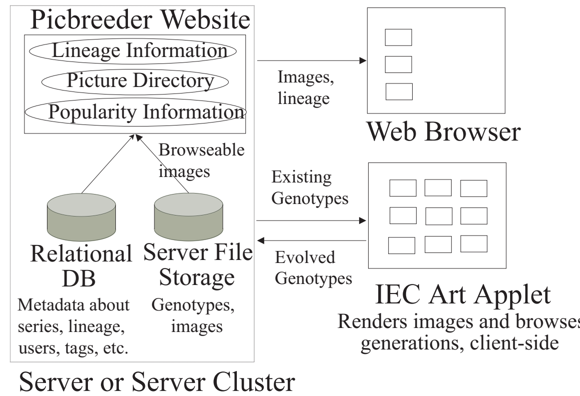


Figure 8. System architecture. The main site, including image views and lineage information, is viewed through a web browser. The user evolves images through a Java client that sends the results back to the server to be stored on the filesystem and database.

can then tag and publish the image. This architecture was designed to facilitate quick, efficient online interaction and dynamic storage.

THE PICBREEDER ONLINE EXPERIMENT

The Picbreeder website [3] opened to the public on August 1, 2007, and by January 1, 2008, had been visited over 12,000 times. There are 196 registered users, 102 of whom have published an image, averaging 19.11 images per active user (sd = 64.09, median = 3). The site is driven by collaboration: each user influences an average of 3.54 other users (sd = 6.99, median = 1) through branching and 86.8% of the images are branched from another image.

Branching does help overcome the limitations that user fatigue places on traditional IEC systems. While published images are evolved for an average of 26.23 generations (sd = 34.47, median = 15) during a single user session, each image has an average of 154.11 cumulative generations (sd = 134.28, median = 114) from an average of 5.95 ancestors (sd = 5.03, median = 5). Of the series in the top rated images, only 4.46% of them were evolved within the typical 20 generation limit for single users [34]; most of the highly appealing images in the system took many more cumulative generations to evolve, and therefore would not have resulted from a traditional IEC process. Branching also facilitated the proliferation of images, evidenced by over 2,100 published images on the site.

Ratings also helped encourage participation. While the average image in Picbreeder is branched by 0.31 users (sd = 0.84, median = 0), images in the top rated category (i.e. the 224 images with at least three rating votes and at least a rating of 3.0) were branched by an average of 1.47 users (sd = 2.01, median = 1).

In the Picbreeder community, there is a correlation between recognition and participation. The ten users who published the most images, while making up less than 5% of the user population, published two thirds of the top rated images. The ten users branched by the most other unique users (who, as a result, had the highest user rankings) created 61.03% of the top rated images.

Tagging is also widely used: The average published image is tagged 1.57 times (sd = 0.83, median = 1). The ten most frequent tags (“Face,” “Eye,” “Alien,” “Animal,” “Creature,” “Spaceship,” “Cool Pattern,” “Bird,” “Insect,” “Head”) include 372 (17.7%) of the published images, indicating that tagging does effectively categorize frequent categories.

Figure 9 features images that were collaboratively evolved in Picbreeder. The images vary from simple geometry (9h) to organic forms (9c). Each image in the figure is identified by a tag given to it by its creator. The variety of images evolved supports the choice of CPPNs as an appropriate image representation, and shows that Picbreeder encourages a proliferation of content.

Picbreeder’s CPPN representation and IEC client allow users to evolve a succession of elaborations on the same theme, as shown in the sequence of faces (each one the representative of one series in the chain) in Figure 10. The images gradually become more elaborate and recognizable as a face as evolution progresses. This sequence demonstrates elaboration through complexification, acquiring new features while preserving the properties of previous generations.

Figure 11 shows a tree of life, collaboratively evolved by 13 different users. Each image represents a branch of its parent series (depicted by its representative). The root image was evolved from random starting images generated by the client. A variety of forms proliferate in breadth and depth. The average number of generations taken to evolve each series (16.35) is consistent with Takagi’s estimate of 10 to 20 generations to expect from an individual IEC user [34]; however, through its entire chain of preceding branches back to the root, the deepest series accumulated an order of magnitude more generations (195), demonstrating that Picbreeder does overcome the problem of user fatigue: Users still spend the expected number of generations evolving each image, but the total number of generations is much higher. Furthermore, while the 13 users were able to collaborate to generate content, they still maintain ownership of their contributions.

DISCUSSION

Picbreeder has already begun to accumulate an online collection of cataloged digital content unlike any CIE system to date. Through its ability to branch and through the complexification of NEAT, Picbreeder has generated several complex images, many of which *resemble real objects* (figure 9). This fact is significant because the search space of images is astronomically large and all evolved objects descend ultimately from *random initial images*. While many genetic art programs generate images that are appealing, most do not evolve recognizable objects. In effect meaningful images are like needles in a haystack. Thus, collaboration through branching has proven effective at harnessing the power of multiple users to search an otherwise prohibitively large space.

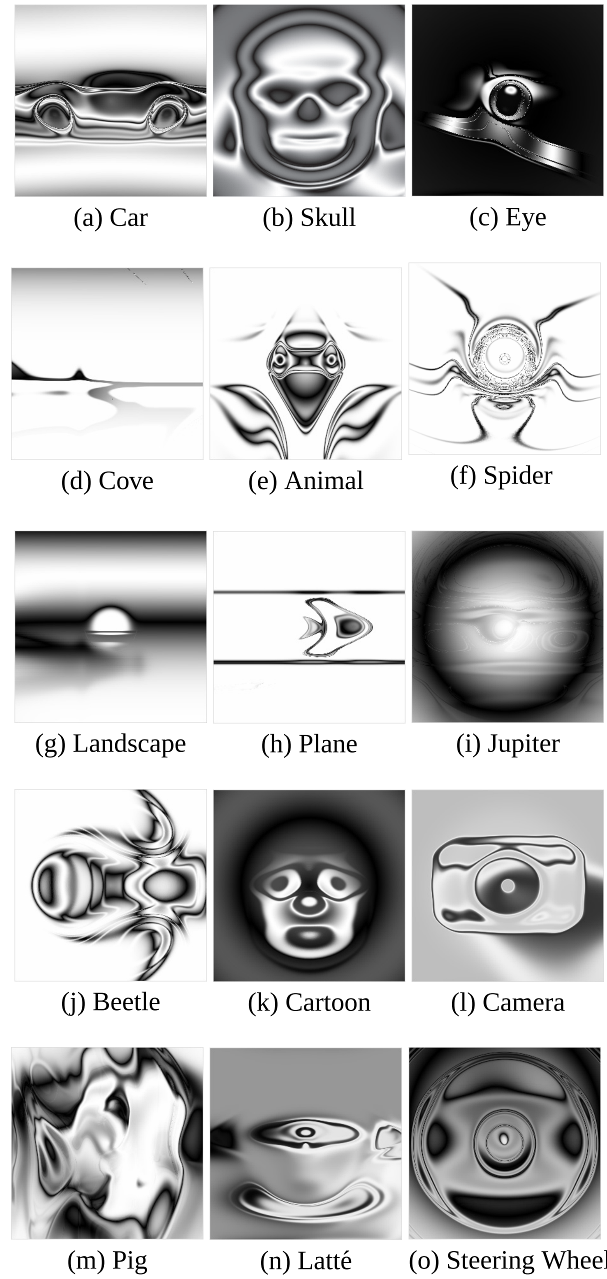


Figure 9. Examples of collaboratively evolved images. These images, all entirely evolved by Picbreeder users, exhibit a variety of themes from mechanical (a) to organic (f).

Systems like Picbreeder naturally raise the question of whether such technologies purport to replace skilled artists or redefine the meaning of *art*. Yet such controversy is unwarranted because CIE does not aim to replace skilled artists and designers, but instead to empower users to create and design artifacts within otherwise inaccessible domains (i.e. to find the needle in the haystack). While the utility of the results is an important contribution, the unique social and creative experience is equally significant.

CONCLUSIONS AND FUTURE WORK

This paper introduced Picbreeder, a novel approach to CIE. To address the main challenges in CIE, several new techniques were developed for Picbreeder. The Picbreeder site

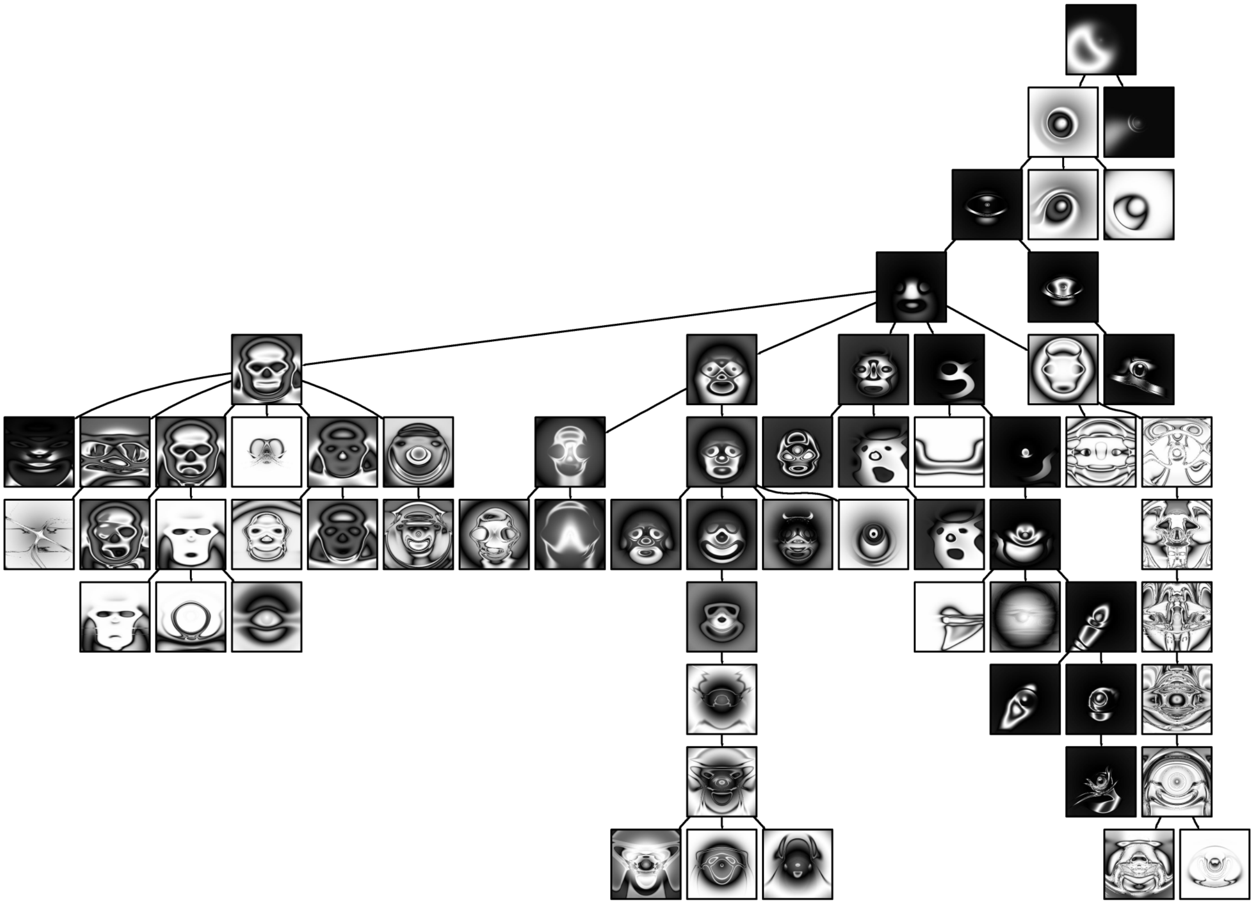


Figure 11. Tree of life. This actual phylogenetic tree, whose root image is at the top, was created by 13 users branching and evolving.

has already actively engaged many users in a collaborative process of design exploration despite the online experiment only being active for less than five months. As images are branched further, the NEAT algorithm will ensure that their complexity increases, continually creating new directions for users to explore. We look forward to seeing what images will be evolved after thousands of generations by hundreds of users. We also hope that the Picbreeder design inspires others to develop similar systems that proliferate digital content through the combined effort of an online community.

Picbreeder will be augmented in several ways in the future. First, color will be added, which requires specially-designed CPPNs. Second, Picbreeder will also allow branching by mating different images. Branching from more than one image would allow combining popular concepts.

The system architecture that supports Picbreeder is not limited to evolving images. Any artifact that can be evolved through IEC can also be evolved collaboratively through a system like Picbreeder, including music, three-dimensional images, synthesized voices, and possibly even intelligent agents.

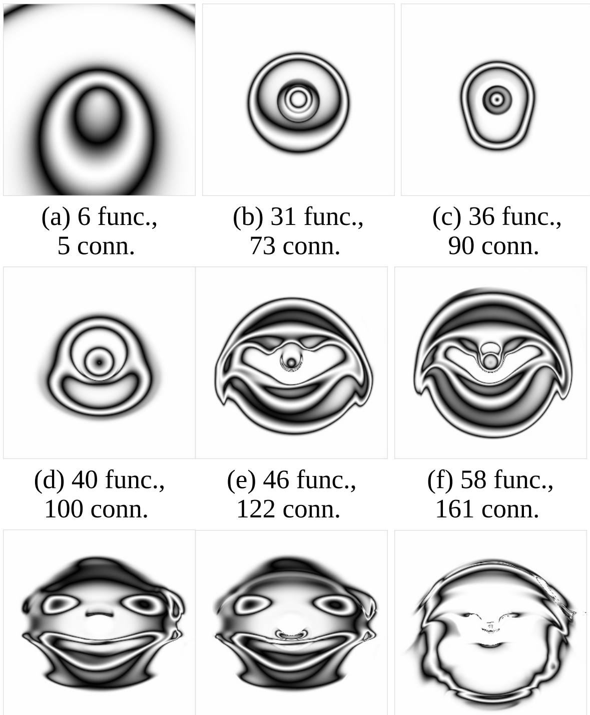
Perhaps Picbreeder-like systems will become common in the future. For example, a car company might deploy a Picbreeder-like system to evolve new car designs, and offer to manufacture the most popular ones. More likely in the near-term, as the popularity of personal rapid prototyping machines increases, Picbreeder-like systems can evolve three-dimensional objects that are later downloaded and rendered by a machine. The popularity of innovations like mobile computing and blog widgets will also provide new outlets for Picbreeder-like systems in the future.

ACKNOWLEDGMENTS

Jimmy Secretan was supported by a National Science Foundation Graduate Research Fellowship. Special thanks to Michael Whiteley and Ben Vanik for their input.

REFERENCES

1. "Living image project," http://www.w-shadow.com/li/, 2006–2007.
2. "Apophysis," http://www.apophysis.org, 2007.
3. "Picbreeder," http://www.picbreeder.org/, 2007.
4. "Sourceforge," http://sourceforge.net/, 2007.
5. "Wikipedia," http://www.wikipedia.org/, 2007.
6. J. Balogh, G. Dubbin, M. Do, and K. O. Stanley, "Dance evolution," in *Proceedings of the Twenty-Second National Conference on Artificial Intelligence (AAAI-07) AI Video Competition.* Menlo Park, CA: AAAI Press, 2007.
7. P. J. Bentley and S. Kumar, "The ways to grow designs: A comparison of embryogenies for an evolutionary design



(a) 6 func., 5 conn.
(b) 31 func., 73 conn.
(c) 36 func., 90 conn.
(d) 40 func., 100 conn.
(e) 46 func., 122 conn.
(f) 58 func., 161 conn.
(g) 66 func., 192 conn.
(h) 72 func., 215 conn.
(i) 106 func., 310 conn.

Figure 10. Sequence of Descendant Face Patterns. The chronological sequence (a)-(i) displays successive progeny evolved in the Picbreeder client. The number of functions and connections in the generating CPPN is shown below each pattern. The sequence exhibits a continual elaboration of regular form.

problem," in *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-1999)*, 1999, pp. 35–43.

8. J. Biles, "Genjam: A genetic algorithm for generating jazz solos," in *International Computer Music Conference*, 1994, pp. 131–137.
9. V. C. L. Collaborative User Experience, "Many eyes," http://services.alphaworks.ibm.com/manyeyes/app, 2007.
10. R. Dawkins, *The Blind Watchmaker.* Essex, U.K.: Longman, 1986.
11. M. Fagerlund, "DelphiNEAT homepage," http://www.cambrianlabs.com/mattias/DelphiNEAT/, 2003–2006.
12. ——, "DelphiNEAT-based genetic art homepage," http://www.cambrianlabs.com/mattias/GeneticArt/, 2005.
13. H. Ferstl, "SharpNEAT-based genetic art homepage," http://www.cs.ucf.edu/~kstanley/GenArt.zip, 2006.
14. J. Grudin, "Computer-supported cooperative work: Its history and participation," *Computer*, vol. 27, no. 4, pp. 19–26, 1994.
15. ——, "Groupware and social dynamics: Eight challenges for developers," *Communications of the ACM*, vol. 37, no. 1, pp. 93–104, 1994.
16. S. Hammond, "Imagebreeder," http://www.imagebreeder.com/, 2007.
17. E. Hastings, R. Guha, and K. O. Stanley, "Neat particles: Design, representation, and animation of particle system effects," in *Proceedings of the IEEE Symposium on Computational Intelligence and Games (CIG-07)*. Piscataway, NJ: IEEE Press, 2007.
18. G. S. Hornby and J. B. Pollack, "Creating high-level components with a generative representation for body-brain evolution," *Artificial Life*, vol. 8, no. 3, 2002.
19. S. Jorda, "Faust music on line: An approach to real-time collective composition," *Leonardo Music Journal*, vol. 9, no. 5, pp. 5–12, 1999.
20. K. Krause, "Kai's power tools 3," [Software], 1996.
21. W. B. Langdon, "Pfeiffer – A distributed open-ended evolutionary system," in *AISB'05: Proceedings of the Joint Symposium on Socially Inspired Computing (METAS 2005)*, B. Edmonds, N. Gilbert, S. Gustafson, D. Hales, and N. Krasnogor, Eds., 2005, pp. 7–13.
22. A. P. Martin, "Increasing genomic complexity by gene duplication and the origin of vertebrates," *The American Naturalist*, vol. 154, no. 2, pp. 111–128, 1999.
23. S. Rooke, *Eons of Genetically Evolved Algorithmic Images*. Morgan Kaufmann, 2002, ch. 13, pp. 339–365.
24. K. Sims, "Artificial evolution for computer graphics," in *Proceedings of the 18th Annual Conference on Computer Graphics and Interactive Techniques (SIGGRAPH '91)*. New York, NY: ACM Press, 1991, pp. 319–328.
25. ——, "Genetic images interactive exhibit," *Centre Georges Pompidou Revue Virtuelle Notebook*, vol. 5, 1993.
26. ——, "Galapagos interactive exhibit, Decordova Museum, Lincoln, MA," http://www.genarts.com/karl/geneticimages, Lincoln, MA, 1997.
27. J. R. Smith, "Designing biomorphs with an interactive genetic algorithm," in *Proceedings of the 4th International Conference on Genetic Algorithms (ICGA-91)*, R. K. Belew and L. B. Booker, Eds. San Mateo, CA: Morgan Kaufmann, July 1991, pp. 535–538.
28. K. O. Stanley, "Exploiting regularity without development," in *Proceedings of the AAAI Fall Symposium on Developmental Systems*. Menlo Park, CA: AAAI Press, 2006.
29. ——, "Compositional pattern producing networks: A novel abstraction of development," *Genetic Programming and Evolvable Machines Special Issue on Developmental Systems*, vol. 8, no. 2, pp. 131–162, 2007.
30. K. O. Stanley and R. Miikkulainen, "Evolving neural networks through augmenting topologies," *Evolutionary Computation*, vol. 10, pp. 99–127, 2002.
31. ——, "A taxonomy for artificial embryogeny," *Artificial Life*, vol. 9, no. 2, pp. 93–130, 2003.
32. ——, "Competitive coevolution through evolutionary complexification," vol. 21, pp. 63–100, 2004.
33. S. R. Szumlanski, A. S. Wu, and C. E. Hughes, "Conflict resolution and a framework for collaborative interactive evolution," in *Proc. of the 21st National Conf. on Artificial Intelligence (AAAI)*. AAAI Press, 2006, pp. 512–517.
34. H. Takagi, "Interactive evolutionary computation: Fusion of the capacities of EC optimization and human evaluation," *Proceedings of the IEEE*, vol. 89, no. 9, pp. 1275–1296, 2001.
35. S. Todd and W. Latham, *Evolutionary Art and Computers*. London: Academic Press, 1992.