

Gaudii: An Automated Graphic Design Expert System

C. Glez-Morcillo V. J. Martín D. Vallejo J. J. Castro-Schez J. Albusac

Escuela Superior de Informatica
University of Castilla-La Mancha, Spain

Abstract

Graphic Design is the process of creating graphics to meet specific commercial needs based on knowledge of layout principles and esthetic concepts. This is usually an iterative *trial and error* process which requires a lot of time even for expert designers. This expert knowledge can be modelled, represented and used by a computer to perform design activities. This paper describes a novel approach named *Gaudii* (standing for “*Intelligent Automated Graphic Design Generator*”) which utilizes principles and techniques known from the fields of Evolutionary Computation and Fuzzy Logic to automatically obtain design elements. Experimental results that demonstrate the potential of the proposed approach are presented in the area of poster design.

Introduction

Graphic designers plan, analyze and create visual solutions to communications problems finding the most effective way to show the information by means of colors, types, illustrations, photos and various printing and layout techniques. For this reason, the printing and electronic media industry is demanding skilled people. According to the Bureau of Labor Statistics (U.S. Department of Labor), the employment in graphic design is expected to grow up to 13 percent in the next 10 years (Statistics 2010).

In contrast to other disciplines in visual arts, Graphic Design simply looks for clarifying a message and showing it aesthetically pleasing by means of a creative approach (Samara 2007). In the last few years, the discipline of Computational Creativity has come of age (Colton and Lopez 2009). In some specific creative areas (such as Graphic Design), there is also a strong dependence of experience and expert knowledge to create design items. Nevertheless, there are some basic rules and principles followed by designers to obtain a better design (Samara 2007), (Williams 2008). These general rules are vague, incomplete and very dependent on the particular design situation. In addition, these principles are strongly interconnected.

Within this context, this paper introduces a novel, fully automatic approach named *Gaudii*¹ to create graphic designs. This is achieved by means of Genetic Productions

and Local Optimization methods which search for valid solutions by using Expert Knowledge represented by means of fuzzy rules. Thus, the approach tries to maximize the design quality making use of cost functions and following some design principles and constraints.

The remainder of the paper is structured as follows. The next section overviews the state of the art and the current methods for automatic computer graphic design. Thereby, the focus is on the most promising issues related to the main research line in this field, named Evolutionary Design. *Architectural Overview* section describes the key points of *Gaudii*. In the next section the empirical results are shown, which have been obtained by addressing a typical design problem. The last section offers a careful discussion and concluding remarks.

Related Work

The design process is usually divided into three main stages: i) the preliminary conceptual phase, ii) the detailed design step and, finally, iii) the evaluation and iterative redesign (Samara 2007). This iterative nature of graphic design fits well with classic local search optimization algorithms.

The most widely used methods for automatic graphic design are in the field of Evolutionary Art (Romero and Machado 2008). These methods try to model the human creativity as a Darwinian process, generating new design ideas through Blind Variation and Selective Retention. In these methods, the current trend consists in randomly altering in different ways, selecting the best variants and repeating the process until a good thought is obtained (Simonton 1999).

The design problem to be solved by a computer must be numerically represented. In Evolutionary Design, the representation determines the method. The existing approaches can be roughly divided into two groups of methods: Genetic Algorithms and Genetic Programming (Lewis 2008).

If **Genetic Algorithms** (GA) (Holland 1992) are used, each solution of the *population* is represented by a fixed length string of numbers (called *genotype*). The values of each component of the string (*gene*) determine the final appearance of the design (called *phenotype*). In this way, a *population* of design is created by setting different values for the *genes* of the design. Normally, the user of the system judges the population by selecting the most interesting creations and the system creates a new *generation* of designs, using normally a *crossover* operator. Within this context, previous works in this area have been developed.

Copyright © 2010, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

¹Funded by JCCM and MCYT Projects (TIN2009-14538-C02-02, PIA12009-40, PII2109-0052-3440 and PII1C09-0137-6488).

The system conceived by (Geigel and Loui 2003) uses a GA to evolve album page layouts by using a 4-tuple for the genotype (coordinates, scale and rotation). The design tool “*evoDesign*”, developed by (Anderson et al. 2008), employs GA to evolve the tiles used for fabrics and wallpapers. In this work, the authors focused on creating intuitive genotype-phenotype mappings, allowing the user to lock good elements on the design. In (Quiroz et al. 2009), the individuals of the population are evaluated on a set of objective heuristics for document design (page layout) and subjectively by the user of the system.

The other family of methods belongs to the **Genetic Programming** (GP) (Koza 1992) field. Unlike the previous methods, the representation is based on a hierarchical graph, usually representing a mathematical expression. In this expression, the operators are located on internal nodes and constants and variables at leaves. The *phenotype* is obtained applying the function of the graph to every pixel in the image. The results obtained from GP methods are usually organic (see (Romero and Machado 2008)) and fractal-landed and inappropriate for many graphic design applications.

(Bentley and Corne 2002) and (Romero and Machado 2008) have excellent surveys on Creative Evolutionary Systems, including expression-based imaginary, fractals, image processing and other related techniques.

Gaudii significantly differs from the aforementioned works in its general and extensible approach, and the *black-box* approach from the point of view of the final user. No previous knowledge of design or evolutionary computing is needed. Most of the evolutionary design systems require a qualitative background of knowledge of these techniques to be used. There are some hard concepts for inexperienced users, such as the set up of related probabilities (crossover, mutation, etc). Also the existing solutions tend to be very specific for the design problem. *Gaudii* identifies multiple independent modules which can be suitable for a wide range of graphic design scenarios. Combining key features of Evolutionary Computation and integrating an Expert System with Local Optimization methods configure *Gaudii* as a novel approach to generate Graphical Design elements.

Gaudii Architectural Overview

The most widely used approach in the automatic generation of designs is based on evolutionary techniques. As previously discussed, this kind of method shows some disadvantages that can be overcome thanks to a hybrid approach. In this way, *Gaudii* can be understood as a hybrid system composed of four key stages: Analysis and Preprocessing, Genetic Production, Expert Knowledge Kernel, Composition and a general Knowledge Acquisition Module (see Figure 1). This module is used to acquire knowledge from users when needed. This module requires the interaction with the user in few stages because of the autonomous nature of the *Gaudii* approach. Some of this functionality will be explained in the context of the related stages.

The key stages are designed taking into account future extensions and provide loose coupling by means of the definition of intermediate data that allows to make changes and integrate new functionality in each module without affecting the rest. Thus, it is possible to independently run and debug

each developed module. Next, the functionality of the devised stages, including the related modules that form them, will be described.

Analysis and Preprocessing

This stage takes the data provided by the user as input; texts and images which can be semantically grouped, as well as keywords (optional for the automatic search of related images on the Internet) and preferences. The definition of semantic groups allows to arrange the related items and give a considerable importance to each item of the group. Each group is formed by one or more elements of text or images within five categories: title, subtitle, important, regular, and notes (see the example in the Experimental Results section).

The Knowledge Acquisition Module (KAM) supports the acquisition of preferences by means of five initial questions:

1. **(Q1)** Do you prefer portrait or landscape designs?
(Portrait / Landscape / I don't mind)
2. **(Q2)** What kind of typography do you like?
(Formal fonts / Fancy fonts / I don't mind)
3. **(Q3)** Do you prefer dark or light designs?
(Dark / Light / I don't mind)
4. **(Q4)** Do you want a grayscale or color design?
(Grayscale / Color / I don't mind)
5. **(Q5)** Where does *Gaudii* get the key tones?
(From the image / I want to force a schema / I don't mind)

The output of this stage consists of a set of Raw Design Elements and the user's preferences (obtained by the KAM). These Raw Design Elements are the set of images chosen for the design with their definition of the Area of Interest (explained later) and the semantic groups (text and related images), and the user's preferences.

Obtaining Visual Elements. This module is responsible for obtaining the images of the design. These design items can be directly provided by the user or can be obtained from the list of keywords supplied by the user. The KAM is used to ask the user for selecting the key image of the design. This image is sent to the next module.

Visual Interest Analysis. This module defines the most interesting visual area of the key image of the design. This module uses a set of submodules specialized in the image analysis and an *Interest Composition Function* (ICF) that combines the output of such submodules, obtaining the final Area of Interest. The current version of *Gaudii* implements three analysis submodules:

- **Face Detection**, which detects human faces using a Haar Classifier. In the case of multiple detection, the submodule returns the best one (defined as the bigger and well-located, following the compositional *rule of thirds* (Samara 2007) used in photography).
- **Object Detection**: which searches for isolated objects in the image, using the previous explained criteria in the case of multiple detection.
- **Manual**: the KAM module provides an Ajax web interface for the user to manually obtain the area.

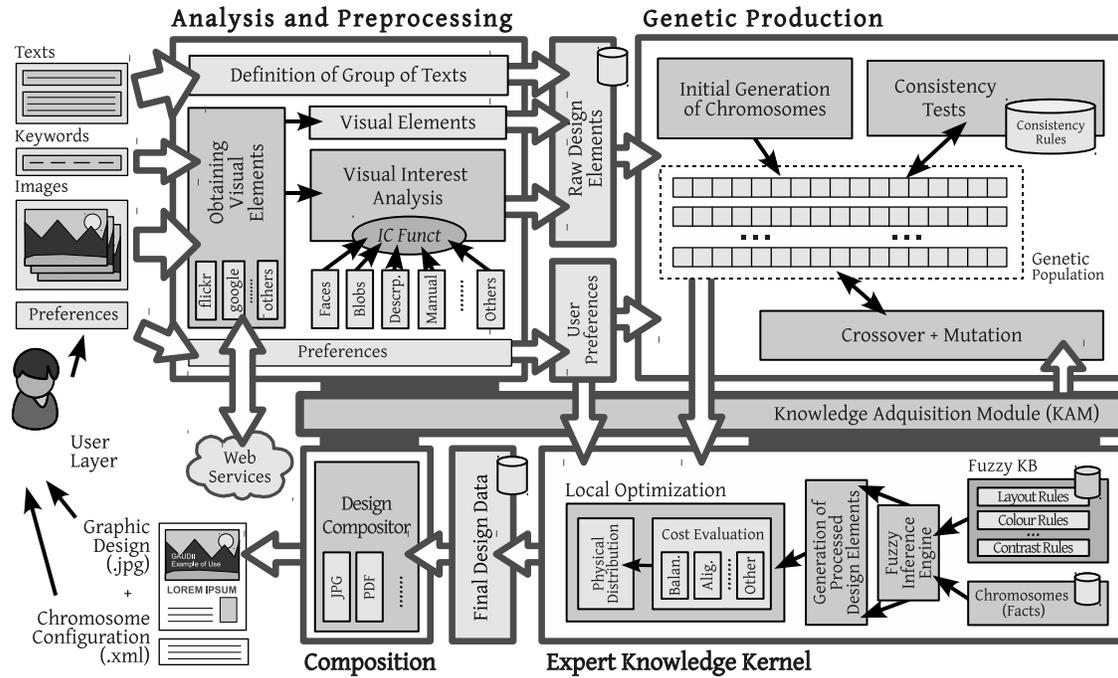


Figure 1: Representation of the main modules of the *Gaudii* architecture.

Genetic Production

In this stage, a population of genotypes (denoted from now as chromosomes) is created. Each member of the population defines the general attributes of a graphic design that will be concreted in the next stage. The specific attributes of each Design Element (such as size, color or position) will be decided in the next stage.

Each gene of the chromosome defines a general characteristic of the design, taking integer values that range from 0 and 9. In *Gaudii*, the chromosome has a fixed size of 24 genes grouped by purpose in 4 categories: Document, Contrast, Repetition and Composition (Table 1 shows some of them). These genes have a correspondence with some of the input variables used in the Knowledge Base of the next stage (Expert Knowledge Kernel).

G	Group	Description
1	Document	Aspect Ratio (0 = Square, 9 = Panoramic)
2	Document	Portrait or Landscape
3	Document	Main Font Style (Graphic, Sheriff...)
6	Document	Use main image as background of design
7	Contrast	Contrast level between size of fonts
8	Contrast	Contrast level between colors of fonts
10	Contrast	Use fancy text decoration for contrast
13	Contrast	Use Capitals in text
14	Repetition	Use similar size of fonts
15	Repetition	Use similar colors in fonts
16	Repetition	Use same shape in fonts (as Main image)
18	Repetition	Repeat the main font as secondary
23	Composit.	Crop key image around the Area of Interest
24	Composit.	Use text or image borders

Table 1: Short description of some genes.

The Genetic Production stage is a controlled Genetic Algorithm. Some chromosomes are not allowed because of the inter-dependencies of some genes. These dependencies are checked in the Consistency Test module. This stage takes as inputs the outputs of the previous stage (Raw Design Elements and the User's Preferences) and produces as output a Genetic Population (set of chromosomes) that will be used in the next stage.

Initial Generation. This module, as in most evolutionary design-based applications, randomly generates the population by assigning values to the free genes of the chromosome (between 0 and 9 in *Gaudii*). Some of the genes can be fixed by the KAM (because of the User's Preferences or during the generation) and cannot be changed in the whole generation process. This random initialization guarantees the diversity of the initial population and the phenotypes (final designs).

Consistency Tests. This module makes use of a base of consistency rules to detect and resolve inconsistencies between genes that deal with a common aspect. For instance, Table 1 shows that the genes 8 and 15 can be contradictory if they take opposed values (e.g. a high contrast level between font colors and a high repetition level of such colors). These tests are used when a chromosome changes (in the initialization process or crossover and mutation).

Crossover and Mutation. By means of the KAM, the system allows the user to choose a set of interesting final design items and combine them to create a new generation of individuals. This can be done only after the four stages have finished (and a set of final designs are shown to the user)².

²In the current implementation of *Gaudii*, the crossover is done with two or more parents using a drag&drop Ajax web interface.

The size of the population is small to avoid the user's fatigue (Takagi 2001) (in *Gaudii* is fixed to 12). The probabilities of crossover and mutation are dynamically adjusted by the KAM module with no user's interaction. There is no fitness function; the chromosomes only represent general attributes. The quality of each solution depends on the concrete properties (position, size, color...) of each element, which will be optimized in the next stage.

Expert Knowledge Kernel

In this stage, the Raw Design Elements and the chromosomes of the population of the previous stage are processed in order to get the Final Design Data (items formatted according to concrete attributes). The chromosome associated to each design is evaluated by making use of the expert knowledge existing in the Knowledge Base (KB). This KB is formed by a set of fuzzy rules that models general graphic design principles. Each potential design solution is optimized using a Local Optimization module that decides the final Physical distribution of the design elements.

The output of this stage is the specific information for each design element. In other words, the Raw Design Elements are converted into Final Design Data that contains the information needed by the Composition stage to produce the final image. The concrete individual properties of each element are obtained in the Generation of Processed Design Elements Module and the physical distribution of each element in the document is obtained in the Local Optimization Module.

Fuzzy Knowledge Base. A set of fuzzy rules (Zadeh 1975) is used to model the expert knowledge and obtain particular values of some of the properties of the design items that correctly behave according to basic design principles. The sets of fuzzy rules are widely used due to the descriptive power and extensibility when modelling expert knowledge.

The current version of *Gaudii* uses a set of 128 rules, generated by a human expert, grouped into the categories of Contrast, Repetition, Layout and Composition (Williams 2008). These rules are loaded into the system through and XML file³. Next, an example of two rules of contrast used by the system is shown:

$$R_{18} : \text{If } CS \text{ is } \{S, M\} \wedge RS \text{ is } S \rightarrow HO \text{ is } B$$

$$R_{19} : \text{If } CS \text{ is } B \wedge RS \text{ is } \{S, B\} \rightarrow HO \text{ is } VB$$

The variables (antecedents and consequents) used in the previous rule are as follows:

- **Contrast of Size** (Antecedent) $[CS]$, defined over the set of values $\{S, M, B\}$ ⁴. This parameter corresponds to the gene 7.
- **Repetition of Size** (Antecedent) $[RS]$, defined over the set of values $\{S, B\}$. This parameter corresponds to the gene 14.
- **Size of Title** (Consequent) $[HO]$, defined over $\{B, VB\}$. It defines the size of the text of class "Title".

³The full list of rules and linguistic variables can be reached at <http://www.esi.uclm.es/www/cglez/gaudii/>

⁴The notation used for the linguistic variables is typical in some works with Fuzzy Sets. This is the correspondence of the linguistic variables: S is Small, M is Medium, B is Big and, finally, VB is Very Big.

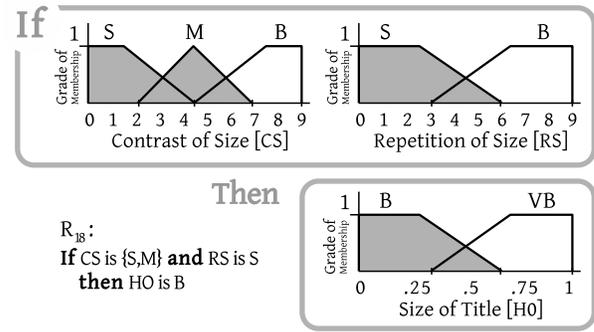


Figure 2: Definition of the fuzzy sets in a rule.

Each variable has its own fuzzy sets. Triangular and trapezoidal functions were used for each linguistic label associated to the linguistic values, as shown in Figure 2. To obtain the output of each rule, Mamdani's Inference System with minimum T-norm and maximum T-conorm is used.

Generation of Processed Elements. This module uses the output (consequents) of the fuzzy rules and obtains concrete attributes for the Raw Design Elements. This information will be sent to the Local Optimization module to obtain the final physical distribution.

These concrete attributes for the design elements are obtained using four submodules of functions:

- **Document:** These functions generate the final size of the document (using the document ratio), the size of margins and the final size (in mm) of design elements.
- **Color:** The final color scheme is generated. These functions also check if the text is readable over the background.
- **Typography:** Using, at the most, two families of fonts (Samara 2007) these functions select the text attributes.
- **Pre-composition:** These functions prepare the results with some extra information required by the Local Optimization Module (such as free white space).

Local Optimization. This module obtains the final physical distribution of the design elements using a Simulated Annealing algorithm. In local search methods, the definition of a cost evaluation function is needed. In *Gaudii*, this evaluation is done by composing two cost functions:

- **Balanced Cost:** Each design element has a Visual Weight determined by its type (title, subtitle, main image, etc). The Balanced cost of the composition is obtained using the mean point of each design element multiplied by its Visual Weight. A polygon is constructed with a weighted vertex (see Figure 3.h). Then, the centroid is calculated. If the centroid is close to the geometric center of the composition (represented in Figure 3.h through a white circle), the design is well-balanced.
- **Alignment Cost:** This function forces to correctly align the design elements to get a visual order (Samara 2007). This cost is computed by means of the number of design elements aligned to the alignment lines (see Figure 3.g).



Figure 3: Case Study: (a-f) Selected examples obtained with the same input data. (g) Alignment lines for (f), defined automatically and used by the alignment cost function. (h) Visual balance by minimizing the value of the balanced cost function.

By default, the margins, the borders of the area of interest, are defined. Some specific aligned lines can be also defined by this function, such as the eyes level in the case of using a face as an area of interest.

Since the implemented modules are independent each other, it possible, for instance, to use Genetic Algorithms to align the design elements. Within this context, the cost evaluation function when using the Simulated Annealing algorithm can be easily replaced by a *fitness* function of GA.

The neighbor solutions are obtained by changing the position of the design elements. When the temperature of the Simulated Annealing is high, large translations are allowed. The more the temperature decreases, the smaller the modification of the position is.

Composition

In this last stage, the formatted design items are processed to get the final image. This stage is independent of the previous stages of generation and allows for the integration of multiple engines of image synthesis, raster or vectorial. The KAM provides to the user the configuration of the chromosome (a XML file) in order to reuse it in future designs. For users that are reluctant to blackbox approaches such as *Gaudii*, the composition stage allows to easily include a module to generate editable designs (e.g. by using as output the ODG standard file format).

Experimental Results

These results were obtained with the implementation of *Gaudii*, which is available free for download at <http://code.google.com/p/gaudii/> under GPL3 Free Software License, on a computer with the following specifications: Intel Dualcore 2Ghz, 3GB RAM running Debian, Mongrel Server 1.1, Ruby 1.9, Ruby on Rails 2.3 and OpenCV 2.0.

In the following example, three groups of design items were defined:

- **Keywords**
Graphic Design, Paint, Atlanta, Artificial Intelligence
- **User's Preferences**
Q1 (Portrait/Landscape): *Don't mind*, Q2 (Formal/Fancy Fonts): *Formal*, Q3 (Light/Dark Design): *Light*, Q4 (Grayscale/Color): *Grayscale*, Q5 (Key tones?): *From the image*
- **Group 1**
Text - **Title** - *Gaudii*
Text - **Subtitle** - *Automated Graphic Design Expert System*
- **Group 2**
Image - *IAAI Logo (.PNG Transparent)*
Text - **Emph** - *IAAI 2010*
Text - **Normal** - *22nd IAAI Conference on AI*
Text - **Notes** - *Innovative Applications of Artificial Intelligence /n Atlanta, Georgia, USA, July 11-15, 2010*
- **Group 3**
Text - **Emph** - *Paper Contents*
Text - **Normal** - *Introduction /n Related Work /n Architectural Overview /n Experimental Results /n Discussion and Conclusion*

The user only entered the previous data in the system. Some of the images provided by Flickr⁵ are shown to the user to choose the main one. The Visual Interest Analysis module, using an OpenCV Haar Classifier, detects a human face and establishes the most interesting visual area.

Some selected examples obtained with this data are shown in Figure 3 (a-f). The Genetic Production assures the variety of the obtained elements, configuring some characteristics of the final item which could be inherited by their offsprings. For example, the chromosome (f) C_f is obtained as offspring of the chromosomes of (c) C_c and (e) C_e of Figure 3:

$$C_c = \{731919 : 785612 : 823516 : 570002\}$$

$$C_e = \{680913 : 584421 : 252606 : 672523\}$$

$$C_f = \{731919 : 785421 : 222609 : 672523\}$$

⁵Using the Ruby plugin flickr.rb, searching (using the keywords provided by the user) the images with Creative Commons license, ordered by Interestingness. Photo in Figure 3 by Carlo Nicora.

In this example, C_f inherits (random crossover) from C_c the first 9 genes which determines for example the landscape distribution of the creation (the value of the second gene is 3, close to 0, which means a high probability of landscape distribution), and the high contrast level between size of fonts (gene 7). The Consistency Test module corrects the value of gene 14 of C_f because, as previously was explained, there is a conflict with gene 7. There is also a mutation in the value of gene 18 (value 9) which forces to use the same font in the primary and secondary texts (using only one typography).

In Figure 3.g the align edges are shown. The elements exactly situated in these edges obtained a better value in the Alignment Cost function. Figure 3.h shows graphically the computation of the balanced cost function. Each design element has its own visual importance (in the figure, the most important, the darker) that is used to calculate the centroid of the polygon. If the centroid is close to the center of the design, the composition is well-balanced.

In the previously discussed example, the average time (in seconds) for design item in each phase is: Genetic Production 0.003, Expert Knowledge Kernel 2.017 and Composition 0.901. The interactive time to search for the image in Flickr and to specify the user's preferences depends on the user, but is usually shorter than 10 seconds for the full set of design items (12 per iteration). Thus, the final time per design item is around 5 seconds, but it can be easily improved by using distributed computing.

In order to get the final users' feedback, a web system (<http://www.esi.uclm.es/www/cglez/gaudii/>) for evaluating *Gaudii* has been developed so that any user, distinguishing between the roles *novice*, *amateur* and *professional*, can grade up to 6 designs randomly generated by *Gaudii*. The statistics gathered by the web system shows that the generated designs are evaluated between *good* and *excellent*⁶.

Discussion and Conclusion

Due to its economic relevance, the Graphic Design occupation is predicted to have a long-term growth. In some cases, the requirements of this in-demand occupation can be automatically achieved by computers (for example in the generation of posters to announce events). The expert knowledge needed to perform these activities requires a new extensive approach to model, represent and generate design solutions. *Gaudii* has been developed in response to this challenge.

The hybrid model presented in this work, based on Evolutionary Computation principles, Fuzzy Rule-based Expert Systems and Local Optimization offers several desirable features which together makes this approach unique and of highest practical value:

- *Gaudii* is independent of the user knowledge and only needs the input text to automatically provide some solutions for the design problem.
- Thanks to the independence of the implemented layers, the system is intuitive (allowing several User Interfaces), useful (in a wide range of application domains) and practical (solving real design problems).
- In contrast to Interactive Evolutionary Systems, *Gaudii* is fast and does not need continuous human interaction.

⁶<http://www.esi.uclm.es/www/cglez/gaudii/results.php>

- The proposed system combines good solutions and incorporates the subjective esthetic skills of the human as a selective agent.

Extending *Gaudii* by means of some machine learning technique opens a promising research line. For example, the use of an inductive algorithm allow for the automatic learning of the dependency rules in mutations, and the dependencies of variables to obtain valid designs. The contents of the knowledge base are critical for the performance of the system and can be improved by learning from processed examples by expert designers. Future work also concentrates on using more techniques from the Computer Vision field to identify interesting areas in images, such as the use of visual image descriptors, foreground/background isolation, etc.

References

- Anderson, C.; Buchsbaum, D.; Potter, J.; and Bonabeau, E. 2008. Making interactive evolutionary graphic design practical. volume 88. Springer. 125–141.
- Bentley, P., and Corne, D. 2002. *Creative evolutionary systems*. Morgan Kaufmann Pub. ISBN: 1-55860-673-4.
- Colton, S., and Lopez, R. 2009. Computational creativity: Coming of age. *AI Magazine* 30(3):11–14.
- Geigel, J., and Loui, A. 2003. Using genetic algorithms for album page layouts. *IEEE multimedia* 10(4):16–27.
- Holland, J. H. 1992. *Adaptation in Natural and Artificial Systems: An Introductory Analysis with Applications to Biology, Control, and Artificial Intelligence*. The MIT Press.
- Koza, J. R. 1992. *Genetic Programming: On the Programming of Computers by Means of Natural Selection (Complex Adaptive Systems)*. MIT Press.
- Lewis, M. 2008. *Evolutionary Visual Art and Design (The Art of artificial evolution)*, volume 226. Springer. 1–37.
- Quiroz, J.; Banerjee, A.; Louis, S.; and Dascalu, S. 2009. Document Design with Interactive Evolution. *New Directions in Intelligent Interact. Multimedia Systems* 309–319.
- Romero, J., and Machado, P. 2008. *The art of artificial evolution: A handbook on evolutionary art and music*. Springer-Verlag. ISBN: 978-3-540-72876-4.
- Samara, T. 2007. *Design Elements: A Graphic Style Manual*. Rockport Publishers. ISBN: 978-1-592-53261-2.
- Simonton, D. 1999. *Origins of genius: Darwinian perspectives on creativity*. Oxford University Press, USA. ISBN: 978-0-195-12879-6.
- Statistics, B. L. 2010. Occupational outlook handbook. 2010-2011 Edition (Updated December 2009). Available on the Internet at <http://www.bls.gov/oco/ocos090.htm>.
- Takagi, H. 2001. Interactive evolutionary computation: Fusion of the capabilities of EC optimization and human evaluation. *Proceedings of the IEEE* 89(9):1275–1296.
- Williams, R. 2008. *Non-Designer's Design Book*. Peachpit Press. ISBN: 978-0-321-53404-0.
- Zadeh, L. A. 1975. The concept of a linguistic variable and its applications to approximate reasoning. part i, ii, iii. *Information Science* 8(3):199–249.