Artificial Design

Paper presented at Design Research in the Netherlands 2005 (see www.designresearch.nl)

Automatic style generation

Jos de Bruin[1], Remko Scha[2]

[1] Department of Artificial Design, Institute of Artificial Art Amsterdam
jos@iaaa.nl

[2] Institute of Artificial Art Amsterdam &
Institute for Logic, Language and Computation, University of Amsterdam
scha@uva.nl

1. Introduction

The Institute of Artificial Art Amsterdam (IAAA) works in the area where art & design intersects with computer science and artificial intelligence. The institute started in the 1980's as a primarily artistic endeavor, with a mission to explore the possibilities of computers and other machines for automatic art generation. In recent years, the IAAA has increasingly focussed its research and development efforts on design generators for architecture, information visualization and graphic design. Designs in these fields are to some extent constrained by functional requirements, but there is always considerable room for purely aesthetically motivated variation. We are particularly interested in computer-supported methods for exploring design-spaces of this sort. This paper gives an overview of our theoretical positions and our design-related projects (section 2 and section 3). Special attention is given to an ongoing project, sponsored by *Premsela* (the Dutch Design Foundation), which investigates the application of our approach to corporate identity (house style) design (section 4). This project also involves the development of new interface techniques for our design tools (section 5).

2. Meta-design

Our perspective on design may be called *meta-design.* In graphic design as well as in product design, there is an obvious need to design *styles* rather than individual products. The identity of companies, brands and even individuals is defined by their style: specific designs should be recognizable instances of such a style.

In our research we investigate computational methods which support the process of designing styles: automatic and semi-automatic style generation algorithms. Such algorithms presuppose that the notion of a "style" has been construed in a mathematically precise way - or, at least, that a suitable mathematical analogon of the "style" notion has been devised. How to do that is not immediately obvious, since styles are not natural objects whose structure can be directly observed. In the real world we only encounter *examples* of styles, and we may even feel that a style is *constituted* by its examples. The situation is rather analogous with the study of language, where the only "objective" data are concrete utterances (Saussure's "parole"), while the science of linguistics is necessarily based on the assumption that different people make largely the same generalizations from these data (Saussure's "langue"; cf. Saussure, 1915.)

In Chomskyan linguistics, a *language* is therefore defined as a (usually infinite) set of sentences, generated by a finite grammar which also specifies, for every sentence of the language, what its internal constituent structure is. In thinking about formal theories of visual form, the analogy with verbal language has turned out to be useful. Researchers in the "shape-grammar" tradition have followed Chomsky in defining "visual languages" by means of "shape grammars". Just as a language can be concisely specified in terms of a grammar and a lexicon, we can specify an unlimited number of possible graphics by using a limited set of basic terms, plus a set of operators for constructing complex terms or categories out of the basic ones. Shape-grammars and similar formalisms have been used to define freely invented "meta-artworks", as well as to simulate the specific historical styles of cultural communities or individual artists or designers (Stiny & Gips, 1978).

In our turn, we follow the example of the shape-grammar tradition, and adopt the linguistic approach to the study of visual form. Especially useful for our purposes is the notion of a "sublanguage", due to Chomsky's teacher Zellig Harris (Harris 1991, Chapter 10). A "language" need not be construed as one monolithic object; rather, it may consist of several "sublanguages", characterized by grammars which "instantiate" the grammar of the "superlanguage". That styles have sub-styles and super-styles is an important consideration if we want to think about exploring "spaces of possible styles", because this property induces a particular, lattice-like structure on such spaces.

3. Research areas

Our visual design research ranges from random image generators for art exhibitions to formal theories of visual structure and their application to automatic information visualization, architecture or house style design. We first describe the scientific context of our research into formal models of Gestalt perception. Then we indicate some application areas in architecture and graphic design.

Perceptual structures

In the context of art and design, the most important property of an image is what structure the human perceptual processes will assign to it. This structure is needed to predict its similarity to other images or its likely interpretation. Therefore, the formalisms used to specify visual designs should have what linguists call "*strong* generative power". They should not just generate all and only those forms belonging to a certain visual style, but they should also assign a structure to the form that reflects the visual structure perceived by a human observer of that form.

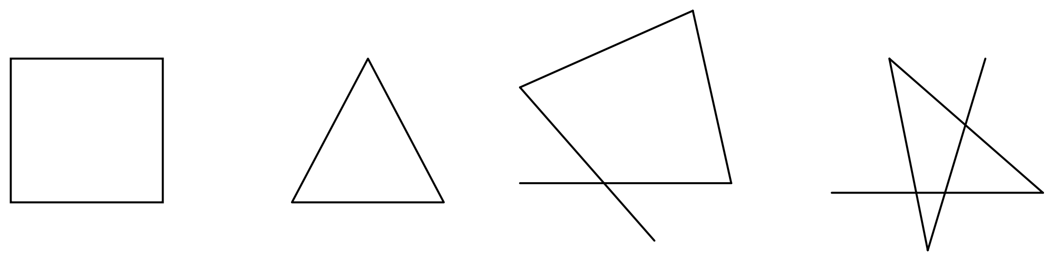
But what does it mean for a formal expression to reflect a visual structure? How do you test this requirement? Can we build on the results of the study of human perception?

The theory of Gestalt perception tried to explain why people prefer particular structural interpretations of visual stimuli, even though these stimuli almost always allow multiple (perhaps even infinitely many) structural characterizations. Wertheimer (1923) identified a number of important organizing principles such as proximity, continuity and similarity. Many of these can be subsumed under one umbrella-principle, called *simplicity* or *Prägnanz*: human perception chooses the simplest analysis among all possible alternatives (Koffka 1935, pp. 171-174). The principle of Prägnanz was formalized by Leeuwenberg (1971) in a framework called *Structural Information Theory* (SIT). Leeuwenberg conjectures that perceiving an image consists in encoding it as an expression of a particular image-description language. Perception is thus a disambiguation problem: an image can usually be described by many different expressions of the coding language, but our perceptual processes tend to choose only one of these. Prägnanz can now be turned into a formal disambiguation criterion: among all the expressions that encode an input image, our perception prefers the shortest one.

Structural Information Theory is essentially a theory of the perception of linear patterns. Its empirical claim is that such patterns are perceived in terms of a limited number of regularities. In recent versions of the theory, these regularities are: symmetry, repetition, and alternation. In the coding language proposed by Structural Information Theory, primitive perceptual elements can thus be combined by simple concatenation and by three composition operators: *Repetition* (which repeats the same sequence a number of times), *Symmetry* (which repeats a sequence in the reverse order) and *Alternation* (which alternates the steps in a sequence with some other sequence).

To apply SIT to visual patterns, severe restrictions on the class of possible inputs are needed. The primitive elements in SIT are best seen as basic instructions for a "turtle graphics" program (*draw line, move pen, turn*). By using the earlier-mentioned structural operators, the sequence of basic turtle steps leading to a particular line drawing can be rewritten. For example, the sequence *ABABA* (with A and B standing for some basic turtle instruction) could be rewritten as *sym(AB, A)*, i.e. a symmetrical structure in which AA is mirrored around an A. Of course, any non-trivial sequence can be reproduced by many different SIT expressions: in this case for instance, 2 * AB, A (i.e. two repeats of AB followed by an A), would lead to the same sequence of turtle steps, but to a different structural analysis. In SIT, the complexity of an expression (determined by its length) is taken as a measure of the complexity of the corresponding perceptual structure. Humans are claimed to prefer the interpretation corresponding to the least complex SIT expression.

The SIT analysis of various classes of visual input patterns has been successfully validated. (Cf. Buffart et al. 1981, Boselie 1988, Boselie & Wouterlood 1989, Van Leeuwen & Buffart 1989, Van Lier 1996). However, as a theory of visual perception, SIT is severely limited by its one-dimensional "turle" perspective. Essentially a formalism for describing sequences, it cannot handle "proximity", as it lacks a notion of distance, nor can it handle even simple 2-D elements such as crossings. A fourfold repetion of an angle A and a line B (ABABABAB) can lead to very different 2-D structures, depending on angle size: a square if A is 90 degrees, a triangle (with one line drawn twice) if A is 120 degrees, or a self-crossing polyline if A is greater than 90 degrees (excluding 120).



In a Ph.D. project in cooperation with the University of Amsterdam, we have started to deal with some of these limitations. We envisage the design of a formal image description language for 2D-patterns which allows us to apply Leeuwenberg's ideas about regularity and simplicity in Euclidean 2D-space, without first mapping it to a linear structure. (Cf. Dastani, 1998; Dastani & Scha, 2003.)

Information visualization

Automatic systems for information visualization or information design presuppose principled and effective methods to reflect the *structural* properties of the information in *structural* properties of the graphics being designed. (Cf. Bertin, 1983; Mackinlay, 1986; Roth, 1990; Roth et al., 1994; Kamps et al., 1996; Zhou and Feiner, 1996; Engelhardt et al., 1996, 1998; de Bruin, 1996; Tufte, 1997; Dastani, 1997; Wang et al., 1997.) An important first step is a systematic analysis of the meaning of the spatial structure of a picture or graph. We achieved significant progress in this area in a Ph.D. project with the University of Amsterdam (Engelhardt, 2002).

Architecture

For Archipel Ontwerpers (The Hague) we wrote an algorithm generating random buildings within certain stylistic constraints. The algorithm was geared towards generating penthouses to be realized by means of a steel construction method which allows considerable freedom in the angles of walls and roofs. Archipel Ontwerpers used our algorithm interactively and chose one of its designs for further elaboration. The resulting design is expected to be realized next year as a penthouse on top of an existing building in the Witte de Withstraat in Rotterdam. (Cf. Melet & Vreedenburgh, 2005.)

Corporate identity design

A good example of meta-design is corporate identity design. Corporate identity covers anything that might affect the public image of an organization, from the styling of its products to the architecture of its buildings, the decoration of its stores, the uniforms of its personnel or its association with certain sports or famous people. In a project funded by *Premsela*, the Dutch Design Foundation, and several large companies, we look at one small but important aspect, the design and management of the visual styles used in corporate communications: company brochures and flyers, reports, business cards, company websites, ads.

The design and management of a house style has become a complicated and costly affair, as media and communication channels multiply and desktop publishing makes it ever easier for departments as well as individual workers to generate their own formats. Corporate communication departments need to strike a balance between the need to control consistency, quality and costs (which seems to call for strict guidelines and predefined formats), and the need for sufficient diversity and local autonomy.

A house style is usually prescribed by a combination of fixed, template-based formats for certain types of documents (e.g. business cards), much more flexible design examples for e.g. company brochures, and purely anecdotal examples of the mood or emotion that e.g. pictures used in company ads would have to convey. Our project is aimed at the design of documents that cannot be given a frozen template-based format, because of the variation in content, but that still need to conform as much as possible to common layout guidelines. Because designers lack a powerful, formal language for specifying visual structures, style guide writers rely on a combination of examples and low-level formatting instructions, such as "always use *this* grid" or "always put the logo at precisely 4 mm from the left and bottom margin". As a result, these style guides leave much to the imagination of the designers. If they take the examples and instructions literally, they will miss intended opportunities for variation, but if they take full advantage of the room left for generalizations, they may well come up with all sorts of unintended variations.

4. Artificial Design project

The goals of the *Artificial Design* project are to develop a formalization of the visual style of company brochures, show how the specific styles of three participating companies (KPN, NS and Océ) can be specified as substyles of this general document design style, and build a prototype that enables the partners in the project to test the implications of our approach for their house style management.

House styles

Generally speaking, the visual style used for company brochures and flyers is rather uniform, with little room for variation. This is at least partly due to functional considerations. These documents often contain a fair amount of text and tables, so readability requirements severely constrain both the placement of items and their rendering. While readable text can still be formatted in different ways, using the same lay-out for different folders makes sense as it eases the search for particular information. There are also economical reasons: by using fixed formats, the cost of applying a style can be kept down. Last but not least is the desire to present a consistent image. Corporate communication managers like to see their folder racks filled with similar looking folders, for aesthetic reasons, but also to show that their organization is in control, knows what it is doing.

The challenge in this project is to show that given all these restrictions, there is still room for variation, that in order for individual items to be seen as variations on the same theme, it is not necessary for each of them to share the same properties. (Cf. Wittgenstein's notion of family resemblance: styles are like natural categories, concepts that cannot be defined in terms of necessary and sufficient properties, even though there is a broad consensus on what objects would fall under their extension.)

Content versus form

In data visualization, the data and their visualizations should (and can) be kept separate and treated distinctly. Automatic visualization techniques rely on this. They use rules to map the semantic properties of the data onto appropriate visual properties and layouts: e.g. a relation between two variables can be mapped to a line graph if the relation corresponds to a table listing two quantitative properties of a set of items; but if one of these variables is a category label and the other variable the number of items in that category, as in a frequency table, then a histogram would be more appropriate.

In document design this distinction between content and form is often harder to make. It can be made when dealing with table-based data, i.e. when formatting catalogues or similar highly structured documents, where the appearance and formatting of various elements can be inferred from their semantic properties. However, when dealing with magazines, books, flyers or advertisements this distinction between content and form is often impossible to make. Isolating the meaning of text from its actual wording is still mostly beyond reach for current technology. The title of a text cannot be automatically inferred from its contents in the same way that for instance a correlation coefficient can be computed for two numeric variables.

As a result, the semantics available for textual documents is mostly restricted to what could be called *presentational* semantics: structural (e.g. title, subtitle, author, footnote etc.) and pragmatical (e.g. *emphasize*) mark-up. While the actual appearance of the documents can thus be kept separate in formatting rules and/or style sheets, their basic structure and format is already presentational: the decision to represent the information contained in the document as a text, and how to structure that text, has already been made and is taken as the starting point for the design process.

For this reason, we treat graphic design not so much as the visualization of an idea or function, but primarily as a matter of reformatting, starting from a simple sequence or graph of textual and other graphical elements. In what is – somewhat confusingly in light of the above remarks - known as *content-based formatting*, i.e. the formatting of books or catalogues, this formatting is a sequential process in which text and other elements flow onto a sequence of pages, the number of which is determined by the amount of text to be formatted. While this may involve some clever computation to deal with widows, hyphenation etc., it is basically a matter of repeatedly filling in page templates. In *layout-based formatting* on the other hand, the overall layout and visual impact of the design is often more open, but the total amount of space available is fixed. This is true of e.g. flyers or posters. In this type of formatting, designers use a trial-and-error approach to achieve the desired impact, going back and forth between different formats and adding elements or suggesting changes to the elements to be included. The *Artificial Design* project is focussed on this type of formatting, which is clearly much harder to formalize.

System architecture

The kernel of the system is a generator that takes a style grammar and an initial expression, and rewrites the non-terminals in this expression until only terminals are left. The resulting intermediate design consists of graphical elements of which the visual rendering has been filled in, but the layout may still be specified in relative terms (e.g. "align vertically with center of predecessor in parent structure, horizontally 5 mm to the left of the right border of the containing graphic"). In the next cycle, these lay-out instructions are applied to the current design, leading to a design in which the absolute size and position of all graphical elements has been determined. This design can be output as an XML document, which can then be rendered by any lay-out program for which an appropriate script has been written. In the prototype systems, instantiated designs can either be rendered directly by a renderer written in Java, or by Adobe InDesign, using a simple Visual Basic script.

Design representation

Non-terminals are like frames or classes (or complex types, in XML terminology), with attributes whose value restrictions are again specified as (subtypes of) non-terminals. The set of non-terminals will be open-ended, but any new types will have to be specified as subtypes of a small set of basic types. Fully instantiated designs are specified using only basic types, to ensure that they can be rendered using the scripts provided for various renderers. Some basic types are:

- *graphic*, any element to be placed somewhere in the design, with attributes such as color, size, alignment, scaling (examples of subtypes: paragraph, image);
- *structure,* a collection of graphic and/or structure elements (and itself a kind of graphic); in addition to providing a hierarchical nesting that provides context when evaluating the scaling and alignment instructions, structures may also come with their own layout (e.g. a grid);
- *alignment,* an instruction to align some graphic relative to some other graphic.

We use the XML Schema standard to specifiy these basic types. A second XML Schema specifies the formalism used to define the actual style grammar, for instance to declare a brochure as a kind of structure consisting of a cover spread (itself a structure consisting of pages) and several inner spreads etc. These complex type definitions restrict the value range and cardinality of attributes, including their probability distributions. Style definitions can thus be maintained as separate XML documents.

5. Interface

Traditionally design software is seen as a tool for a designer working on the visual layout and appearance of a particular product. Consequently, the user interfaces of these tools are stuck in the toolbox metaphor: Photoshop, InDesign, AutoCad or QuarkXpress, they all start with a blank sheet and a rather overwhelming array of menus and dialogs. However appropriate this approach may be for expert designers who know what they want to achieve (and how to achieve it), it is not the most efficient way to interface with style generators, neither for users of such generators nor for their developers.

Style definitions can only be developed by looking at significant and representative samples of the sort of designs they would give rise to. When applying the style definitions, users should be able to get a good impression of the variation allowed by the style before committing to a particular design. Most users will lack the expertise needed to infer the implications of a style definition, let alone of changes to such a definition, without being able to look at concrete examples.

A simple approach would be to generate a large number of example designs in some random order. This is the approach followed in our Artificial image generation programs. For artificial art this is an effective way to give an overall impression of a style, but when users are looking for a good design for a particular document, this is not very efficient. Design is often described as the search for a fitting solution to a set of constraints. Using automatic generation this becomes literally true. The solutions are all there, the problem is knowing where to look for likely candidates.

To be able to quickly zoom in on the right area, we are exploring a technique we call flashing. Flashing is like automatic browsing (as in a slide show), but with a more flexible method of selecting the items to be shown, providing a better sense of the space of possibilities of which the item shown is just one instance. Direct manipulation of, and navigation through, some visual representation of information spaces is a standard tactic in data visualization. New in our approach is that we show the actual items, instead of just a few visual indicators of some of their attributes. When visualizing collections of non-visual information such non-iconic indicators are unavoidable, but here we are dealing with data whose visual appearance is their main, if not their only point of interest. To convey the impact of a particular style, certainly to non-experts and in most likelihood also to experts, it will be more effective to show the samples themselves, and not try to abstract away from their actual appearance. Designers communicate almost exclusively by example, not just to their clients but also among colleagues.

Flasher was originally developed as an alternative way to browse the internet for non-textual, graphical items. Searching for such items with current tools is rather cumbersome and inefficient, not only because these searches are mostly severely restricted by the limited number of available keywords, but also because only a limited number of results of such searches are shown, ordered on a single relevance measure.

Flasher assumes a more active approach, continually collecting items, ordering them in a multi-dimensional space and actively displaying a random slide show of the items falling within the area of the search space currently selected by the user. Items are ordered in a limited number of categories on each selected dimension. Two of these are used to subdivide the presentation panel in an n-by–m cross-tabulation, each sub panel showing a selection of items that fall within that particular subgroup. Users move through a search space by switching perspective, i.e. by choosing different 2-D slices through the n-dimensional space, and by zooming in and out of the sub-divisions. When zooming in, the items in that particular cell are again distributed over an n-by-m cross tabulation that further divides the attribute ranges zoomed into.

6. Discussion

The overall aim of our research is to come to grips with the elusive notion of visual style, using a formal, algorithmic approach, centred on the development of automatic generators of visual designs and taking human perception of visual structure as pour inspiration. The *Artificial Design* project attempts to make the step from specialized, hand-tuned generators for specific designs, i.e. generators with a built-in style, to a more general generator that uses declarative style specifications to guide its designs. Later this summer we expect to demonstrate a prototype written in Java and consisting of a generator, a Flasher-based interface, a specification of three different house styles for the design of brochures and, in addition to the bundled Java renderer, an option to render documents in InDesign.

Focussing on corporate styles greatly restricts the variation in visual designs to be handled. Corporate styles tend to be simple, uniform and rather similar, ruled by the regular grid of modernistic design. However, this lack of variability is momentary: corporate styles change over time, mostly in sync with general fashions regarding preferences for straight versus curved lines or the use of decorative elements. An adequate style specification needs to handle not only the ostensibly restricted styles occurring at a specific moment in time, but also at least some of their variations over time.

As indicated in section 3, the search for perceptually relevant structure-building operators is on-going. A major challenge is satisfying the constraints imposed by top-down generated hierarchical structures with the constraints imposed by the elements to be positioned in these structures. In terms of the linguistic metaphor used earlier: if visual structures are seen as expressions of a visual grammar, generation is not just a matter of finding a sequence of words that expresses the intended meaning, but also of using pre-determined words and phrases as part of that expression, and of making sure that the length of the total expression stays within a particular range.

The usefulness of the proposed Flasher interface depends on the availability of useful dimensions to categorize the items to be flashed. In this automatic design system, information about these dimensions is available from the style definition file. The main issue in this type of application is whether users can actually use these dimensions effectively to zoom in on the desired designs. This will depend both on the structure of the overall space (i.e. can designs that are similar in terms of what the user is looking for, actually be found together in the space used) and on the ability of users to infer a sense of direction towards such groups in this space.

Artificial graphic design is conceivable because graphic design is a matter of reformatting. Such design-as-repackaging is not restricted to graphic design. It occurs whenever design is not determined solely by function, but also by considerations of taste, fashion or aesthetics. We are convinced that designers will increasingly use automatic design tools to explore this "room for aesthetic maneuver", formulating the rules once instead of repeatedly creating new samples.

7. References

Boselie, F. (1988). Local versus global minima in visual pattern completion. *Perception & Psychophysics,* **43**, pp. 431-445.

Boselie, F. and D. Wouterlood (1989). The minimum principle and visual pattern completion. *Psychological research,* **51**, pp. 93-101.

Bruin, J. de, P. de Vink and Jarke van Wijk (1996). M - A Visual Simulation Tool. In: *Simulation in the Medical Sciences,* The Society for Computer Simulation, San Diego, 181-186.

Bruin, Jos de and Remko Scha (1998). A republic of information designers. In: *Proceedings Vision Plus 4,* Carnegie Mellon University, Pittsburgh, 26-29 March 1998.

Bruin, Jos de and Remko Scha (2003). Algoritmische kunst spot met museale waarden." *Automatisering Gids,* March 14, 2003, p. 15. [Reprinted in: Cees de Boer (ed.): *Concrete Kunst & Mathematiek*. Reader for an International Symposium at the Mondriaanhuis, Amersfoort, 25 October 2003.]

Bruin, Jos de and Remko Scha (2003). Algoritmische architectuur is toegepaste toevalskunst." *Automatisering Gids,* April 25, 2003, p. 17.

Buffart, H., E. Leeuwenberg and F. Restle (1981). Coding theory of visual pattern completion. Journal of Experimental Psychology: Human Perception and Performance. Vol. 7, pp. 241-274.

Collard, R., P. Vos and E. Leeuwenberg (1981). What Melody Tells about Metre in Music. *Zeitschrift für Psychologie*, 189.

Dastani, M. (1997). A Formal Framework for Visual Representation. *TVL 97, International Workshop on Theory of Visual Languages*, Capri, Italy, pp. 117-126.

Dastani, Mohammad Mehdi (1998). *Languages of Perception.* Ph.D. Dissertation, University of Amsterdam, ILLC Dissertation Series 1998-05.

Dastani, M., B. Indurkhya and R. Scha (2003). Analogical projection in pattern perception. *Journal of Experimental and Theoretical Artificial Intelligence,* December 2003.

Dastani, Mehdi and Remko Scha (2003). Languages for Gestalts of Line Patterns. *Journal of Mathematical Psychology,* **47**, pp. 429-449.

Engelhardt, Y., J. de Bruin, T. Janssen and R. Scha (1996). The visual grammar of information graphics. In: *Artificial Intelligence in Design (AID '96), Workshop on Visual Representation, Reasoning and Interaction in Design*, June 1996, Stanford University.

Engelhardt, Y. (1998). Meaningful space in diagrammatic graphics. In: *Proceedings Vision Plus 4,* Carnegie Mellon University, Pittsburgh, 26-29 March 1998.

Engelhardt, Y. (2002). *The Language of Graphics. A framework for the analysis of syntax and meaning in maps, charts and diagrams.* Ph.D. Dissertation, University of Amsterdam,. ILLC Dissertation Series 2002-03.

Harris, Zellig (1991). *A Theory of Language and Information. A Mathematical Approach.* Oxford: Clarendon Press.

Helm, P. van der and E. Leeuwenberg (1991) Accessibility: a criterion for regularity and hierarchy in visual pattern code. *Journal of Mathematical Psychology,* **35**, pp. 151-213.

Koffka, K. (1935). *Principles of Gestalt Psychology.* London: Routledge and Kegan Paul.

Leeuwen, C. van and H. Buffart (1989). Facilitation of retrieval by perceptual structure. *Psychological Research,* **50**, pp. 202-210.

Leeuwenberg, E.L.J. (1971). A Perceptual Coding language for Visual and Auditory Patterns, *American Journal of Psychology,* **84**, 3.

Lier, R.J. van, (1996) *Simplicity of visual shape. A structural information approach.* Ph.D. Dissertation, NICI, University of Nijmegen.

Liere, R. van and J.J. van Wijk (1996). CSE - A modular architecture for computational steering. 7th *Eurographics Workshop*, Prague.

Mackinlay, J. (1986). Automating the design of graphical presentations of relational information. *ACM Transactions on Graphics*, **5**, No. 2.

Melet, Ed and Vreedenburgh, Eric (2005). Rooftop Architecture, Rotterdam: NAi Publishers.

Saussure, F. de (1915). *Cours de Linguistique Générale.* English translation: (1966) *Course in General Linguistics.* New York: McGraw-Hill.

Scha, R. (1989). Computer/Art/Photography, *Perspektief*, **37**, pp. 4-10.

Stiny G., and J. Gips (1978). *Algorithmic Aesthetics. Computer Models for Criticism and Design in the Arts.* Berkeley: University of Califormia Press.

Tufte, E.R. (1997). *Visual Explanations.* Graphics Press.

Twyman, M. (1979). A Schema for the Study of Graphical Language. In: P.A. Kolers, M.E.Wrolstad, H. Bouma (eds.) *Processing of Visible Languages.* **1**, New York: Plenum Press.

Vreedenburgh, E. and R. Scha (1994). Vers une autre Architecture. *Zeezucht* 7, 8, pp. 6-14.

Wang, D., Y. Engelhardt and H. Zeevat (1997). Formal Specification of a Graphic Design Theory *TVL 97, International Workshop on Theory of Visual Languages*, Capri, Italy; pp. 97-116

Wertheimer, M. (1923). Untersuchungen zur Lehre von der Gestalt. *Psychologische Forschung* **4**, pp. 301-350. (English translation of important parts of this paper: Laws of Organization in Perceptual Forms. In: W. Ellis *A source book of Gestalt psychology.* London: Routledge & Kegan Paul, 1938, pp. 71-88.)

Zhou, X.Z. and S.K. Feiner (1996). Data Characterization for Automatically Visualizing Heterogeneous Information. In: *Proc. InfoVis'96*, 13-20, San Francisco: IEEE.