

A Computer Model of Aesthetic Industrial Design

by

David R. Wallace

Bachelor of Mechanical Engineering
Carleton University
June, 1989

Bachelor of Industrial Design
Carleton University
June, 1986

Submitted to the Department of Mechanical Engineering
in Partial Fulfillment of the Requirements for the

MASTER OF SCIENCE
in Mechanical Engineering

at the

Massachusetts Institute of Technology

August, 1991

© Massachusetts Institute of Technology, 1991, All rights reserved.

Signature of Author _____
Department of Mechanical Engineering
August 9, 1991

Certified by _____
Mark Jakiela
Assistant Professor, Mechanical Engineering
Thesis Supervisor

Accepted by _____
Ain A. Sonin
Chairman, Department Committee

MASSACHUSETTS INSTITUTE
OF TECHNOLOGY

FEB 20 1992

LIBRARIES

ARCHIVES

A Computer Model of Aesthetic Industrial Design

by

David R. Wallace

Submitted to the Department of Mechanical Engineering
on August 9, 1991 in partial fulfillment of the requirements for the degree of
Master of Science in Mechanical Engineering

ABSTRACT

A computer model of the industrial design process for mass produced consumer products is proposed and described. The model can be used to design products subject to ergonomic, manufacturing, aesthetic, and style constraints. The four stages of the model, component organization, surface styling, product detailing, and graphical design are discussed. The model is verified through a computer-based implementation for the design of injection moulded consumer electronics products. As input to the system, users choose a selection of components to be included in the product (e.g. speakers, displays etc.) and choices for product attributes, including an aesthetic style. Three styles — Braun-like, High-Tech, and Art Deco — have been implemented. The system automatically configures the components, generates a product housing, applies details such as buttons or speaker grills, and colours the product. The model can generate designs ranging from calculators to stereos and televisions using general design principles. Prototypes or templates for specific kinds of products are not used. A valuable feature of the system is its ability to propose multiple design alternatives. The work has confirmed the validity of the proposed model for rapid generation of aesthetic preliminary product designs which are conceptually correct for their intended market, use, and manufacture.

Thesis Advisor: Prof. Mark Jakiela

ACKNOWLEDGEMENTS

The author wishes to acknowledge the Leaders for Manufacturing Program at the Massachusetts Institute of Technology for its support of this work. Additional support was received from the Massachusetts Institute of Technology Sloan Foundation. This support is greatly appreciated. The computing facilities made available through the Massachusetts Institute of Technology Mechanical Engineering CADLAB were indispensable.

The guidance from my advisor, Mark Jakiela, has been enlightening. Support received from members of the Computer Aided Design Lab and friends at MIT is most appreciated.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS	3
TABLE OF CONTENTS	5
LIST OF FIGURES.....	8
LIST OF TABLES.....	11
NOMENCLATURE.....	12
1 INTRODUCTION.....	15
1.1 Goals and Motivation.....	15
1.2 Outline of the Thesis.....	16
1.3 Terminology	17
1.4 Need to Automate and Factors to be Considered.....	18
1.5 Domain of the Proof of Concept	19
1.6 Overview of the Proposed System.....	21
1.7 Role in the Overall Design Process	22
1.8 Potential Benefits.....	23
2 BACKGROUND.....	24
2.1 The Key to Aesthetics and Style.....	24
2.1.1 Aesthetics.....	24
2.1.2 Style.....	28
2.1.2.1 Shape and Structure Grammars.....	28
2.1.2.1.1 Shape Grammars.....	28
2.1.2.1.2 Structure Grammars	31
2.1.2.2 Analogy	32
2.1.2.2.1 Letter Spirit	32
2.1.2.2.2 Morphological Analysis	33
2.1.2.3 Hybridization of Surface Styles	35
2.2 Manufacturing and Ergonomics.....	37
2.2.1 Manufacture	37
2.2.2 Ergonomics.....	39
3 MODEL FOR FORM GIVING IN INDUSTRIAL DESIGN.....	41
3.1 The Overall Structure	41
3.2 Four Level Model for Industrial Design Form Giving.....	42
3.3 Requirements of the Receptor and the Product Structure.....	44
3.4 Requirements of the Organization Level.....	46
3.4.1 Overview and Background.....	46
3.4.2 Production System Architecture	48
3.4.3 Organization of the Organization Level	49
3.4.3.1 Stage 1	
Conceptually Blocking Out the Product	51
3.4.3.1.1 The Product Matrix	52
3.4.3.1.2 Rules for Filling the Product Matrix.....	53
3.4.3.1.3 Component Taxonomy and	
Requirements	53
3.4.3.1.4 Rules and the Product Matrix.....	55
3.4.3.2 Stage 2	
Aesthetic Positioning of the Components	56

3.5	Requirements of the Surface Level	58
3.5.1	Definition of a Prototype.....	59
3.5.2	Requirements of the Prototype and Utilization.....	60
3.5.3	Designing a Prototype Surface Form Organization.....	62
3.5.3.1	The Methodology.....	62
3.6	Scaling Effects.....	68
3.7	Requirements of the Detail Level.....	69
3.7.1	Reasons to Apply Details	70
3.7.2	Requirements of the Prototype and Utilization.....	71
3.8	Requirements of the Graphical Level of the Model	73
3.8.1	Colouring Considerations	74
3.8.2	Application of Colour Palettes to the Detail Product.....	75
4	IMPLEMENTATION	76
4.1	Global Architecture of the Implementation.....	77
4.2	The Product Structure	78
4.3	The Component Library.....	79
4.4	Organization Level	81
4.4.1	Stage 1 Filling the Product Matrix	82
4.4.1.1	The Matrix Structure	83
4.4.1.2	Production Rules	84
4.4.1.2.1	Component Locations in the Matrix	85
4.4.1.2.2	Some Example Rules	89
4.4.2	Stage 2 Aesthetic Positioning of the Components.....	91
4.4.2.1	Estimating the Product Size and Format.....	92
4.4.2.2	Positioning Components within a Matrix Element ...	96
4.4.2.3	Positioning Elements to Complete a Product Face ...	97
4.4.2.4	Positioning the Matrix Faces.....	99
4.4.2.5	Building the Bounding Hull.....	100
4.5	The Surface Level.....	101
4.5.1	Prototype Styles	102
4.5.2	Designing Surfaces with Form Organization.....	103
4.5.3	Using the Surface Prototype Library	106
4.5.3.1	Prototype Anatomy and Selection of a Prototype.....	106
4.5.3.2	Mapping of the Surface Prototype.....	107
4.6	The Detail and Graphical Levels.....	110
4.6.1	The Detail Library	110
4.6.2	Operation of the Detail and Graphics Level.....	111
4.6.3	Determining the Detail Size.....	113
4.6.4	Computing Details.....	114
4.6.4.1	Acoustic Class Details	114
4.6.4.2	Visual Display Class and Mechanism Details	115
4.6.4.3	Keypad, Jack, and Switch Class Details	116

5	RESULTS AND DISCUSSION.....	117
5.1	The Capture of Style using the Four Level Model	117
5.1.1	Comparison to Existing Style Archetypes	123
5.2	Range of Products in Proof of Concept	125
5.3	Production of Multiple Product Variations.....	129
5.4	Organization Level	135
5.4.1	General Discussion.....	135
5.4.2	Effects of Product Attributes on Configuration	137
5.5	Surface Level	139
5.6	Detail and Graphical Level	141
6	CONCLUSIONS AND FUTURE WORK.....	143
6.1	Conclusions.....	143
6.1.1	Organization Level.....	144
6.1.2	Surface level.....	145
6.1.3	Detail and Graphics Levels	146
6.2	Future Work.....	147
6.2.1	Development in the Context of Aesthetic Design	147
6.2.2	Expansion within the Overall Design Process.....	148
	REFERENCES	150
	APPENDIX A Elements in the Component Library	154
	APPENDIX B Prototypes in the Surface Style Library	157
	APPENDIX C Prototypes in the Detail Library	160

LIST OF FIGURES

Figure 1.1	Overview of an automated aesthetic industrial design system	15
Figure 1.2	The scope of the proof of concept	19
Figure 1.3	Major constituents of the proposed system	21
Figure 1.4	Role of the proposed system in design and manufacture.....	22
Figure 2.1	Illustration of the idea of pragnanz.....	25
Figure 2.2	The gestalt laws of grouping	25
Figure 2.3	Methods to create unity and harmony	26
Figure 2.4	Form and perception.....	27
Figure 2.5	Elements of a simple shape grammar	29
Figure 2.6	Mechanics of a simple shape grammar	29
Figure 2.7	Possible application of a shape grammar to design a product enclosure.....	30
Figure 2.8	Example of a Structure Grammar	31
Figure 2.9	How details change style.....	32
Figure 2.10	Results from letter spirit.....	32
Figure 2.11	An application of morphological analysis.....	33
Figure 2.12	Using ideas of morphology to generate an enclosure.....	34
Figure 2.13	General application of the morphology analogy	35
Figure 2.14	Hybridization based upon combining features and on global averaging.....	36
Figure 2.15	Design fixer upper	37
Figure 2.16	The weakness of fixer uppers	37
Figure 2.17	A typical part from a straight draw mould with no side holes	39
Figure 3.1	General model for the proposed system.....	41
Figure 3.2	The four level hierarchy for form giving.....	42
Figure 3.3	Example of the stages in the four level model	43
Figure 3.4	The product structure slots.....	45
Figure 3.5	Elements of the organization level	47
Figure 3.6	Combinatorial approach to configuration design.....	47
Figure 3.7	Simplified production system architecture	49
Figure 3.8	The two stages of the organization level	50
Figure 3.9	Improved second stage of the organization level.....	50
Figure 3.10	Activities of the first stage of the organization level	51
Figure 3.11	The concept of the product matrix	52
Figure 3.12	The component structure.....	54
Figure 3.13	Inputs and outputs of the aesthetic positioning stage	56
Figure 3.14	Aesthetic concerns in component positioning	57
Figure 3.15	Elements of the surface level of the model.....	58
Figure 3.16	Prototypic enclosure.....	59
Figure 3.17	Surface prototype structure	61
Figure 3.18	The control points to define a conic segment	63
Figure 3.19	The use of conic control curves to define a surface.....	63
Figure 3.20	Nomenclature for points defining a conic segment	64
Figure 3.21	Parameter spacing weighting function.....	65
Figure 3.22	The shoulder point of a conic segment.....	66
Figure 3.23	Effect of the weighting function on a curve with six data points.....	67
Figure 3.24	Illustration of how corner style changes under scaling.....	68
Figure 3.25	Input and output of the detail level of the model	69

Figure 3.26a	Different groupings of components under details.....	70
Figure 3.26b	Applying details to integrate product faces.....	71
Figure 3.27	Detail prototype structure	71
Figure 3.28	Possible methods to incorporate details into a surface.....	72
Figure 3.29	Input and output of the graphical level of the model.....	73
Figure 4.1	Major elements of the program.....	77
Figure 4.2	Component library data file structure.....	81
Figure 4.3	How the product matrix is filled	82
Figure 4.4	Faces and normals of the product matrix.....	83
Figure 4.5	Structure representation of the product matrix.....	84
Figure 4.6	Location of acoustic components in the product matrix	86
Figure 4.7	Location of jack components in the product matrix.....	87
Figure 4.8	Location of visual display components in the product matrix	88
Figure 4.9	Location of keypad membrane components in the product matrix	88
Figure 4.10	Location of cassette mechanism in the product matrix.....	89
Figure 4.11	Location of switch components in the product matrix.....	89
Figure 4.12	Component positioning procedures.....	92
Figure 4.13	Estimating the size of a matrix element.....	93
Figure 4.14	Estimating the size of a matrix face	93
Figure 4.15	Swapping matrix elements in hand held products.....	96
Figure 4.16	Context dependent ordering of components within a matrix element....	97
Figure 4.17	Building rows or columns in a matrix face	97
Figure 4.18	Row or column justification options.....	98
Figure 4.19	Direction of movement for product faces	99
Figure 4.20	Diagram of the surface level implementation	102
Figure 4.21	Form organization control lines for designing the Braun-like enclosure.....	103
Figure 4.22	Form organization data to encode a surface.....	104
Figure 4.23	Control curve slicing methods in the form organization module	105
Figure 4.24	Corners characteristic of the three styles	106
Figure 4.25	Desk top and hand held enclosures	106
Figure 4.26	Enclosures with horizontal and vertical parting lines	106
Figure 4.27	The enclosure prototype library file.....	107
Figure 4.28	Enclosure prototype nomenclature.....	108
Figure 4.29	Characteristics of the scaling function.....	109
Figure 4.30	The detail file format.....	111
Figure 4.31	Nomenclature for the detail file.....	111
Figure 4.32	Example of the decision process for sizing a detail	114
Figure 5.1a	A Braun-like style portable stereo.....	118
Figure 5.1b	A High-Tech style portable stereo	119
Figure 5.1c	An Art Deco style portable stereo	120
Figure 5.2a	A Braun-like style calculator.....	121
Figure 5.2b	A High-Tech style calculator	122
Figure 5.2c	An Art Deco style calculator	123
Figure 5.3	Braun Archetypes	124
Figure 5.4	High-Tech	124
Figure 5.5	The Art Deco archetypes	125
Figure 5.6a	Braun-like mono cassette player design	126
Figure 5.6b	Handset design.....	127
Figure 5.6c	Pocket computer design.....	127
Figure 5.6d	Stereo television design	128
Figure 5.6e	Tall television design	128

Figure 5.6f	Mono television design.....	128
Figure 5.7	Product versions with different semantic interpretations.....	130
Figure 5.8a	Alterative stereo version 1	131
Figure 5.8b	Alterative stereo version 2	131
Figure 5.8c	Alterative stereo version 3	132
Figure 5.8d	Alterative stereo version 4	132
Figure 5.8e	Alterative stereo version 5	132
Figure 5.8f	Alternative stereo version 6.....	133
Figure 5.9a	Poor alternative stereo version 1	133
Figure 5.9b	Poor alternative stereo version 2.....	134
Figure 5.10	Modified organization level with an evaluation mechanism.....	134
Figure 5.11	Components prior to organization.....	135
Figure 5.12	Output from the organization level.....	135
Figure 5.13a	Stereo with a horizontal parting plane	137
Figure 5.13b	Alternative horizontal design	137
Figure 5.14a	Desk top design	138
Figure 5.14b	Hand held design	138
Figure 5.15	The effect of product height on the configuration.....	138
Figure 5.16	Example output from the surface level	139
Figure 5.17	Distorted technical corner due to uniform scaling.....	140
Figure 5.18	Output from the detail level	141
Figure 5.19	Panel inserts used to simplify mouldings	142
Figure A1	Acoustic components.....	154
Figure A2	Keypad membranes.....	154
Figure A3	Visual displays	155
Figure A4	Visual display	155
Figure A5	Switches	156
Figure A6	Jacks.....	156
Figure A7	Mechanism.....	156
Figure A8	Jacks.....	156
Figure B1	Braun-like desk top prototypes.....	157
Figure B2	Braun-like hand held prototype	157
Figure B3	High-Tech desk top prototypes	158
Figure B4	High-Tech hand held prototype	158
Figure B5	Art Deco desk top prototypes.....	159
Figure B6	Art Deco hand held prototype	159
Figure C1	Panel inserts	160
Figure C2	Speaker grill used for visual displays	160
Figure C3	Typical mask details used for visual displays	161
Figure C4	Buttons used for keypads and switches	161
Figure C5	Toggle switches.....	162
Figure C6	Jack covers for speaker and headphone jacks	162
Figure C7	Cassette Door.....	163

LIST OF TABLES

Table 4.1 The components and their attributes80

NOMENCLATURE

\mathbf{r}_i	position vector in Cartesian coordinates
\mathbf{r}_s	shoulder point on a conic segment
w_i	weighting factor
\mathbf{n}	normal vector in Cartesian coordinates
\mathbf{t}	tangent vector
t	spline parameter
u	conic segment parameter
u_s	shoulder point conic segment parameter
μ	tangent length
Ω	tangent angle
m	number of points on a conic segment

To my parents

"Epitaph of an unfortunate artist"

He found a formula for drawing comic rabbits:
This formula for drawing comic rabbits paid,
So in the end he could not change the tragic habits
This formula for drawing comic rabbits made.

by Robert Graves in
Matt Groening, 1990, *The Big Book of Hell*,
Matt Groening Productions Inc.

1 INTRODUCTION

1.1 Goals and Motivation

Consider a computer aided design system that, given a selection of standard components, automatically generates designs based upon manufacturability, ergonomics, aesthetics, and style. The system would link with existing computer aided engineering packages to allow the aesthetically naive engineer to pursue mechanical engineering and industrial design in a unified process. The goal of this thesis is to consider a system as depicted in figure 1.1, and demonstrate its feasibility in the proof of concept design domain of consumer electronics. To achieve this goal it is necessary to propose a model for industrial design and devise methods to address aesthetics, ergonomics, corporate style, and manufacturing in a computer aided design system.

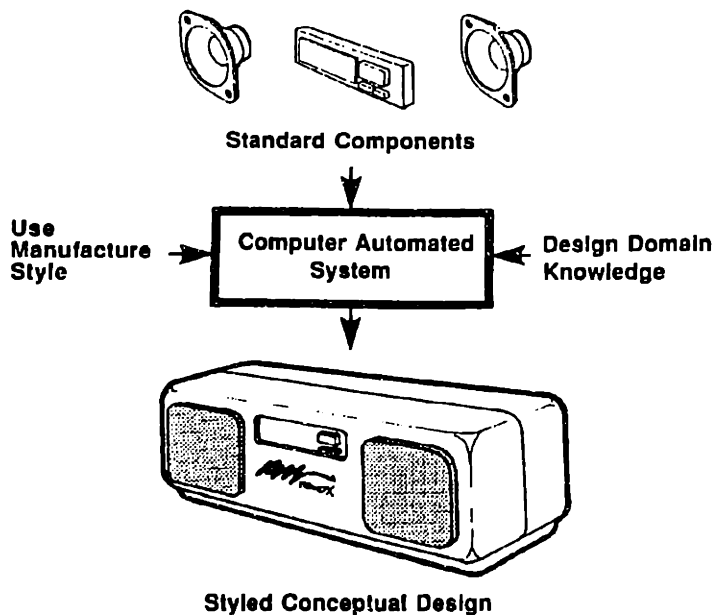


Figure 1.1 Overview of an automated aesthetic industrial design system

Belief that such a system would be possible arose through observations as an industrial designer about the evolution of products. One can observe that the form of new products rarely (if ever) pops up out of the blue, but rather is borrowed or hybridized from some other familiar existing product. For example, the first microwaves borrowed their look from televisions, while the original televisions looked more like furniture.

Thus, it is felt that a large part of product design creativity involves drawing upon familiar artifacts or experiences and applying them to new situations. If this is true, it is my supposition that a system which incorporates standard design practice within a chosen domain can be used to formulate a general layout method for creating new products with appropriate semantics. In a sense, this system would automate a form of creativity which,

until now, has been reserved only for human activities. If this goal is achieved, it will help the engineer to design something beautiful rather than just functional.

1.2 Outline of the Thesis

The body of the thesis is divided into six chapters. The intent of chapter 1 is to provide introductory insight about terminology, the proposed system, the proof of concept domain, possible benefits, and where the system would fit in the overall design and manufacturing environment.

Chapter 2 provides background material which addresses elements critical to aesthetics and style in product design. Related work regarding the computer automation of style is discussed. Finally, the chapter reviews the basic manufacturing and ergonomic factors which play an important role in the conceptual design of consumer products.

A model for the automated industrial design system is proposed in chapter 3. In this chapter, the goal is to present the model in a more general design context, rather than in the specific domain of the implementation. A four level model for aesthetic design is advocated and issues and methods related to the implementation of this model are discussed.

Chapter 4 addresses the model of chapter 3 within the specific context of the proof of concept implementation. The material should provide the necessary knowledge to understand how the system designs the products using the four level model.

Results from the implementation demonstrating the model's ability to capture styles and produce reasonable designs are presented in chapter 5. The performance and merit of the implementation is reflected upon in this chapter.

Finally, in chapter 6 conclusions about the proposed system's feasibility and limitations are drawn based upon the results presented in chapter 5. The system is regarded as a first step in the venture to automating aesthetic design — suggestions for future work are provided.

1.3 Terminology

The design related terms used with specific implications throughout the thesis are clarified here.

The thesis is devoted to automating a type of design which I have referred to as *industrial design*. In the thesis, the term *industrial design* encompasses the design of everyday products which are mass produced for the consumer market. In this type of design, functionality is considered from the perspective of the user. Human factors or ergonomics, aesthetics, style, and manufacturing are particularly important. The main objective of the *industrial design* process is what shall be called *form giving*.

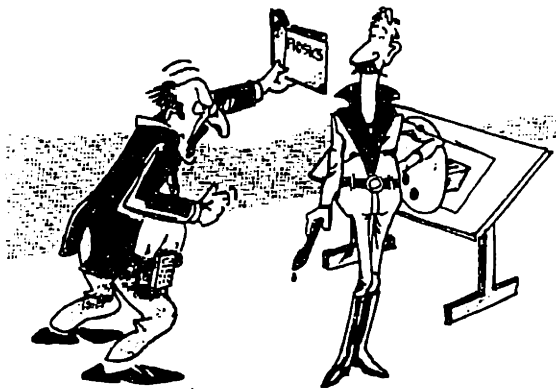
Form giving refers to the generation of the form, or seen surface of a product. From the consumer's viewpoint, the form is the product's identity. The primary interest of the thesis is to automate the process of *form giving*. When discussing form, the words *aesthetics* and *style* are used frequently.

Aesthetics refers to beauty and addresses characteristics which will make an object be perceived as appealing or unappealing. When discussing *aesthetics*, I am concerned about issues such composition and proportion. *Aesthetic* issues are considered as fundamental common concerns to be addressed in the configuration of all products. For my purpose, *aesthetic* decisions are independent of *style* (this shall be justified later).

I use the term *style* to refer to more superficial characteristics or features of a form which can be used to differentiate similar products (two cars on the same chassis can be made to look different through *styling*) or can be used to make dissimilar objects look like they belong together. A *corporate identity* is a style which is consistently used by a company to make all of their products identifiable as their own. The establishment of standard practice is the key to corporate product identity.

1.4 Need to Automate and Factors to be Considered

Technical design often suffers from a duality of approach depending upon whether the design is to be judged as useful or beautiful (Benton, 1975). The engineer is literate in function only, while beauty is the responsibility of the Industrial Designer.



It is common for *stylists* to work in isolation, conjuring fantastic product images to which engineering and manufacturing must conform. (Note the use of the name *stylist* rather than industrial designer. I use *stylist* to describe those who create form solely upon visual criteria rather than the totality of the product, including manufacture, ergonomics, etc.)

The stylist philosophy can lead to marginally functional products which, because of their arbitrary form, rapidly become aesthetically dated. In the converse scenario, the industrial designer is called upon to impart aesthetic improvements after the engineer has nearly completed the design. The 'improvements' often must take the form of ornament or surface embellishments. As such additions are costly, aesthetic considerations are frequently ignored altogether. Regardless, ornament will do little to rescue a bad form.

I suggest that the myriad of unappealing and difficult to manufacture products evident in the marketplace may, in part, be attributed to the separation of function and form design. It should be the goal of product designers to combine both beauty and function in consumer products. To realize such a goal would require the unified cooperation of the artisan and the engineer. Unfortunately, in many companies, the differences in opinion between the two groups is a constant source of friction. The system studied in this thesis tries to integrate the industrial designer's expertise with conceptual engineering considerations, thereby facilitating the unification of the design process.

Form design cannot be viewed in isolation, but must be shaped by its context or environment. Without such notions arbitrary forms are probable, and the designer will

have adopted the stylist philosophy! Thus, in addition to encoding the essence of a style and aesthetics, the automated system must acknowledge the product's context if it is to produce functional, attractive, timely, and producible products. Therefore, the proposed system must minimally consider how the product is used, and how the form is influenced by materials and manufacturing processes.

In this effort, the scope shall be limited to automating a design process through the use of form rules established by the industrial designer, in addition to manufacturing and ergonomic considerations. Design automation related to functional concerns, such as housing strength, internal housing features, and wire routing etc. have been ignored to focus upon how the conceptual design process is influenced by aesthetics, style, ergonomics, and manufacture.

1.5 Domain of the Proof of Concept

The feasibility of the computer automated industrial design system (CAID) is studied in the context of a specific design domain — desk top or hand held injection moulded consumer electronic products that have visual, auditory, or tactile interactions with the user — as indicated in figure 1.2.

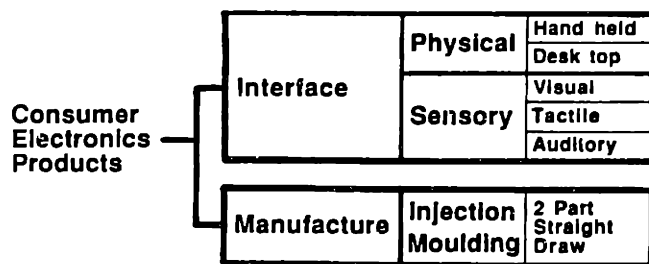


Figure 1.2 The scope of the proof of concept

The proof of concept is identified at the onset as I feel that, because design requirements vary so widely between design domains, a generic multi-domain automated system is not realizable. The chosen domain encompasses a very wide variety of products with similar design requirements.

Example products would include stereos, televisions, calculators and microwaves. This is not to say that the model is intended for application to this domain only. In the development of the model, the tailoring of the basic framework to different domains has been kept in mind. I can envision a similar system for designing the overall layout of an automobile,

and yet another system designed to create automobile dashboards. The remainder of this section will discuss the rationale for the chosen proof of concept domain.

In the design of consumer products aesthetics and style are critical to success. In many instances, these considerations dominate the overall layout of the product. Industrial design plays an important role in this domain.

The so called functional design of electronics products is embodied primarily by the organization of selected standard components which are interconnected through wiring. Thus, spatial constraints and functional sharing (Ulrich and Seering, 1988) between components is minimal. As the components can be positioned fairly autonomously, electronic products allow the focus of the work to be dedicated to aesthetics, ergonomics, and manufacture. In contrast, the position of components in a domain such as mechanical linkages would be determined purely by spatial interactions. Further, the proposed system uses a catalogue of standard components or sub-assemblies which are the ingredients for new designs. This design approach suits the electronics industry. Some companies now offer computer catalogues of their components. Perhaps some day such catalogues will come complete with the geometric data required by the proposed system.

Injection moulding is the manufacturing process of choice for several reasons. It is highly suited to mass produced products, and yields the high tolerance, high quality surfaces desired for cosmetic parts (Hanada and Liefer, 1989). Also, the process affords a wide range of aesthetic freedom — materials have a relatively low influence on the form of a part. Conversely, the form of sheet metal products is governed by material considerations.

The choice of two use types, *hand held* and *desk top*, serves to illustrate the effect of different ergonomic requirements on the configuration of a product. *Desk top* refers to products which are used while they sit on a desk (e.g. a radio or television). *Hand held* refers to products which are used while being held in *one hand* (such as a television remote control).

Finally, I chose to include components in the system which provide a sufficient range of sensory interaction with the user to permit the design of a variety of reasonably realistic products.

1.6 Overview of the Proposed System

A slightly more detailed diagram illustrating the actions of the proposed system is shown in figure 1.3. The system is shown as an autonomous design tool to emphasize its capabilities. In practice an application which allows both the user and the system to participate in configuring the design is possible.

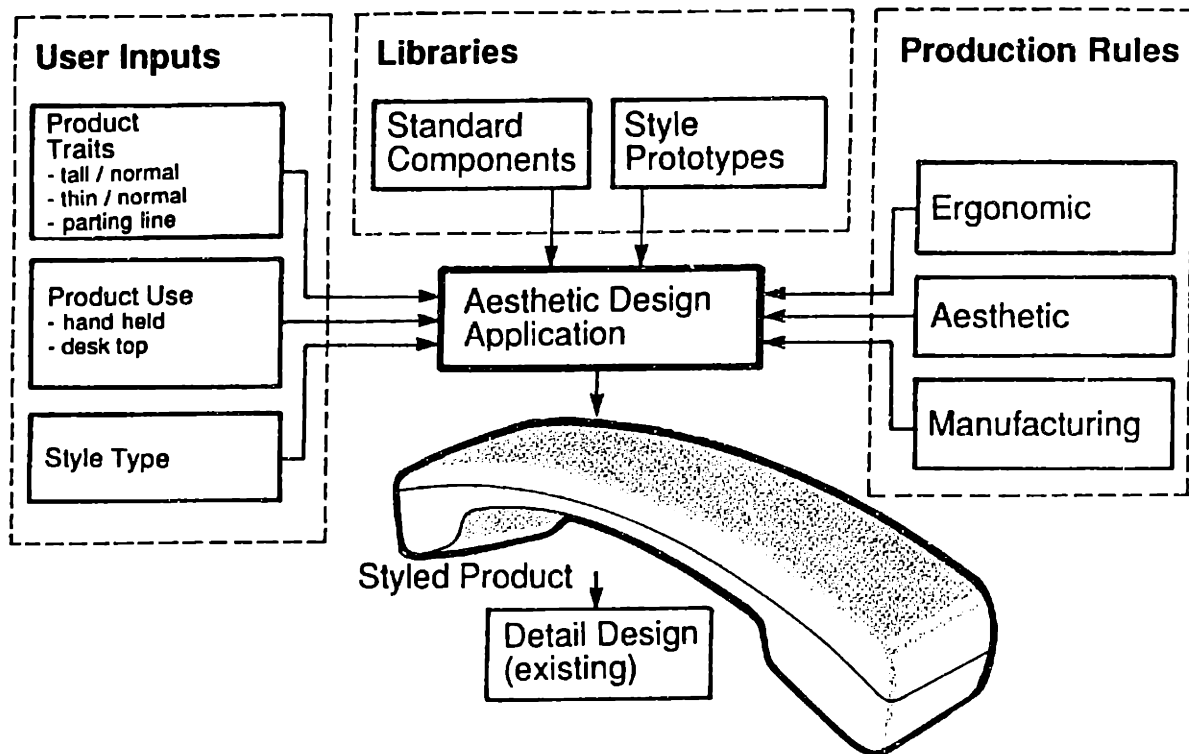


Figure 1.3 Major constituents of the proposed system

In the proof of concept domain, the user first selects the components which are desired in their design from a library. The library includes acoustic components such as speakers, tactile components such as keypads, or visual displays such as a liquid crystal displays. The user also specifies how the mould parting plane is to be oriented, and whether the product is hand held or desk top. The designer also provides qualitative input about physical characteristics such as product thickness or height, and importantly, the desired style or corporate identity. Given this information, the system completes the conceptual design process without further actions on the part of the program user. The system executes the design in a four stage process: locating the components in three dimensional space relative to the mould parting plane; enclosing the component layout in an

appropriately styled surface; adding style specific details to the surface such as panel inserts, buttons, grills, or vents; and applying graphical details such as logos or colouring schemes. The output of the system is graphical images of the product concepts which are analogous to design renderings. Complete geometric surface data and the positions of the components is known to the system.

The automated design process uses manufacturing, ergonomics, aesthetics, and style considerations to derive as many appropriate design alternatives as possible. The system does not employ prototypes or templates of what certain kinds of products look like. For example, the system has no notion of whether it is designing a telephone or a calculator. Instead, the system uses general principles and domain specific standard practice to determine the product layout.

1.7 Role in the Overall Design Process

Figure 1.4 indicates where the proposed system is intended to fit into the design/manufacturing scheme. This figure is derived from Hanada and Liefer (1989).

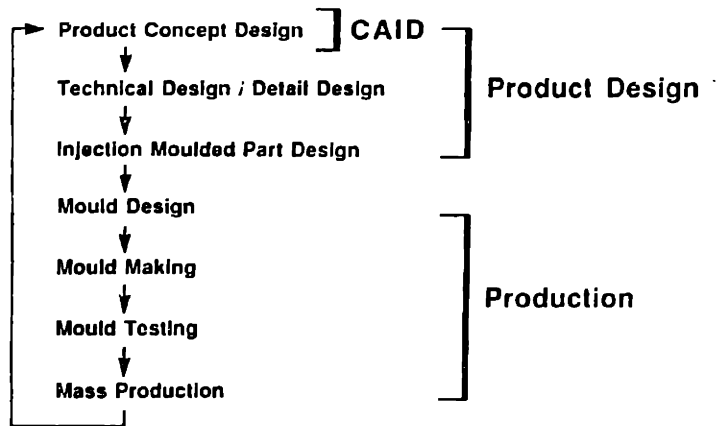


Figure 1.4 Role of the proposed system in the design and manufacture picture

The computer automated industrial design system (CAID) is part of the conceptual design stage. The goal of the system is to propose conceptually correct design layouts — the internal details of the design are incomplete. It is felt that by following simple manufacturing, aesthetic, and ergonomic guidelines at the onset will reduce the likelihood of major design problems in subsequent detail design phases.

After the automated industrial design system produces the product layout and exterior surface, this information is passed to subsequent technical design stages. Internal features such as ribs, bosses, and component attachment surfaces are added to the design in

these later stages. The remainder of the existing design process is unaltered, as shown in the figure.

1.8 Potential Benefits

Thus far, I have identified the proposed system's potential to bridge the gap between engineering and industrial design as an important benefit. There are however, several other compelling reasons for the study of an automated aesthetic design system.

The system can aid productivity as it will help to apply industrial design expertise over many products, and eliminate the repetitive re-design of common product details such as buttons or masks. The industrial designers can focus their creative energy on the design of styles, aesthetic protocols, and ergonomic considerations while the system applies this knowledge to a variety of products.

While an obvious benefit of the system is that it will allow the aesthetically naive to achieve acceptable aesthetic designs, a more significant advantage might be the system's ability to produce multiple product versions. The system can provide a range of plausible designs in an instant. The experienced designer could use the concepts as a starting point. Rapid prototyping would be facilitated as, in addition to producing conceptual designs in a few seconds, the system possesses the complete geometric data needed for numerically controlled production of models or moulds. This could reduce the time consuming iterations between the designer and the pattern/mould maker. The geometry is also available for subsequent design stages and analysis.

Marketing advantages may be accrued through a strong corporate identity enforced by the system, while the occurrence of products with disastrous aesthetics could be prevented. Finally, by considering ergonomics and manufacturing at the onset of the design process, the system should help to prevent downstream production difficulties.

2 BACKGROUND

Preliminary material required to understand decisions made in the development of the model is presented in this chapter. A set of characteristics for products with *good* aesthetics is formed, and methods to capture style in a computer system are both reviewed and suggested. Finally, check lists for manufacturing and ergonomic considerations relevant to the conceptual design of injection moulded consumer electronics products are composed.

2.1 The Key to Aesthetics and Style

2.1.1 Aesthetics

In this section ideas important to achieving desirable aesthetics in product design are discussed. These ideas are used as a basis for the aesthetic considerations built into the automated system.

Tjalve (1979) believes that the appearance of a product is the consequence of the designer's choice of structure, form, material, dimension, and surface treatment. While the judgement of appearance is subjective, aesthetics — which is the study of beauty — tries to identify elements or characteristics of beauty. Through the study of aesthetics, it is possible to identify general characteristics common to attractive products. Although possession of such characteristics does not guarantee beauty, they greatly enhance the probability of the product attaining an acceptable appearance.

Historically, many formalisms (for example, *The Modulor — a harmonious measure to the human scale universally applicable to architecture and mechanics* (Le Corbusier, 1954)) have been developed to systematically design forms. It was common for such methodologies to be rooted in religion. It was, in my opinion, however, the work of gestalt psychologists that paved the way for the establishment of the product design guidelines such as those clearly outlined by Tjalve (1979). In their study of organization and perception, gestalt psychologists introduced the principle of *pragnanz* (Rock and Palmer, 1990, Kubovy and Pomerance, 1981). Simply stated, the principle says that when people perceive objects, they try to find as *good* an interpretation as possible, where *good*

means simple, regular, or symmetric. That is, a *good* perception is one with the least information.

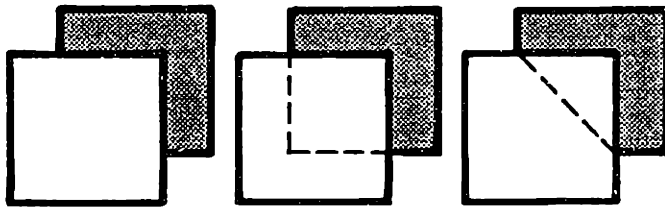


Figure 2.1 Illustration of the idea of pragnanz: ambiguous shapes (left) are perceived as simple forms (centre) over more complex ones (right)

The idea of pragnanz is illustrated in figure 2.1. Related to this principle are the Gestalt laws of grouping, as shown in figure 2.2. Awareness of these principles is helpful in understanding the aesthetic design guidelines provided below.

Characteristics for beauty in product design are identified as *unity* and *order* (Tjalve 1979). *Unity* refers to the combination of elements and details in a *harmonic* way. A *harmonic* arrangement is congruent and pleasing. Unity is achieved through colour, shape, and structure relationships between elements. *Order* implies a systematic or underlying theme to the layout of the product. However, an overly repetitious (or simple) layout may become boring.

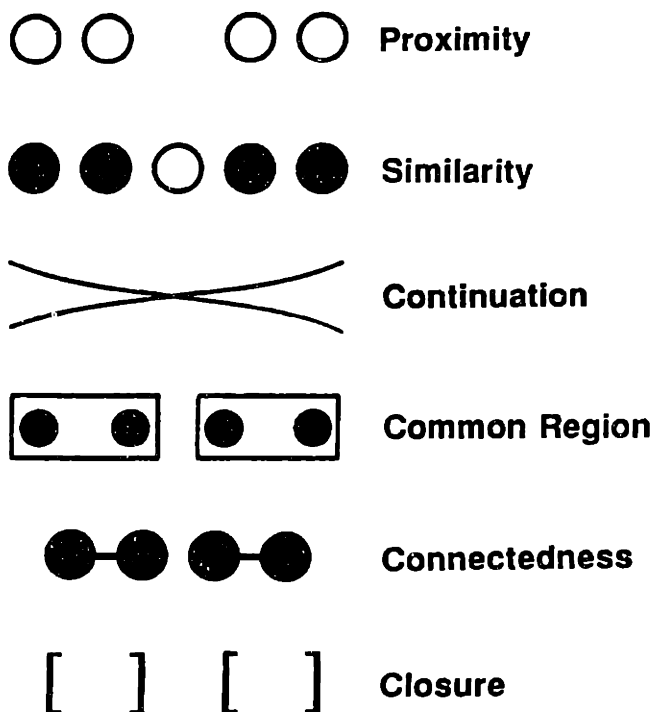


Figure 2.2 The gestalt laws of grouping: (redrawn from Rock and Palmer, 1990)

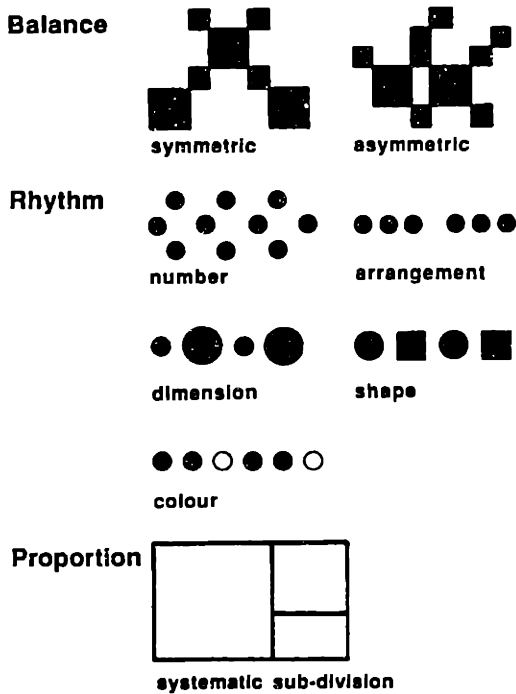


Figure 2.3 Methods to create unity and harmony : (based upon Tjalve, 1979)

Unity or harmony in combined elements can be achieved through the means shown in figure 2.3. Visual balance is obtained, in both symmetric and asymmetric designs, by ensuring that the layout has the same visual weight on both sides of the object's centre line. Visual weight can be manipulated through both form and colour. *Rhythm* refers to the repetition of patterns or themes throughout the product (Lauer, 1979), and can be achieved through variations of arrangement, dimension, number, colour, and the form of the elements. Rhythm can prevent a highly ordered product from becoming boring. Proportion and systematic subdivision and grouping using common ratios can greatly enhance harmony.

The Golden section (the ratio of 5/8 advocated by Leonardo Davinci) historically is the perfect aesthetic, but dimensional ratios of 3/4, 3/5, and 4/5 (suggested by Palladio) are also recommended. For very large objects, the proportion of 1/2 is preferred (Flursheim, 1983). The Bauhaus movement studied the use of such ratios extensively.

Order, or organization, is not independent of the characteristics already explored. Order is achieved through the systematic application of the traits for unity. Although not explicitly mentioned in the references, I feel that the careful and consistent spacing of elements in the product helps to create an ordered appearance.

There are other factors important to how a produced is perceived by the user. Tjalve (1979) believes that we intrinsically perceive vertical and horizontal as principle visual directions. Thus, orthogonal lines and planes are preferred form elements in product design.

Forms can express or imply other characteristics about the product, or even how the product is to be used (form follows function). Figure 2.4 shows how weight, stability, height, and direction or motion can be implied through form or colour.

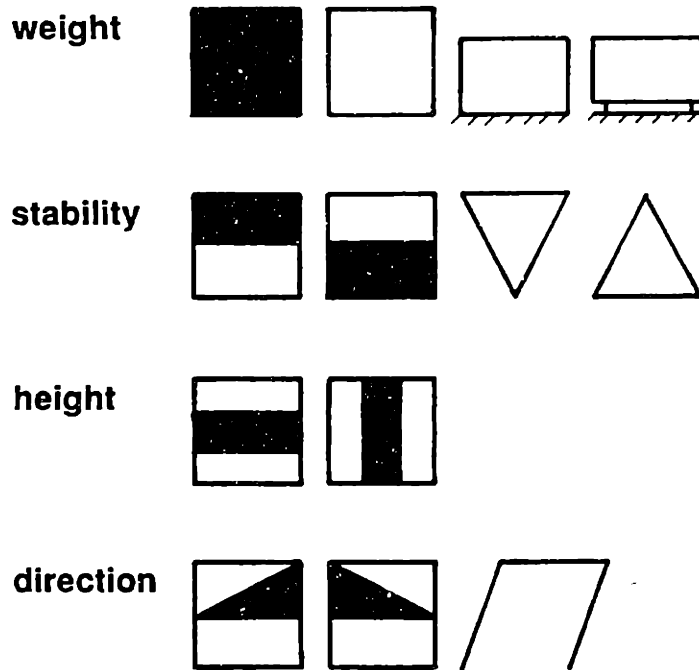


Figure 2.4 Form and perception:
(based upon Tjalve 1979, Flurschein, 1983)

The visual weight of a product can be easily manipulated, as shown in the figure 2.4. For desk top products, it is very common to use a foot as shown prevent the design from looking too chunky or clumsy. Stability is important if the product is to be perceived as balanced or safe. Visual stability may be achieved by making the top half of the product either lighter or smaller than the bottom, and through the careful choice of the overall proportions. Lastly, notions of height, direction, or motion can provide the consumer with insight about how to use the product.

The last aesthetic consideration I wish to discuss is related to the joints in a product where plastic parts meet. High quality 'invisible' joints demand very high mould precision and, for the most part, are difficult to achieve. Thus, it is recommended that products are designed so that the joint is either emphasized or placed in a location where it is not noticeable. One would typically try to place the mould parting line in less visible locations of the product housing.

2.1.2 Style

Douglas Hofstadter (1985) states that stylistic consistency among different letters in a typeface is the by-product of modularity in the roles of elements such as serifs or strokes. In a similar manner, I feel that a style can be maintained over a variety of different products through consistent conventions and forms. In this section, I shall review possible methods to capture style in a computer system, and identify the approach adopted in the proof of concept. There are two general approaches to this problem which I feel are noteworthy: shape or structure grammars; and analogy.

2.1.2.1 Shape and Structure Grammars

Surface and structure grammars are computational formalisms for the representation of shapes and spatial structures. These related methods have been used fairly widely in architecture to develop formal approaches for producing designs in specific styles. Applications in computer programs include their use to design Frank Lloyd Wright style prairie houses (Koning, Eizenburg 1981), Queen Anne houses (Flemming, 1987), Mughul gardens (Stiny, Mitchell 1980), and Hepplewhite style chairs (Knight 1980). They have also been used to layout rectangular floor plans (Flemming, 1989).

The method borrows ideas from formal language theory developed by Chomsky (1956), and was first applied to design by Stiny and Gips (1978) in the form of shape grammars. I will begin by describing shape grammars, and then move on to discuss structure grammars.

2.1.2.1.1 Shape Grammars

In the article *Introduction to shape and shape grammars* Stiny (1980) defines shape as "a limited arrangement of straight lines defined in a Cartesian coordinate system". A shape grammar is made of four different components. First, the grammar must have a finite set of shapes which are the building blocks from which other shapes are to be constructed. Further, a finite set of symbols, and a set of rules to transform one shape and symbol to another is required. In many cases, the symbols may be regarded as markers which indicate where shape rules are to be applied. Finally, the grammar requires an initial

shape from which to start. Figure 2.5 illustrates the components of a very simple shape grammar.

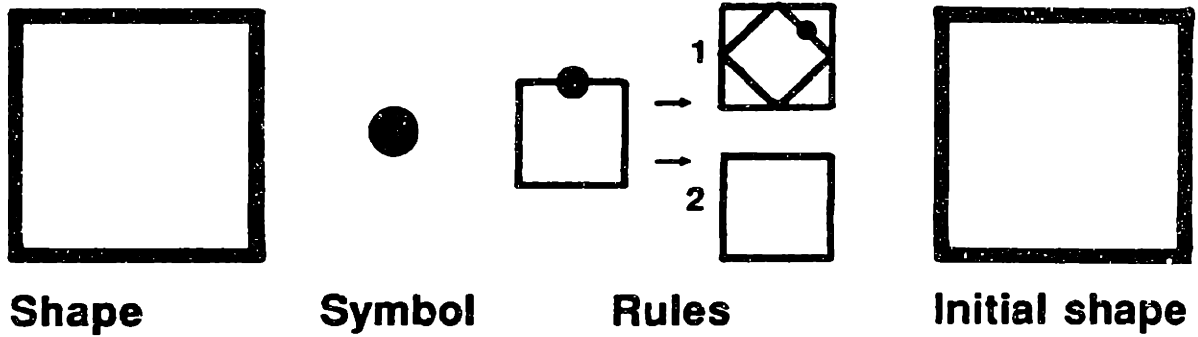


Figure 2.5 Elements of a simple shape grammar: (re-drawn from Stiny, 1980)

The shape grammar builds a language of forms by combining shapes until no symbols remain. An example of a possible outcome of this process, using the elements of figure 2.5, is shown in figure 2.6.

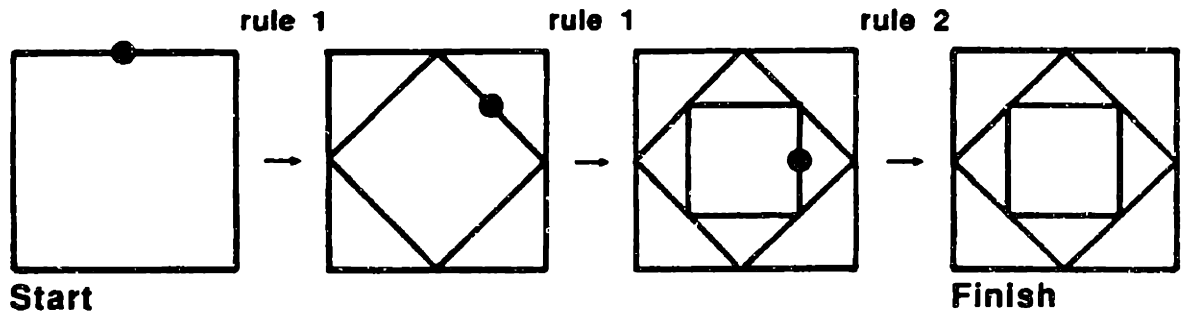


Figure 2.6 Mechanics of a simple shape grammar: (re-drawn from Stiny, 1980)

Shape grammars have been used in two dimensions, but it is not difficult to imagine their application in three dimensions. How a shape grammar might be used to produce styled surfaces for an enclosure is shown in figure 2.7. The shape grammar adds surface corner treatments. The term corner treatment is used to describe where one plane of an enclosure meets another. It is hypothesized that the corner treatment of a surface is a very

important aspect of a product's style. This hypothesis is to be demonstrated in the proof of concept.

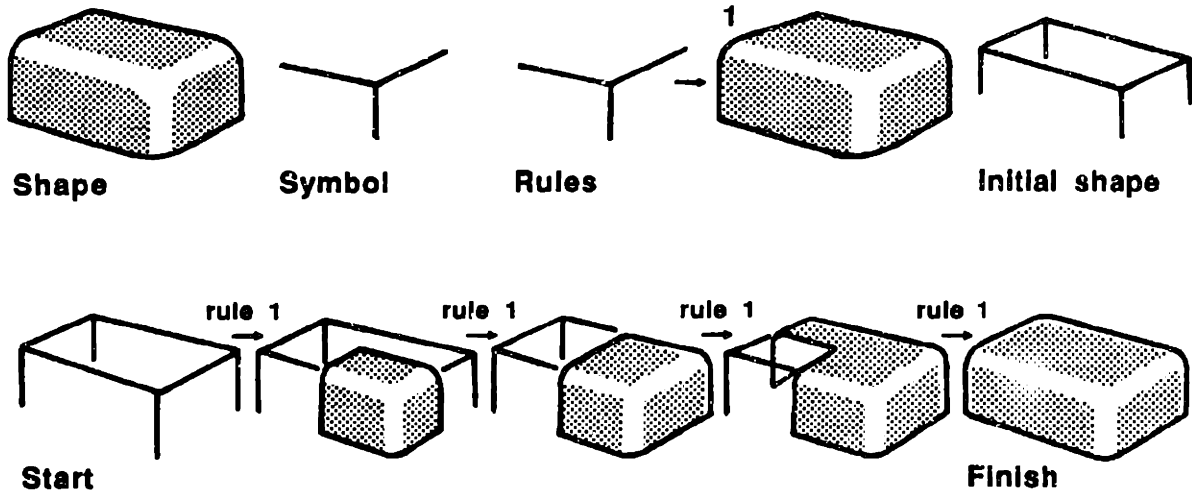


Figure 2.7 Possible application of a shape grammar to design a product enclosure

Attempts to develop shape grammars for surface generation have been undertaken recently (Longenecker and Fitzhorn, 1990). However, the method has serious problems in the context of an automated product design system. Designing and building a shape grammar is not trivial, and thus it will be difficult to add new styles or make refinements. Further, the approach to form synthesis is foreign to the designers who will be required to define the shapes used by the grammar. This would be another impediment to developing a style repertoire for the program. Perhaps most significantly, there is a need to consider the enclosure as a totality. That is, while corner treatments are critical to the surface style, I do not believe that the simplistic systematic incorporation of corner details will be adequate to produce designs with acceptable appearance (the whole is greater than the sum).

2.1.2.1.2 Structure Grammars

Structure grammars are an adaptation of shape grammars. Shape grammars combine shapes to create new forms without *apriori* knowledge of the emergent outcome. Unlike shape grammars, structure grammars specify how a set of primitives is assembled to achieve a predetermined goal (Carlson and Woodbury 1990). For example, a structure grammar can be written to combine elements so that human-like configurations arise (see figure 2.8).

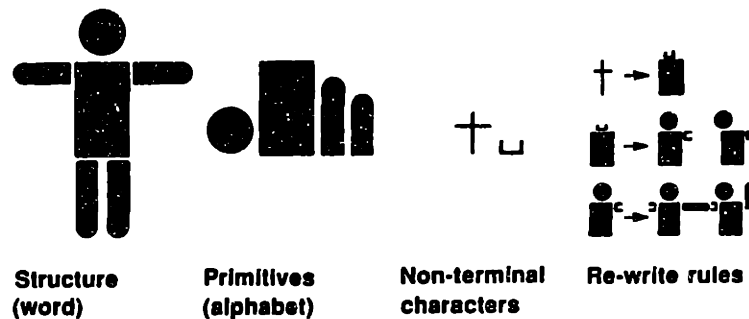


Figure 2.8 Example of a Structure Grammar: (re-drawn from Carlson and Woodbury, 1990)

A good analogy for understanding structure grammars is to imagine the set of primitives as characters in the alphabet. Rules of spelling are used to assemble the letters into meaningful words or structures.

A grammar for laying out a specified type of product — stereos for instance — could be written. The limitation of the method is its specificity. The example shows a grammar designed to create people. A different grammar would be required to make dogs, even though dogs and people are quite similar (both are vertebrates, have four appendages, a body, and a head). Likewise, it is necessary to construct separate grammars for televisions and calculators, even though both are electronic products with moulded housings. An important objective is to develop a system that can design similar products without knowing exactly what they are. This will allow the single system to design televisions, calculators, and new, yet unknown, types of products.

Although structure grammars are not appropriate for the proposed system, they can be used to illustrate that, for a given configuration, dramatic changes in style can arise through a change in primitives.

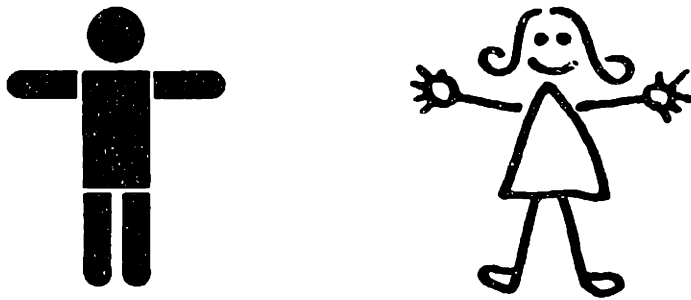


Figure 2.9 How details change style:
(re-drawn from Carlson and Woodbury, 1990)

This fact, exemplified by figure 2.9, is used to justify the simplifying assumption which separates style from product configuration. Styles are reflected in surface design only. The component layout for a product will be the same regardless of style.

While this assumption is motivated primarily to simplify the implementation, it is justifiable from a manufacturing viewpoint as well. Many companies develop standard layouts for their products, relying upon a variety of housing designs to differentiate them for specific market segments.

2.1.2.2 Analogy

Analogies are abstractions which contain information about the situations they represent. The use of analogy eliminates the need for deduction in problem solving. Problems are resolved through comparison to the analogy (Barr, Feigenbaum, eds. 1981).

2.1.2.2.1 Letter Spirit

Douglas Hofstadter *et al.* (1987) used the framework of analogy to devise a program called letter spirit.

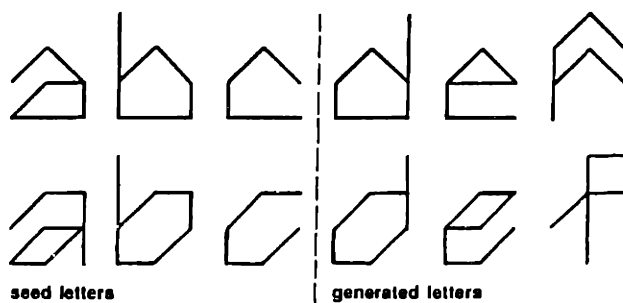


Figure 2.10 Results from letter spirit:
(re-drawn from Douglas Hofstadter
et al., 1987)

This program used three seed letters as inspiration to generate a complete alphabet of consistent style. Examples of the program's output are shown in figure 2.10. They believe letter style is maintained by the analogous transfer of elements in the seed characters to the rest of the alphabet.

In letter spirit, the computer system formulates analogies — it must determine the characteristics of a style — and applies them. Feature recognition constraints limit letter spirit to simple characters defined on a coarse grid, but the results show that through analogy styles can be captured.

2.1.2.2.2 Morphological Analysis

An idea borrowed from biology, called morphological analysis, provides a straightforward method for the design and transfer of product surface style analogies. Morphology refers to the science of possible forms. Morphological analysis was devised by D'arcy Thompson (Bonner ed., 1961) to evaluate the relatedness of animal forms within a zoological class. By superimposing a grid over animal skeletons and performing various transformations he was able to assess evolutionary closeness, as illustrated in figure 2.11.

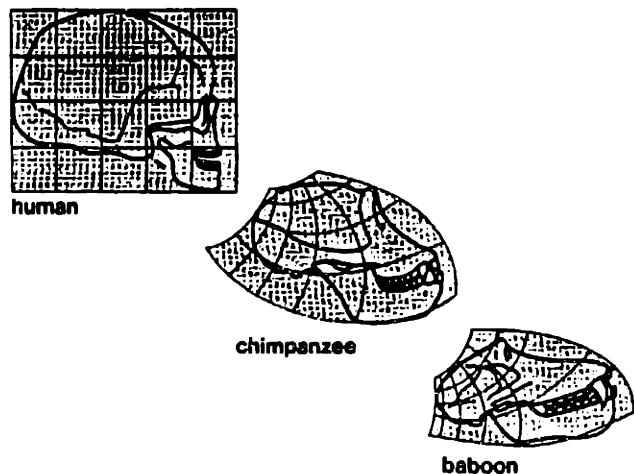


Figure 2.11 An application of morphological analysis: (re-drawn from Bonner ed., 1961)

Morphological analysis separates the unique description of a configuration from the infinite variations possible through scaling or distortion (Steadman, 1983). Each different transformation of the human skull results in a different style of ape skull. Using morphological analysis, a style is encoded with a characteristic surface prototype (an analogy), and a transformation function which maps the prototype to specific products.

The mapping functions must be designed to preserve the style — the skull example shows transformations which alter style. A similar two step process has been applied to generate human caricatures (Brennan, Carey, 1987). The caricature program compares a prototypic face to an individual face (a specific style), and then computes the

transformation function. With the characteristic transformation defined, the computer exaggerates the function to create a caricature.

There is more than one interpretation of how morphological analysis can be applied to the proposed system. The most literal adaptation is to create prototypes which have a direct mapping to the product's enclosure — if the product is six sided, then the prototype must also have six sides. The application of a prototype is reduced to a matter of scaling, as depicted in figure 2.12.

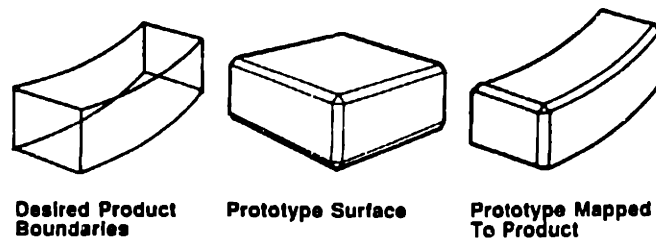


Figure 2.12 Using ideas of morphology to generate an enclosure

This method is simple and intuitive, but would require many enclosure prototypes for each style. Significantly, it permits designers to create prototypes using the traditional design methods of their choice. The prototypes are merely a collection of housing designs. Thus, it should be relatively easy to design a large number of prototypes.

Once a prototype enclosure is defined, the computer system adapts the prototype to suit different sizes of products. Manufacturing considerations (such as draft angles or corner radii) can be incorporated into the prototype enclosure design directly. The mapping process must be executed so that it does not destroy this information. Further, the prototype transformation — which must maintain the style over a wide range of sizes — will probably be non-linear. The visual characteristics of a form will change with size, not unlike physical properties such as strength or stiffness. A chair drawn at 1/8 scale may look quite beefy but, in fact, appear spindly at full scale. Therefore, the scaling of corner details and surface features must be carefully considered. This issue is addressed empirically in the implementation section.

A more general approach could map prototypes to a surface with an arbitrary number of sides. A simple example is shown in figure 2.13.

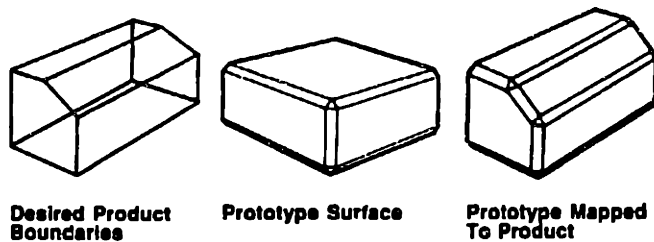


Figure 2.13 General application of the morphology analogy

The energy minimizing surfaces developed by George Celniker (1990) would be an ideal candidate for this approach. The automated system would produce a bounding box — or character lines — for the product. An energy minimizing surface would be ‘draped’ over the product outline to produce the enclosure.

Style would be encoded in the both character lines (which govern the overall configuration) and the design of the surface energy function (which controls surface fairness and corner treatments). A different surface representation that could be employed in a similar manner is outlined by Sederberg and Parry (1986). The general approach derives its versatility by using the transformation functions to create the style defining corner treatments. This approach is less intuitive and would complicate the design of different styles.

2.1.2.3 Hybridization of Surface Styles

The hybridization — or merging — of styles is not central to the thesis, but issues related to the topic shall be discussed for two reasons. It is not uncommon for the industrial designer to be commissioned for designs that combine the image of one company with another. For example, a General Motors designer might be asked to make a distinctly Chevrolet dashboard which reminds the consumer of a Mercedes. Interest in hybridization is also kindled by its relation to creativity. Douglas Hofstadter (1985) states "variations upon a theme is the crux of creativity". This statement is consistent with my own opinion about the design process voiced in the introduction. To a large extent, design is a borrowing or adaptation activity. Thus, the idea of hybridization may be the key to the

development of a truly creative computer system. In this section, it is my intention to outline approaches and barriers to the development of a hybridization system.

When Douglas Hofstadter (1985) considers hybridization, he envisions a black box with knobs that allow the user to adjust the amount of blending between designs. His example indicates that the process of hybridization must be guided. The user must determine the degree of hybridization for best results. This type of evaluation is very difficult for a computer to perform, as it requires sophisticated analogy making capabilities. Consequently, like Douglas Hofstadter, I feel that a literate user will be integral to a hybridization system.

Two different approaches to the style hybridization problem are foreseen: one which combines, blends, or replaces features; and another which blends entire objects. The two possible approaches are shown in figure 2.14.

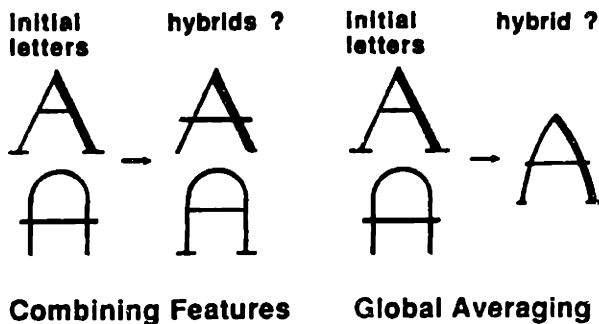


Figure 2.14 Hybridization based upon combining features and on global averaging

The former approach was used by Donald Knuth (1982) to develop a system for generating hybrid letters. This approach requires a joint parameterisation between the objects being hybridized. In other words, the objects must have common features. It is not at all clear how this method would be used to interpolate between two radically different objects, say a letter and an automobile.

This may seem ridiculous, but lettering, automobiles, and buildings all can be designed to have a common look — Art Deco for example. Therefore, hybridizing between an Art Deco letter and a Porsche automobile involves hybridizing between Porsche and Art Deco.

The second approach to hybridization — the physical blending of two objects — circumvents the issue of joint parameterisation, but it is hard to imagine how such a naive approach could work between dissimilar artifacts.

2.2 Manufacturing and Ergonomics

The application of computer aided design systems to improve manufacturability is of ongoing interest in mechanical engineering. In this section it is planned to review the general trends of current systems, and indicate how the goals of the proposed system differ. The simple underlying assumptions about manufacturing and ergonomics in the proof of concept domain will also be outlined in this section. Specific details of how these assumptions are affected by the system, and how they apply to specific components will be discussed during the development of the computer model.

2.2.1 Manufacture

The current trend in design for manufacture computer programs, exemplified by Hanada and Liefer (1989), Duffy and Dixon (1988), or Jakiela (1988), is what I describe as *design fixer uppers*. In this approach, the designer provides a skeletal part based upon functional criteria, and the computer system modifies the design for manufacturability. This process is illustrated in figure 2.15 for an injection moulded piece.

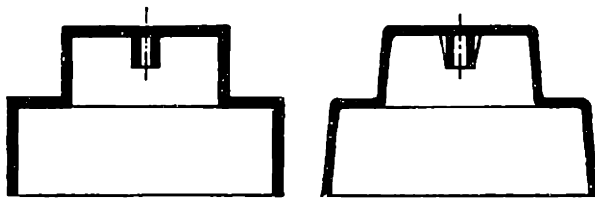


Figure 2.15 Design *fixer upper*: input topology (left) and fixed design (left) (re-drawn from Hanada and Liefer, 1989)

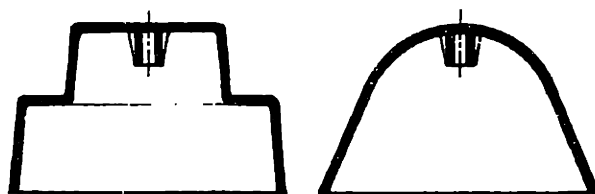


Figure 2.16 The weakness of *fixer uppers*: fixed design (left) and optimal design (right)

Hanada and Liefer (1989) claim that the initial skeleton is designed without regard to manufacturing considerations. I believe that such a statement is fundamentally incorrect. The inherent weakness of the *fixer upper* approach is that, although it will produce makeable parts, it cannot guarantee optimally mouldable designs. The ultimate mouldability of the design is dependent upon the initial skeleton. This has been recognized by researchers working in structural optimization.

Papalambros and Chirehdast (1990) state that "the initial choice of topology greatly restricts the achievable optimality of the designs, and is mostly based upon the designer's

experience". The example of figure 2.16 illustrates this concept in the injection moulding domain. If the designer can provide a skeleton that permits the dome-like design, the result is more mouldable.

The proposed system will consider manufacture in the initial stages of conceptual design. The goal is to produce products configured such that, when detail design occurs, a highly manufacturable part will be realized. The program is not intended to create part details such as wall thickness or bosses (as provided by the fix up programs). This objective differs from the more common analytically based design by manufacturability approach (see Grosse (1989), whose system considers mould flow, for example). The system shall strive towards designing an outer shape conducive to injection moulding. The program's objectives are analogous to the goals of the industrial designer. The industrial designer follows general manufacturability guidelines to avoid problems downstream in the detail design process. They are not concerned with detailed mould design.

Some injection moulding manufacturing considerations important at the initial concept design stage are listed below. This subset of guidelines is developed from Beall (1983), Postings (1985), and Boothroyd and Dewhurst (1983). The list is used as a basis from which specific product configuration rules are formed in the proof of concept. Specifics of these rules are discussed in chapter 4.

- aim for simple mouldings
- design generous draft angles to aid extraction
- avoid undercuts
- design for a flat parting plane
- design generous corner radii to avoid flow problems
- avoid very large flat surfaces to reduce warpage
- design generous draw or cavity to avoid warpage
- avoid side holes to reduce mould complexity
- design for easy flash removal (tumbling)
- design parting lines to avoid unnecessary tolerances
- avoid closely spaced oblique holes to limit freeze marks
- avoid depressed or raised features on curved surfaces

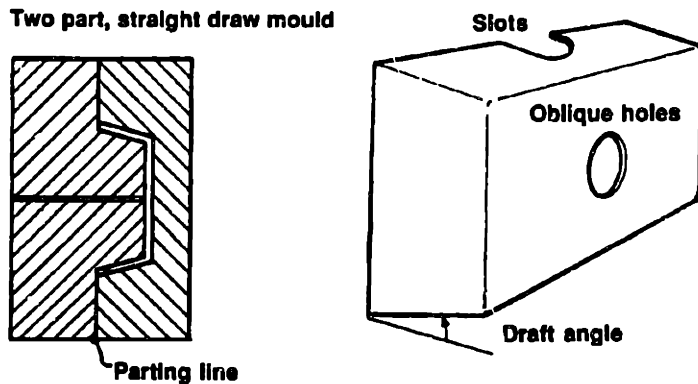


Figure 2.17 A typical part from a straight draw mould with no side holes

In the proof of concept domain the system is limited to housings which can be made in a straight draw two part injection mould with no side holes. All housings will have a straight parting line. A simplified sketch of this type of mould and a typical part from such a mould is provided in figure 2.17.

2.2.2 Ergonomics

Like the manufacturing considerations, the program is intended to consider ergonomics at a high — so called 'common sense' — level. This differs from expert programmes which address specific ergonomic problems in great detail. For example, the system is not intended to configure individual buttons — it will receive multiple button keypads and control panels as pre-designed entities. The system must be able to layout the product so that the keypad assemblies are usable.

Ergonomic issues play a large role in the spatial configuration of consumer electronics products, and unfortunately, simple ergonomic guidelines are often overlooked. The ergonomic texts compiled by McCormick and Sanders (1982) and Woodson (1981) were used as references while formulating the guidelines below.

- design so components are not obscured during use
- hand held objects must be easy to grasp
- organize components into functional groupings to avoid confusion
- design the product so that it is stable during operation
- locate components so that they are appropriately accessible
- avoid cluttering the primary surface of interaction

- frequently used components belong on primary product surfaces, less frequently used on secondary surfaces
- products should be configured to avoid uncomfortable use positions

While this list seems to be almost ridiculously simple, it cannot be over-looked during conceptual design if useful products are to result. The application of the ergonomic guidelines to the proposed system is discussed in chapter 4.

3 MODEL FOR FORM GIVING IN INDUSTRIAL DESIGN

3.1 The Overall Structure

Pivotal to the proposed system is the design of a unified model for industrial design form giving. The representation must lend itself to computerization and divide the process into manageable sub-tasks. In this chapter, I first propose such a model, and then proceed to discuss particulars in detail. Previous work is cited when appropriate. It is attempted, as far as possible, to present the model in a non-domain specific fashion. The details of the domain specific implementation are presented in chapter 4.

Since the middle seventies, there have been several attempts to develop computer models for design and criticism in the arts. A discussion of related philosophical issues and attempts — such as the rule based drawing program *Aaron* — may be found in *The Creative Mind* (Boden, 1991). One of the pioneering efforts was forwarded by Stiny and Gips (1978).

While it is not felt that the details of their model are appropriate for industrial design, the high level structure shown in figure 3.1 is a reasonable starting point for the proposed system.



Figure 3.1 General model for the proposed system:
(based upon Stiny and Gips, 1978)

Simply stated, the receptor is an input device. The receptor translates real world initial conditions into a form useable by the aesthetic system. In this thesis, the receptor is a user interface which allows the designer to select components and define product attributes (such as style, height, thickness, use, or parting plane orientation). The receptor locates and incorporates the data necessary to determine how the components should be positioned. This information includes component geometry and attributes. Generally, this will also include global geometric product constraints. For example, interface requirements with other assemblies would limit the global geometry. In this study such restrictions were ignored for simplicity. The effector transforms the output of the aesthetic system into a form useful to the world environment. Output might include geometric data for analysis,

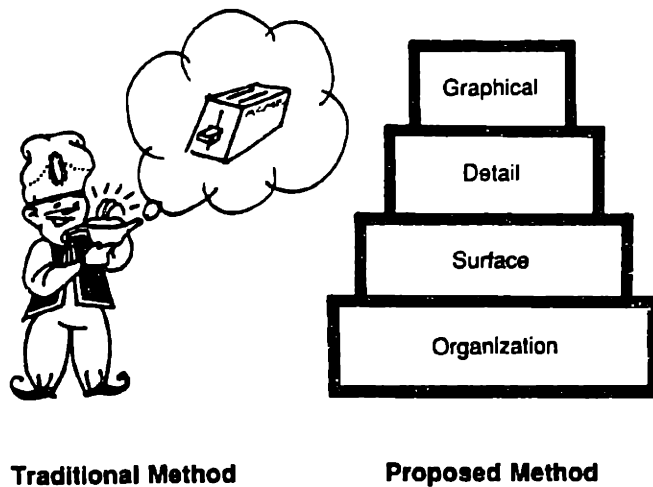
detail design, or computer controlled rapid prototyping machinery. In the proof of concept, the effector interacts with the user through three dimensional computer graphics renderings.

The aesthetic system uses the initial conditions provided by the receptor to produce valid product interpretations. In the proposed system, this entails the synthesis of either single or multiple product concepts. The number of versions generated will depend upon the initial conditions. The validity of interpretations is to be derived through ergonomic, manufacturing, style, and aesthetic considerations. An evaluation mechanism which compares the concepts against a measure of acceptability will guarantee the results. In the proof of concept, however, the issue of evaluation is avoided. The simplification allows the research to focus on the synthesis of product forms. Consequently, the system will always propose a design. The designs are created using methods which should generate — but cannot ensure — good results.

The implementation of the receptor and the effector is relatively routine, while the constituents of the aesthetic system are yet undefined. The remainder of this chapter shall be devoted to developing a model for the aesthetic system, and identifying the model's demands on the receptor.

3.2 Four Level Model for Industrial Design Form Giving.

The process of form giving is defined as indicated in figure 3.2.



The model is contrived to discretize the ill-defined and mystical form-giving process into hierarchical problems. The actions associated with each of the four levels of the model are shown in figure 3.3.

Figure 3.2 The four level hierarchy for form giving

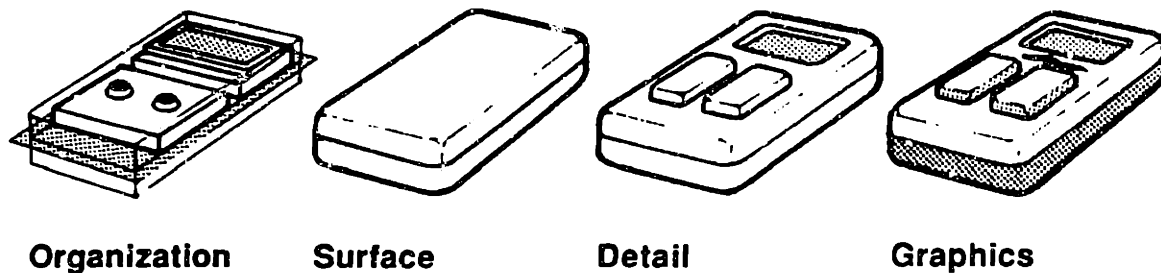


Figure 3.3 Example of the stages in the four level model

The first stage of the model, the organization or structure level, spatially positions components relative to each other. This step provides an organized groundwork upon which the exterior of products are constructed. The layout should reflect manufacturing, ergonomic, and aesthetic aspects such as proportion or composition. The organization level is style independent. It is assumed that the component positions are style invariant.

The second phase of the model generates a housing over the component assembly. The product style, or corporate identity, and manufacturability are particularly important at this level.

The detail level completes the product surface. Additions are made subject to manufacturing considerations and the requirements of components in the product. For example, buttons are added to keypads, and cooling vents are installed over heat liberating components. The details are very important in determining the style.

Lastly, graphical surface embellishments complete the form giving process. The graphical level includes colouring, surface textures, text, logos, and hot stampings. Issues related to style and aesthetics are considered to be dominant. Graphical additions are tempered by the chosen style and user preferences.

The four level process for form giving arose through personal observations. Tests using a series of different telephones were integral to the development of the model. The case study used a 'reverse design' process on styles of commercial telephones to extract configuration rules and procedures. After writing the rules, they were passed to a non-industrial designer (Mark Jakiela) who emulated a computer by blindly following the instructions. The exercise resulted in styles of telephones similar to the original products. Generalization of the case study led to the four level approach. Notably, the model proceeds in an order that is the reverse of how an industrial designer often works. Typically, the designer will start with a holistic idea — or gestalt — for the product, and then proceed to organize the components to match the idea. It is not immediately obvious that the model, which works from the inside out, will be able to emulate this form giving process. One of the goals of the thesis is to demonstrate that this approach is viable — provided the system is limited to a consistent product domain. By considering only consumer electronics products, the rules used by the model can be designed to reflect standard practice. In other words, a very general schema (or plan) for the class of products designed by the system is encoded in the rules (in the proof of concept domain, the product designs are assumed to be box-like).

With the model for form giving introduced, the following sections proceed to detail the requirements of the system receptor and the four levels of the aesthetic system.

3.3 Requirements of the Receptor and the Product Structure

The receptor receives world design constraints, and transforms the said information into initial conditions for aesthetic system. The initial conditions are incorporated into the product structure. The product is represented as an object with slots for data. An example of the representation is illustrated in figure 3.4. All italicized items are initial conditions for the aesthetic system, and must be filled by the receptor mechanism. The initial conditions are considered as a minimal set of data to be provided by the designer. The structure shown is specific to the proof of concept, but in the remainder of this section the product slots will be discussed from a more general standpoint.

The product requirements are initial conditions corresponding to manufacturing, ergonomics, aesthetics, and style. In a more general system, the requirements might include information such as global geometric constraints, cost estimates, and functional specifications.

Requirements:	parting line	<i>vertical</i> <i>horizontal</i>
	use	<i>desk top</i> <i>hand held</i>
	traits	<i>thickness</i> <i>height</i>
	style	<i>Braun-like</i> <i>Art Deco</i> <i>High-Tech</i>
	demographics	<i>professional</i> <i>sport</i>
Components:	component 1	<i>identity</i> <i>position</i> <i>orientation</i>
	component 2	. . .
	:	
Surface:	bounding hull enclosure	geometry geometry colour
Details:	detail 1	associated components position orientation colour
	detail 2	. . .
	:	

Figure 3.4 The product structure slots

Manufacturing initial conditions specify the desired process, materials, and process specific information. In this thesis, injection moulding is assumed, material specifications are ignored, and process specific information is limited to the mould parting plane orientation. The mould parting plane may be either vertical or horizontal. The use requirements provide the system with the information needed for ergonomic classification. In this case, the overall product classifications of hand held and desk top are allowed. Other systems could require more specific data about product use, and perhaps even ask for the products' identity. The product traits, style, and demographics all influence the visual

appearance of the product. The small list of options in the figure are given simply to illustrate some possibilities.

The remaining user inputs identify the components to be included in the product. Given this, the receptor locates the necessary information (attributes and geometry) to describe each component. This implies that a library of components is available to the system. Details about the components in the library are covered in section 3.4.3.1.3.

Given a complete set of initial conditions, it is the task of the organization level of the aesthetic system to fill the component position and orientation slots. The organization level also produces a bounding box for the product. The surface level generates the product enclosure geometry using the bounding box as a guideline for the overall size. The detail level considers the components in the product, and both modifies the surface geometry and adds additional elements as required. Finally, the graphics level fills the surface and detail colour slots.

3.4 Requirements of the Organization Level

3.4.1 Overview and Background

The function of the organization level is to spatially position the components in a product. The input to and output from the organization level is illustrated in figure 3.5. The component configuration is important because the layout determines a product's semantics - particularly in the proof of concept domain. For example, a collection of speakers and other electronic components is interpreted as stereo only when assembled in an appropriate fashion. Consequently, I feel that the product organization is the most critical stage in the model for form giving.

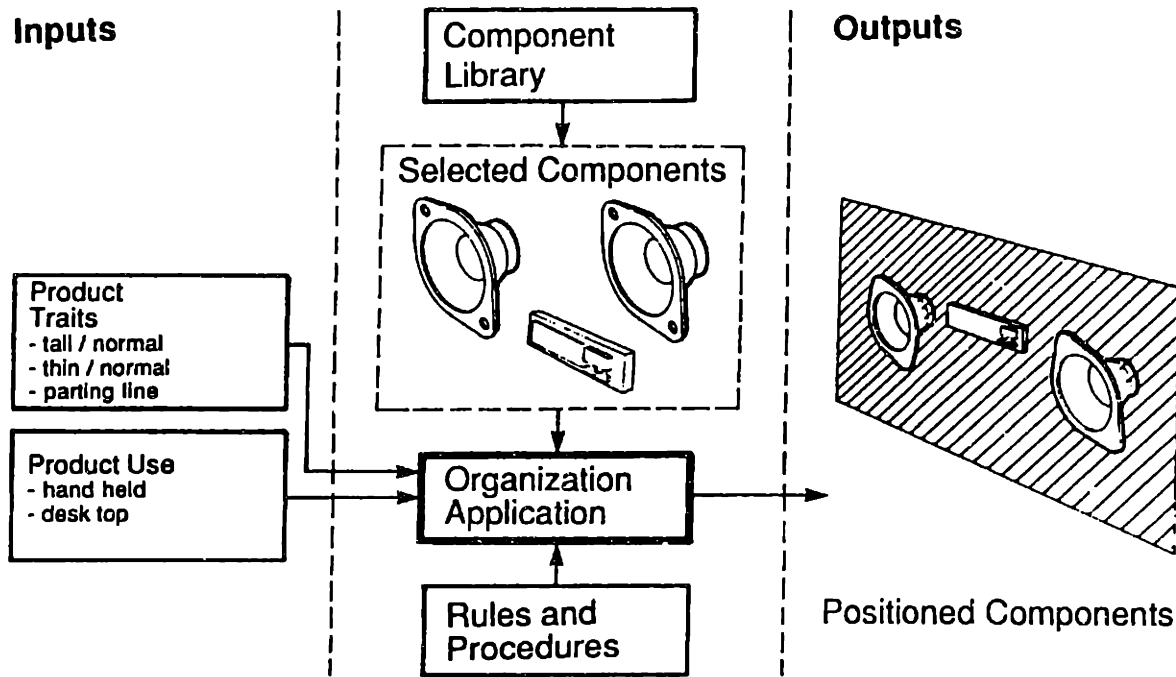


Figure 3.5 Elements of the organization level

Before discussing the approach adopted, it is instructive to highlight prior solutions to the configuration problem. Probably the oldest systematic methods for developing the overall product configuration are the so called combinatorial methods (Tjalve, 1979).

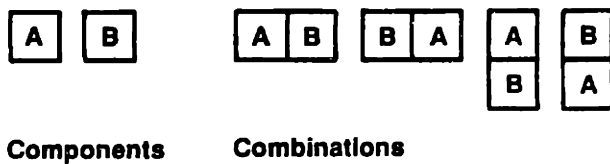


Figure 3.6 Combinatorial approach to configuration design

In this approach, as illustrated in figure 3.6, the components are labelled and all possible variations are tested. Even ignoring combinatorial explosion, blind use of this model is not appropriate for computerization. The method relies upon an evaluation process to filter out poor designs.

While humans are quite good at this, it would be a very difficult task to implement in a computer. I can imagine programming a computer to make pizza, but cannot foresee the program determining which recipe tastes the best.

Another approach to the component packaging problem was developed by Kim and Gossard (1989). They automated the process of positioning components in a restricted design space while optimally satisfying spatial relationships between parts. While related to the topic of this thesis, their work tackles the layout problem from an opposite direction. The goal of their system is to fit an assembly within a pre-defined enclosure space — the goal of this thesis is to *produce* an enclosure given the unassembled internal components. In reality, a practical system should lie somewhere between the two. In most design problems, it will be necessary to synthesize a product housing subject to various geometric constraints (such as interfaces with other products or elements). The proof of concept domain is limited to one without rigid spatial restrictions between either the individual components or the housing and its environment. This limitation allows the work to focus upon its primary objective (being the automation of aesthetics and style in product design).

Expert systems have been applied to various configuration problems. The program R1, which was developed for Digital Equipment Corporation (McDermott, 1980a, 1980b, 1981), might be considered a prototypic example of such an application. This program features a production system architecture. Production systems use data sensitive unordered rules rather than sequenced procedures. The representation is useful when the domain knowledge lacks unifying theory, and can be formulated as heuristic rules of thumb (Brownston *et. al.*, 1985). The systems are very modifiable because the rules are separate from the control structure of the program.

The proposed aesthetic system will not remain static once created, thus adaptability is essential. Further, much of the industrial design domain knowledge can be represented as rules of thumb. Therefore, the production system model is used as a basis for configuring components in the organization level.

3.4.2 Production System Architecture

The architecture of production systems is very briefly discussed in this section. A more complete discussion may be found in Brownston *et. al.* (1985) and Hayes-Roth *et. al.* (1983). A simplified diagram of the production system architecture is provided in figure 3.7.

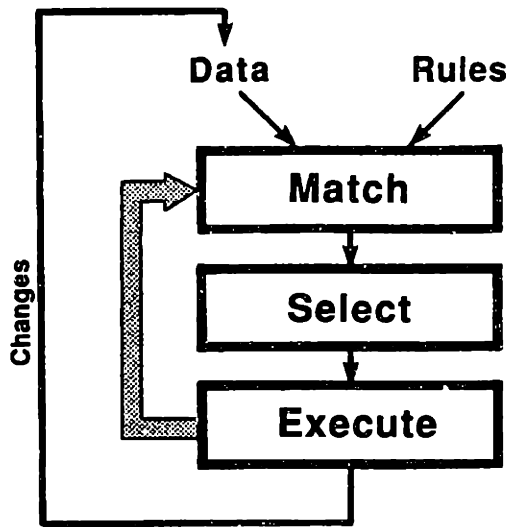


Figure 3.7 Simplified production system architecture

The data consists of information contained within the product structure — initial conditions plus any information in other product slots. Rules contain left hand side conditions (or antecedents) and right hand side consequences. The conditions (if) determine applicability, and the action (then) alters the data. The rules do not reference each other. In forward chaining systems the left hand side specifies a combination of facts or situations which are matched against the data memory. The task of *matching* applicable rules with the data, *selecting* which rule of the set to apply, and *executing* rules is performed by the inference engine.

The inference engine re-evaluates the data continually until applicable rules cannot be found. The control flow is indicated by the shaded arrow in the figure 3.7.

Before progressing to address the specific architecture of the organization level, it should be noted that production systems have the potential to be difficult. Problems can arise through the obscure form of program control. It is often very difficult to understand what the system is doing. This can result in slow operation, complicated debugging, and problematic unforeseen rule interactions.

3.4.3 Organization of the Organization Level

The organization of the components is in itself a two stage process. A rule-based system first fills an intermediate structure, called the *product matrix*, based solely upon ergonomic, manufacturing, and standard practice criteria. The production system uses component and product attributes to decide how to fill the matrix. The product matrix is formally introduced in section 3.4.3.1.1. Once the product matrix is complete, general aesthetic procedures spatially position the components and define the product volume. A high level representation of the approach is depicted in figure 3.8.

The two part design of the organization level emphasizes manufacturing and ergonomics prior to aesthetics.

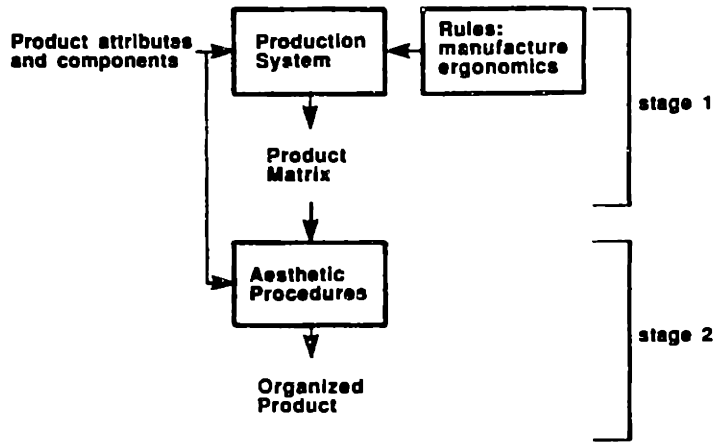


Figure 3.8 The two stages of the organization level

This reinforces the notion that good aesthetics must be achieved without sacrificing product usability or producibility. Manufacturing and ergonomic factors can be identified as heuristics which can be written as individual rules. Therefore, the first stage of the organization level has a rule-based architecture.

The aesthetic concerns addressed in the second stage are more conceptual. This makes the articulation of rules difficult. For instance, the concept of unity cannot be expressed as a single rule. Further, the order of the aesthetic positioning events appears to be important. Unlike the manufacturing and ergonomic criteria, the aesthetic positioning process is most suited to a procedural architecture. It should be noted, however, that after gaining the experience of implementing the model, a slightly different structure for the second stage is recommended.

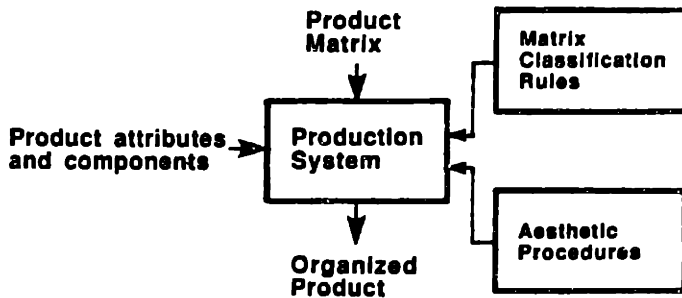


Figure 3.9 Improved second stage of the organization level

Figure 3.9 shows an architecture that uses a rule-based approach to determine the most appropriate set of aesthetic procedures for a given product. The improved model has not been implemented.

3.4.3.1 Stage 1: Conceptually Blocking Out the Product

A pictorial representation of the first stage in the organization level is shown in figure 3.10. As illustrated, ergonomic and manufacturing rules are used to determine the placement of components in the product matrix.

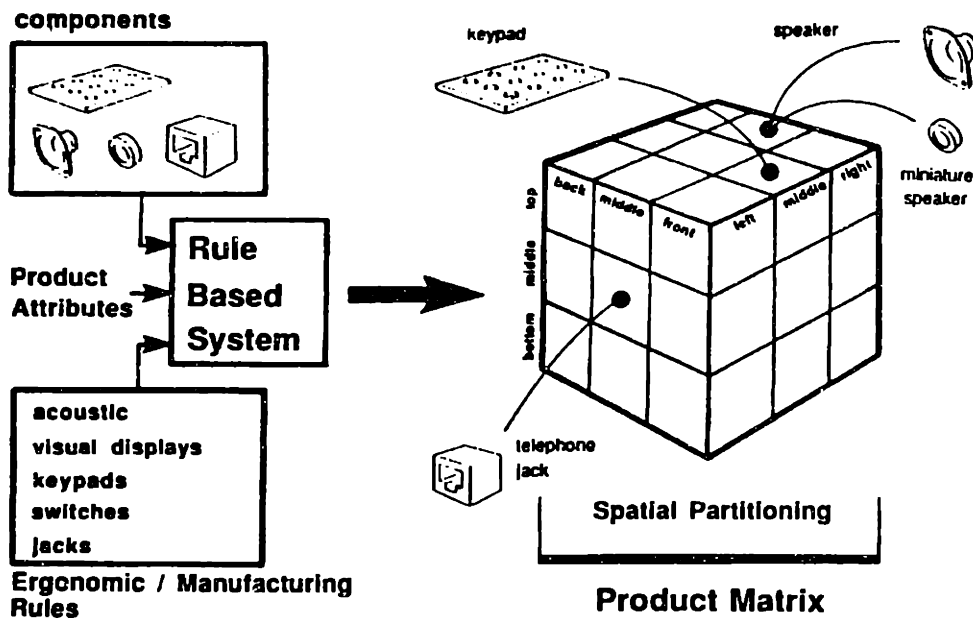


Figure 3.10 Activities of the first stage of the organization level

The placement process is geared towards deciding where components belong in the product conceptually. This task is accomplished without worrying about product dimensions or the physical location of components. The approach allows the system to approximately "flesh out" the product before making detailed refinements, just as a human designer does. The process is made possible by the product matrix. In the following sections, the constituents of the figure are explained.

3.4.3.1.1 The Product Matrix

The product matrix is a spatial partitioning tool which can be visualized as the rubic's cube structure shown in figure 3.11.

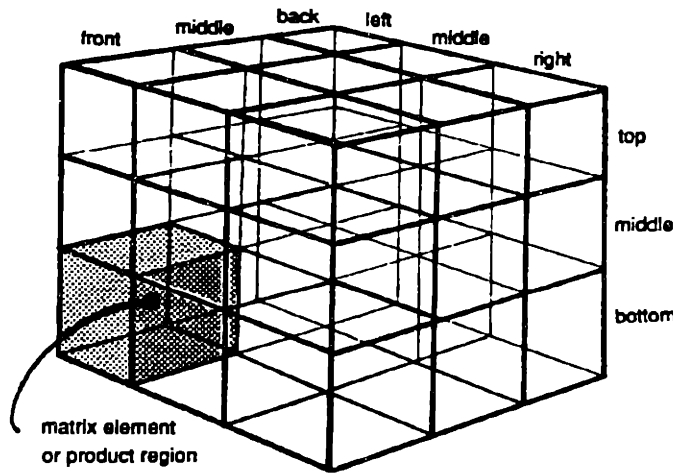


Figure 3.11 The concept of the product matrix

Each individual cube, or element of the matrix, corresponds to a conceptual region of the product. For example, the shaded matrix element (1,1,3) corresponds to the left front bottom. Filling the matrix involves determining which element each component belongs in. The components are oriented within their assigned element to face appropriate directions.

As filling the matrix is concerned with conceptual location only (not physical location), several objects can be placed in a single element. For example, the two acoustic components shown in figure 3.10 both reside in the right middle top of the matrix. The completed matrix provides an outline indicating where each component belongs in the product. This outline is used as a flexible guide during the physical placement process that occurs in the second stage of the organization level.

The product matrix has sixty four outward facing conceptual regions — nine on each face of the product. How was this number of elements arrived at? The precision of the matrix is intended to match the level of discrimination used by people when they describe consumer products. Observations indicate that the readily identifiable regions are limited to the top, front, back, left, right, etc. For example, people describe the disc drive of a computer as being in the front right corner of the machine, or that the on/off button is in the back left. I decided that a 3 x 3 x 3 matrix would allow a similar level of discrimination. In a more complex product domain a larger matrix may be desired, but the use of a matrix divided too finely will begin to specify where components belong precisely

and defeat its purpose. The vagueness of the matrix is intended to create the positioning freedom needed by the aesthetic procedures.

The placement of components determines the essence of the product. A portable stereo typically has speakers facing the front on both the left and right, while the cassette mechanism is located in the middle between the speakers. It is felt that general ergonomic, manufacturing, and domain specific standard practice provide a sufficient range of criteria for the system to devise semantically correct product matrices. Semantically correct matrices will produce products which the user can understand or recognize. As indicated in figure 3.10, the matrix is filled using a rule-based approach. Issues related to writing the rules are considered in the following section.

3.4.3.1.2 Rules for Filling the Product Matrix

The production rules determine the placement and orientation of individual components in the product matrix. While the placement is based primarily upon manufacturing and ergonomics, there are instances when these considerations are insufficient to locate a component. Standard practice is invoked to address this problem. For example, consider a power plug in a desk top product. The power plug is used infrequently, and thus ergonomic considerations suggest that it belongs in the back of the product facing the rear. If the mould parting plane is horizontal, manufacture dictates that the plug is placed in the middle corresponding to the location of the parting plane. In doing so, a side hole is avoided. This type of side hole is undesirable as it requires an actuated pin in the mould. Manufacturability and ergonomics have dictated that the component should be located in the back middle of the product matrix, but cannot determine if it belongs in the left, right, or centre of the product. However, power plugs are commonly placed on the left side. Standard practice is used to fully locate the component in the matrix.

Clearly, the issue of how the system recognizes components and the writing of rules must be addressed. Such concerns lead to the issue of component taxonomy.

3.4.3.1.3 Component Taxonomy and Requirements

Writing specific rules for every single component is unreasonable. The development of a general component taxonomy is prerequisite to a viable system. When

developing a taxonomy, it is desired to define groupings or classes of components that are subject to similar matrix placement. Then, rules can be written for classes of components, rather than individual components. As a result, the total number of rules required for the system can be greatly reduced. In the consumer electronics domain, the components are treated as autonomous black boxes differentiated through ergonomic interactions with the user. In this case, a taxonomy based on ergonomics is the most appropriate. Rules are required for categories of components such as visual displays or acoustic devices. Each component in the library is structured with attributes which reflect the taxonomy, as indicated in figure 3.12. The figure includes characteristics representative of the components used in the proof of concept.

Attributes:	identity	<i>name of component</i>
	classification	<i>visual displays</i> <i>acoustic</i> <i>keypads</i> <i>jacks</i> <i>switches</i> <i>mechanism</i>
	features:	<i>buttons</i> <i>vents</i> <i>mask</i>
	special	<i>produces heat</i> <i>type</i> <i>etc.</i>
Data:	<i>surface geometry</i> <i>orientation</i>	

Figure 3.12 The component structure

The component identity or name is required so that the system can add the correct components to the product. When filling the product matrix, however, the system needs only to know the classification of the component and any special characteristics that might differentiate the component within its class. Therefore, very small class specific rule bases can be written. The features attribute is used in the detail level of the model. The component's surface geometry, and information about its orientation must be provided also. The system must know the face of the component with which the user interacts. Other information, such as the button locations in keypads, is encoded in the component data.

While most components fit well into the chosen taxonomy, the classification of mechanical components is problematic. In the proof of concept, different rules bases are required for each type of mechanism. As consumer electronics products have few mechanisms, this is not a great shortcoming, but this issue will demand attention in other design domains. With the classification of components addressed, it is now possible to consider the rules used in the production system.

3.4.3.1.4 Rules and the Product Matrix

The rules for filling the product matrix all have the general form

if (conditions) then (put the component in a specific plane of the matrix)

or

if (conditions) then (orient the component to face a specific direction)

The conditions are boolean expressions made from comparisons to either component or product attributes. Component class or special attributes are considered. Product attributes such as use or the parting plane orientation are important also. The rules perform one of two actions. The first action places a pointer to the component in the product matrix. An example of a matrix plane is the entire left face. A component is positioned when it is located in three planes. The second action transforms the component geometry data so that the component is oriented correctly. For example, a keypad in the top of a product must be oriented so that the buttons face up. The necessary rotations can be determined using robotics formulate for inverse kinematics (Wolovich, 1986, Hearn and Baker, 1986).

Three types of rules are identified: unconditional rules, conditional rules, and conditional either-or rules. Unconditional rules are those with no antecedents within a component class. This type of rule is independent of product attributes. For example, all keypads are placed on the top of products for easy access. Conditional rules are applicable depending on product attributes and the current state of the product data. Only one component location is possible in a conditional rule. Conditional either-or rules are like conditional rules, but have more than one acceptable matrix placement. The either-or rules are used to create product layout variations.

3.4.3.2 Stage 2: Aesthetic Positioning of the Components

The second phase of the organization level positions the components spatially and defines the overall shape of the product. The system must propose an aesthetically pleasing layout while resolving physical interferences and reflecting product attributes. The aesthetic positioning process completes the organization level of the model. The input to and output from the second stage is illustrated in figure 3.13.

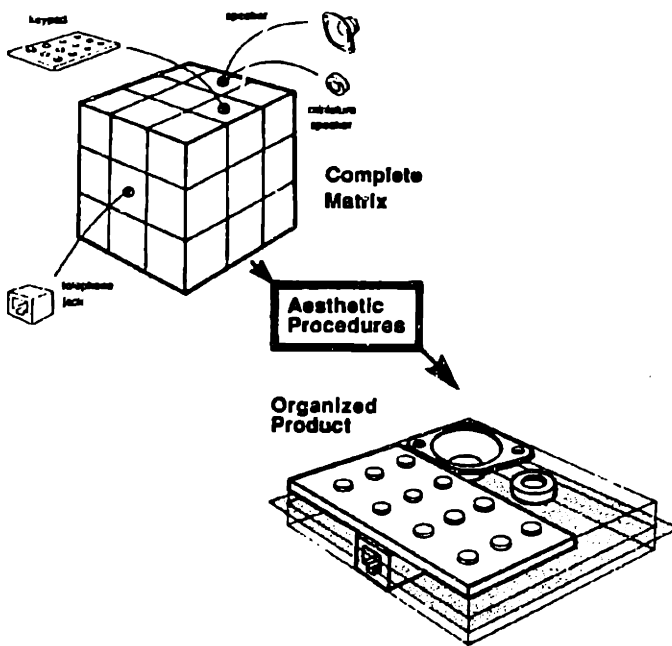


Figure 3.13 Inputs and outputs of the aesthetic positioning stage

The component positioning process is akin to formatting text on a word processor. The main difference is that the computer, not the user, must select the most appropriate formatting method. In order to do this, the system must have some knowledge of the overall shape it is trying to achieve. In the proof of concept, all the products are assumed to be box-like. The system tries to produce the best configuration for a box-like design which matches the product attributes.

In a more general program, the desired shape of the product and other factors, such as world interfaces, will influence the strategies chosen for positioning the components. Once the general approach to the problem is decided upon, the system can proceed to position the components, blindly enforcing the chosen formatting procedures. Details of how this process is implemented are presented in chapter 4.

In light of the aesthetic guidelines formalized in chapter 2, it is felt that the four characteristics of stability, balance, rhythm, and organization — shown in figure 3.14 — are of primary importance when positioning the components. The aesthetic procedures used by the system must be designed to achieve these characteristics.



Figure 3.14 Aesthetic concerns in component positioning

One can achieve stability through a low visual centre of gravity, rhythm through systematic ordering, and organization through consistent spacing and grouping of components (the elements of the product matrix provide a useful way to determine component groupings). Visual balance is important, but does not necessarily mean symmetry. Symmetry might be described as a poor man's aesthetic (meaning anyone can use symmetry to obtain an acceptable layout). However, it is not possible to cast all products into symmetric forms. Symmetry cannot be used exclusively. Further, a system which designs only symmetric products would become stale quickly.

The four characteristics are not independent of one and other, and I cannot foresee explicit algorithms for such properties. The most appropriate approach is to define a series of steps that should, in combination, produce layouts which exhibit the desired aesthetic characteristics. Details of such procedures are included in the implementation section. Considerable work is required in devising aesthetic procedures.

3.5 Requirements of the Surface Level

The surface level of the aesthetic system creates a housing which, although devoid of component specific features such as surface holes, is compatible with the overall size, shape, and intended use of the product. The surface must provide an appropriate basis for the desired style. Figure 3.15 depicts the activities of the surface level. This level of the model receives the configured components from the organization level and, using a pre-defined style library, completes the product housing data.

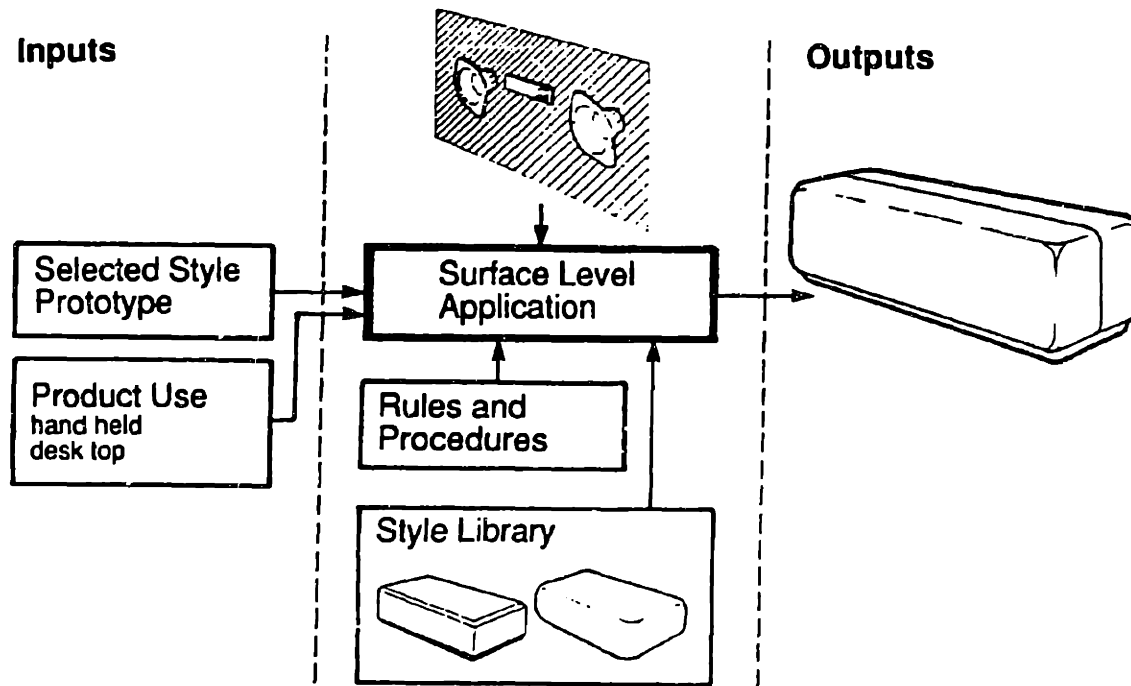


Figure 3.15 Elements of the surface level of the model

In chapter 2 it was decided that the use of analogy — in particular the model provided by morphological analysis — is the most appropriate representation for encoding surface styles. The method uses prototype solutions to obtain product housings.

The knowledge representation used in the surface level is not consistent with the rule-based approach adopted in the organization level. It would be desirable to cast the entire system within a single representation to maintain a uniform approach. Unfortunately, rules are not sufficient to represent the complexities of cosmetic surface design — the use of analogy is

essential. This opinion is consistent with that expressed by Gero *et. al.* (1988) about complex design situations in general. The appropriateness of using prototypes for this application should outweigh additional programming effort.

In the following sections, the definition and use of prototypes to represent surface styles is discussed within the context of the aesthetic model.

3.5.1 Definition of a Prototype

Prototypes are an important design tool. For instance, Moneo (1978) views architectural design as the adaptation of known prototypes. In *Designing by prototype refinement in architecture*, Oxman and Gero (1988) describe a prototype as "a stylized or parameterized design description or design description generator". The morphological analysis approach is an example of a parameterized prototype, while shape grammars can be described as a design description generator. An example of a prototypic enclosure for a technical looking style is shown in figure 3.16.

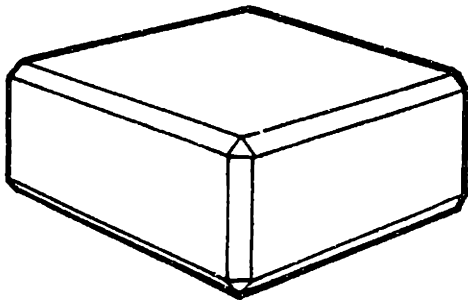


Figure 3.16 Prototypic enclosure for the surface level

Embedded in the prototype are manufacturability considerations, plus a collection of features which, as a whole, characterize a particular style and use. The term surface style prototype is used to describe an example enclosure which captures relevant design knowledge. It should be noted that (in general) the number of sides is not essential to the prototype. The prototype should carry information such as appropriate draft angles, corner/surface treatment, and use specific features.

The process of mapping the prototype to specific products is of critical importance. The mapping must be done in a manner which does not destroy the information embodied by the prototype. Unrestricted deformations during the mapping process can negate manufacturability characteristics (draft angles for example). Further, the mapping must

preserve the style of the prototype. In the proof of concept, I have asserted that the scaling of the corner treatments is especially important.

The morphological approach requires topological similarity between the prototype and the instance of the prototype (the product housing). For example, if a box-like product is to be designed, then the system must have a box-like prototype. This is required even though the concept box-like may be non-essential to the knowledge that the prototype represents (say a specific style). This very literal interpretation of a prototype may seem limiting, but after further thought does not seem too far-fetched. Consider a system which automatically styles automobile bodies. The anatomy of automobiles is well defined and homogeneous, so a literal interpretation of body prototypes should not be a problem. Similarly, the box-like domain is adequate for a wide variety of electronics products. A direct interpretation of prototypes is acceptable for classes products whose general characteristics are well established. It cannot invent new topologies or ideas. The approach promotes the standardization of product housings, which can reduce production costs. For the system to create arbitrary shaped enclosures from a single prototype, a generative method (such as shape grammars) is required.

If the chosen approach is not to become overly predictable, it will be necessary to develop prototypes on an on-going basis. The system will need a sizable quantity of prototypes. However, the method permits the designer to develop the prototypes in a conventional or familiar fashion. Designing a prototype is not much different than designing a housing for an individual product. The use of prototypes does not limit the types of surfaces which can be incorporated into the system. If highly free form surfaces are desired, then free form prototypes are required.

3.5.2 Requirements of the Prototype and Utilization

The prototype data encodes manufacturing, ergonomic, and stylistic information. The prototype library files must provide the information needed by the system to select the appropriate prototype and map it to the product. A general structure for a prototype in the surface library is shown in figure 3.17. The figure is intended to suggest the flavour of information to be contained in a prototype file.

The applicability attributes are necessary to determine if the prototype is appropriate for the current design. A product which is injection moulded will require an injection

moulded prototype. Additionally, the intended use of the product must be considered during the prototype selection. It may be desired to have different prototypes for hand held and desk top products. The desk top prototype would be designed with a foot, while the hand held prototype would not. The overall characteristics of the enclosure (such as box-like) are needed to match the topology of the housing with the overall configuration of the components. The proof of concept considers only box-like designs, and the components are laid out with this in mind. If the components are positioned for a different overall shape — perhaps a product with a curved parting plane — then a different prototype may be required. Finally, the prototype style must match the desired style of the product.

Applicability Attributes:	manufacture	<i>injection moulding</i>
		<i>mould type</i>
		<i>parting plane</i>
	use	<i>hand held</i>
		<i>desk top</i>
	characteristics	<i>box-like</i>
	style	<i>Art Deco</i>
		<i>High-Tech</i>
		<i>Braun-like</i>
	scaling	<i>reference points</i>
		<i>mapping functions</i>
Mapping Instructions:	features	<i>geometric constraints</i>
List of Geometric Data		

Figure 3.17 Surface prototype structure

Features which constrain component locations (such as a free form surface) must be identified. In the box-like domain, the components can be placed anywhere on the faces of the product, so this is not an issue. If enclosures which restrict component placement are desired, it will be necessary to choose the enclosure before the organization level. Reference or datum points which map to the overall dimensions of the product are used to determine what transformations need to be applied to the prototype. The prototypes will have their own transformation functions. Consider a situation where the system has used the reference points to determine how the prototype must be scaled to fit the product. Scaling functions contained within the prototype will be used to perform this task.

The last item in the prototype is the geometric data used to create the product's enclosure.

3.5.3 Designing a Prototype Surface: Form Organization

The use of surface prototypes permits designers to create styled enclosure prototypes in media with which they are conversant. The prototypes can be developed using conventional drafting board and modelling techniques. This feature should facilitate the definition of prototypes and minimize retraining requirements for industrial designers. Regardless of the design method, the final prototypes must be mathematically encoded for use by the system. A complete physical (i.e. solid) model is necessary to interface with engineering analysis and the design of internal features. If the outer appearance is of interest only, a surface representation is adequate.

Ideas for general prototype shape design are suggested in chapter 2. In this section a simple design approach — form organization — is presented. The technique allows prototypes to be defined on a drafting board but, unlike the possibilities discussed in chapter 2, is not suitable for a very general application. An introduction follows.

3.5.3.1 The Methodology

Form organization is a way of imagining free form surfaces. A summary is provided here, but complete details may be found in the book *Form Organization* (Gilles, Willem, 1991). The method was conceived in the 1950's by Wim Gilles and, during the 1980's, was developed by Professor Gilles, David McAleer, Terry Sanders, and myself into a tool for designing monolithic product forms. The method is an alternative way to think about shapes and provides the data necessary to loft a surface. The technique is more restrictive than fully parametric representations, but its utility for designing smooth shapes is derived through such constraints.

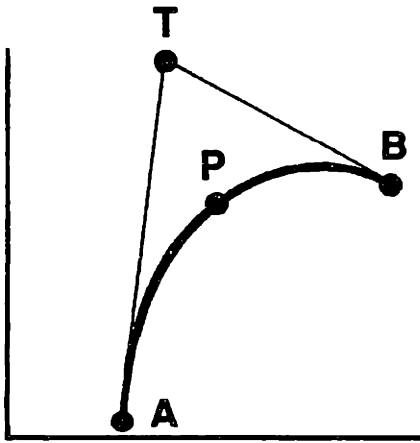


Figure 3.18 The control points to define a conic segment

Form organization is based upon conic segments. A conic segment can be computed using four points: the end points of the curve; the point where the tangents intersect, and any point on the curve. We call these the A, B, T and P points respectively. They are sketched in figure 3.18. This approach to defining conic segments was applied in the aircraft industry long ago (Liming, 1944). The appeal of conic segments is that, due to their low order, they are very predictable and can be sketched rapidly. It is possible to use data measured from sketches to define curves analytically.

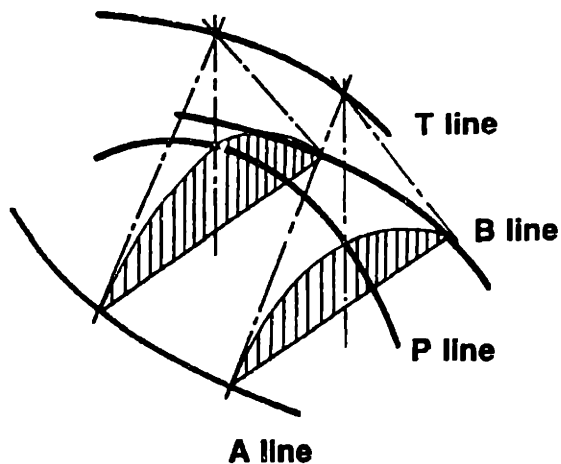


Figure 3.19 The use of conic control curves to define a surface

If four conic segments are drawn in space, as indicated in figure 3.19, the segments form control curves for a surface. The control curves are called A, B, T, and P lines. A planar slice through the control curves provides the four data points necessary to compute the surface section at a location. The control curves are composed of any number of conic segments. Surface points are obtained by computing several cross sections. Conic segments are planar, but space curve control lines can be represented by projecting two conics onto each other.

The mathematics which follows pertaining to conic sections is derived from Faux and Pratt (1987). The material on splines may be found in Rogers and Adams (1990).

The parametric equation for a conic section based upon four vectors as shown in figure 3.20, is given by

$$r = \left\{ \frac{w_0 r_0 (1-u)^2 + 2w_1 r_1 u(1-u) + w_2 r_2 u^2}{w_0 (1-u)^2 + 2w_1 u(1-u) + w_2 u^2} \right\} \quad 0 \leq u \leq 1 \quad (3.1)$$

where w_0 , w_1 , and w_2 are weighting factors.

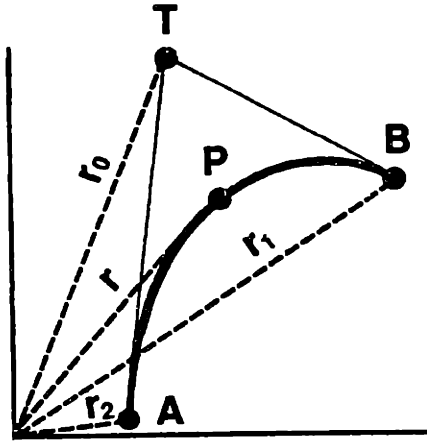


Figure 3.20 Nomenclature for points defining a conic segment

Given four points on the curve and assuming the parameter $u = 0.5$ at the p point, the weights may be calculated as below.

$$w_0 = 2 \left\{ \frac{n \cdot [(r-r_1) \times (r-r_2)]}{n \cdot [(r-r_2) \times (r-r_0)]} \right\} \quad (3.2)$$

$$w_1 = 1.0 \text{ (assumed)} \quad (3.3)$$

$$w_2 = 2 \left\{ \frac{n \cdot [(r-r_0) \times (r-r_1)]}{n \cdot [(r-r_2) \times (r-r_0)]} \right\} \quad (3.4)$$

$$n = [(r_0-r_1) \times (r_2-r_1)] \quad (3.5)$$

The tangent in the plane of the curve is given by the oblique coordinate equations

$$a = \frac{w_0(1-u)}{w_0 + u(w_1 - w_0)} \quad (3.6)$$

$$b = \frac{uw_2}{w_1(1-u) + uw_2} \quad (3.7)$$

Converting to a Cartesian system, the in plane tangent vector is given by

$$t_{in \ plane} = a(r_1-r_0) - b(r_1-r_2) \quad (3.8)$$

At each point on the surface a second tangent is required for normal computation. The second tangent must not be in the plane of the surface defining conic section. The corresponding data points in the previous and subsequent surface cross sections are used to form a cubic spline. From this spline the out of plane tangent can be approximated.

The out of section tangent for the i^{th} section and the j^{th} data point in the section is given by

$$t_{\text{out plane}} = \frac{-6 \frac{t_3}{t_2} r_{1-1:j} + (6 \frac{t_3}{t_2} - 6 \frac{t_2}{t_3}) r_{1:j} + 6 \frac{t_2}{t_3} r_{1+1:j}}{6(t_2 + t_3)} \quad (3.9)$$

For the first section in a surface, the special case equations below applies

$$t_{\text{out plane}} = \frac{-(9 + 6 \frac{t_3}{t_2}) r_{1:j} + (9 + 6 \frac{t_3}{t_2} + 3 \frac{t_2}{t_3}) r_{1+1:j} - 3 \frac{t_2}{t_3} r_{1+2:j}}{6(t_2 + t_3)} \quad (3.10)$$

while for the last section

$$t_{\text{out plane}} = \frac{3 \frac{t_3}{t_2} r_{1-2:j} - (9 + 3 \frac{t_3}{t_2} + 6 \frac{t_2}{t_3}) r_{1-1:j} + (9 + 6 \frac{t_2}{t_3}) r_{1:j}}{6(t_2 + t_3)} \quad (3.11)$$

The factor t_2 is the distance between the points on the i^{th} section and the section with the lower index. The factor t_3 is the distance between the points on the i^{th} section and the section with the higher index. The cross product of the two tangents yields a surface normal at the point.

Using the equations presented above, it is possible to compute a number of surface points m for a predetermined number of cross sections. For each point the normals are calculated to complete the surface. Evenly spacing the surface points through the parameter space u of the cross sections is, in many cases, very inefficient. It is desirable to concentrate surface points in areas of high curvature to achieve adequate resolution with a minimum amount of data.

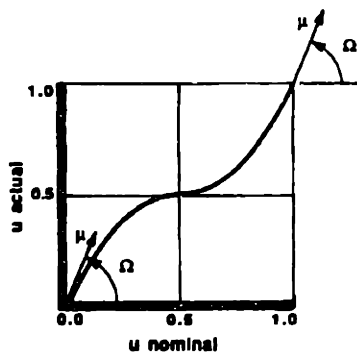


Figure 3.21 Parameter spacing weighting function

A spline function, as shown in figure 3.21, can be used to concentrate points in areas of high curvature. In describing the weighting function, it is necessary to introduce the shoulder point of a conic segment. The shoulder point of a conic, which occurs at $u = u_s$, is the deepest point of the curve (see figure 3.22). The shoulder is usually close to the point of maximum curvature. The weighting function, as shown in figure 3.21, assumes that the point of maximum curvature is located at $u = 0.5$.

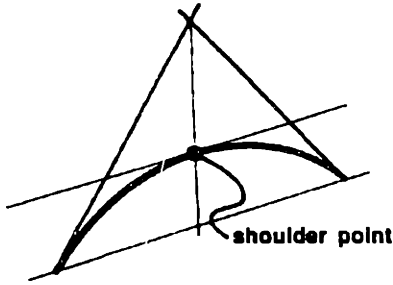


Figure 3.22 The shoulder point of a conic segment

Hence, it is convenient to parameterize the curve so that $u = 0.5$ at the shoulder. The shoulder parameter u_s can be calculated using the relationships

$$k = \left\{ \frac{w_0 w_2}{w_1^2} \right\}^{\frac{1}{2}} \quad (3.12)$$

and

$$u_s = \left\{ \frac{1}{1 + k \frac{w_1}{w_0}} \right\} \quad (3.13)$$

The value of k is an indicator of the conic segment's maximum curvature. Therefore, k can be used to determine the degree to which the data should be focused about the shoulder point. (Recall that the shoulder point is approximately at the point of maximum curvature, and the curve should be reparameterized such that $u_s = 0.5$.) Given the value of k and the reparameterized curve, the weighting function in figure 3.21 is applied as follows.

If m points are to be calculated along a segment, then the nominal parameter is

$$u_{\text{nominal}} = \left\{ u_{\text{nominal}} + \frac{1}{m - 1} \right\} \quad 0 \leq u_{\text{nominal}} \leq 1.0 \quad (3.14)$$

The nominal parameters are evenly spaced along the segment. The actual parameters — which are used to calculate the points on the curve — are obtained by solving for the weighting function parameter t in terms of u_{nominal} .

$$\left\{ \mu \cos \Omega - \frac{\sqrt{2}}{2} \right\} t^3 + \left\{ \frac{3}{2} - \frac{3}{\sqrt{2}} \mu \cos \Omega \right\} t^2 + \mu \cos \Omega t - u_{\text{nominal}} = 0 \quad (3.15)$$

$$u_{\text{actual}} = \mu \sin \Omega t + \left\{ \frac{3}{2} - \frac{3}{\sqrt{2}} \mu \sin \Omega \right\} t^2 + \left\{ \mu \sin \Omega - \frac{\sqrt{2}}{2} \right\} t^3 \quad (3.16)$$

for $0 \leq t \leq \frac{1}{\sqrt{2}}$, $0 \leq u_{\text{nominal}} \leq 0.5$

and

$$\left\{ \mu \cos \Omega - \frac{\sqrt{2}}{2} \right\} t^3 + \left\{ \frac{3\sqrt{2}}{4} - \frac{1}{2} \mu \cos \Omega \right\} t + 0.5 - u_{\text{nominal}} = 0 \quad (3.17)$$

$$u_{\text{actual}} = \left\{ \mu \sin \Omega - \frac{\sqrt{2}}{2} \right\} t^3 + \left\{ \frac{3\sqrt{2}}{4} - \frac{1}{2} \mu \sin \Omega \right\} t + 0.5 \quad (3.18)$$

$$\text{for } 0 \leq t \leq \frac{1}{\sqrt{2}}, \quad 0.5 < u_{\text{nominal}} \leq 1.0$$

The symbol μ represents the length of the weighting function tangents. The angle of the spline tangents is calculated using the curvature of the segment.

$$\Omega = \frac{\pi}{4} \left(1 + e^{-\frac{k^a}{b}} \right) \quad (3.19)$$

The parameters μ , a , and b are adjusted to provide good point distribution for curves ranging from nearly straight to almost a 90 degree corner. Using $a = b = 1.0$ and $\mu = 1.1$ gives reasonable results. The effect of the weighting function is shown in figure 3.23.

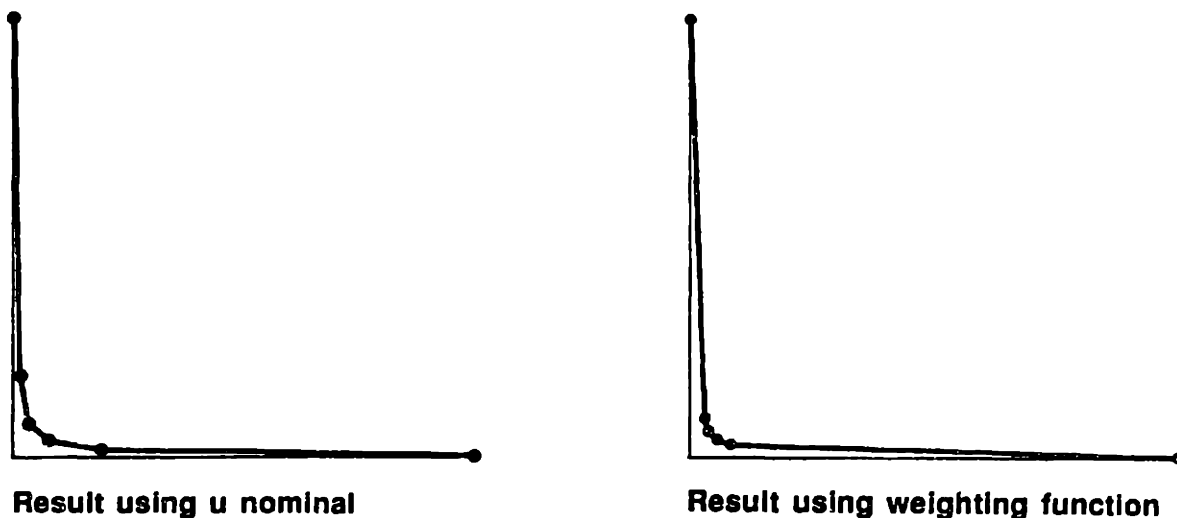


Figure 3.23 Effect of the weighting function on a curve with six data points

The equations provided in this section are sufficient to write a surface lofting program using conic segment control curves. Now that a method for encoding surface

prototypes has been discussed, the important issue of scaling a prototype should be addressed.

3.6 Scaling Effects

Designing the style prototypes is only half of the problem. The system must also know how to transform the prototypes to fit specific products. In this thesis, the application of prototypes is limited to scaling.

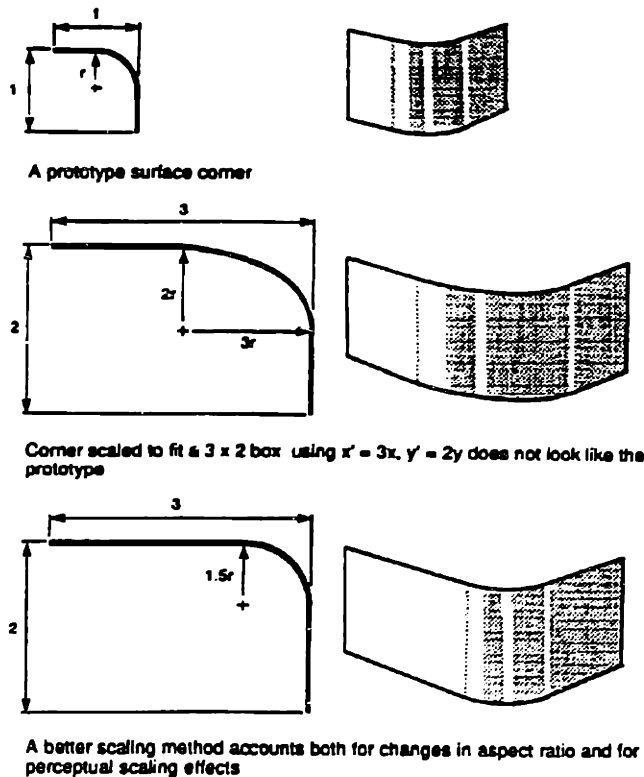


Figure 3.24 Illustration of how corner style changes under scaling

Human perception is non-linear with scale. An equation which distorts the prototype to compensate for scaling effects is necessary to ensure that style is preserved. The example in figure 3.24 illustrates the inadequacy of blind uniform scaling. Without reference to literature on scaling effects in product design, experience suggests that when the prototype size is increased, the corner features need to be made proportionally smaller. A universal mapping formula will, most likely, be implausible. The scaling requirements for smooth and faceted products will be different.

It will be necessary to teach the system how different styles of prototypes should be mapped to various sizes and shapes of products. The method for designing mapping functions must not be abstract. The designer should focus on how the prototypes transform rather than how to encode or develop a transformation function. An encoding scheme in which the industrial designer shows the system how to fit prototypes to a small range of characteristic products is envisioned. A number of control knobs would be used to modify the scaled prototypes to achieve the desired appearance in each instance. Then, the system would use the example cases to determine appropriate scaling functions.

3.7 Requirements of the Detail Level

The third stage of the four level model for form giving builds upon the monolithic product housing created in the previous stage. Details refer to features, elements, or local surface alterations which must be imparted to the enclosure. Examples of such additions are holes and grills for speakers, or holes and buttons for keypads. Figure 3.25 depicts the input and output for this phase of the model.

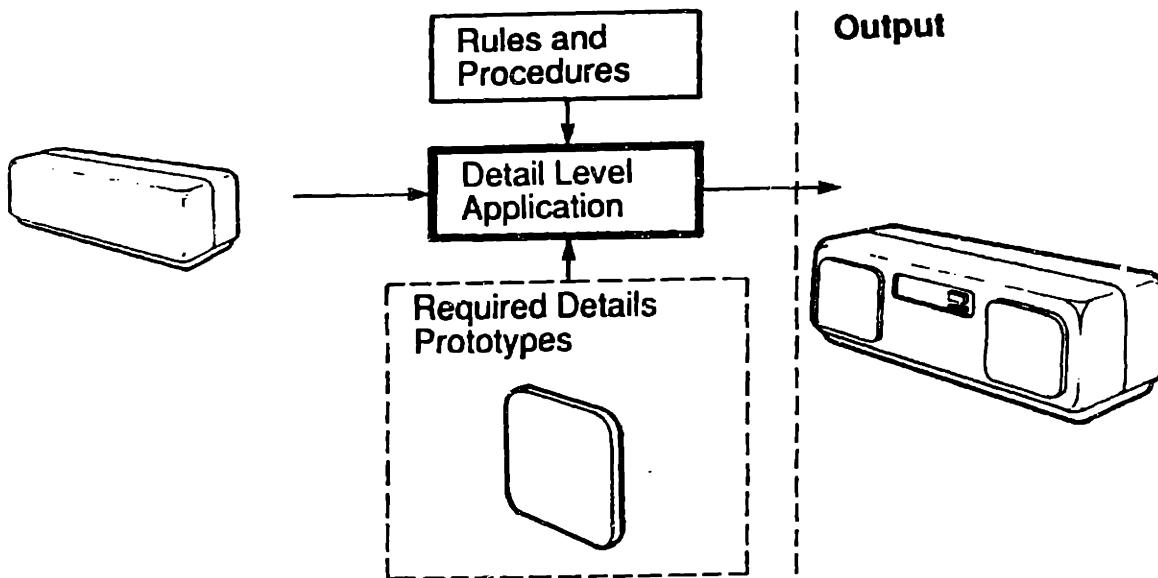


Figure 3.25 Input and output of the detail level of the model

The system receives the enclosure from the surface level and, using a library of pre-defined style specific detail prototypes, applies features as required by the components within the product. When necessary, components will be re-positioned relative to the surface to accommodate newly added elements. The completed surface geometry is output for cosmetic finishing in the graphical level of the model.

Detail prototypes are used for the same reasons as discussed in the surface level. Issues covered in section 3.5 about the design of prototypes and their adaptation applies to this section as well. The designer needs to create a certain type of detail only once. The system will modify and apply the prototypes to a wide range of products. For example, if a designer in an automotive company creates a door handle insert for the library, the system will take responsibility of placing the insert in specific car body designs. The need for

designers to perform this task repeatedly is eliminated. In the remainder of this sub-section reasons for adding details, possible types of details, and requirements of the prototype library are discussed.

3.7.1 Reasons to Apply Details

Details are added to the housing subject to the components included in the product design. Three reasons for the adding details — within the scope of the model — have been identified: function (ergonomics); manufacture; and aesthetics or style.

The need to add details for functionality is obvious. Surface details are the ports through which the user interacts with internal components. For example — buttons are needed for keypads — grills allow sound produced by speakers to reach the user. Ventilation slots need to be incorporated into the design if heat liberating components are contained.

Adding details can augment manufacturability. Panel inserts are used to avoid freeze marks or poor flow characteristics in moulded parts when closely spaced holes are required. Often, the inserts are made from stamped material. Further, panels can be used to avoid side holes in favour of less complex side slots.

Component Layout



Possible Detail Configurations

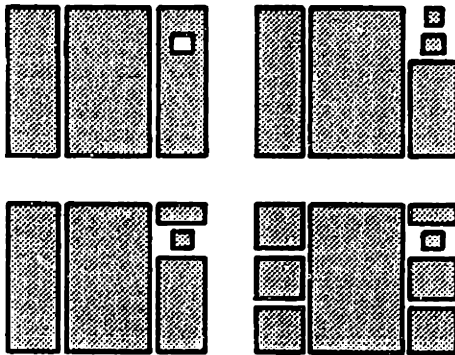
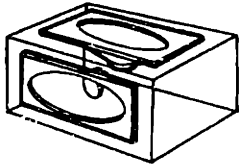


Figure 3.26a Different groupings of components under details

The details are expected to play a major role in the style of a product. They must make products with identical layouts but different styles appear unique (the model assumes that the component layout is style independent). The styling of individual details is determined by the designers through the definition of prototypes. However, adding details can enhance or maintain the grouping and organization imparted to the product layout during the organization level. Figure 3.26 demonstrates how details can be used to create different effects for a given component layout.

Component Layout



Possible Detail Configurations

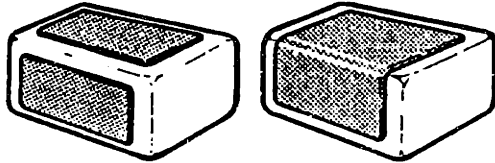


Figure 3.26b Applying details to integrate product faces

Details can accentuate different functional interfaces or integrate several components into one element. (such as a bank of speakers). Details may serve to integrate product faces or imply how the product is to be used.

Typical considerations important in the detailing level of the aesthetic model have been addressed in this section. The requirements of the detail prototypes are discussed subsequently.

3.7.2 Requirements of the Prototype and Utilization

The components and their spatial locations provide the necessary information to determine what details are needed and how big they should be. For example, a keypad component carries the information about button type and locations. The system applies rules of thumb based upon aesthetics and manufacture to determine if components, such as speakers, should be placed under a single detail. Information and instructions for addition to the surface must be included in the detail. Examples of such information is outlined in figure 3.27.

Applicability Attributes:	identity	<i>button</i>
	manufacture	<i>speaker grill</i> <i>injection moulding</i> <i>stamped</i>
	style	<i>Braun-like</i> <i>Technical</i> <i>Art Deco</i>
Mapping Instructions	scaling	<i>reference points</i> <i>method</i>
	installation	<i>add to housing</i> <i>part of housing</i> <i>cutting template</i> <i>component location</i>
List of Geometric Data		

Figure 3.27 Detail prototype structure

Each detail must have an identity and a style. The detail's manufacture may differ from the process used for the housing. Keypad inserts are often made from sheet metal.

Details contain instructions that describe how they can be customized for individual products. The scaling reference points are used to map the detail to the product, just as the enclosure prototype reference points project to the product hull. Like the surface prototype, a style specific scaling method must be provided. The system will use the functions to transform parametric details — such as grills or panel inserts — to suit a particular product. Some details, however, will be of a fixed size. This is desirable for standardization and modularity. It makes sense to standardize items such as keypad buttons.

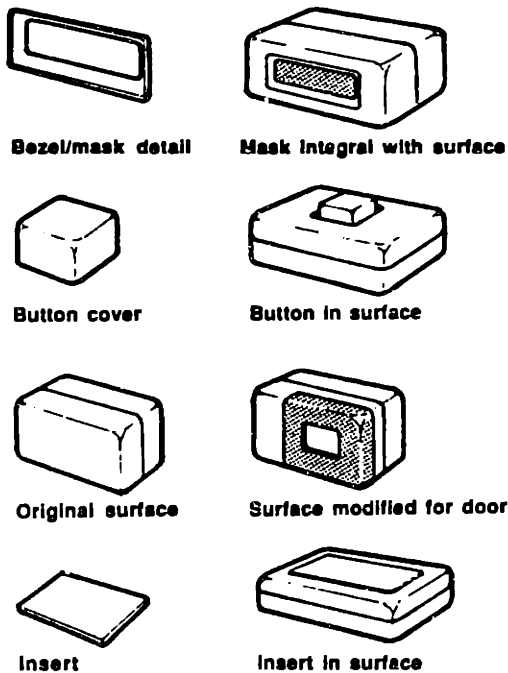


Figure 3.28 Possible methods to incorporate details into a surface

Installation instructions will vary with the detail type and style. For example, a visual display mask could become an integral part of the housing. A button will require an oversized hole in the product surface. A cassette door might be formed through modification of the enclosure surface. Some possibilities are shown in figure 3.28. Performance of these operations will require sophisticated geometric modelling. Even the relatively simple hole cutting operation is non-trivial. These operations are not considered because they are not central to the thesis. Martti Mantyla (1988) provides a good introduction to solid modelling issues.

Once the detail is positioned in the product surface, it may be necessary to adjust a component's location relative to the detail. For example, it might be necessary to raise a keypad membrane to meet a button at the product surface.

3.8 Requirements of the Graphical Level of the Model

The fourth and final level of the model applies graphical elements to the outer surface of the product. The term graphical element refers to two dimensional features such as colours, screenings, logos, hot stampings, and textures. The throughput of the fourth level is illustrated in figure 3.29.

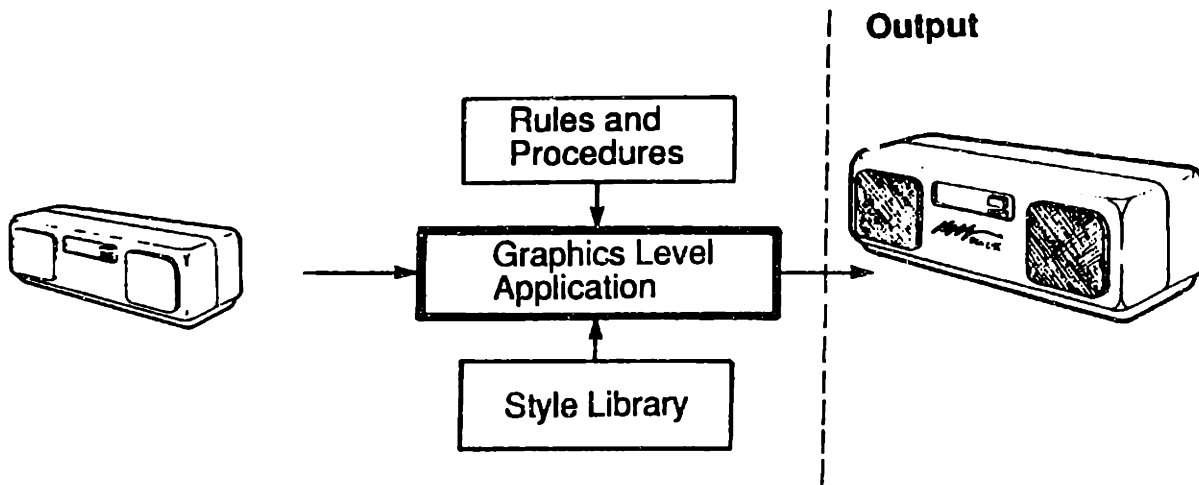


Figure 3.29 Input and output of the graphical level of the model

After completing the surface geometry, it is easy to trivialize this level of the process. Indeed, graphics are only superficially explored in this thesis. It must be emphasized, however, that poor execution of this phase can spoil an otherwise good design. Colour is often critical to a style, and it influences how product forms are perceived. Issues such as part inventory and part matching also affects colouring decisions. Textures can disguise moulding defects such as sink or freeze marks and help the consumer to understand the product's use.

It may be possible for the system to automatically apply colour, logos, and texture subject to style, product use, and aesthetics. In general, however, interactive addition of graphical elements — such as text — will be required. The scope of the thesis is limited to colouring the product exterior. This is probably the most prominent graphical feature of a product style. This limitation restricts the magnitude of the project to a reasonable level. Graphic arts is another field unto itself.

3.8.1 Colouring Considerations

Product colour schemes often play a paramount role in the purchase decision of consumers. Consequently, many companies devote considerable resources to the study of colour preferences. For example, the automotive company Nissan employs a psychologist to determine culture specific tastes (Bonner, 1991). Each culture tends to have its own colour preferences, as do different economic, geographic, religious, and age groups (Mahnke and Mahnke, 1987). The proposed system could provide an ideal mechanism for applying such knowledge.

There are universal mood associations and responses related to colour (Mahnke and Mahnke, 1987). The following examples are from *Psychology of Color and Design* (Sharpe, 1974) and *Color and Light in Man-Made Environments* (Mahnke and Mahnke, 1987). Red is the most dominate colour and is stimulating or aggressive. It grabs attention. Deep red is masculine, while pinks are feminine. Blues or greens are retiring and relaxing, while yellow evokes high-spirits or cheerfulness. Saturated colours are preferred over less saturated colours, which are often associated with cheapness. Light colours recede and help products fade into the background. Dark objects command a stronger presence. Deep colours are associated with weight — a dark base will enhance the visual stability of a design.

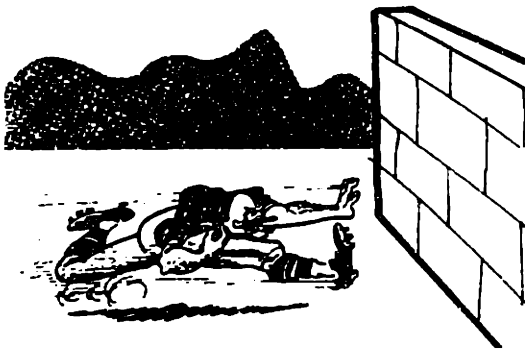
General guidelines are merely such — the choice of colour must be made within the context of specific products. Some colours may be wholly inappropriate for particular forms. In the automated system, the use of style specific colour pallets, defined by graphic artists, is envisioned. The responsibility of creating acceptable colour combinations falls upon the designer. Different pallets should be designed to accommodate the tastes of different user groups within the constraints of the given style.

3.8.2 Application of Colour Palettes to the Detail Product

The system should use the product style and user group as an initial conditions to choose an appropriate colour palette from a library of predefined colour combinations. Given the appropriate set of colours, the system must decide how the surface and individual details should be coloured. This process lends itself to a rule based approach. The colouring process will assign base colours to the product housings, and then add colours to the details based upon detail type and the overall composition of the product.

4 IMPLEMENTATION

Continuing from the general discussion of the model for aesthetic industrial design in chapter 3, this chapter reveals how the model was executed in the domain of injection moulded box-like consumer electronic products. Details of the aesthetic rules and protocols are provided. While the computer code is too large for inclusion here, adequate detail is provided so that one should be able to construct a similar system.



The program is written in the C programming language for a Silicon Graphics Iris 4D/70 GT workstation. The program is realistic enough to demonstrate the proposed system's potential. It is capable of designing simple but convincing box-like products in three different styles: Art Deco, High Tech, and Braun-like.

The program user interactively defines the design initial conditions. The aesthetic design process is completely non-interactive. The primary goal of the program is to demonstrate the ability to automate. As the designs cannot be modified by the user, results unambiguously portray the program's capabilities.

The program is not intended for use as a platform for long term research or developing a commercial system. Many aspects of the model were opaque at the onset, and only became clear after writing and testing code. Only the minimum level of sophistication for acceptable results was strived for. This approach permitted all four levels of the model to be investigated to some degree.

This chapter provides a description which can be used as a starting point for developing a program of sufficient artistry for commercial use or on-going research.

4.1 Global Architecture of the Implementation

Major program modules, the flow of data, and program control are indicated in figure 4.1. The four levels of the model are implemented as autonomous black boxes which communicate through the product data only. This system design was chosen to allow each level of the model to be tackled incrementally. Program control is regulated by the receptor (or user interface). When a level of the model is invoked, control is passed to the module. Each level of the model works towards completing the product data (as outlined in figure 3.4). When the product data for a level is complete program control is returned to the interface. It should be noted that for the proof of concept the detail and graphics levels were combined into a single module.

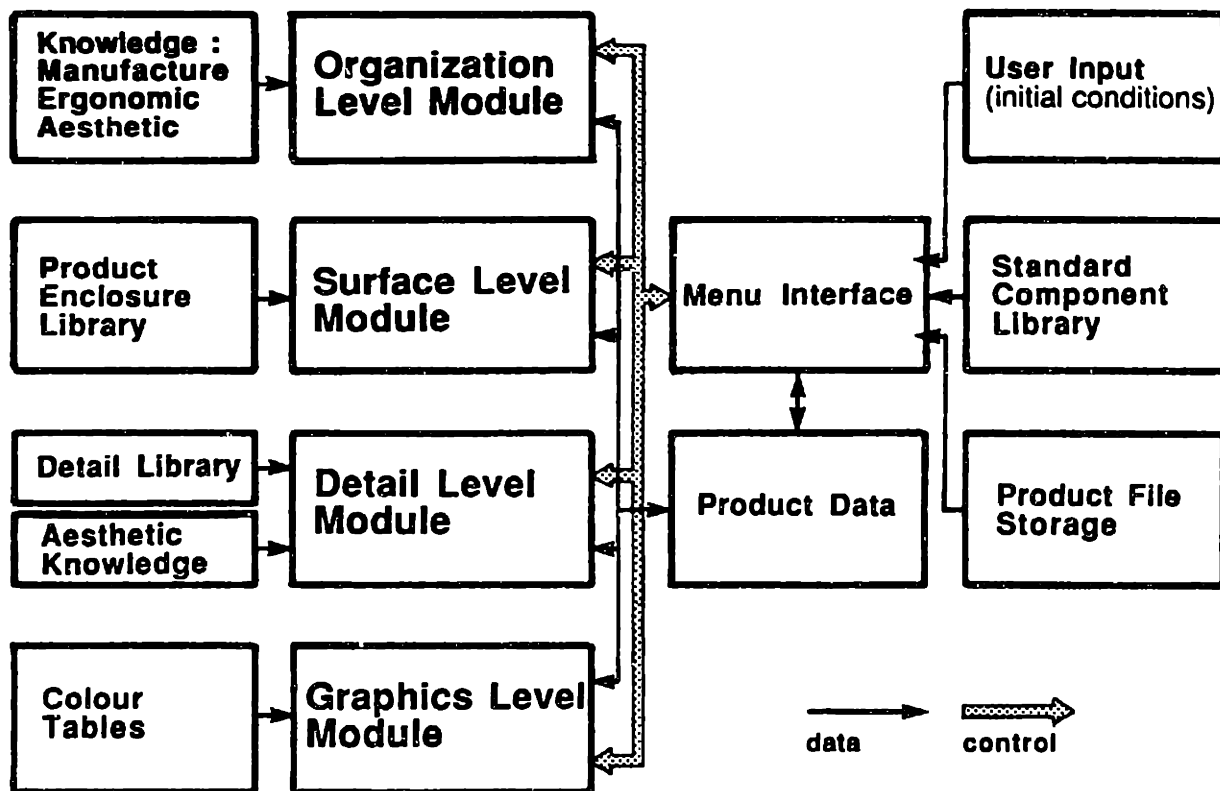


Figure 4.1 Major elements of the program

The interface is implemented using pre-defined iris pull down menus. The user selects product attributes from a menu of options. Similarly, the menus are used to select standard components for inclusion in designs. A means to store or retrieve products from disk in any stage of completion is provided. The user calls the four levels of the model using the menu system. If the product data is not adequately complete for the chosen level, program control is returned to the interface. This approach is convenient for debugging. The system could just as easily call all of the levels itself once the necessary initial conditions are provided. Predefined Silicon Graphics light modelling routines were utilized to produce shaded renderings of the product designs.

The pull down menus are used as a matter of programming convenience. Truthfully, the program should provide a component selection mechanism analogous to a part catalogue — complete with technical information. This issue is secondary to the goals of the thesis.

As mentioned previously, the program does not allow the user to position components directly, or alter the output of the automated system in any way. This implementation decision was made to avoid possible confusion over what the system is capable of doing, and what is the user's input. The results and conclusions suggest that a practical implementation will allow the designer to edit the output.

In the remainder of the chapter, the major program elements shown in figure 4.1 are discussed. These elements are: the product structure; the component library; the organization level and its rules; the surface level and its enclosure libraries; the detail and graphics level.

4.2 The Product Structure

The product structure links the levels of the model through its data, as seen in figure 4.1. The product was encoded as a C structure reflecting the product framework presented earlier in figure 3.4.

Restating, the product structure includes the following attributes: *horizontal* or *vertical* mould parting planes; *desk top* or *hand held* use; *tall* or *normal* height; *thin* or *normal thickness*; and the three styles *Braun-like*, *Art Deco*, or *High Tech*. The height attributes indicate whether taller looking configurations are desired. Choosing a normal height option means that the product height is not of concern. A similar explanation applies

to the thickness attributes. Product thickness is defined as the dimension orthogonal to the mould parting plane.

The components placed in the product form a doubly linked list. Only the identities (pointers to components) and the position and orientation of components reside in the product. Component geometry and attributes are kept in separate objects. Only product specific information is stored within the product structure. This organization has the advantage of being very flexible. For example, if the standard component library is updated, the system automatically incorporates the new components into products designed prior to the revisions.

Unlike components, the enclosure and details must be local to the product. This data is customized for individual products. The surface geometry is represented using arrays of directed polygons. The polygon structure contains vertex, normal, and colour information for individual polygons.

4.3 The Component Library

A library of standard components with appropriate attributes is pivotal to the concept. For companies attempting to standardize — or limit — the number of components they use, developing a library is justifiable. Perhaps some day, topologically complete computerized vendor catalogues will be available. The component library for the proof of concept was fabricated in its entirety. A manageably small set of components was chosen to permit the design of a variety of products.

In all, twenty three different components spanning five different classes are available to the system. The classes are based upon the taxonomy discussed in section 3.4.3.1.3. The components are listed, along with their attributes, in table 4.1. The attributes are used by a rule based system to locate the components in the product matrix, or to determine special surface and detail requirements. The proof of concept uses only three types of attributes. The component class identifies the rule base that applies to the component. The features attribute indicates a component's detailing requirements. The special attribute describes component characteristics that differentiate it from others within its class.

Table 4.1 The components and their attributes

Component	Attributes class	features	special
<i>6x6 cathode ray tube</i>	<i>visual display</i>	<i>mask</i>	<i>hot</i>
<i>2 digit LED</i>	<i>visual display</i>	<i>mask</i>	<i>numeric</i>
<i>12 character LCD</i>	<i>visual display</i>	<i>mask</i>	
<i>30 character LCD</i>	<i>visual display</i>	<i>mask</i>	
<i>radio tuner</i>	<i>visual display</i>	<i>mask</i>	
		<i>buttons</i>	<i>5 miniature</i>
<i>middle range speaker</i>	<i>acoustic</i>	<i>grill</i>	
<i>high range speaker</i>	<i>acoustic</i>	<i>grill</i>	
<i>mini piezoelectric speaker</i>	<i>acoustic</i>	<i>grill</i>	
<i>miniature microphone</i>	<i>acoustic</i>	<i>grill</i>	
<i>single button membrane</i>	<i>keypad</i>	<i>buttons</i>	<i>1 miniature</i>
<i>three button membrane</i>	<i>keypad</i>	<i>buttons</i>	<i>3 miniature</i>
<i>numeric membrane</i>	<i>keypad</i>	<i>buttons</i>	<i>12 miniature</i>
<i>alphanumeric membrane</i>	<i>keypad</i>	<i>buttons</i>	<i>28 miniature</i>
<i>headphone plug</i>	<i>jack</i>	<i>cover</i>	<i>interface</i>
<i>speaker plug</i>	<i>jack</i>	<i>cover</i>	<i>service</i>
<i>power supply</i>	<i>jack</i>	<i>cover</i>	<i>service</i>
<i>telephone plug (handset)</i>	<i>jack</i>	<i>cover</i>	<i>interface</i>
<i>telephone plug (line)</i>	<i>jack</i>	<i>cover</i>	<i>service</i>
<i>1 button press</i>	<i>switch</i>	<i>buttons</i>	<i>1 press</i>
<i>1 button toggle</i>	<i>switch</i>	<i>buttons</i>	<i>1 toggle</i>
<i>5 button press</i>	<i>switch</i>	<i>buttons</i>	<i>5 press</i>
<i>5 button toggle</i>	<i>switch</i>	<i>buttons</i>	<i>5 toggle</i>
<i>cassette</i>	<i>mechanism</i>	<i>buttons</i>	<i>5 press</i>
		<i>door</i>	

The component library is embodied by ASCII character files stored on disk. The files are accessed on demand to add component information to the product structure. The structure of a component library file is provided in figure 4.2. The format is self-explanatory. Storing the component data in character files — like the redundant method for defining polygon vertices — is desirable for debugging the library (the files can be read by a human). The format sacrifices efficiency. The maximum dimensions of a component are included to define a bounding box. The bounding box is used for rapid interference calculations. The bounding box is specified explicitly as it will not always correspond to the the component's overall dimensions. In some cases it may be desirable to provide additional clearance around a component — perhaps for soldering wires onto the terminals of a jack.

<i>string</i>	component name
<i>integer</i>	identity number
<i>integer</i>	number of attributes
<i>string</i>	list of attributes
<i>integer</i>	number of polygons
<i>integer</i>	number of vertices in polygon
<i>float</i>	x y z vertices coordinate list
<i>float</i>	x y z vertices normal list
<i>integer</i>	colour index
<i>integer</i>	number of buttons
<i>float</i>	x y z button locations
<i>float</i>	x y z normal of the top of component
<i>float</i>	x y z normal of the front of the component
<i>float</i>	x y z normal of the main surface with which the user interacts
<i>float</i>	x y z bounding box overall dimensions

Figure 4.2 Component library data file structure

The component geometry is designed using the form organization method described in the sections on the surface level of the model. All components are centred on the origin. The component surfaces are detailed just enough to be recognizable. They are pictorial schematic representations. Photographs of all components made available in the implementation are contained in appendix A. The appendix can be used to identify components in subsequent chapters.

4.4 Organization Level

Recalling chapter three, the organization level locates components relative to the injection mould parting plane and produces a product bounding hull. The positioning process begins after the initial conditions and desired components are selected by the user. The organization level is a two step process which prioritizes manufacturing and ergonomics over aesthetics, as indicated in figure 3.8. The rule based approach for filling the product matrix subject to manufacturing, ergonomics, and standard practice is described in this section. Additionally, discourse on the aesthetic positioning procedures is included.

4.4.1 Stage 1: Filling the Product Matrix

The product matrix is filled using a primitive production system model derived from examples found in *Artificial Intelligence Using C* (Schildt, 1987). The system employs the simplest form of conflict resolution possible — it uses the first applicable rule it can find. This permits some control over rule interaction as the ordering of rules in the knowledge base affects the sequence in which rules are fired.

Pseudocode for the matrix filling process is included below. The same activity is illustrated in figure 4.3.

```

while (there are incomplete product versions)
  make the next version the current product
  while (the current product has unlocated components)
    make the unlocated component the current component
    select the appropriate rule base
    repeat
      match, select, and execute rules
      alter product data, matrix, or create a new product version
    until (no more rules apply)
  end while
end while

```

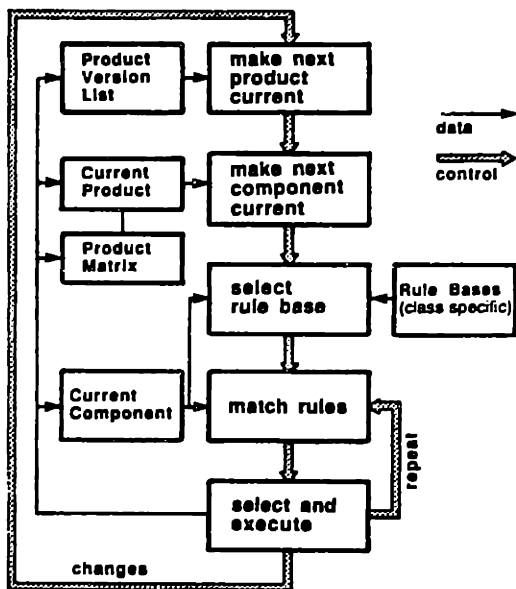


Figure 4.3 How the product matrix is filled

Initially, a single product version containing the components and attributes defined by the user exists. This version begins as the current product. The term current implies that the object is under active consideration. The system sequentially cycles through the list of components in the current product until all components are individually located within the product matrix. For each component a rule base is selected (as determined by the component's classification) and a pointer to the rule base is passed to the production system engine. The system matches, selects, and fires rules until the current component is fully positioned.

This event is recognized when no new rules apply to the current product data or matrix. Product data changes are limited to the current component's orientation. The current component's element position is recorded in the product matrix. The product matrix is discussed in the next section.

At any point in the process a new product version may be appended to the product version list. A new version is created when more than one acceptable component matrix location is found (an applicable either-or rule). The current product is copied into the new version in its present state of completion. The system then resumes to complete the current product. The new version or versions are resolved sequentially after completing the original product.

4.4.1.2 The Matrix Structure

The product matrix is a structure that exists within the organization level only. A unique product matrix is associated with each product version. When completed, the product matrix provides the information required by the aesthetic component positioning procedures. The product matrix establishes the conceptual spatial regions in which the components reside.

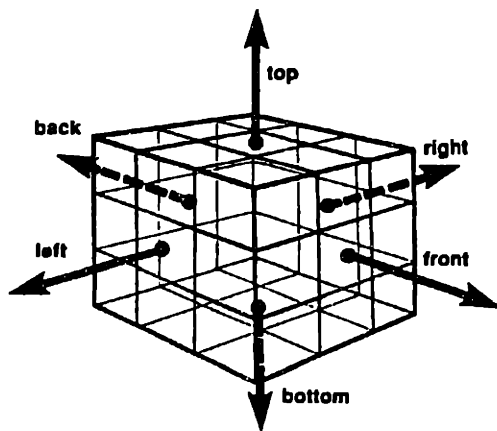


Figure 4.4 Faces and normals of the product matrix

The 'matrix' is implemented as a one dimensional array. Each element of the array is assigned to one of the six outward faces: top; front; left; right; back; and bottom. The faces and their normals are indicated in figure 4.4. The interior cell of the product matrix is not represented as, in the proof of concept, all components interact with the user. The components will reside in outer faces of the matrix.

Each face contains nine elements corresponding to the regions of the product matrix within the face. The nine elements are pointers to arrays of pointers. The arrays of pointers lead to the components located in the given region. A representation of the product matrix structure is included in figure 4.5. In this figure, the product contains three different components: two are located in the top left of the front face; and one is found in the bottom right of the back face.

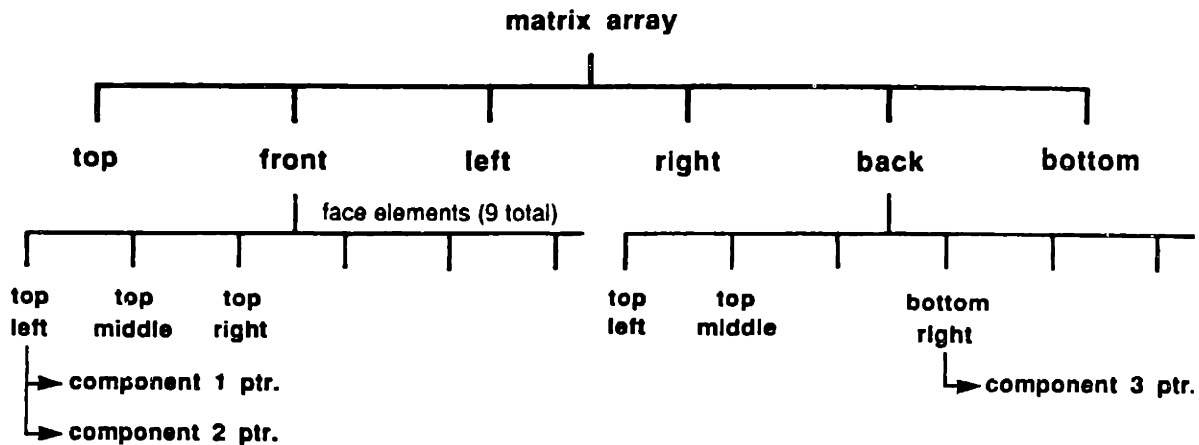


Figure 4.5 Structure representation of the product matrix

A component is fully located when it has been assigned to a face element and appropriately oriented for the chosen face. Placement in the matrix does not involve dimensional positioning of the components. The matrix only groups the components into the conceptual regions of the product identified previously.

4.4.1.2 Production Rules

Explicit rules are not provided here, but the rationale applied behind the matrix filling rules is disclosed. Initial efforts attempting to separate manufacturing and ergonomic rules were abandoned. In many instances, deciding a rule's classification seemed to be arbitrary. Consider a desk top product with a vertical parting plane. A visual display should face the front of the product both for ergonomics (so it can be viewed), and for manufacturability (to avoid side holes in the mould). Therefore, rule bases are delineated by component class only. This approach helps to avoid redundant rules.

A small ten to fifteen rule knowledge base is written for each class of component. The small rule bases facilitate the anticipation of rule interaction. Further, this organization allows for the easy addition of knowledge as the system expands.

The product matrix provides a useful mechanism for gathering rules in an intuitive manner. The first step in writing rules is to decide where a type of component belongs in the matrix. This decision is then used as a guide for writing the production rules. The product matrix should assist knowledge extraction when working with industrial designers. The matrix also helps to visualize the outcome of rules.

When writing rules it is useful to gather a large sample of products within the domain of the implementation. The examples are used to observe standard practices. Lists of manufacturing and ergonomics guidelines, such as those presented in chapter two, also aid the rule writing process.

4.4.1.2.1 Component Location in the Matrix

Before detailing where components belong in the matrix, it is necessary to make assumptions about how the user will interact with the product. These suppositions are based on the mould parting plane orientation. The parting plane will preferably pass through the secondary faces of a product. When designing a product with a horizontal parting plane, it is assumed that the primary visual face is the top. The sides of the product — which will feature a parting line — are secondary surfaces. This assumption is validated on both an aesthetic and manufacturing basis: it helps to conceal a potentially unsightly feature; and it reduces the need for a highly toleranced cosmetic parting plane. If the user is intended to interact with the front of the product a vertical parting plane is preferred. For a product like an answering machine — where the top of the product is the main surface — a horizontal parting plane is desired.

To avoid side holes, components placed in a secondary face of the product will be constrained to the parting plane. (Recall that the parting plane passes through the secondary faces.) This action allows side holes to be replaced with side slots. Side holes require the use of actuated pins in the mould, while side slots do not. Consequently, this restriction will help to reduce mould complexity.

In the present implementation, inter-component spatial relationships are not considered. The system does not acknowledge constraints such as components that require

adjacency. In situations where components must be assembled in a specific manner, they should be joined to form an assembly. The assembly is then included in the system's library as a single component.

The shaded areas of the product matrix in figure 4.6 indicate the desired placements for acoustic components.

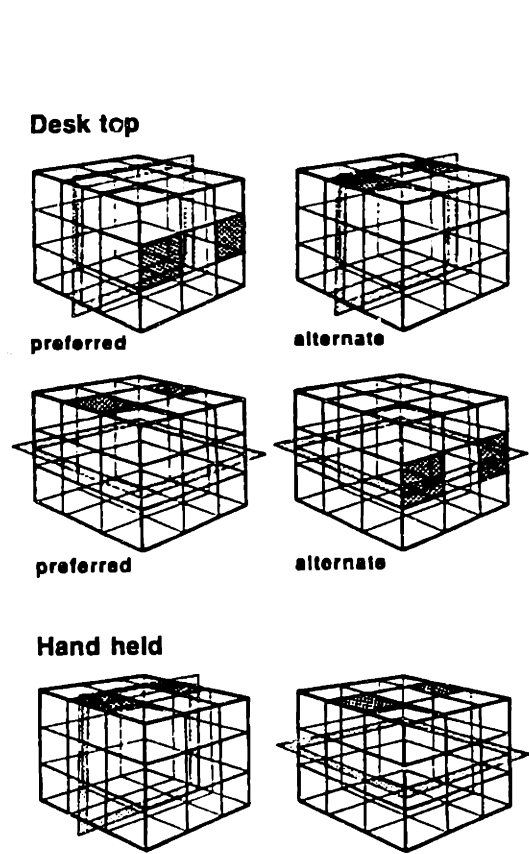


Figure 4.6 Location of acoustic components in the product matrix

In desk top products, acoustic components are placed in either the front left or right, or the top left or right. The preferred configurations reflect the parting line assumption discussed previously. The preferred configurations result in the so called *thin* designs. Alternative locations are acceptable but less desirable because they violate the parting line assumption. The alternative placements are used to create multiple product versions. Components in the alternative locations are constrained to the parting plane. The speakers face the assumed location of the user for ergonomic reasons.

For hand held products, ergonomic factors restrict the speakers to the top of the product. The definition of hand held products is limited to those used while being held in one hand. Although a vertical parting plane is permitted for hand held products, a horizontal orientation is preferred.

The left or right matrix position reflects standard practice derived through stereo considerations. The rules are written to maintain left/right symmetry for identical acoustic components. It is assumed that if two identical speakers are included in the design, they should be configured for stereo sound. This example demonstrates how functional considerations can be built into the system. For example, rules could be written to place components sensitive to electromagnetic interference away from electromagnetic sources.

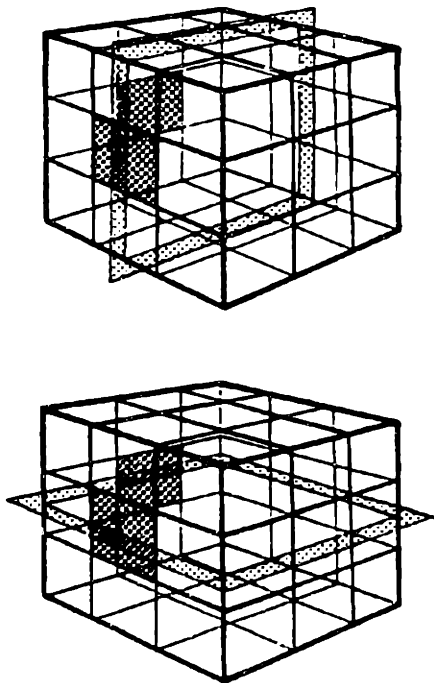
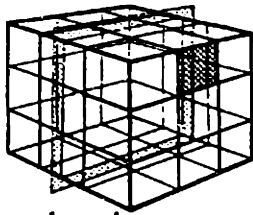


Figure 4.7 Location of jack components in the product matrix

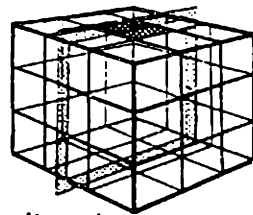
Jack matrix locations are indicated in figure 4.7. Their matrix positions are completely independent of product attributes (parting line orientation, use, or thickness). Jacks with the attribute *interface* are used frequently. Headphone or microphone connections are of this type. Interface jacks are placed in the side so that they are accessible but do not clutter the primary face of the product. *Service* jacks — such as power supplies or speaker terminals — are used seldomly. They are placed in the back of the product. A standard practice that delegates service jacks to the left is adopted.

Figure 4.8 is a similar presentation for the placement of visual displays. Visual displays are situated in the upper regions so that they will not be easily obscured during use or by debris. Like the acoustic components, the alternative configurations are considered when product thickness is not a concern. *Numeric* visual displays are treated as an exception. They are placed in the right front or right top face.

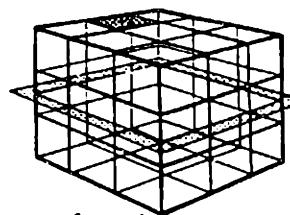
Desk top



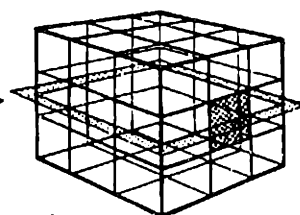
preferred



alternate



preferred



alternate

Hand held

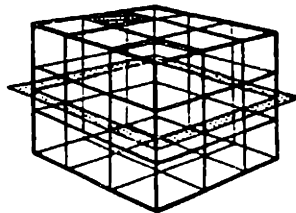
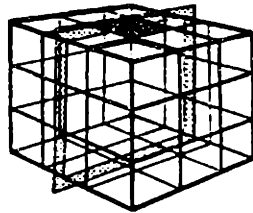


Figure 4.8 Location of visual display components in the product matrix

The regions assigned to keypads are indicated in figure 4.9. Keypads belong in the top of all products for ergonomic reasons. If a keypad is located in the front of a desk top product it will be difficult to press its buttons without moving the product. They are placed toward the front of the matrix (unless constrained by parting plane considerations) for easy access.

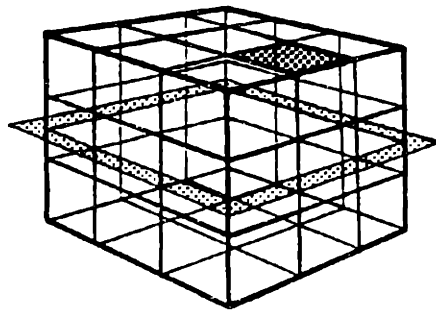
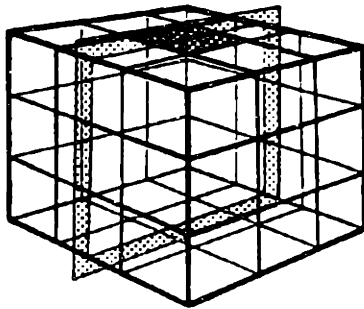
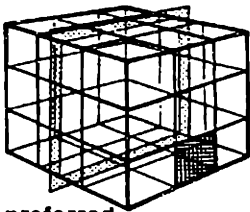


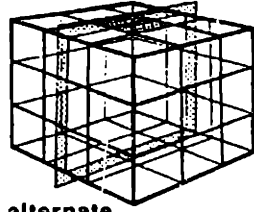
Figure 4.9 Location of keypad membrane components in the product matrix

It is necessary to consider mechanisms individually. The acceptable positions for cassette mechanisms are in figure 4.10. Figure 4.11 reveals the preferred switch locations. Switches are located using logic similar to that applied to keypads. Unlike keypads, the placement of toggle switches in the front of products is permissible.

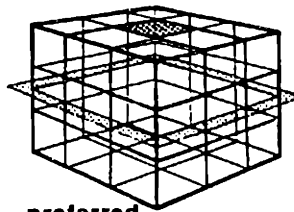
Desk top



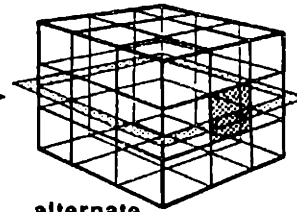
preferred



alternate



preferred



alternate

Hand held

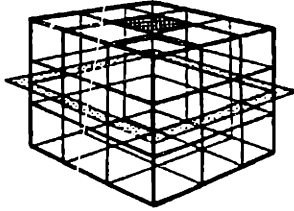
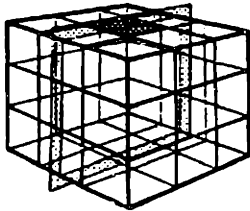
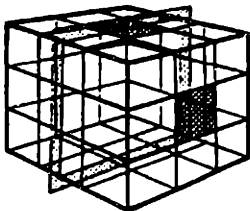
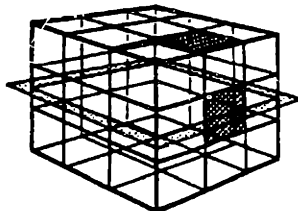


Figure 4.10 Location of cassette mechanism in the product matrix

Desk top



toggle only



toggle only

Hand held

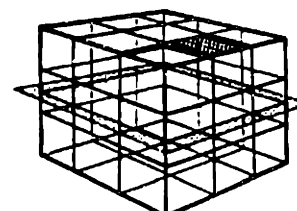
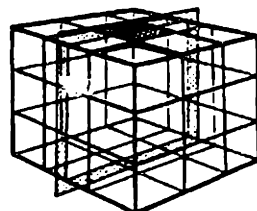


Figure 4.11 Location of switch components in the product matrix

4.4.1.2.2 Some Example Rules

The previous section demonstrates how the product matrix is used to extract knowledge for constructing rules. In this section a few illustrative examples of rules are included.

A typical orientation rule in the acoustic rule base is constructed as follows.

```

if ((acoustic component) and (product parting line horizontal) and (product use desk top) )
then
    component main surface faces the top of the product
    component front surface faces the front of the product
end if
    
```

The rule orients the component to suit the specific product data. The orientation data are stored in the product structure. The rotations required to correctly orient the component are determined using inverse kinematics formulae. The component's main functional surface and front are defined in the component data. The orientation of two perpendicular faces is needed to orient the part unambiguously. The previous selection stated that if thinness is not a concern an alternative orientation is acceptable. The remainder of this either-or rule is

```

or
if (product is not thin)
then
    create a new product version
    new version component main surface faces the front of the product
    new version component front surface faces the bottom of the product
end if

```

The **or** part of the rule adds another product version to the list of product configurations. The component is placed in its alternative position in the additional version. All alternative positions suggested in the preceding section are used to make new product variations when the product is not thin. The rules are written so that the system proposes only one thin product configuration.

The rule determining where an acoustic component belongs within a face must consider where other speakers reside in the matrix.

```

if (acoustic component)
then
    put component either left or right of product
end if

```

The action *put component in either left or right of product* checks for identical acoustic components already assigned to the product matrix. The new component is placed to create left/right symmetry. If symmetry cannot be maintained the component is, by default, put in the right element. (This rule reflects the stereo assumption discussed previously).

A final example illustrates how manufacturing constraints are manifested in a rule.

```

if (component main surface normal is parallel to the parting plane)
then
    constrain the component to the parting line
end if

```

Constraint to the parting line means that the component must be placed in matrix elements through which the parting plane passes. This rule is used to avoid side holes in the product housing design.

4.4.2 Stage 2: Aesthetic Positioning of the Components

The aesthetic positioning process begins with the completed product matrix. In chapter three concerns related to the aesthetic positioning of components is outlined (see figure 3.14).

Clearly, there is no singly correct method for designing a product to achieve a sense of organization, stability, rhythm, and balance. The goal of the thesis is to demonstrate that such formalisms can be devised. It is not to rigorously develop numerous design strategies. The implementation uses a single series of positioning procedures designed for box-like products. Several strategies will be needed in a more practical system. The procedures are a *bag of tricks* that should result in — but cannot guarantee — an adequately composed product. The procedures are designed using personal industrial design experience combined with a trial and error approach.

Once the most appropriate positioning strategies are selected, the aesthetic positioning process is procedural in nature. Unfortunately, ordered procedures do not offer the modularity of a production system approach. To achieve acceptable generality the aesthetic procedures do not distinguish between different classes of components.

A schematic of the positioning process is depicted in figure 4.12. Firstly, the program *estimates* the overall size of the product and chooses the best overall organization to match product attributes (height and use). Next, the system narrows its scope and positions components within individual product matrix elements. The completed matrix elements are shuffled to form the six faces of the product. Finally, the faces are positioned relative to each other and the product bounding box is created.

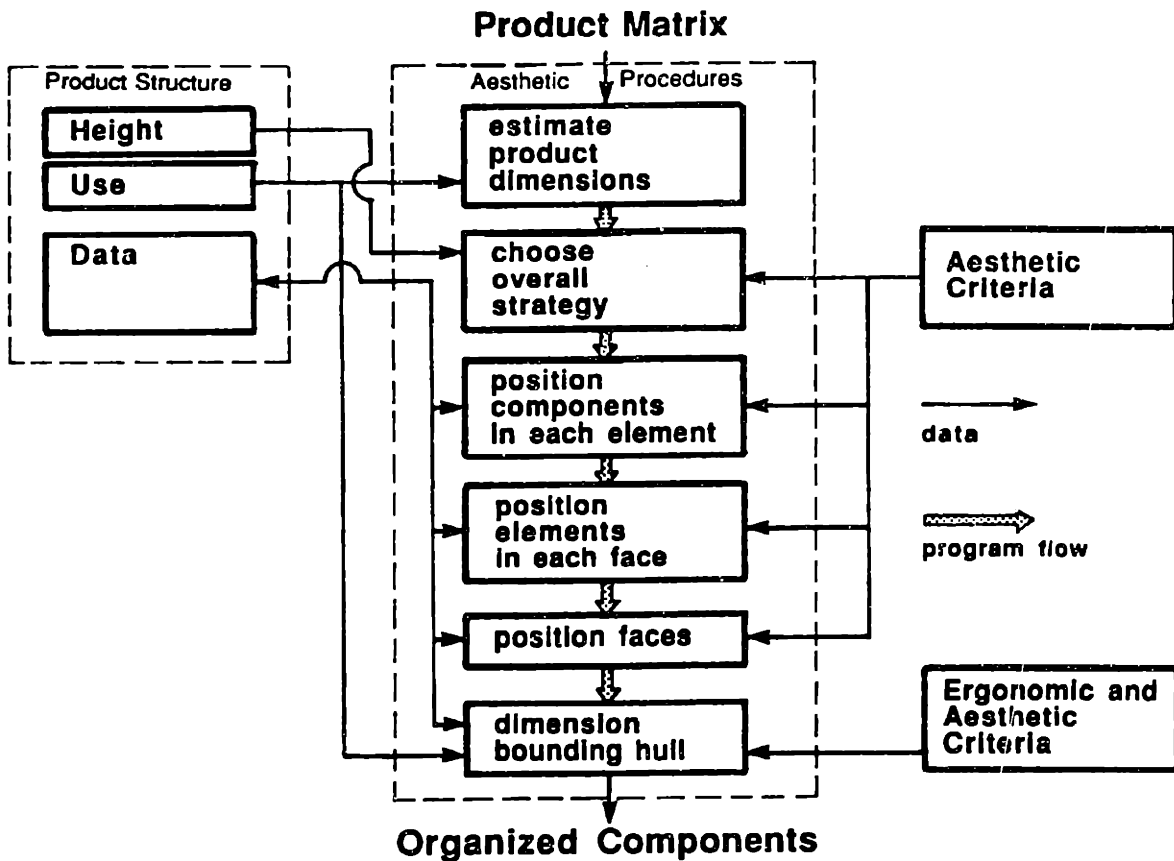


Figure 4.12 Component positioning procedures

4.4.2.1 Estimating the Product Size and Format

The first two blocks in figure 4.12, *estimate product dimensions* and *choose overall strategy* approximate the outcomes of different component organization strategies. This is done without making detailed and time consuming calculations. The task is performed in three steps.

- estimate the possible dimensions of each matrix element*
- estimate the possible dimensions of each face of the matrix*
- use aesthetic criteria to select the best outcome*

When estimating the dimensions of a matrix element, it is assumed that inter-component conflicts will be resolved by moving components in either of the two principle directions of the element's face. This idea is clarified in figures 4.13. Component

dimensions are summed to provide element size estimates for the two configurations shown in the figure.

matrix face

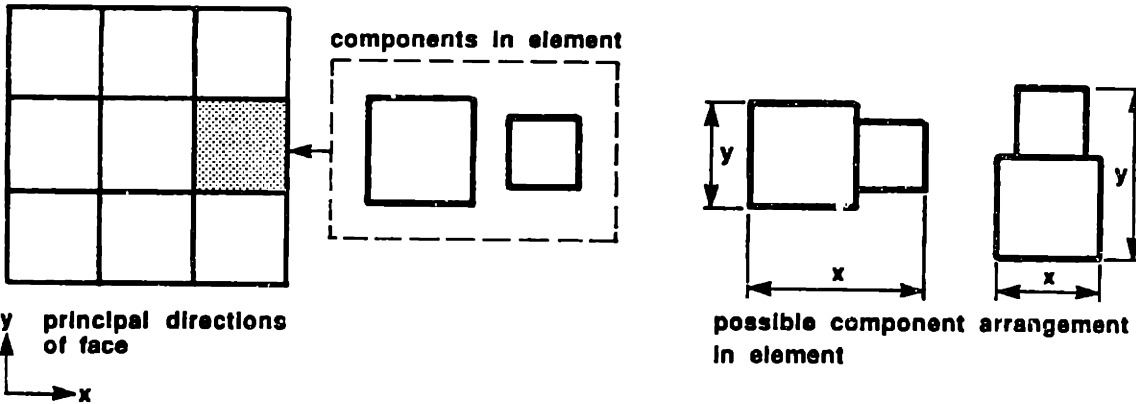


Figure 4.13 Estimating the size of a matrix element

Two sets of overall dimensions for each matrix face are calculated using the estimated element dimensions. Possible dimensions are estimated by combining elements in either a row or column organization, as shown in figure 4.14. Again, interference is resolved through movements along the principle directions of the face. Assembling the elements into rows or columns will create desirable horizontal or vertical visual directions in the product. This type of layout imparts a simple grid-like structure to the face.

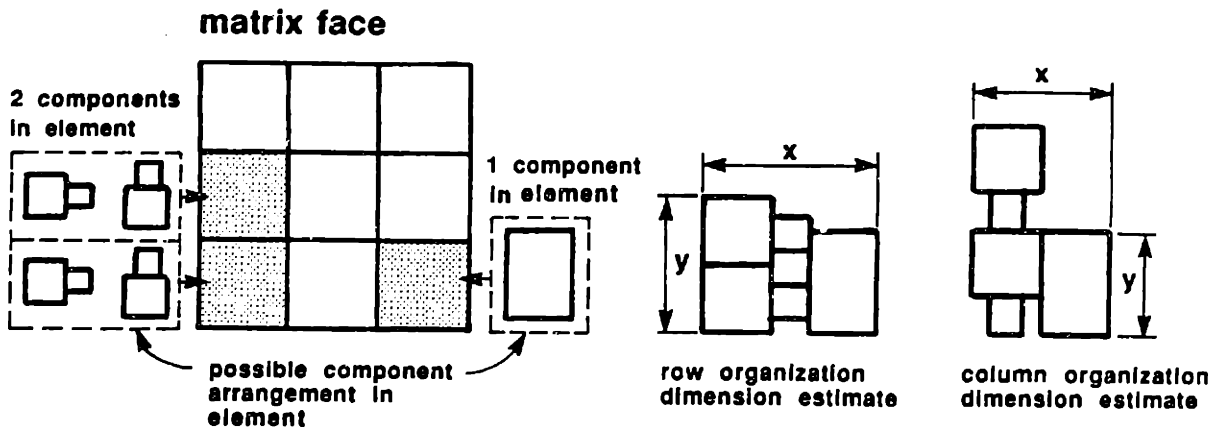


Figure 4.14 Estimating the size of a matrix face

The assumption that all elements in a face are organized in the same way (either columns or rows) is primitive. A more sophisticated approach would combine the matrix elements considering packing efficiency.

A series of aesthetic criteria use the estimated face dimensions to determine which format will result in the most appropriately sized product. The criteria reflect product attributes. The decision process begins by deciding the expansion direction for the primary product face.

```

If ((product use is desk top) and (product height is tall))
then
    If (front face of product has components in it)
    then
        choose the expansion direction that makes the face tallest
        If (row direction expansion height = column direction expansion height)
        then
            choose column expansion direction
        end If
    end If
    If (front face empty)
    then
        choose top face direction to make product deepest
    end If
end If

If ((product use is desk top) and (product height is normal))
then
    If (front face of product has components in it)
    then
        choose expansion direction that makes the face shortest
        If (row direction expansion height = column direction expansion height)
        then
            choose row expansion direction
        end If
    end If
    If (front face empty)
    then
        choose top face direction to make product widest
    end If
end If

```

For desk top products, the product height determines whether the front or the top face will govern the dimensions. In the ambiguous case where both column and row organization yield the same product height, the organization which aids the perception of the desired height is chosen. For products which do not have components in the front face, the system interprets tall as depth. Normal height products are configured with a broad base to enhance stability.

After selecting the primary face organization of the desk top product, the expansion directions for other matrix faces are chosen. The secondary face dimensions are chosen to best match the primary face.

```

if ((product use is desk top) and (the front face expansion direction is determined) )
then
    choose top, bottom, and back face expansion directions to best match the front
    face width
    choose the left and right face height to best match the front face height
else
if (the top face expansion direction is determined)
then
    choose front, bottom, and back face expansion directions to best match the top
    face width
    choose the left and right face expansion directions to best match the top face
    depth
end if

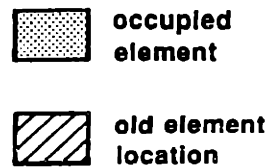
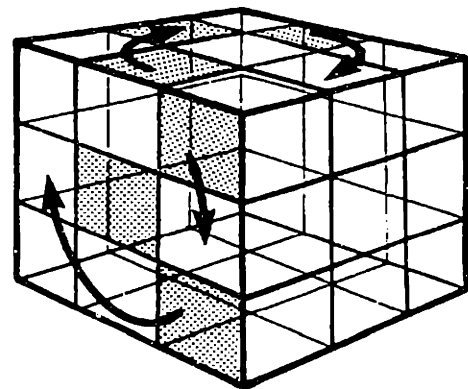
```

The expansion directions chosen for hand held products are, unlike desk top products, governed by ergonomic considerations. The program looks for the layout which most closely matches what can be held in a user's hand. The nominal hand held width is taken to be 2.5 inches (65 mm) square. If an acceptable layout cannot be found, product matrix elements are rearranged to try to find a better solution. An example of element swapping is illustrated in figure 4.15.

```

for (all matrix faces)
    choose the expansion direction which best matches the hand held width.
    if (the face's best dimension is greater than the hand held width)
    then
        condense the face into a single row
        re-estimate the element and face sizes of the product
        if (rearranged face dimensions match hand held width better)
        then
            permanently change matrix
            chose new best expansion direction
        end if
    end if
end for

```

Matrix faces prior to element rearrangement

→ direction of element movement

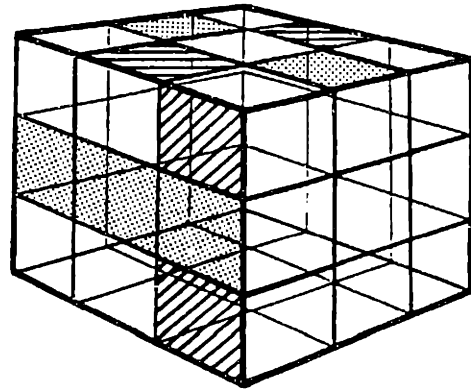
Re-arranged matrix

Figure 4.15 Swapping matrix elements in hand held products

4.4.2.2 Positioning Components within a Matrix Element

With the organization of each face decided (either rows or columns), the physical positioning of the components begins. This section describes actions that occur in the block *position components in each element* (refer to figure 4.12). At this stage each matrix element is considered independently.

Consider a single matrix element. If the element contains one component or fewer nothing needs to be done. Otherwise, a component spacing factor is calculated. The factor is a function of the average estimated length and width of the element. A component spacing of 0.05 times the average dimension is used by the program. The components are spaced by this amount along the chosen expansion direction. If the components in the element are constrained to the parting plane the expansion direction is forced to follow the parting plane. The components are centred on the axis of the expansion direction. Additionally, the parts are ordered by decreasing area. The pattern is sensitive to the element's location in the matrix. Examples are shown in figure 4.16.

The systematic spacing imparts organization to the matrix elements. Spacing variations between different elements form the components into sub-groups. Sub-groups help to avoid overly repetitious designs. Sorting the components by exposed area creates a unifying pattern or rhythm throughout the product and further reinforces a sense of grouping. The sorting pattern is designed to enhance the visual stability of the layout.

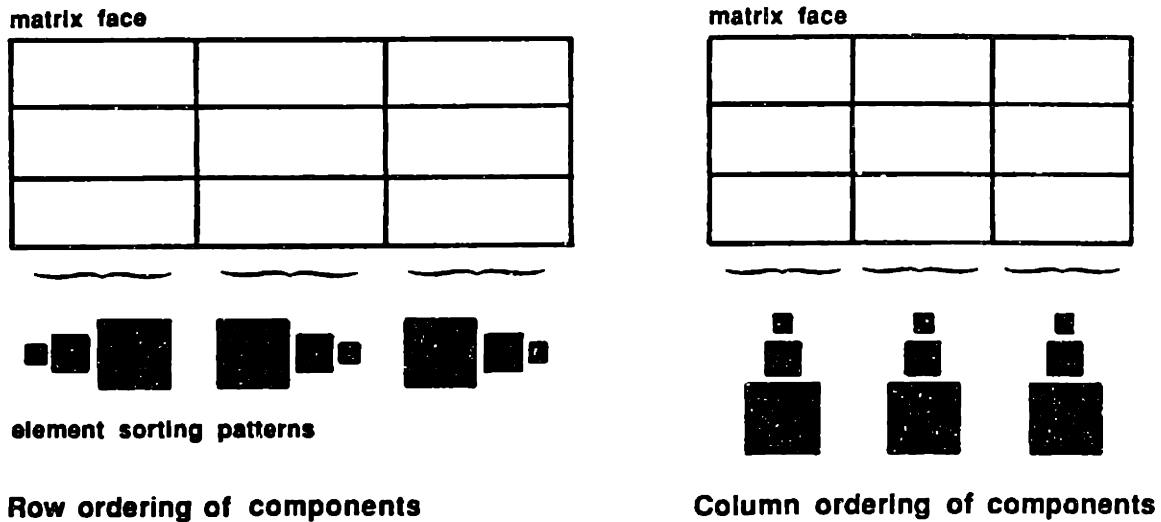


Figure 4.16 Context dependent ordering of components within a matrix element

4.4.2.3 Positioning Elements to Complete a Product Face

In the previous section, intra-element component positioning is performed. Subsequently, inter-element relationships are considered. Figure 4.12 indicates that the face layouts are designed in the module *position elements in each face*. The program moves the component arrangements within matrix elements as rigid bodies to individually layout the six product faces.

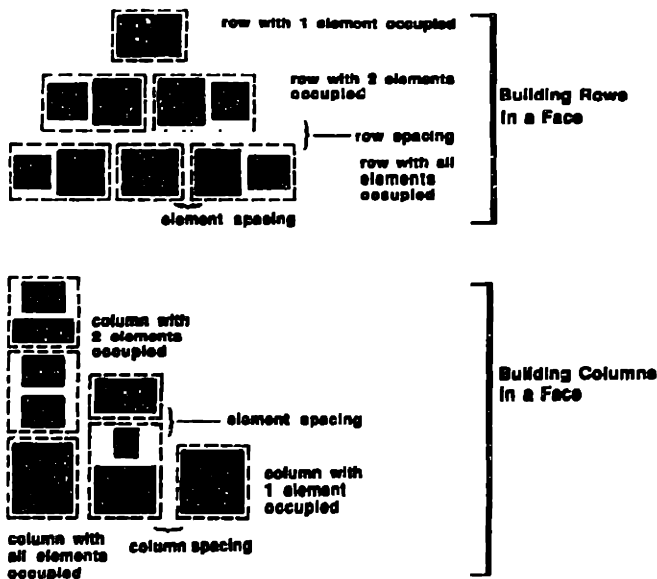


Figure 4.17 Building rows or columns in a matrix face

The product regions (or elements) in a face are assembled into either three rows or columns, as shown in figure 4.17. When creating the rows or columns, the spacing between individual regions is 0.05 times the estimated average dimension of the row or column. It is convenient to refer to a completed row or column as a face line. The lines are separated from each other by 0.05 times the estimated average face dimension.

Next, the lines in a given face are justified relative to each other. Positioning the lines fully locates components with respect to the product face in which they reside. The minimum overall dimensions of the product are computed and used to guide the alignment process. Options such as centre, left/right, or bottom justification — as illustrated in figure 4.18 — are used in the implementation.

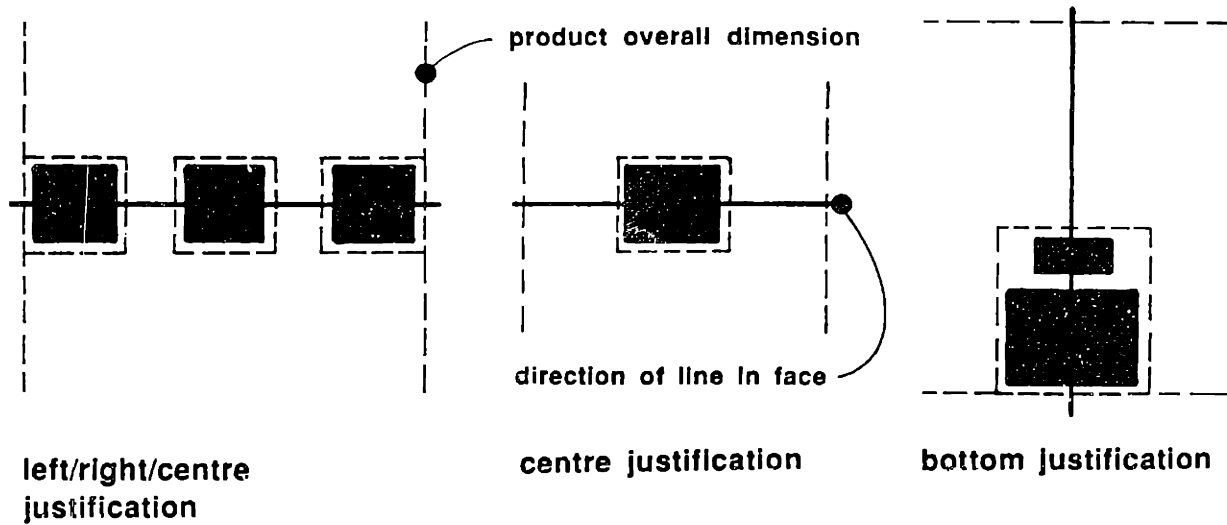


Figure 4.18 Row or column justification options

The justification decision criteria below are established by trial and error. Although they are quite simple, they perform adequately in the proof of concept.

```

if (line length less greater than the justification criterion)
then
    left, right centre justify the elements in the line
end if
else (the line is less than the justification criterion)
    if (working on front, back, left, or right face) and (vertical line))
    then
        bottom justify the line (a stability consideration)
    end if
    if (working on front, back, left, or right face) and (horizontal line))
    then
        centre justify the line
    end if
    if ((working on top or bottom face) and (line is front to back))
    then
        rear justify the line
    end if
    if ((working on top or bottom face) and (line is left to right))
    then
        centre justify the line
    end if
end else
    
```

The justification length criteria is adjusted by trial and error. The program uses one half of the overall product dimension in the line's direction. Justifying the lines relative to the product boundaries helps to create visual coherence between faces. Using a single justification criteria is overly simplistic. Considerable improvement will be accrued by increasing the formatting sophistication at this stage.

Using organizational units of rows and columns builds directional orientation into the product — which is desirable. The linear structure forms a higher level of grouping that can make a composition more interesting. Rows enhance visual stability in desk top products. Columns create the impression of height.

The mould parting plane is assumed to pass through the origin of the coordinate system, on which the product is centered. Each matrix face is centred about the origin so that lines constrained to the parting plane are correctly positioned. Finally, the components are translated so that the outer surfaces of each face are flat.

After completing the process above, component locations relative to their resident product face are fixed. All that remains is to position the faces relative to each other and to create a product bounding hull.

4.4.2.4 Positioning the Matrix Faces

The second last block in figure 4.12 — *position faces* — involves translating the now complete product faces until there are no conflicts between components in different faces. The faces are moved as rigid bodies in the direction of their outward normals, as indicated in figure 4.19.

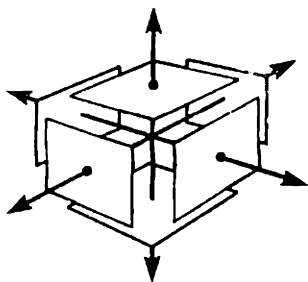


Figure 4.19 Direction of movement for product faces

As a first step, the faces are aligned to the minimum overall dimensions calculated previously. Then, the program checks for component interference between faces. When components in different faces intersect the program must determine which face should be moved. This decision is tricky — a poor choice can ruin the appearance of the product layout. (This was discovered while developing the program). An approach that works satisfactorily follows.

```

if (components in different faces interfere)
then
    if (interference involves a side face (left or right) with a face orthogonal to it)
    then
        if (side face movement < translation factor x other face movement)
        then
            move the side face
        else
            move the other involved face
        end if
    else {any other combinations of face intersections involved}
        move the face which requires the smallest change
    end if
end if

```

For the components provided in the proof of concept a translation factor of 2 yields satisfactory results. The factor reflects a preference to increase the depth or height of a product over increasing frontal length.

4.4.2.5 Building the Bounding Hull

Finally, the process *dimension bounding hull* completes the component organization level (see figure 4.12). The bounding hull is the underpinning to which the product surface will be mapped. The hull determines the overall proportions of a product and should help make the design appear stable. Ergonomics can also play a role in hull generation.

Before building the hull, the overall dimensions of the product — based upon finalized component locations — are computed. The hull is sized so that there is an outer border around the components. The boarder is 0.05 times an appropriate average product dimension. (The value 0.05 is chosen for consistency with other spacing conventions). The pseudocode below describes how the average dimensions are chosen.

```

if (parting plane is vertical)
then
    x average = front face average dimension
    y average = front face average dimension.
    z average = lesser of top face or side face average dimension
end if
if (parting plane is horizontal)
then
    x average = top face average dimension
    y average = top face average dimension.
    z average = lesser of front face or side face average dimension
end if

```

The criteria above are determined through trial and error. The hull dimensions are calculated by adding the x, y, and z borders to the product dimensions defined by the component locations. The hull is then centred about the component layout.

In the last step of the process the hull is checked for adequacy. Desk top products are checked for stability. Product bounding hulls are altered if required. If the product depth is less than 0.6 times its height the design will look disturbingly unstable. The factor 0.6 is chosen to approximate the golden section. Hand held products are checked to see if keypads or switches are in the front of the product. If there are, additional grasp space is provided (1.5 inches or 38 mm). If the extra space is not provided uncomfortable extension of the thumb will be required to press the buttons (consider the design of a television remote, for example).

The factors accounted for during the construction of the bounding box are far from thorough. They serve to illustrate the types of issues that should be considered when establishing the overall size of a product. Computing the bounding hull completes the activities of the organization level of the model. The next step, the surface level, generates an enclosure for the product design.

4.5 The Surface Level

The surface level of the system uses product attributes to select pre-designed enclosure prototypes for mapping to product hulls. The process generates monolithic enclosures for individual products. This level of the program is designed to be as straight forward as possible. It is felt that a simple approach will simplify the addition of styles to the library. Three different styles and twelve prototypic box-like housings are included in the proof of concept enclosure library (2 prototypes for each product use in every style).

The program proposes only one surface for a given component layout. This simplification eliminates the need for a complicated prototype selection mechanism. Multiple prototypes will allow the program to suggest surface design alternatives. In many instances multiple prototypes for a given style and use are desired.

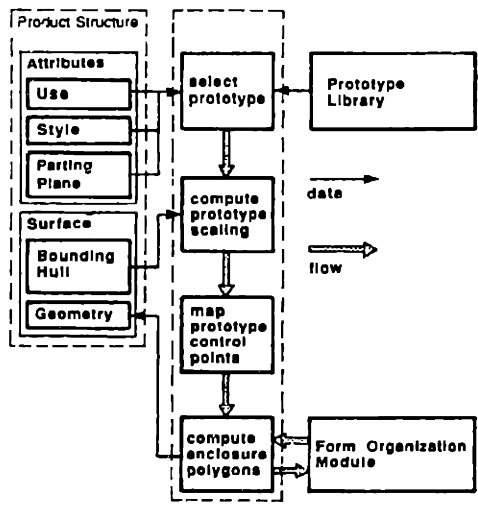


Figure 4.20 Diagram of the surface level implementation

A high level description of the operation of the surface level program is outlined below and illustrated in figure 4.20.

select an enclosure prototype
compute the scaling function
map surface control points to the product hull
compute the surface and update the product

The program reads a suitable prototype from the surface library. The required scaling is calculated by comparing the prototype to the bounding hull. The prototype surface control points are transformed, and then the form organization module is called to loft the enclosure surface.

The surface geometry (in the form of directed polygons) is then copied into the product data.

An introduction to the three styles available in the proof of concept, designing form organization surface files, creating the prototype library, and the mapping function follows.

4.5.1 Prototype Styles

The three styles provided in the proof of concept are Art Deco, High Tech, and Braun-like. Archetypes for the three styles are provided in chapter five (figures 5.3 to 5.5). The monolithic surface prototypes for the different styles are shown in appendix B. The thesis will not dwell upon the history of the styles. Let it suffice that Art Deco is a collection of styles fashionable in the 1920's and 1930's. Old refrigerators or juke boxes are often in this style. High Tech is a utilitarian architectural style of the 1970's and is characterized by hard edges with primary coloured elements. The Braun-like style refers to the strong corporate identity established by Dieter Rams. A brief discussion of these style and many others may be found in *The Conran Directory of Design* (Bayley, ed., 1985)

4.5.2 Designing Surfaces with Form Organization

The prototype enclosures used by the implementation are designed with the form organization method presented in chapter 3. A typical design sketch showing surface control lines — in this case for a Braun-like enclosure — is included in figure 4.21. Recall that the four control lines define conic cross sections of the surface. The control line data are measured from the drawing directly. The control points are encoded in an ASCII character file for use by the form organization module. A surface is lofted from this data. (see chapter 3, section 3.5.3 for details). As form organization is not central to the four level model for form giving, the surface computation module will not be discussed. Only the format to encode a surface for use by the program is provided.

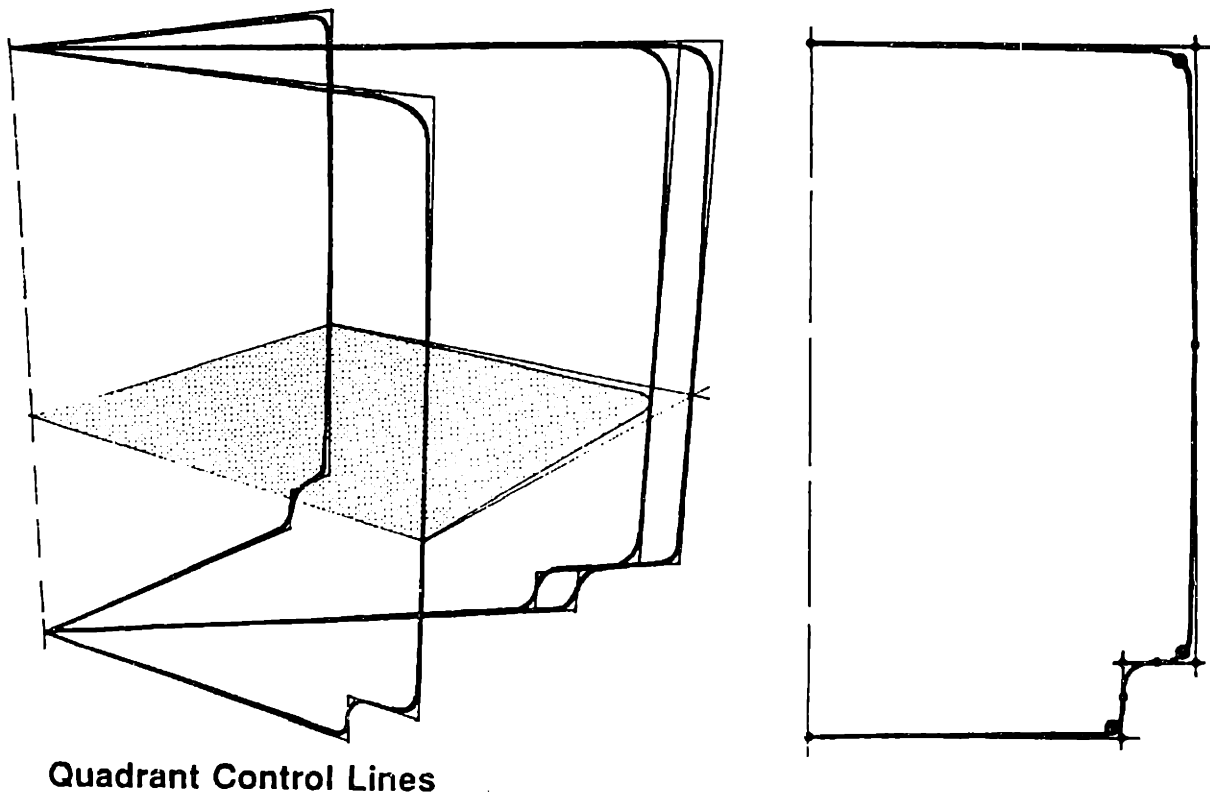


Figure 4.21 Form organization control lines for designing the Braun-like enclosure

The surface control line data are encoded in a file structured as shown in figure 4.22. Although the format is specific to the form organization module, the file structure should help to clarify how the method is used.

```

int    control curve which determines the spacing of cross sections
      ('A' line = 0, 'T' line = 1, 'B' line = 2, 'P' line = 3)
int    flip first cross section normals
      flip last cross section normals
int    surface colour index
int    number of surface points to be computed on each cross section
int    slicing method
      (x axis = 1, y axis = 2, z axis = 3, radial axis = 4)
int    rotation axis (x axis = 1, y axis = 2, z axis = 3)
float  point on axis of rotation (x y z)

'A' control line data points

int    number of conic segments in the control curve
      for each segment in the control curve
float  'A' point (x y z)
      'T' point (x y z)
      'B' point (x y z)
      'P' point (x y z)
int    number of data points to compute on the control curve segment

'T' control line data points (repeat encoding scheme as above)

'B' control line data points (repeat encoding scheme as above)

'P' control line data points (repeat encoding scheme as above)

```

Figure 4.22 Form organization data to encode a surface

An integer index determines which control curve governs the number and location of surface cross sections calculated. The second set of instructional codes are binary indices (1, 0) which indicate if the surface normals are to be reversed on the first or last cross sections. The indices are necessary to resolve ambiguous cases when control lines converge to a point. A colour index is used to assign a colour to the surface polygons. In the program implemented a given surface is limited to a single colour.

The slicing method index instructs how the surface cross sections are to be oriented. The slicing cases implemented presently are shown in figure 4.23. When using radial slicing, the orientation of the rotation axis and its location must be specified.

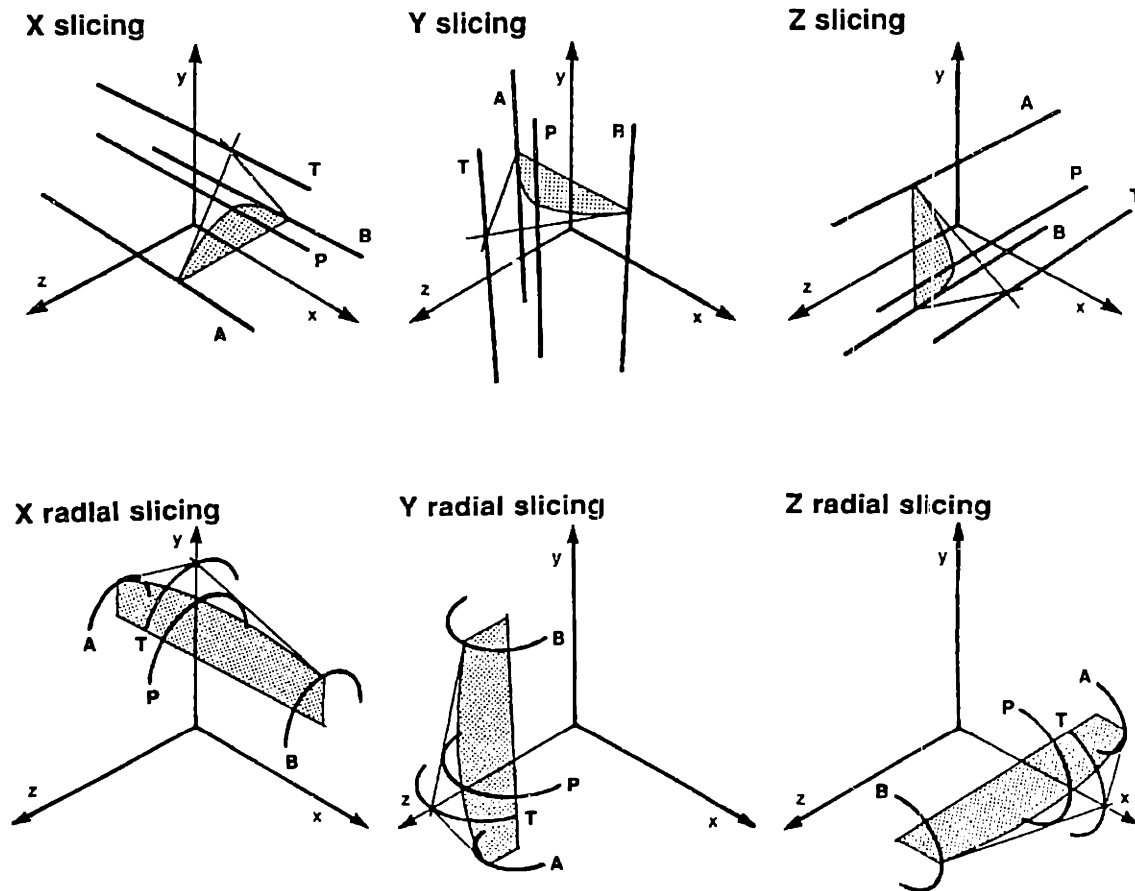


Figure 4.23 Control curve slicing methods in the form organization module

The major weakness of the form organization method should be very apparent at this point. Each slicing method must be encoded as a special case. Further, the designer must plan the control lines for a specific slicing method.

The control curve data follows after the surface lofting instructions. Control lines may be defined using any number of piecewise continuous conic segments.

This section has provided the information needed to define a general conic surface. Conic surface files are part of all prototypes found in the system's style libraries. Before discussing the enclosure prototype library, it is instructive to consider the characteristics of enclosure prototypes.

4.5.3 Using the Surface Prototype Library

4.5.3.1 Prototype Anatomy and Selection of a Prototype

The enclosure anatomy is tempered by three factors: style; use; and parting plane orientation. This section explains how these factors influence the design and selection of prototypes.

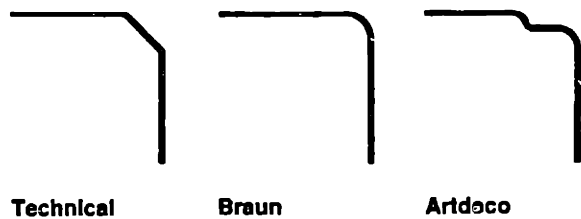


Figure 4.24 Corners characteristic of the three styles

In the proof of concept, style affects the design of enclosure corner treatments. Characteristic corners for the three styles implemented are in figure 4.24. The enclosure designs are restricted to having almost flat faces. This allows components to be placed anywhere within a product face.

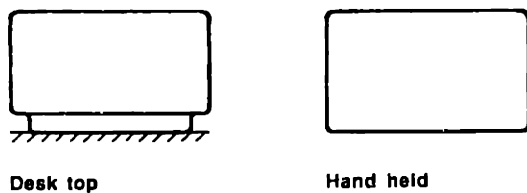


Figure 4.25 Desk top and hand held enclosures

The product's use determines whether the enclosure will have a foot. Desk top housings have a small foot. The foot raises the product off of the ground, creating an impression of lightness (see figure 4.25).

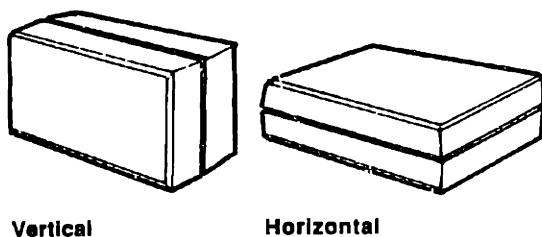


Figure 4.26 Enclosures with horizontal and vertical parting lines

The parting plane determines how the housing will be divided into halves. Draft angles must be placed according to the mould draw. Additionally, the parting line determines the product's primary face (refer to section 4.4.1.2.1). The location of the primary face may affect the orientation of surface edge treatments. An example of how parting plane orientation affects enclosure design is shown in figure 4.26.

In the proof of concept, a specific prototype is provided in each style for all of the use and parting plane alternatives. This minimizes the size of implementation required to demonstrate the surface level of the model. A generative approach capable of adding a foot or parting plane as required by the product attributes would be more powerful. The prototypes available in the surface library are in appendix B.

Providing prototypes for all combinations of product attributes trivializes the selection process. It is not necessary to define prototype applicability attributes as suggested in figure 3.17. The prototype name provides the necessary information. A library file such as *artdeco.handheld.horizontal* fully identifies the prototype's intended application.

```
if ((style is Art Deco) and (product use is hand held) and (parting plane is horizontal))
then
    select prototype file named artdeco.handheld.horizontal
end if
```

4.5.3.2 Mapping of the Surface Prototype

After selecting and reading the enclosure prototype file, the surface is transformed or mapped to fit the product. The product bounding hull and information in the prototype file contain the information needed for the process. The prototype files are described in figure 4.27.

<i>file name</i>	<i>styleName.use.partingPlaneOrientation</i>	
<i>int</i>	<i>number of surfaces in enclosure</i>	
<i>int</i>	<i>number of surfaces in top of housing</i>	
<i>float</i>	<i>normalizing scale</i>	
<i>float</i>	<i>top housing overall dimensions (x y z)</i>	
	<i>bottom housing overall dimensions (x y z)</i>	
<i>mapping points</i>	<i>top housing</i>	<i>maximum (x y z)</i>
		<i>minimum (x y z)</i>
	<i>bottom housing</i>	<i>maximum (x y z)</i>
		<i>minimum (x y z)</i>
<i>surface control line data as per figure 4.22 for each surface</i>		
<i>put surfaces in the top of the housing first</i>		

Figure 4.27 The enclosure prototype library file

The file name is used to identify the appropriate enclosure. The conventions and nomenclature for the prototype enclosure file are clarified in figure 4.28.

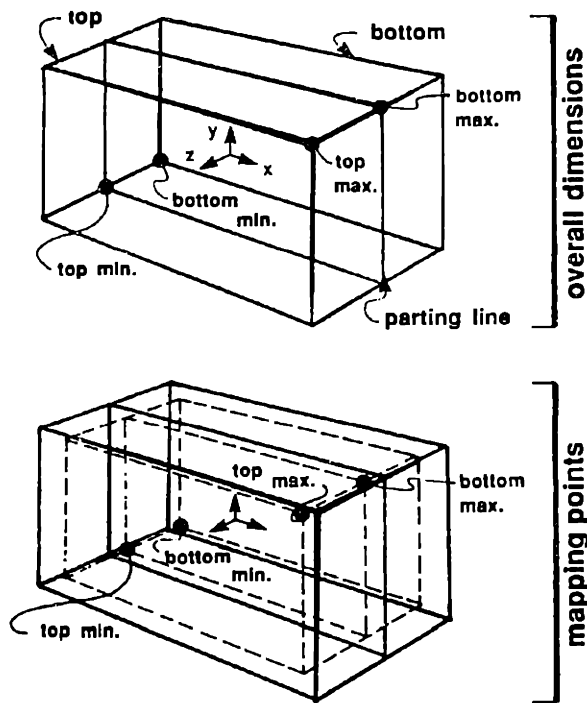


Figure 4.28 Enclosure prototype nomenclature

Each enclosure has a top and a bottom housing. The parting planes are, by convention, defined so that they pass through the origin. The normalizing scale permits prototypes to be designed and encoded at an enlarged size to reduce drafting board measurement errors. Prototype dimensions should be the same order of magnitude as typical products to which they will be applied. This helps to avoid difficulties with scaling effects. The complete overall dimensions of the top and bottom housing are provided in the file, but could just as easily be computed by the program.

The mapping points are used to fit the enclosure to the product bounding hull. The prototype must be transformed so that these points coincide with the hull vertices. The top and bottom halves of the enclosure (which are individual parts) are positioned independently and thus require separate mapping data.

When mapping either the top or the bottom housing prototype to the hull, the overall factors (s_x , s_y , and s_z) for x, y, and z scaling are computed as the ratio of the hull dimensions to the mapping point dimensions. The x values of the control lines are scaled using the following transformation.

$$x' = s_x * x \left\{ 1 + f \left[1 + \frac{|x|}{\text{overall } x} \right] \right\} \tag{4.1}$$

$$\begin{aligned} \text{where } f &= 1 - \frac{1}{s_x} && \text{for } s_x > 1.0 \\ f &= s_x - 1 && \text{for } s_x < 1.0 \end{aligned}$$

Similar transformations are performed on the y and z control point data. The scaling function's behavior is illustrated in the normalized plot of figure 4.29.

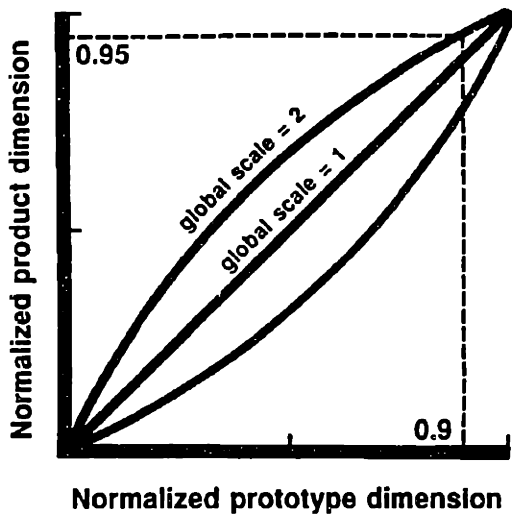


Figure 4.29 Characteristics of the scaling function

When the overall scale is greater than one the outer extremes of the enclosure are compressed. For scale factors less than unity the outer extremes are expanded. The function adaptively compresses or expands prototype corner details based upon the degree of global scaling. A way of visualizing this action is to imagine that the prototype enclosure is made of a deformable material. The material stiffness increases near the edges of the enclosure. Thus, when the enclosure is stretched, the edge regions expand less than the interior area — the corner features become proportionally smaller. Similarly, when the prototype is compressed, the corners contract less than the interior. Consequently, the corners will be proportionally larger than in the undeformed prototype.

The function is established empirically. The previous chapter states that a different scaling function is required for each style (section 3.6). Although believed to be generally correct, one function is adequate here because the housings are so simple. Even still, the scaling function defined above performs best when applied to hard edged styles (such as High Tech). It is not as good for scaling softer forms (such as Art Deco or Braun-like)

After modifying the enclosure control points, the housing is computed by the form organization module and the surface polygons are entered into the product structure. Another approach to scaling the surface would first compute the surface polygons, and then scale the vertices and normals to fit the product. This approach would make the scaling process independent from surface generation method.

4.6 The Detail and Graphical Levels

The detail level of the model adds pre-defined style specific elements — such as buttons or grills — to complete the product surface. The details added to the surface will vary with the components in the design. The graphical level of the model finishes the design by applying surface embellishments. Here, graphical treatment is limited to surface colouring.

In this thesis, practical considerations dictate simplification of the detail and graphical levels. The two levels are combined into a single step. The details are coloured as they are computed. This eliminates the need to write colouring rules for the graphical level. The colour choices are contained within the prototype detail's surface definition or in the procedures that add the details. A single palette of colours was developed for each of the three styles.

The details are restricted to convex shapes that stand proud of the enclosure surface. This aesthetic compromise eliminates the need to compute and trim the enclosure/detail intersections. Only the outermost surfaces are displayed.

In the following sections, the construction of the detail library and the operation of detail and graphics program are described.

4.6.1 The Detail Library

Prototypical details are used to encode different styles and types of details. Details must be designed to satisfy the needs of the components available to the system. The details required in the proof of concept are: a speaker grill for acoustic components; a panel insert; a keypad button; a press and toggle switch button; a jack cover; visual display masks or bezels; and a cassette mechanism door. These details must be available in the three styles. A complete visual reference for the details library is included in appendix C.

Like the surface prototypes, details are designed on the drafting board using the form organization technique. Once the scaling factors for a detail are determined, the computing process is identical to that executed in the surface level. The file format for a detail prototype is given in figure 4.30.

<i>file name</i>	<i>styleName.detailName</i>
<i>int</i>	<i>number of surfaces in detail</i>
<i>float</i>	<i>normalizing scale</i>
<i>mapping points</i>	<i>maximum (x y z)</i> <i>minimum (x y z)</i>
<i>float</i>	<i>main direction (x,y,z)</i> <i>front direction (x y z)</i>
<i>float</i>	<i>location of detail relative to the surface</i>
<i>surface control line data as per figure 4.22 for each surface in the detail</i>	

Figure 4.30 The detail file format

The name of the detail identifies its application. The normalizing scale is used in the same manner as described in section 4.5.3.2.

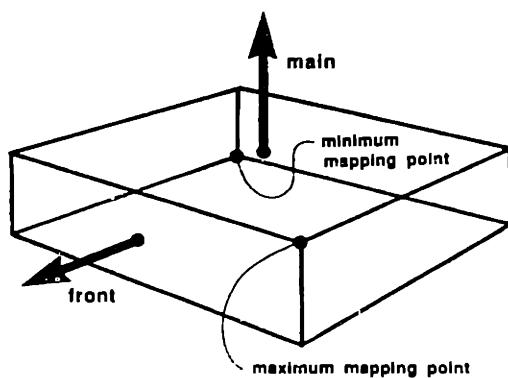


Figure 4.31 Nomenclature for the detail file

The details are one sided so only one set of mapping points is required. The mapping points are simply the maximum and minimum dimensions. The main and front directions are used to orient the details relative to components and enclosure. The height that the detail is to be positioned above the surface is included in the file also. The nomenclature for the detail file is clarified in figure 4.31.

4.6.2 Operation of the Detail and Graphics Level

Ideally, the detail and graphical levels will combine the rule based approach of the organization level with style encoding detail prototypes. The size and location of details will be decided using production rules. The prototypes are then scaled to meet the product's needs. More production rules will be required to assign surface colours. The process implemented is highly simplified. The detailing and colouring procedure executed in the thesis may be described as follows.

```

for (each region of the product matrix)
  form a set of components in the region
  sub-group like and adjacent components in region
  for (each sub-group of like components in region)
    if (components cannot be grouped into one detail)
      for (each component in sub-group)
        determine correct detailing function
        compute detail and add to product data
      end for
    end if
  else
    determine the best size for detail
    determine correct detailing function
    compute details and add to product data
  end else
end for
end for
apply base colour to the enclosure surface

```

The organization level matrix assignments are used to guide the detailing process. When possible, components in the same element are grouped into a single detail. The program does not consider grouping components in different faces or between elements of the product matrix.

Detailing is performed on an element by element basis. Sub-groups of adjacent components of the same class are formed within the element. For each sub-group, the program determines if the components should be unified under a single detail. Components such as visual displays and mechanisms are not candidates for grouping. Their details must fit the components exactly. The details for these components are computed directly and added to the product data. A specific detailing function is written for each class of component.

Components with the class of acoustic, jack, switch, or keypad can be grouped under common grills or panel inserts. A sub-group of components will be of only one class. The system uses an aesthetic criterion to decide the best size of the detail that groups the components. The common detail is added to the enclosure and then details for each component within the group are computed as required. For example, two keypads might be grouped in a panel insert. In addition to computing the insert, the system must add buttons to both of the keypads.

After the details are computed for every element of the product matrix, the design is completed with the style specific colouring of the housing surface.

The remainder of this chapter first describes how the size of the details are determined and then how the details are computed.

4.6.3 Determining the Detail Size

The previous section states that a component can require either a fixed size detail or be grouped with other components. Fixed size details are designed for specific components. For example, individual details are designed for the 12 character LCD, the 30 character LCD, and the CRT. Sizing such details is a non-issue (the scale factor is unity).

Establishing the size of a detail that groups several components — say a speaker grill covering several speakers — is a more interesting problem. A very simple approach intended to enhance the organization and unity of the product is implemented as follows

```

for (each direction in the resident face of the sub-group)
    determine if any components are between the sub-group and the product hull
    if (components are in the way)
        set the detail boundary to the edge of the sub-group
    end if
    else
        if (the distance to the hull is less than criterion)
            set the detail boundary to the product hull
        end if
        else
            set the detail boundary to the edge of the sub-group
        end else
    end else
end for

```

The decision process is demonstrated in figure 4.32. If the distance from the edge of the sub-group to the hull is less than the justification factor times the dimension of the sub-group, the detail is extended to the hull of the product. A factor of 0.7 is used in the program. The value of the justification factor is derived empirically for satisfactory results.

When the size of the detail is determined, component class specific detailing functions are invoked.

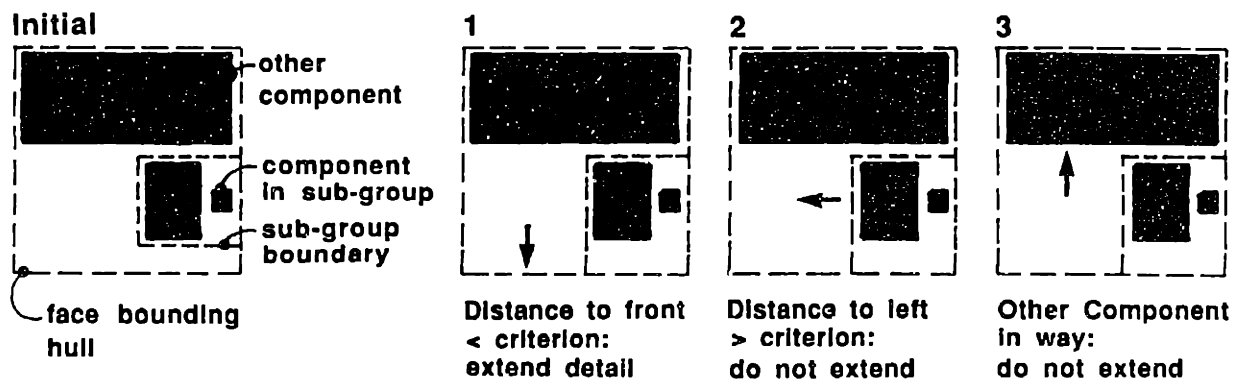


Figure 4.32 Example of the decision process for sizing a detail

4.6.4 Computing Details

After the detail size is established, the system calls class specific functions to apply the necessary details. The general approach involves: reading detail prototypes from the library; scaling control points using the style preserving scaling function; computing the detail surfaces; and orienting and positioning the details relative to the surface. The functions for each component class are outlined in the following sub-sections.

4.6.4.1 Acoustic Class Details

Acoustic components require grills. The grills may be sized arbitrarily to cover a collection of components. The detailing function is passed the detail's size and location. Also, the position of the housing surface under the detail is needed. Colour information is contained in the detail file.

*select correctly styled speaker grill file
 compute the detail
 orient the detail to match the components in the group
 position the detail relative to the surface
 add the detail to the product data
 label components in the sub-group as detailed*

The compute detail module first determines the scaling required for mapping the detail to the desired location and size. The overall scale factors are used by the surface level scaling function to transform the detail control points. The form organization module

written for the surface level of the program is called to loft the detail surface. This module returns the detail's surface polygons.

The detail surface is oriented so that its main and front directions match the components in the sub-group. This is performed by the inverse kinematics and rotation routines written for the organization level. Finally, the detail is positioned relative to its resident housing face.

4.6.4.2 Visual Display Class and Mechanism Details

Finishing masks are required for visual displays. Different masks are designed for each component. Similarly, the cassette mechanisms have fixed door designs. The detailing functions for the two classes are essentially the same.

```

select correctly styled file for specific component type
compute the detail
orient the detail to match the component
position the detail relative to the surface
add the detail to the product data
for (each button location in the component)
    select correctly styled and type of button file
    compute, orient, and position the button
    colour the button for style dependent button colours
    add button to product detail data
end for
reposition the component relative to the surface
label component as detailed

```

Prototypes for these details do not required scaling. The detail colour is encoded in the prototype file. Some components — such as a radio tuner or cassette — will require buttons in addition to a mask or door. If buttons are needed the routine adds the required number of fixed size keys to the detail. Additionally, the component must be repositioned to mate with the detail properly.

4.6.4.3 Keypad, Jack, and Switch Class Details

The detailing process for keypads, jacks, and switches is similar. Sub-groups of these components may be located under a single panel insert sized using the criterion previously discussed. Fixed size buttons or details are added to individual components in the sub-group. The system does not apply a panel insert if the sub-group contains only one component with one button or fewer.

```

If (panel insert desired)
    select correctly styled file for panel insert
    compute the detail
    orient the detail to match the component
    position the detail relative to the surface
    add the detail to the product data
end If
for (each component in the sub-group)
    for (each button location in the component)
        select correctly styled and type of button file
        compute, orient, and position the button
        colour the button for style and component dependent button colours
        add button to product detail data
    end for
    reposition the component relative to the surface
    label component as detailed
end for

```

When detailing jacks, finishing covers — not buttons — are placed over the components.

5 RESULTS AND DISCUSSION

In this chapter the performance of the system is reviewed, critiqued, and, in some instances, improvements are suggested. The primary goal is to demonstrate the feasibility of the industrial design model using examples in the chosen domain of consumer electronics. It is necessary to show that the aesthetic system of the proposed model is capable of creating designs which are both meaningful and exhibit distinctive styles over a range of products.

The first three sections of this chapter are devoted to demonstrating capabilities with respect to such general questions. The subsequent sections then proceed to review the performance of each of the four levels of the aesthetic system with regard to many of the smaller questions and problems introduced through the course of the thesis. The results are obtained through observation of the graphical output from the proof of concept implementation.

5.1 The Capture of Style using the Four Level Model

Figures 5.1 and 5.2 show two products, a portable stereo and a simple calculator, in all three of the styles afforded by the proof of concept. The figures demonstrate the ability to create recognizable styles for a single component configuration and, to generalize a single style over different component layouts or products. This capability supports the hypothesis that the proposed four level approach to form giving can indeed capture style. While the component layout establishes what kind of product the human observer perceives, the surface, detailing, and product colour are very important in establishing the style. These styles can be compared with their archetypes which are presented in section 5.1.1.

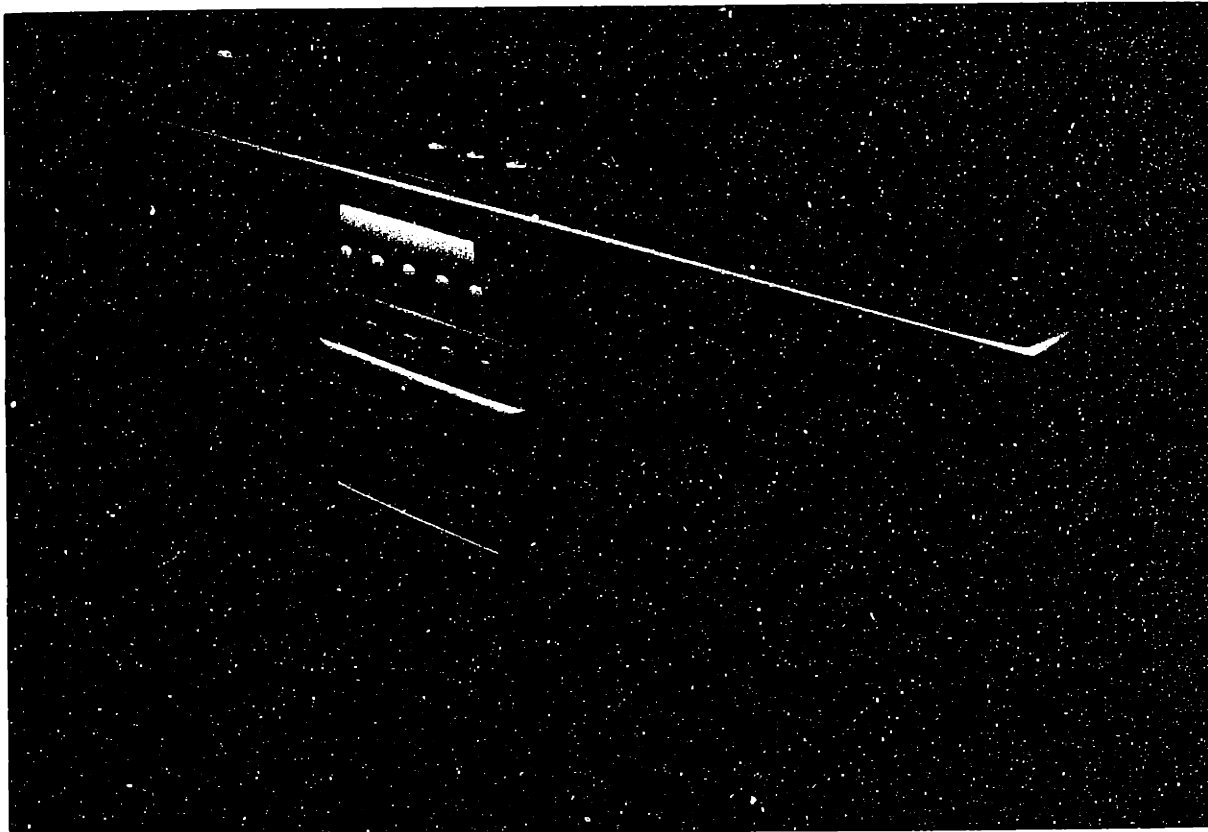


Figure 5.1a A Braun-like style portable stereo

This portable stereo design contains a cassette mechanism, a radio tuner, and a total of four speakers (2 large speakers - 2 small speakers). Also included are a power supply and jacks which are not apparent in the view presented. To obtain this design the program user first selects the standard components from the component library, and then defines the product attributes. This is a thin desk top product with a vertical parting line and a Braun-like style. After providing the system with these initial conditions, the program arranges the components, generates a surface, and adds surface details and colours without user intervention. The program presents the design shown above. The Braun-like style employs subtle variations of cool blue blacks with soft corners. The speakers are accented with a chrome red band underneath the speaker grill.

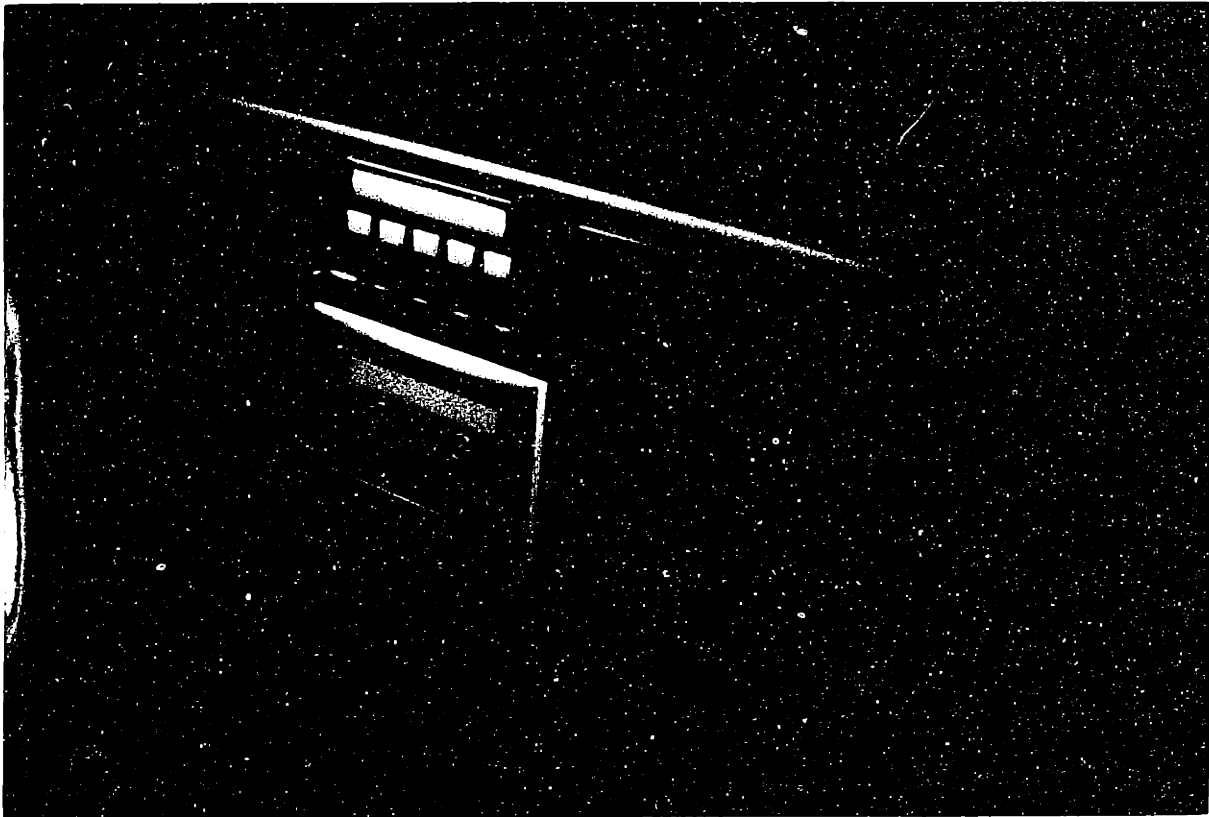


Figure 5.1b A High-Tech style portable stereo

The High-Tech stereo contains the same components with a layout identical to the Braun-like design in figure 5.1a. Only the surface style and the style of details are different.

The proof of concept assumes that the component layout is independent of the style. The components are oriented under the assumption that the user of the product will interact primarily with its front. The principal face is determined by the parting line orientation. The components are positioned so that the housing can be made in a straight draw two part mould with no side holes.

The High-Tech style is distinctive through its use of faceted and chamfered linear elements. Features are delineated from a predominantly black background with brightly coloured accents.

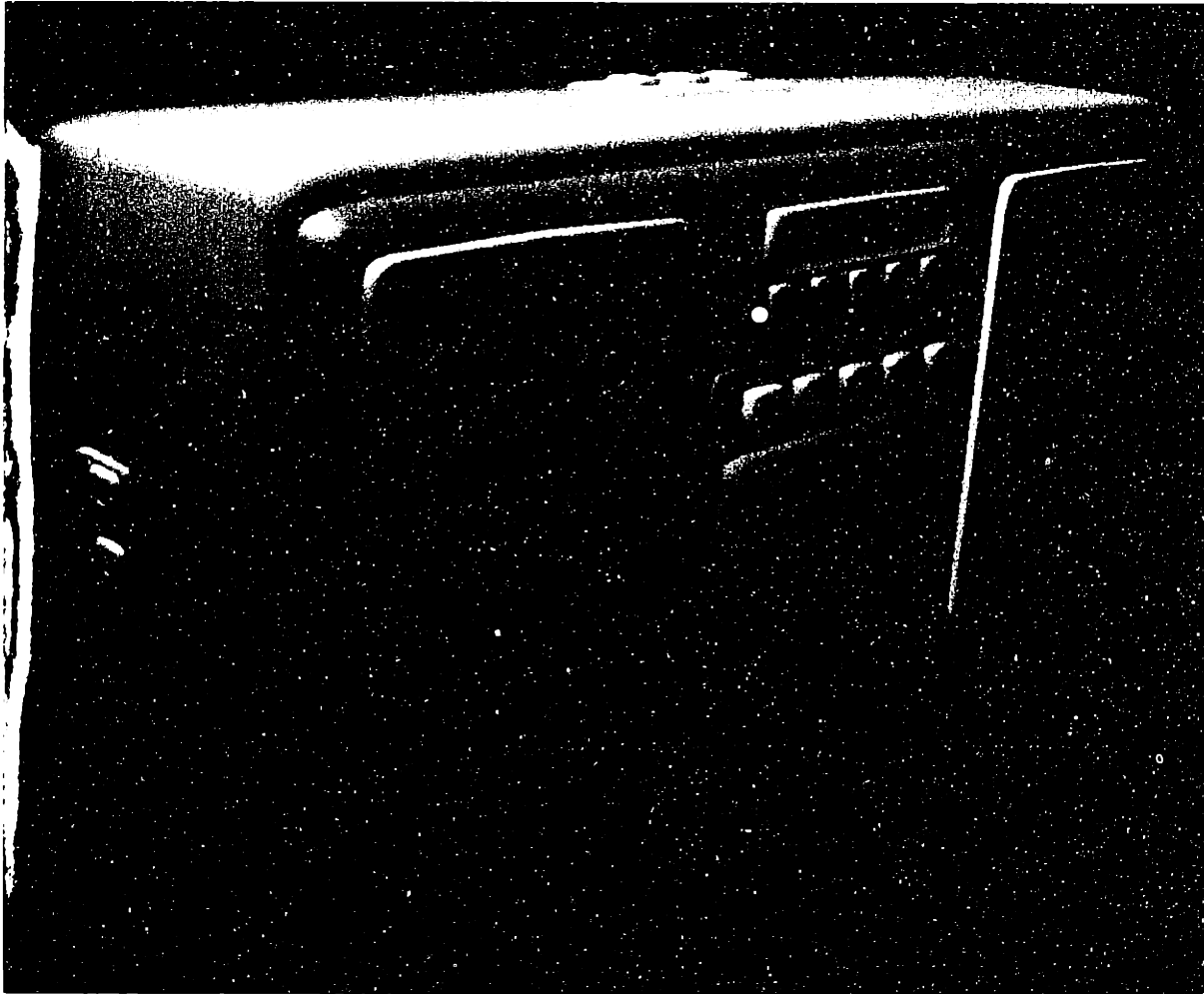


Figure 5.1c An Art Deco style portable stereo

The Art Deco style for a similar stereo design has a much softer corner treatment than the previous two examples. The form suggests a cascading water theme that is very common in Art Deco designs. The styled surfaces and details are encoded by the designer in the prototype libraries. The system modifies and applies the prototypes to individual designs.

Two headphone jacks are grouped in a panel insert and constrained to the housing parting line due to manufacturing considerations. They can be seen in the left of the figure. Jacks which are frequently used are placed on the side of the product, while the power supply, which is unplugged less frequently, is placed on the back of the design.

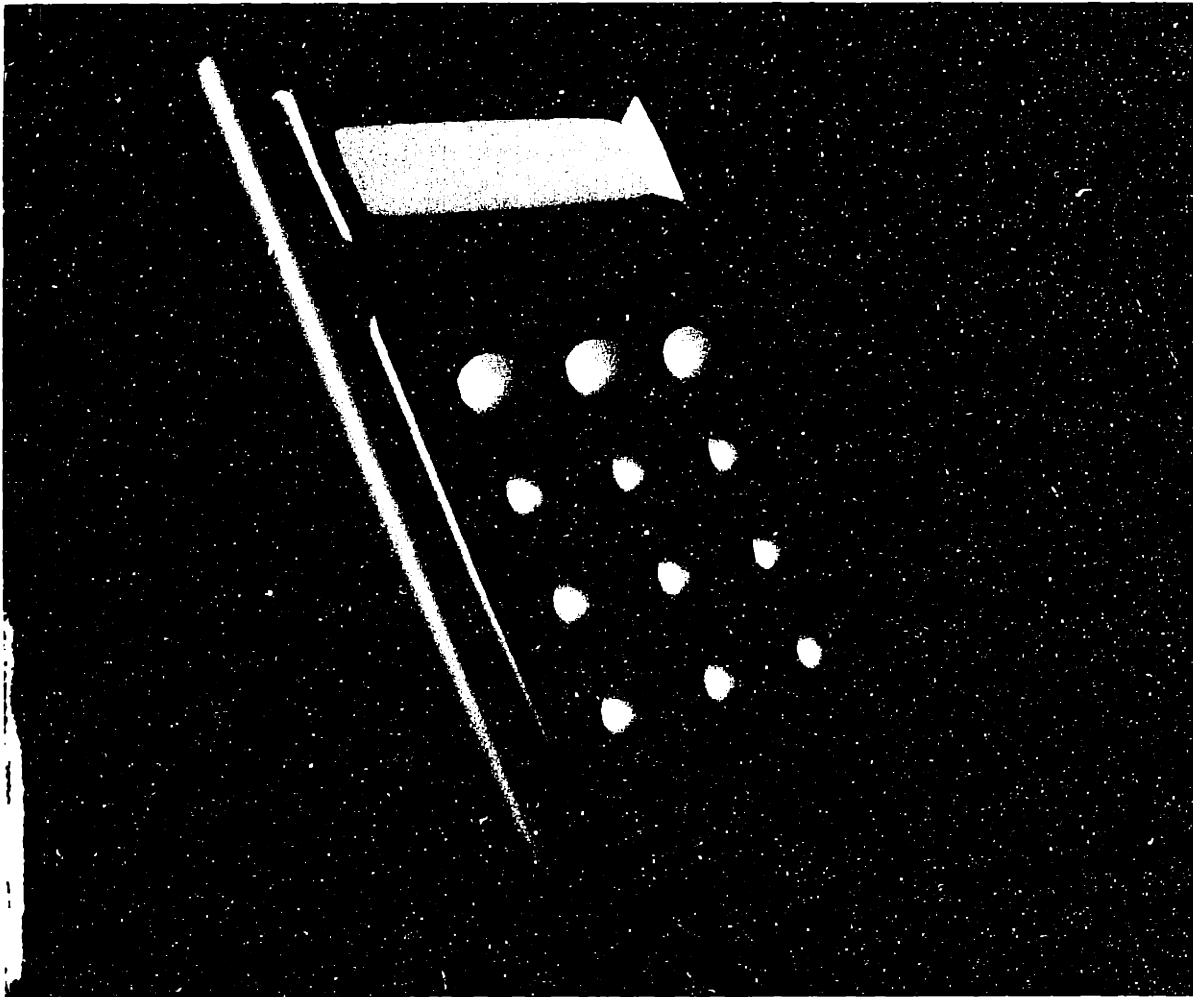


Figure 5.2a A Braun-like style calculator

The calculator design is a thin desk top product with a horizontal mould parting plane. It is clearly recognizable as a calculator even though the system uses the same rules as those employed in the stereo design. Further, the stylistic relatedness to the Braun-like stereo is readily apparent.

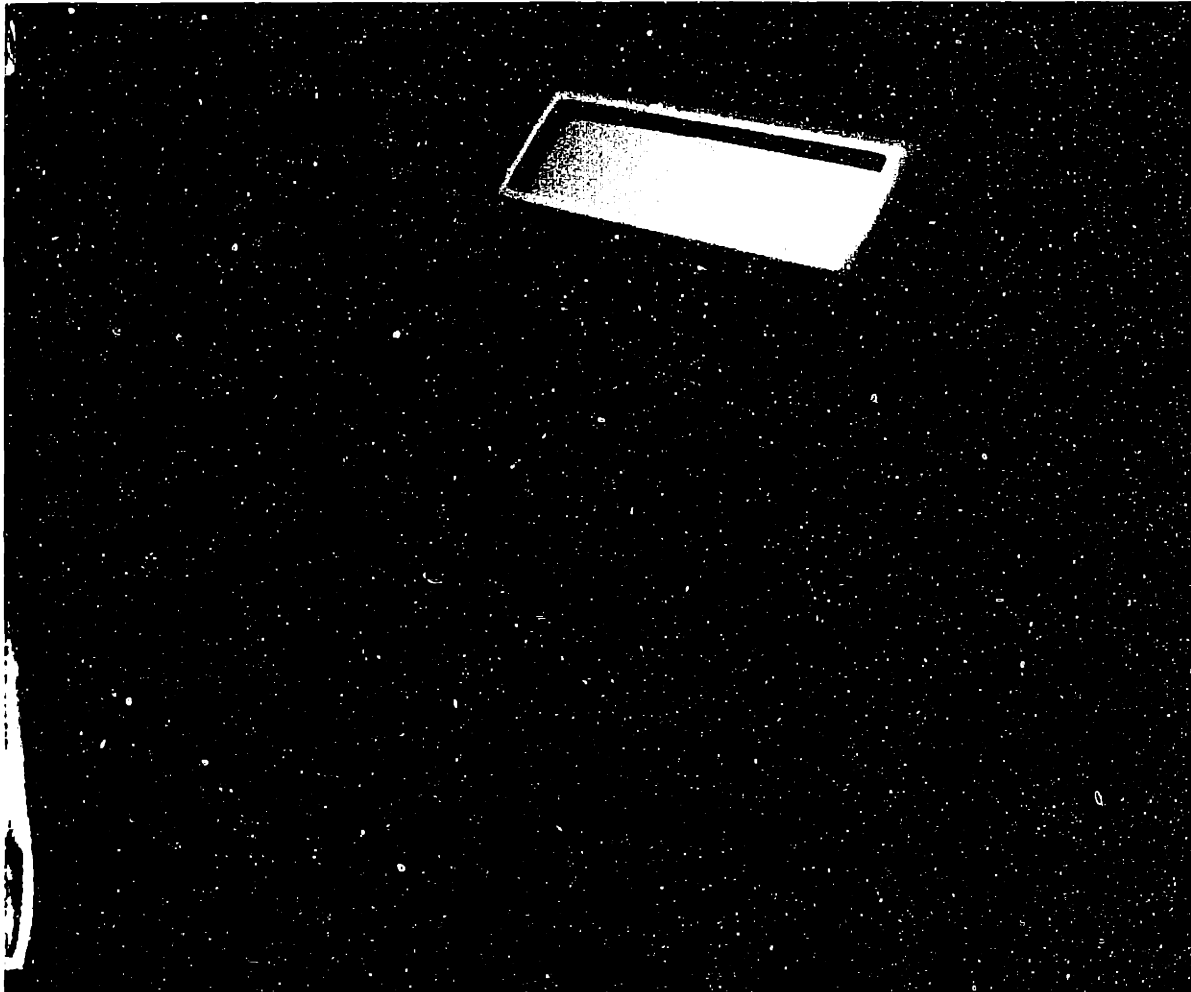


Figure 5.2b A High-Tech style calculator

The same set of components in the High-Tech style results in a very different looking calculator, even though the design contains only two elements: a numeric keypad membrane and a 12 character liquid crystal display.

The system can position components with which the product user interacts to provide conceptual layouts for a design. Locating components that the product users will not interact with is not addressed in this thesis.



Figure 5.2c An Art Deco style calculator

The Art Deco style is equally distinctive in the calculator design. Note that the upper row of buttons in the keypad are not highlighted with a different colour as they were in the previous two styles of calculators. The style specific colouring patterns are encoded in the detailing functions.

5.1.1 Comparison to Existing Style Archetypes

The results have demonstrated that the proposed model of industrial design permits the system to produce designs in both distinct and repeatable styles. The styling of products forwarded by the system is determined by the surface and detail prototype libraries, which the designers define. The style specific colour palettes used by the system are prepared by either an industrial designer or a graphic artist.



Figure 5.3 Braun Archetypes
(from Bayley, 1985, p. 62-63)

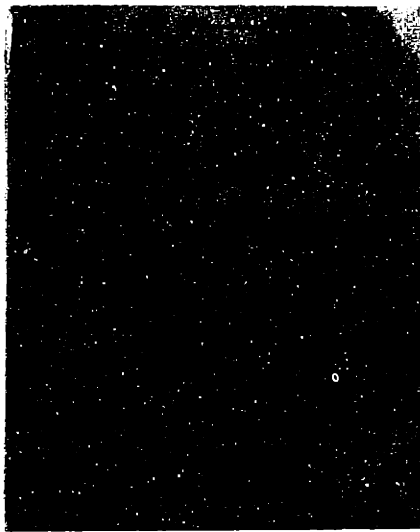


Figure 5.4 High-Tech
(from Bayley, 1985, p. 153)

When developing the style libraries for the proof of concept, I chose a small number of existing products or images to serve as archetypes for each style. The icons used for Braun-like, High-Tech, and Art Deco styles are shown in figures 5.3 through 5.5. I feel that the visual characteristics of the products designed by the system are consistent with these icons. While this consistency is obtained through the careful design and mapping of the prototypes, the results demonstrate that the proposed model not only permits the system to design products of different styles, but that it can reproduce specific pre-defined styles. The surface corner treatments of the product enclosures were designed to reflect those of the style icons. The shapes of the details were created to reinforce the traits of each style.

The Braun-like archetypes designed by Dieter Rams are refined and subdued. Subtle shades of cool dark colours with an occasional accent contribute to the appearance of refinement and precision.

The High-Tech style interior design example at the left features delineated surfaces with angular corners. The choice of colours enhances these characteristics. The bright primary colour accents used on the details stand out on the black base colour to emphasize structure and severity.

The pastel colours of the Art Deco designs match the softer forms. Art Deco designs also frequently use bright metallic accents which are difficult to reproduce on the graphics workstation. In the proof of concept Art Deco colours were limited to green, pink, and yellow pastels. The prototype surfaces were defined using designs such as the refrigerator by Raymond Loewy as inspiration.



Figure 5.5 The Art Deco archetypes
(from Capitman, 1988, p. 41 and Bayer, 1988, p. 165)

5.2 Range of Products in Proof of Concept

Another characteristic central to the thesis is the ability of the automated system to produce appropriate design configurations without knowledge of what the product is. Generality could be an important factor in justifying the initial overhead of implementing a system as proposed.

Within the proof of concept domain the system does not use pre-encoded templates or designs for specific products. Distinction between calculators, stereos, or any other kind of device is not made. The configuration of a specific product is driven by the components which it contains and general design principles.

In section 5.1 we saw stereos and calculators designed by the proof of concept implementation. Figure 5.6 provides an indication of the system's capacity to produce a variety of recognizable products using combinations of the limited number of components implemented presently.

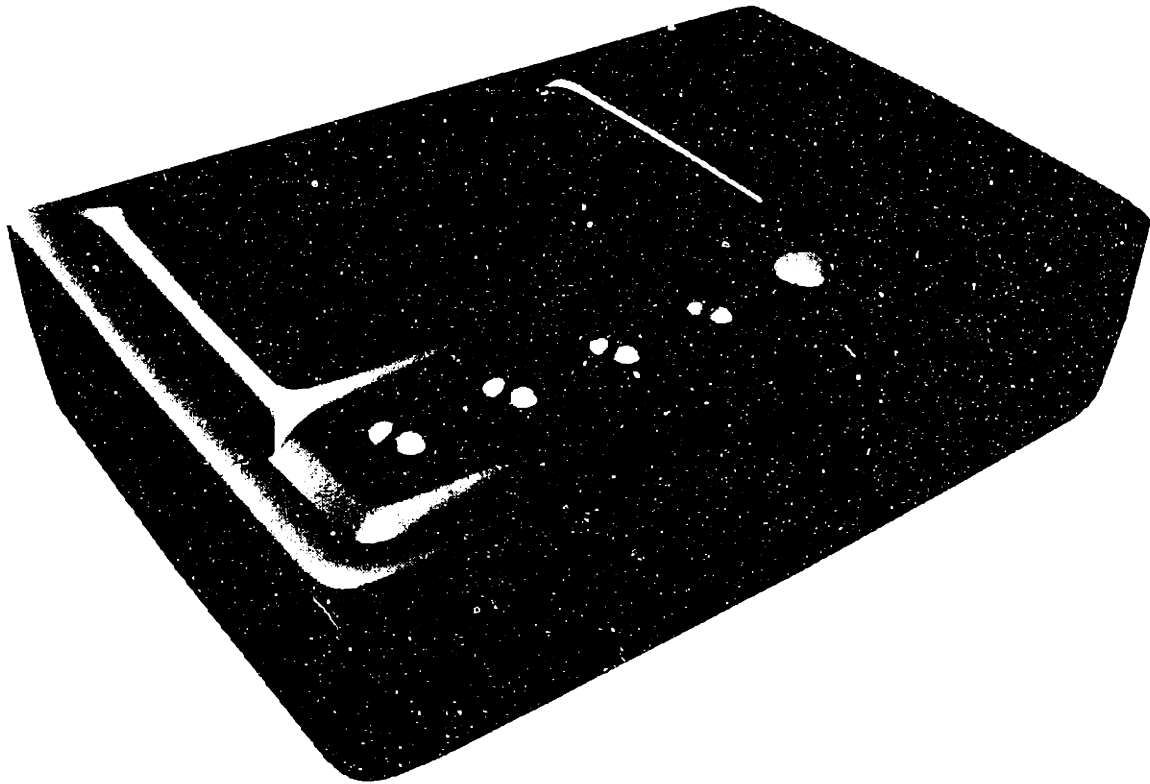


Figure 5.6a Braun-like mono cassette player design

The cassette player above is a thin, desk top product with a horizontal parting line and a Braun-like style. The design contains: a cassette mechanism; a single small speaker; a miniature microphone; two speaker jacks; and a power supply. The speaker and microphone are grouped together under the grill. The jacks and power supply are positioned in the rear face of the product. The design could be made into an answering machine by adding telephone jacks to the arrangement.

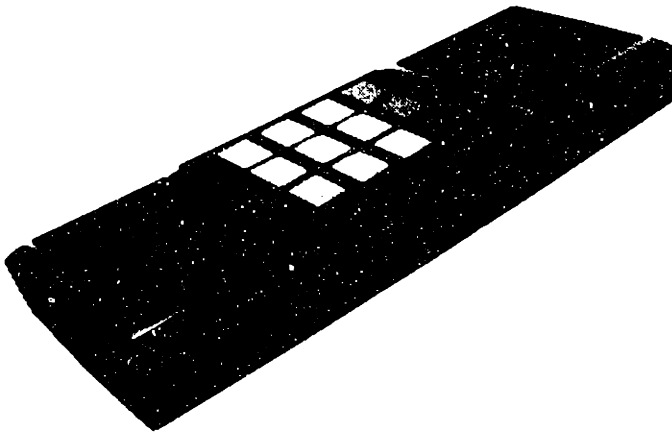


Figure 5.6b Handset design

The High-Tech design on the left contains a numeric keypad, a small speaker, and a small microphone. The telephone jack, which is not visible in the photo, is located at the top end of the housing.

The product has a horizontal parting line and is configured for hand held use. The panel insert added over the keypad is desired for ease of manufacture.



Figure 5.6c Pocket computer design

The pocket computer, finished in the High-Tech style, is very similar to the calculators shown previously. The panel insert unifies the separate numeric and alpha-numeric keypads. This desk top device is thin and uses a 30 character liquid crystal display.

The Art Deco stereo television is made from a cathode ray tube, four speakers (two large speakers and two small speakers), a set of toggle switches, and a power supply. The design — intended for desk top use — has a vertical parting line and looks like a stereo.

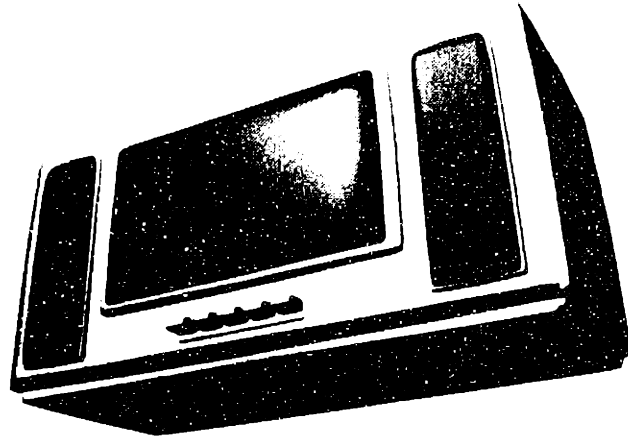


Figure 5.6d Stereo television design

By removing the two large speakers and setting the height attribute to tall, a quite different looking Braun-like television in figure 5.6e is designed by the system. Substitution of a single miniature speaker for the two small speakers yields the mono television below. The different products can be designed without product specific rules because they all share a common language which makes the general approach possible. In chapter 1, it was speculated that this commonality arises through product evolution — ideas from well known existing products are transferred to

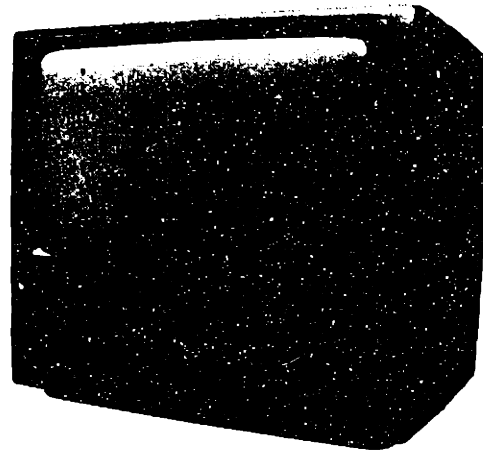


Figure 5.6e Tall television design

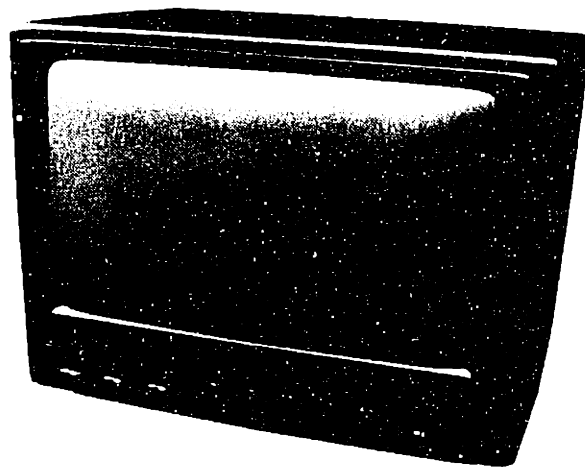


Figure 5.6f Mono television design

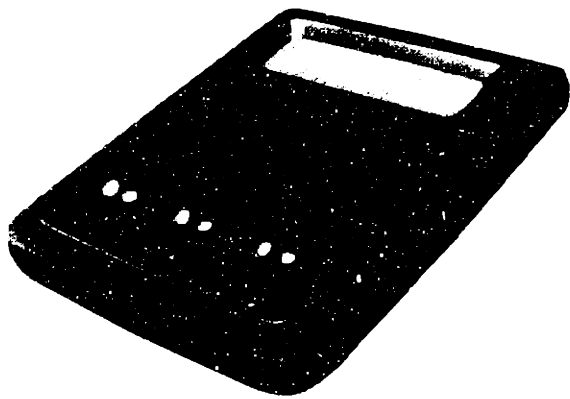
new yet unknown designs. Similar manufacturing and ergonomic considerations are also unifying influences in the designs. It must, however, be acknowledged that system generality is obtained at an expense. The program implemented applies a very conservative (and perhaps unsophisticated) approach to designing products. (In fact, the approach is so conservative that it is assumed that style does not affect the layout of components.) If a higher level of design performance is desired, it would likely be necessary to target the system for a less diverse product range. For example, one could probably create a system with a very subtle aesthetic vocabulary for workstations or automotive dashboards exclusively. This degree of specialization may be acceptable when the product is produced in a wide variety of lines.

The effects of manufacturing and ergonomic considerations are apparent in the example designs. All components are positioned so that no undercuts or side holes are required. Components on faces through which the parting plane passes are constrained to the housing parting line to avoid side holes. When a large number of closely spaced holes are required (such as over a keypad) a panel insert is incorporated to avoid problems with mould flow or cosmetic blemishes from freeze marks. Components are located so that they are accessible to the product user and not likely to be obscured while the product is in use.

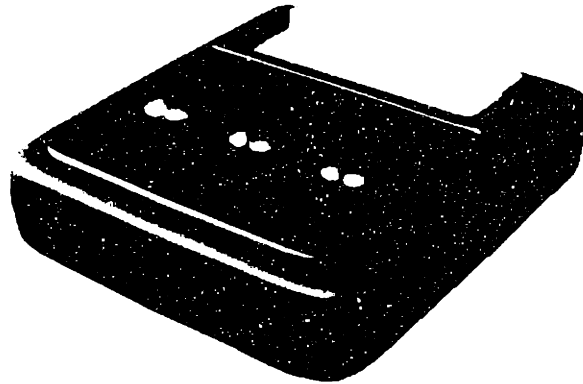
5.3 Production of Multiple Product Variations

A valuable feature of the proposed system, as stated in chapter 1, is its ability to suggest multiple designs for a product. In an instant, the designer is provided with a series of alternative product configurations which can serve as conceptual seeds for the design process. As was explained in section 4.4.1.2.2, multiple versions arise through the execution of *if then or* rules during the organization level of the model. Illustrative examples of design alternatives for two different groups of components are considered in this section.

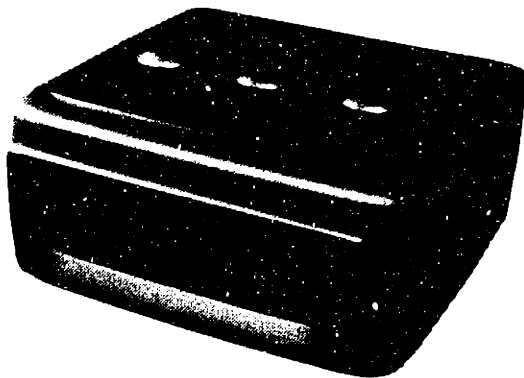
The four designs proposed by the system in figure 5.7 exemplify that it does not have a pre-encoded understanding of what the product is. While the first two versions are calculator-like, the last two are perceived as travelling alarm clocks. The meaning of the layouts is entirely due to human interpretation. All four alternatives contain a 12 character liquid crystal display, a miniature speaker, and a three button keypad. The housing parting line is horizontal.



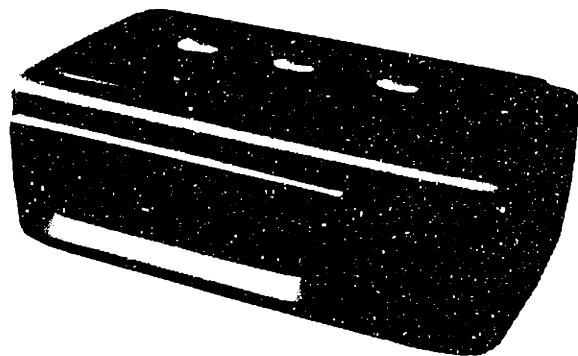
version 1



version 2



version 3



version 4

Figure 5.7 Product versions with different semantic interpretations

A range of portable stereo designs — as an example of variations on a theme — are shown in Figures 5.8. The designs use the same components as the stereos in section 5.1.

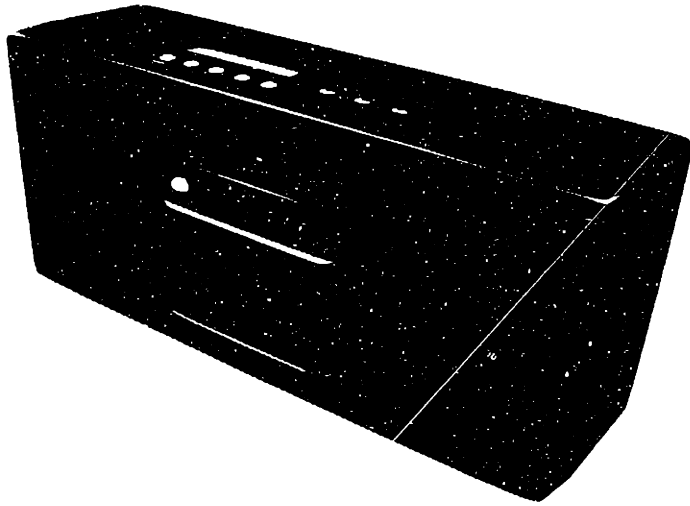


Figure 5.8a Alternative stereo version 1

All of the design variations are shown in the Braun-like style for ease of comparison.

Version 1 is a minor change from the previously shown layouts as the radio tuner has been moved to the top face of the product. All components on the top surface are constrained to lie on the housing's vertical parting plane.

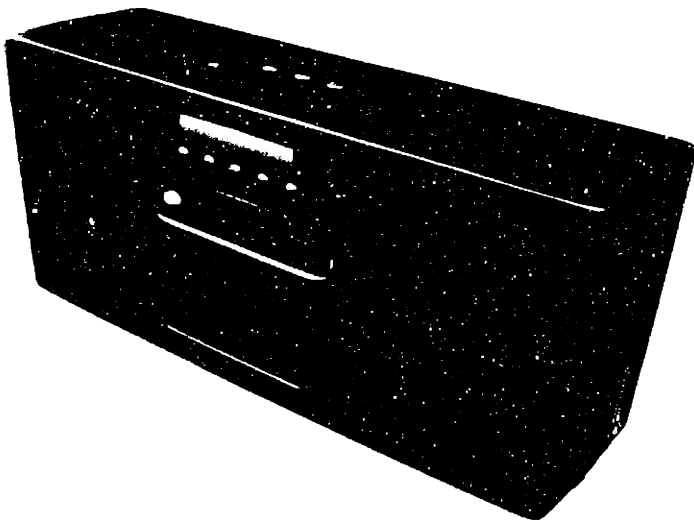
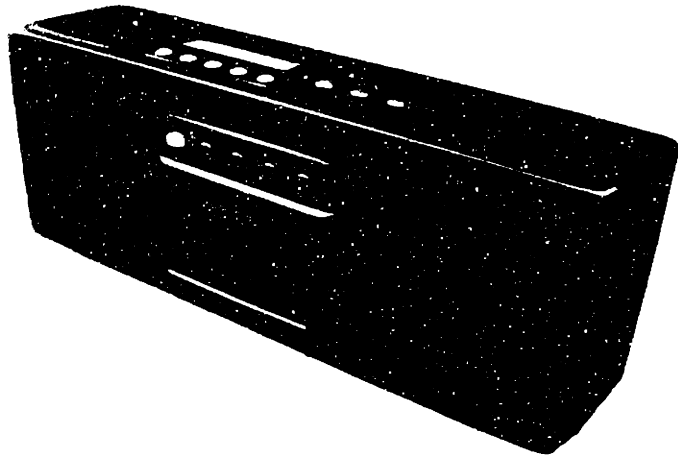


Figure 5.8b Alternative stereo version 2

In version 2, the two small speakers are placed in the top surface rather than the radio tuner. Note that the grills on the top and front face are not aligned. The detail sizing algorithm which locates the grills does not consider interaction between faces of the product. While this assumption was made to simplify the program, an organization that is too rigid would become boring.



The third version at the left combines elements of the first two variations and results in a very compact and unified appearance.

Versions four and five demonstrate more unusual design alternatives. Because the system is naive to what a stereo looks like, it can create unanticipated designs. Such surprises could lead the designer to new ideas.

Figure 5.8c Alternative stereo version 3

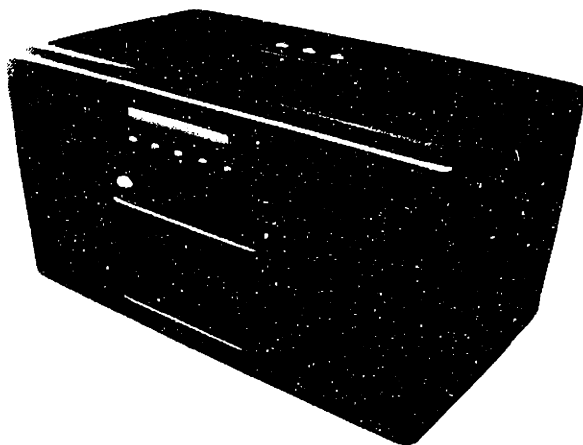


Figure 5.8d Alternative stereo version 4

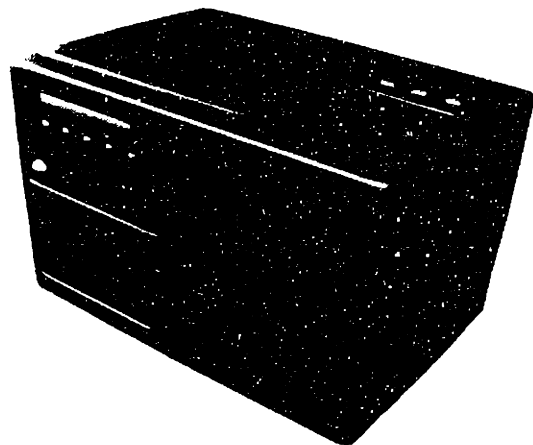


Figure 5.8e Alternative stereo version 5

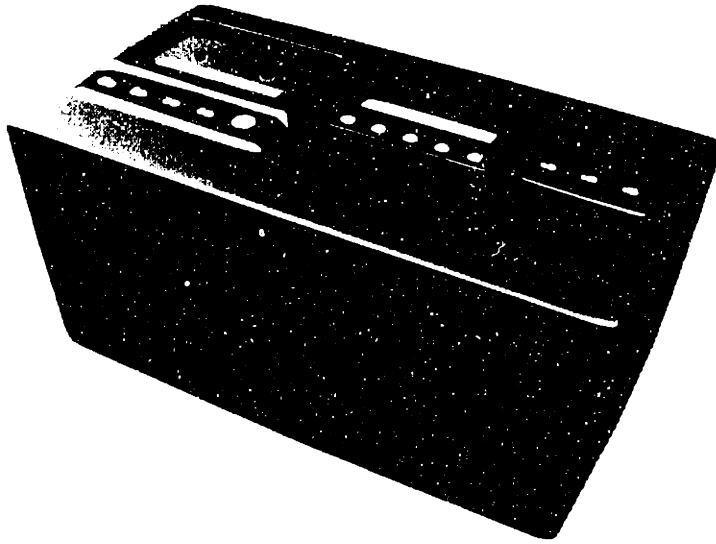


Figure 5.8f Alternative stereo version 6

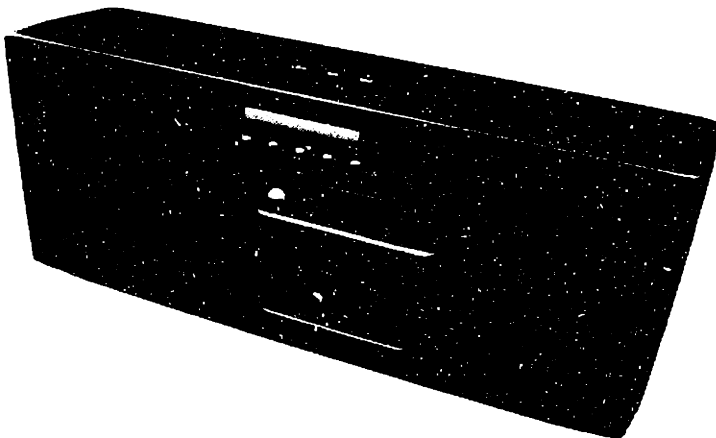


Figure 5.9a Poor alternative stereo version 1

Version 6 has a very spartan appearance. The components in the top face are forced to lie on the parting line. This configuration would be better suited to a horizontal parting plane.

The designs in figure 5.8 represent a small sample of versions proposed by the system. Figure 5.9 shows some of the proposed designs which I feel are failures. These failures occur because the organization level of the system performs *open loop* without post component positioning evaluation. As illustrated in figure 5.9a, the system is creating some product matrices (based only upon manufacturability and ergonomics) for which the limited aesthetic positioning procedures cannot produce good layouts. A disorganized unbalanced design results.

A second example of a design forwarded by the system which exhibits a lack of coherence between product faces is in figure 5.9b.

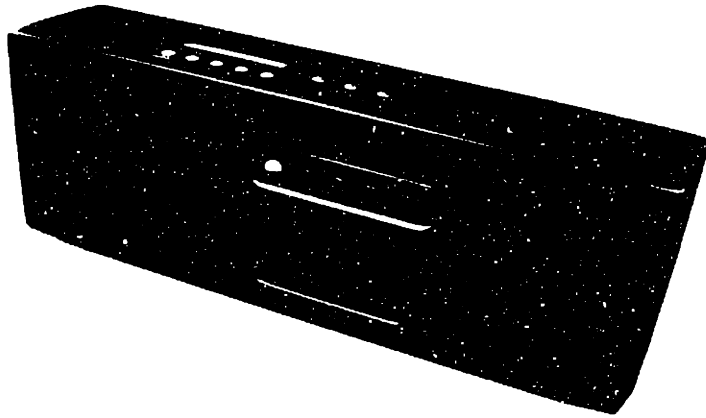


Figure 5.9b Poor alternative stereo version 2

Such problems can be reduced by introducing an evaluation process into the organization level (see figure 5.10). Also, a wider variety of aesthetic positioning strategies is needed. The evaluation mechanism could be used to prevent the explosion of possible layouts which occurs for products containing large numbers of components.

Here, creation of product variations is limited to occur only at the organization level.

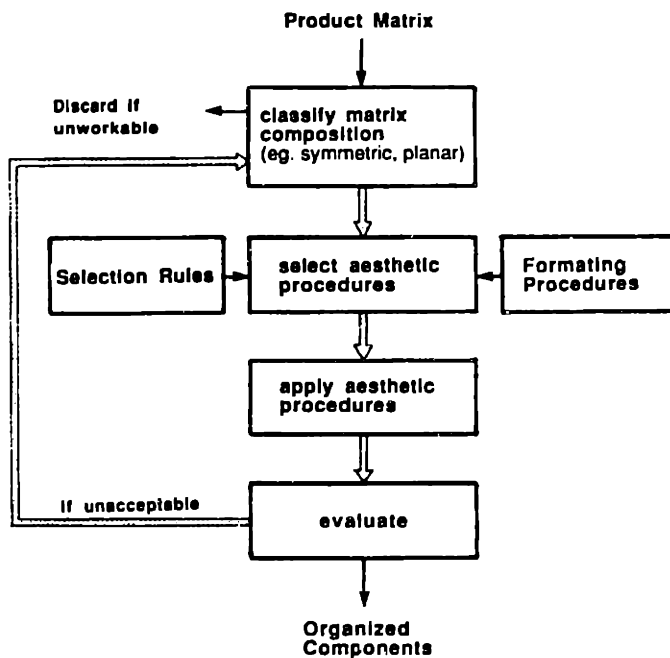


Figure 5.10 Modified organization level with an evaluation mechanism

In fact, design alternatives could be presented at almost every level of the process through the provision of multiple prototypes and colour palettes for each style. In practice, there will be a happy medium which provides an adequate quantity of alternatives without swamping the user with so many options that the best designs are obscured.

5.4 Organization Level

5.4.1 General Discussion

An example of the input to and the output from the organization level is provided in figure 5.11 and figure 5.12.

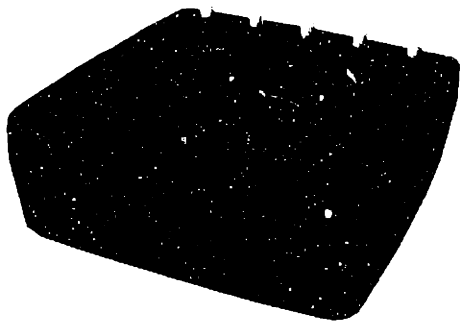


Figure 5.11 Components prior to organization

The user selects components as desired from the library using a menu system. The unpositioned components are centred on the origin and interfere with each other, as shown in the figure. This example contains: a cassette; 2 large speakers, two small speakers; a radio tuner; a three button keypad; a headphone jack; two speaker jacks; and a power supply. After defining the product attributes, the system spatially locates the components.

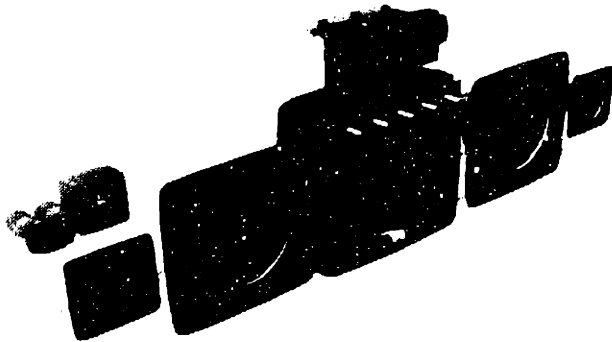


Figure 5.12 Output from the organization level

The process of positioning and orienting the components relative to the mould parting plane typically takes one or two seconds. The figure shows the positioned components and the product bounding hull. The example is a thin product with a vertical parting line. The row-like organization is used to create the impression of stability.

In addition to accounting for manufacturing and ergonomics (refer to chapters 3 and 4), the organization level imparts the structure which will (hopefully) make the design meaningful. The examples provided in the previous sections illustrate that the proposed two stage architecture of the level — utilizing the product matrix and aesthetic procedures — can perform this task competently. Using the product matrix as a space partitioning mechanism in combination with a rule based approach appears to be helpful in both rule construction and knowledge extraction through examples. Encoding the aesthetic principles as procedures produced results that tend to exhibit desirable visual properties such as stability, rhythm, balance, and organization.

A notable weakness of the organization level is that the burden of choosing appropriately matched components is placed on the user. The acceptability of the design is highly dependent upon the selection of well matched components. This is a manifestation of the saying "garbage in equals garbage out". If the user selects a group of either poorly matched or inappropriate components for the product use, the system will propose a poor design (for example, a CRT is inappropriate for a hand held product). While the responsibility of selecting matched components should not be a problem for experienced designers, it might be for the aesthetically naive. An improved system could resolve this problem by choosing the exact dimensions of the components, allowing the user to specify the component types only.

Other limitations — such as no component interaction or the rudimentary nature of the aesthetic positioning procedures — were discussed in chapter 4. The remainder of the organization level discussion will illustrate specifically how product requirements influence design layouts.

5.4.2 Effects of Product Attributes on Configuration

The output in figure 5.13 demonstrates the dramatic effect that the parting plane has on the layout of the product. The components and product attributes are identical to those of the design in figure 5.1, except the parting plane is horizontal instead of vertical. The difference in the layout is a manifestation of the assumption about how the product is used in relation to the parting plane (see chapter 4, section 4.4.1.2.1). For manufacturing and aesthetic reasons the system assumes that the parting plane is not located in the the primary visual surface of the product.

The system can propose alternate designs regardless of the parting plane orientation. One design variation for the thin stereo with a horizontal parting plane is shown in figure 5.13b.

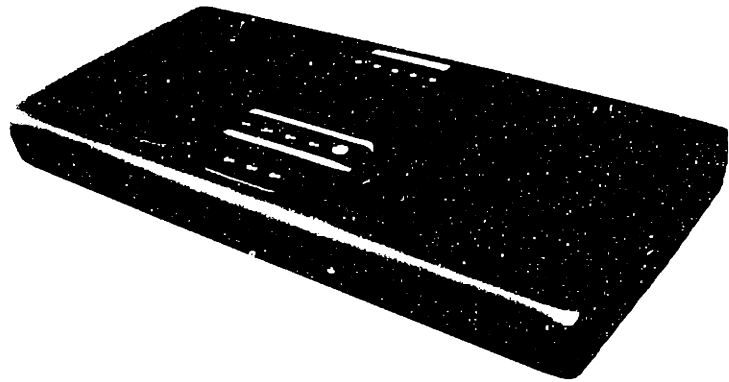


Figure 5.13a Stereo with a horizontal parting plane

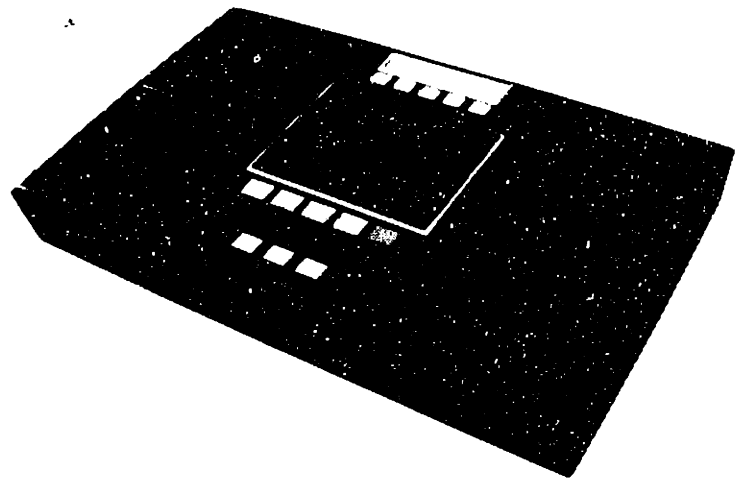


Figure 5.13b Alternative horizontal design

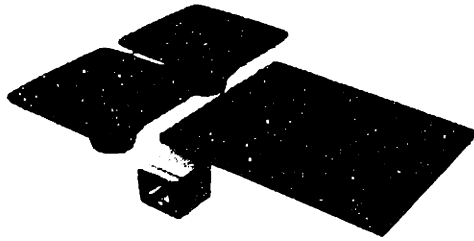


Figure 5.14a Desk top design

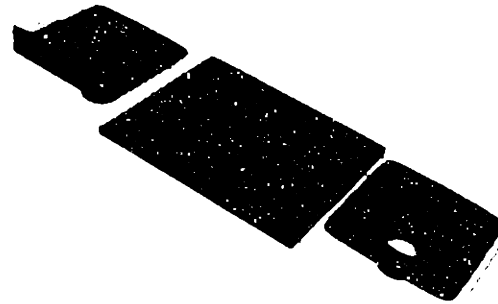


Figure 5.14b Hand held design

Figure 5.14 depicts the difference in layout between hand held and desk top thin products with a horizontal parting line. The hand held layout is dominated by the requirement that it must fit in the user's hand.

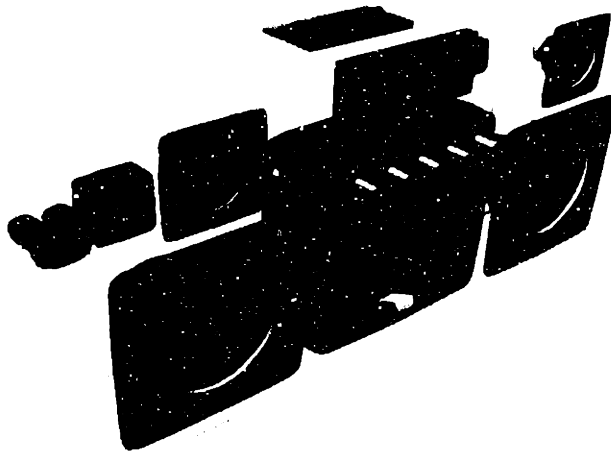


Figure 5.15 The effect of product height on the configuration

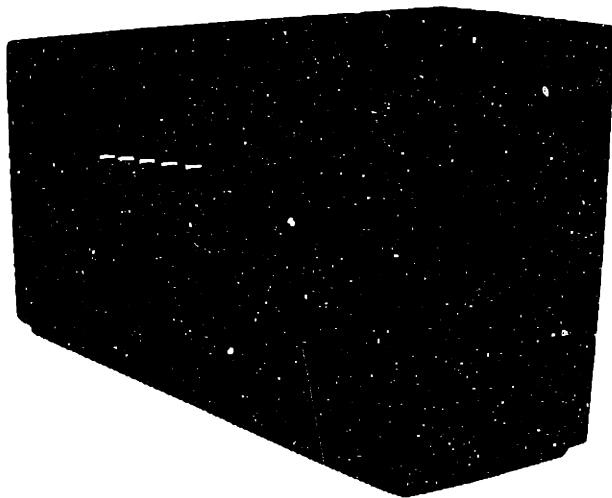
When the product height attribute is tall, the component positioning is chosen to make the design physically taller. The system uses a column arrangement in figure 5.15 to create the impression of a tall stereo.

Only the height attribute has been changed to arrive at the tall configuration from the stereo in figure 5.12.

The rules are constructed so that when a product is thin, the layout will be nearly two dimensional. This means that, when possible, the components are positioned so that the product has only one predominant face with which the user interacts. The face which dominates will depend upon the parting plane orientation. When writing the matrix filling rules, I decided that the so called thin attribute would be used to control whether multiple product versions are proposed. If the product is 'thin' only one configuration will result, while if it is not, the 'thin' version along with additional variations will be generated. The preferred matrix locations give the thin designs (see section 4.4.1.2.1). When the product is not thin the alternate matrix locations are filled to produce multiple layouts. The rules were written in this manner to reflect my opinion that thin configurations are preferable for electronic products.

5.5 Surface Level

The output from the surface level for the components in figure 5.12 is shown in figure 5.16.



The surface level simply scales an appropriate enclosure prototype to fit the product bounding hull and house the components within a monolithic form.

Attempts to capture surface styles using the two step process modelled after morphological analysis — style prototype definition coupled with a style preserving mapping function — performs acceptably in the proof of concept.

Figure 5.16 Example output from the surface level

The trivial method of designing case specific prototypes adopted in the proof of concept is sufficient only if a homogeneous class of product designs is acceptable (i.e all cameras, all car dashboards, or all box-like products) and the number of specific cases is limited. In the proof of concept, four prototype enclosures are required for each style. These prototypes are: hand held with a vertical parting plane; hand held with a horizontal parting plane; desk top with a vertical parting plane; and desk top with a horizontal parting plane. This very simple approach is appealing because it permits the designers to create the prototype enclosures using the method of their choice. Manufacturability characteristics are designed into the prototypes directly, so the implementation of this level is straight forward — provided the mapping functions are designed to preserve the prototype draft angles. A larger variety of product surfaces could be afforded by designing more prototypes for each style. This would allow the system to propose multiple styling alternatives for a single component configuration.

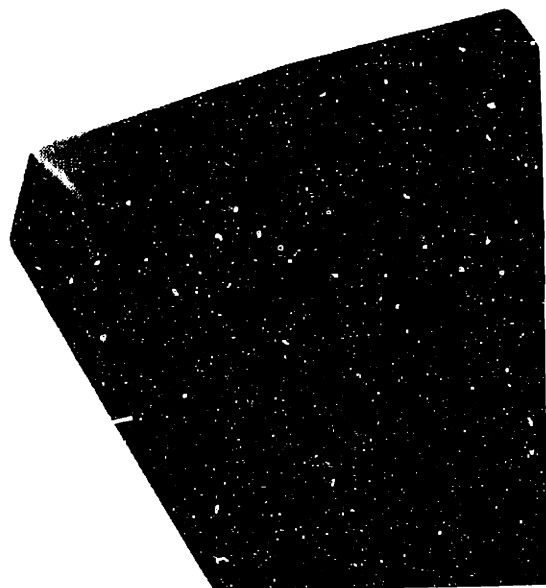


Figure 5.17 Distorted technical corner due to uniform scaling

In Figure 5.17, the corner chamfer is lost on one edge through linear prototype scaling. This provides evidence that simple scaling of enclosure prototypes is, in general, not acceptable if the product style is to be adhered to over a large range of dimensions. The single scaling function used in the proof of concept worked well for the technical style, and acceptably for the Braun-like style. For softer forms, it was not clear whether the non-linear scaling preserved the enclosure style over various sizes and aspect ratios better than linear scaling. This suggests that it will be necessary to design scaling functions on a style specific basis.

Mapping the enclosure to the product hull was greatly simplified by the hypothesis that style can be encoded principally in the corner treatments. This assumption appears to be valid, especially since details and colouring are such an important ingredient in different styles. The effects of product attributes on the surface enclosure is discussed in chapter 4 in the section on the anatomy of the prototype designs (see chapter 4).

5.6 Detail and Graphical Level

The surfaces added by the detail and graphical levels are shown in figure 5.18 relative to the positioned components for the portable stereo of figure 5.12. The detail level customizes the product enclosure for the components in the design.

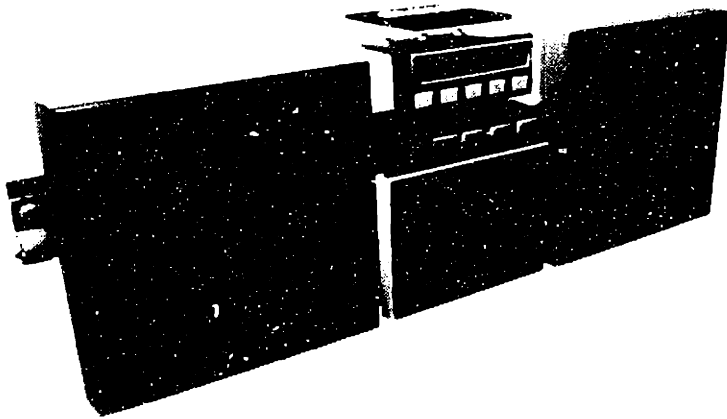


Figure 5.18 Output from the detail level

The results presented throughout this chapter emphasize that the details and product colouring are very important to the style and appearance of the product. The colouring simplification which pre-encoded the detail colours did not seem to seriously compromise the composition of products designed by the program. Even so, I feel that a rule based colouring system which assigns colour to details based on the overall composition will be necessary to prevent the graphics from becoming too predictable.

Figure 5.6c is an example where two keypads are grouped under a panel insert. While panel inserts are used with keypads for manufacturability, the visual grouping created by the insert is equally important. The panel helps to integrate the two components into the design and unify the product. Similar grouping through the application of details is apparent in the stereos and other designs shown previously. Product organization is enhanced by justifying details with the outer perimeter of the housing when appropriate, and through their application to emphasize component sub-groupings. Only components of similar class are placed in a single insert to form a sub-group. This helps to delineate components with different functionality.



Figure 5.19 Panel inserts used to simplify mouldings

In Figure 5.19 a group of jacks constrained to the housing parting plane are combined in a panel insert. In this case the grouping allows the housing to be moulded with a single side slot rather than a series of slots.

Even with detail grouping some of the designs look a bit 'add-on' due to a lack of integration between details and different faces of the product. In some cases it would be desired to establish relationships between details. For example, the cassette door and the speaker grills of the stereos could be more integrated.

The ability to group like components on different faces of the product could further integrate the appearance of products designed by the system. The wrap around speaker grill sketched in figure 3.26 suggests what this type of detailing would look like.

The simplifying assumption that all component details must protrude from the surface contributes to the 'add-on' appearance of the designs. Applying some details which are recessed or integrated with the surface would help to eliminate this problem. Implementation of such details would not require major changes in the programming logic, but would demand solid modelling capabilities.

6 CONCLUSIONS AND FUTURE WORK

6.1 Conclusions

In the thesis a framework for the computer automation of industrial design is developed. The four level hierarchy proposed for aesthetic form giving is central to the model. Results for the injection moulded consumer electronics design domain indicate that the model can produce meaningful product configurations through the consideration of ergonomic, manufacturability, and aesthetic rules of thumb without the use of product specific prototypes. The system can create designs in recognizable styles or corporate identities (Art Deco, High Tech, and Braun-like). The styles are sufficiently distinctive that they can either unify a collection of dissimilar products or differentiate a single design layout. The aesthetic system model will allow the proposal of multiple design alternatives at any one of the four stages of the design process: the organization level, the surface level, the detail level, or the graphics level. The proof of concept implementation offers design alternatives at the component organization level only.

The goal of the system is to create aesthetic preliminary product designs which are conceptually correct for their intended market, use, and manufacture. Such issues need be considered in the initial design phase if inherently suitable concepts are to be produced. Intrinsically flawed concepts which require significant re-design or design effort in later detail design phases can be avoided at the onset, thereby reducing the likelihood of cost overruns or marginal products. The proof of concept was able to create simple hand held or desk top designs well suited to production in straight draw two part moulds with no side holes or undercuts.

The model assumes that the design process begins with the selection of standard components or subassemblies which are to be included in the design. This implies that the model is intended for a type of design in which the general characteristics of the product domain are well established. The premise that classes of products (such as consumer electronics) share common properties which allow the system to be designed for product categories rather than individual types of products is exploited. It is suggested that this commonality is related to how new designs are often created by borrowing ideas and visual language from other known and related artifacts. The system, in its present form, uses conventions observed in existing products to design new ones. As suggested by the poem

"Epitaph of the unfortunate artist", the system automates aesthetic design precedence or formulae, thus freeing the industrial designer to partake in truly creative design. The system relies upon ongoing input from the designer to increase its knowledge and style repertoire through the expansion of its prototype libraries. The implementation is not a learning program.

The ability to create a design system which can use a single method and set of design rules for a category of products is important to the viability of the system as a practical design tool. I feel that, while this is possible, the level of design sophistication will decrease as the generality of the program is increased.

The proof of concept implemented for consumer electronic products, although simplified, has been able to both demonstrate the feasibility of an automated system as proposed and validate the appropriateness of the four level model for form giving. In the remainder of this section conclusions specific to each level of the model are drawn.

6.1.1 Organization Level

The first stage of the design process, the organization level, locates the standard components included in the product relative to a mould parting plane subject to product attributes, manufacturing, ergonomics, and aesthetics. The configuration is assumed to be style independent. This assumption is an oversimplification made to limit the size of the implementation. However, it is a quite reasonable one if a degree of standardization in product lines is desired. Evidence suggests that a specific style can be achieved through the product surface and its details. Another simplifying assumption at this level of the design is that the components do not share functional interaction.

The positioning of the components is divided into two separate processes: a rule - of - thumb based stage which considers manufacturing and ergonomics; and a procedure based phase which addresses aesthetics. Integral to the rule based stage is the product matrix. The product matrix is a structure which permits the system to conceptually block out a design without having to worry about detailed positioning. A 3 x 3 x 3 matrix that allows the system to make distinctions such as 'the component belongs in the front-right-middle of the product' was adequate resolution for the spatial partitioning process. The matrix also proved to be a useful visualization tool which facilitated the process of writing the rules and anticipating their consequences. A simple taxonomy of components based

upon ergonomic interactions was developed to permit the construction of small 10-to-15 rule knowledge bases for each taxonomic grouping. This approach should simplify the process of adding new kinds of components to the system.

The aesthetic procedures receive the product matrix, which can be imagined as the blocked out product, and physically positions the components to impart a sense of rhythm, balance, organization, and stability to the layout — while ensuring that interference does not occur. To achieve this goal the system must try to anticipate the outcome of various positioning strategies and then decide which method will yield the best results. The set of procedures written to do this in the proof of concept, although quite primitive, were adequate for designs containing small numbers of components. The system should have several strategies available so that it can determine the best approach for the given product matrix. With the overall strategy decided, the system can systematically work through regions of the product matrix until the components are fully positioned. The positioning procedures can be likened to applying word processor formats to a text file.

An inherent weakness of the approach is that, if given a poorly matched set of components, the system will produce designs which do not exhibit acceptable aesthetics. The success of the organization level is tied to the selection of suitable components. Other instances in which visually poor designs arose can be attributed to the limited positioning strategies implemented in the proof of concept.

6.1.2 Surface level

The surface level receives the positioned components from the organization level and creates a monolithic enclosure in a chosen style, subject to manufacturing and product use considerations. It was found that analogies — in the form of enclosure prototypes — are an appropriate method to achieve these goals. The advantage of using prototypes is that knowledge related to both manufacturing and style is encoded implicitly. The use of style prototypes will allow the designer to create the enclosure libraries using the methods of their choice.

The biological tool known as Morphological Analysis reveals that creating the surface for a product from a prototype is a two part process. The procedure involves first the selection of a prototype enclosure, and then the application of a transformation function which maps the prototype to the product while preserving the style. It was found that the

surface style could, to a high degree, be captured in the corner treatment of the prototypes. Thus, the scaling functions designed to enforce the style focus on maintaining correct corner features. A different scaling function will be required for different styles. In general, as the prototype is scaled up the corner features should become proportionally smaller. When the prototype is scaled down they should become larger.

The extremely literal interpretation of the surface prototypes used in the proof of concept trades simplicity of implementation for the expense of a larger prototype library. As the design of prototypes is relatively straight forward this may not be too large a sacrifice, but generative approaches to combine prototypic features are worthy of investigation.

6.1.3 Detail and Graphics Levels

The detail and graphical levels of the model augment the product enclosure and fine position the components to complete the product design concept. The two stages of the process are very important in establishing the style and consumer appeal of the product. The addition of details can be used to enhance the organization and component groupings which are imparted to the configuration during the organization level of the model.

The detail level of the system utilizes a library of style specific elements to complete the components contained within the product. Example details would include grills for acoustic components, buttons for keypads, or doors for the cassette mechanism. The system applies aesthetic criteria to determine the best size for a particular detail and which components should be grouped together under a single detail element. Given detail size and location information, component class specific functions apply the details to the surface of the product until all components have been satisfied. The method of applying the detail prototypes in the library is consistent with the approach used at the surface level of the model. The utility of the library concept is that the designer is required to create a certain type of detail only once. The system takes care of the adaptive application of details to fit specific products, and thus the designer is relieved from the chore of repeatedly modifying semi-standardized elements to suit different products.

The approach of adding independent details to the housing based upon the components under the surface can lead to a product which lacks overall aesthetic integration

and appears 'add-on'. The grouping of components under common details such as panel inserts is important in reducing this problem.

The graphical level of the model incorporates colours, applies surface textures, and adds graphic logos or other surface embellishments to the detailed product designs. The graphical level encompasses knowledge which bridges the industrial design domain and graphic arts. In the proof of concept this very important level of the model is limited to surface colouring and combined with the detail level. In addition to affecting appearance, decisions made in the graphical level — such as the use of texture and colour — can have important ergonomic and manufacturing consequences.

6.2 Future Work

The material presented in this thesis is a first exploration in the automation of aesthetic product design, and identifies central issues and possible solutions to the problem. In an effort to propose a complete model for the form giving process and demonstrate its validity, the work merely touches upon the exploration of individual details — hopefully deep enough to encourage future developments. Almost every aspect of the form giving process merits more involved study. This section is divided into two parts. First, possibilities within the scope of aesthetic design are discussed. Subsequently, expansion of the automated system within the overall design process is considered.

6.2.1 Development in the Context of Aesthetic Design

The system should be developed into an interactive design tool which can quickly provide design alternatives to be expanded upon, adjusted, or discarded by the designer. I feel that this will be the most powerful application of an automated design system. Further, user interaction is essential in the graphical level of the model and thus, in the proof of concept, this aspect of the model is investigated at a trivial depth only.

A weakness in the system as implemented is its inability to produce good results if poorly matched components are chosen for the product. It would be useful to provide the system with the capability of choosing the best matched parts, given the kinds of components to be included in the design. The problem of producing poor layouts in addition to good ones, as encountered in the proof of concept, could be resolved by developing internal evaluation mechanisms during the configuration process. I suspect that

creating an evaluator which is not highly arbitrary will be difficult. Another improvement would be to increase the repertoire and sophistication of the aesthetic positioning strategies available to the system. It is desirable to find more fundamental algorithms or principles for stability, balance, organization, and rhythm.

Another improvement would be to transcend the limitation of the box like domain, and include highly free-form surface prototypes. This requires that the system be able to consider spatial restrictions associated with the surface prototype during the component positioning process. A more general surface level of the system might be generative — meaning that it can assemble prototypes to create housings with an arbitrary number of sides or features. For example, the system could combine a foot prototype with an enclosure to generate a desk top housing.

It would be interesting to investigate possibilities of automatically extracting the production rules to fill the product matrix. The idea behind this approach would be to "show" the system a variety of products which the program uses to assess typical component placement in the product matrix. Another approach might be to allow the designer to interactively assign matrix locations to components.

Other augmentations to the system would include expanding the detail application capabilities so that elements which are recessed or integral to the surface can be added. The system must be able to cut holes in housings to achieve this goal. This would help to unify the appearance of products designed by the system. Additionally, it would be beneficial consider the inclusion of details which extend over more than one face of the product.

The possibilities associated with colouring and adding graphics to the product design merit more exploration. A very simple extension would be the inclusion of user group sensitive colour palettes. If this is done, the system could become an effective platform to survey user group preferences. One could also consider colour hybridization or interchanging between different styles.

6.2.2 Expansion within the Overall Design Process

Two issues need to be addressed in the expansion of the system's capabilities. First, the system should consider so called functional issues in the conceptual positioning

process. Secondly, the system needs to interface with and perhaps assist subsequent design processes. Ideas about these issues are discussed below briefly.

A limiting assumption made in the proof of concept is that of no component interaction. One of the first extensions of the proposed model would be to include other design factors in the configuration process. Concern about packing efficiency or mating with other products and parts is not unrealistic. Work to try to incorporate an approach such as proposed by Kim and Gossard (1989) might be a starting point. Other issues such as wire routing or field interference between components should be addressable within the production system framework. In the proof of concept, the components were positioned solely upon their interaction with the user. The system does not represent purely internal components. Inclusion of such parts will require dealing with component interaction and geometric constraints during the positioning process.

Tied to the issue of coping with external constraints is the idea of standardization. For some types of design it would be desirable to provide the system with a library of standard but dimensionally fixed enclosures. In this case, the job of the system would be to design a product subject to the constraints of a standardized housing.

Another logical extension would be to develop a system which can generate the inner surface for the product enclosure. The aesthetic system provides the outer shell and the component positions for a product design. The next step would be to determine the necessary internal mountings to complete the housing design for injection moulding. In addition to considering purely functional considerations, this system should also address cosmetic issues such as the creation of sink marks in the outer surface of the moulding.

REFERENCES

- Barr, A., Feigenbaum, E. A., eds., 1981, *The Handbook of Artificial Intelligence*, vol. 1, Heuristech Press, California, ch. 3.
- Bayer, Patricia, 1988, *Art Deco Source Book*, Phaidon Press Ltd., Oxford, 1988.
- Bayley, Stephen, ed., 1985, *The Conran Directory of Design*, Villard Books, New York
- Beall, G., 1983, *Plastic Part Design for Economical Injection Molding*, Borg-Warner Chemicals Corporation, Parkersburg, West Virginia 26101.
- Benton, Tim and Charlotte, 1975, *Form and Function*, Granada Publishing Ltd., ch. 3.
- Boden, Margaret A., 1991, *The Creative Mind*, Basic Books, U.S.A., pp. 135 - 181.
- Bonner, David, L., 1991, *Personal communication*, Massachusetts Institute of Technology Computer Aided Design Laboratory.
- Bonner, J. T., ed., 1961, *On Growth and Form*, abridged edition, Cambridge University Press.
- Boothroyd, G., Dewhurst P., 1983, *Design for Assembly, A designer's Handbook*, Department of Mechanical Engineering, University of Massachusetts, Amherst Massachusetts.
- Brennan, Susan, Carey, Susan, 1987, 'Identification and Ratings of Caricatures: Implications for Mental Representations of Faces', *Cognitive Psychology*, vol. 19, pp. 473-497.
- Brownston, L., Farrell, R., Kant, E., Martin, N., 1985, *Programming Expert Systems in OPS5, An Introduction to Rule-Based Programming*, Addison-Wesley Publishing Company, ch. 1.
- Capitman, Barbara Baer, 1988, *Deco Delights*, E. P. Dutton, New York, p. 41.
- Carlson C., Woodbury R. F., 1990, "Structure Grammars and their application to design", *Proceedings of the First International Workshop on Formal Methods in Engineering Design, Manufacturing, and Assembly*, Colorado State University, January 15-17, pp 99-131.
- Celniker, George, 1990, *ShapeWright: Finite Element Base Free-Form Shape Design*, Doctor of Philosophy Dissertation, Department of Mechanical Engineering, Massachusetts Institute of Technology.
- Chomsky, Noam, 1956, "Three models for the Description of Language", *IRE Transactions on Information Theory*, 2:3 pp. 113-124.

- Duffy, M. R., Dixon, J. R., 1988, "Automating the Design of Extrusions: A Case Study in Geometric and Topological Reasoning in Mechanical Engineering Design", *Proceedings, 1988 ASME Computers in Engineering Conference*, v.1, August 1988.
- Faux, I. D., Pratt, M. J., 1987, *Computational Geometry for Design and Manufacture*, John Wiley and Sons, pp. 31-36, 138-144.
- Flemming, U. 1987, "More than the sum of parts: the grammar of Queen Anne houses", *Environment and Planning B: Planning and Design*, volume 14, pp. 323-350.
- Flemming, U., 1989, "More on the representation and generation of loosely packed arrangements of rectangles", *Environment and Planning B: planning and Design*, volume 16, pp. 327-359.
- Flurscheim, Charles H., 1983, *Industrial Design in Engineering: a marriage of techniques*, Springer-Verlag, pp. 105 - 156.
- Gero, J. S., Mather, M. L., Zhang, W., 1988, "Chunking structural design knowledge as prototypes", *Artificial Intelligence in Engineering Design*, Elsevier, co-published with Computational Mechanics Publications, pp. 3-21.
- Gilles, Willem, 1991, *Form Organization new design procedure for numerical control*, Butterworth-Heinemann Ltd., Oxford.
- Grosse, I. R., 1989, "A Design for Manufacturability Approach for Computer-Aided Configuration Design", *NSF Engineering Design Research Conference*, College of Engineering, University of Massachusetts, Amherst Massachusetts, June 11-14.
- Hanada, H., Liefer, L.J., 1989, "Intelligent Design System for Injection Moulded Parts Based on the Process Function Analysis Method", *NSF Engineering Design Research Conference*, College of Engineering, University of Massachusetts, Amherst, June 11-14, pp. 597-612.
- Hayes-Roth, F., Waterman, D., Lenat D., eds., 1983, *Building Expert Systems*, Addison-Wesley Publishing Company.
- Hearn, Donald, Baker, M., 1986, *Computer Graphics*, Prentice-Hall p. 223.
- Hofstadter, Douglas R., 1985, *Metamagical Themas: Questing for the essence of mind and pattern*, Basic Books Inc., U.S.A., pp. 280, 232- 260.
- Hofstadter, Douglas, Mitchell, Melanie, French, Robert M., 1987, *Cognitive Science and Machine Intelligence Laboratory, Fluid Concepts and Creative Analogies: A Theory and its Computer Implementation*, Technical Report no. 10, University of Michigan, pp. 17-19.
- Jakiela, Mark, 1988, *Intelligent Suggestive CAD Systems*, Doctor of Philosophy Dissertation, University of Michigan, Department of Mechanical Engineering.
- Kim, Jay J., Gossard, David G., 1989, "Toward Automation of Assembly Packaging", *NSF Engineering Design Research Conference*, College of Engineering, University of Massachusetts, Amherst Massachusetts, June 11-14, pp. 267-284.

- Knight, T., 1980, "The Generation of Hepplewhite-style Chair-back Designs", *Environment and Planning B*, volume 7, pp. 227-238.
- Knuth,D., 1982, "The Concept of Meta-Font", *Visible Language*, vol. 16, No. 1., pp. 3-27.
- Koning, H., Eizengberg, J., 1981," The language of the prairie: Frank Lloyd Wright's prairie houses", *Environment and Planning B*, volume 8, pp. 295-323.
- Kubouy, Micheal, Pomerance, James R.,editors 1981, *Perceptual Organization*, Lawrence Erlbaum Associates.
- Lauer, David A., 1979, *Design Basics*, Capital City Press.
- Le Corbusier, 1954, *The Modular*, Harvard University Press, Cambridge.
- Liming, R. A., 1944, *Practical Analytical Geometry with applications to Aircraft*, Macmillan.
- Longenecker, Steven, Fitzhorn, Patrick,1990,"A Shape Grammar for Modeling Solids", *Proceedings of the First International Workshop on Formal Methods in Engineering Design, Manufacturing, and Assembly*, Colorado State University, January 15-17, pp. 70-97.
- Mahnke, Frank H., Mahnke, Rudolf H.,1987, *Color and Light in Man-Made Environments*, Van Nostrand Reinhold Company, New York, U.S.A., pp. 10-21.
- Mantyla, Martti, 1988, *An introduction to solid modelling*, Computer Science Press, Inc., Maryland U.S.A.
- McCormic, Ernest J., Sanders, Mark S., 1982, *Human Factors in Engineering Design*, Fifth Edition, McGraw-Hill.
- McDermott J., 1980a, "R1: An Expert in the Computer Systems Domain", *AAAI* , pp. 269-271.
- McDermott J., 1980b, *R1: A Rule-based Configurer of Computer Systems*, Technical Report CMU-CS-80-119, Department of Computer Science, Carnegie-Mellon University.
- McDermott J., 1981, 'R1: The Formative Years', *AI Magazine* , no. 2, pp. 21-29.
- Moneo, R., 1978, "On typology", *Opposition*, 13.
- Oxman, R., Gero, J. S.,1988, "Designing by prototype refinement in architecture", *Artificial Intelligence in Engineering Design*, Elsevier, co-published with Computational Mechanics Publications, pp. 395-411.
- Papalambros, Panos, Chirehdast, Mehran, 1990, " An Integrated Environment for Structural Configuration Design", *Journal of Engineering Design*, vol. 1, no. 1, pp. 73 - 95.

Postings, J. H., 1985, *Plastics Mouldings*, Materials Engineering Design Guides no. 24, published for the Design Council, the British Standards Institution, and the Council of Engineering Institutions, Oxford University Press.

Rock, Irving, Palmer, Steven, 1990, "The legacy of Gestalt Psychology", *Scientific American*, December.

Rogers, David F., Adams, J. A., 1990, *Mathematical Elements for Computer Graphics*, McGraw-Hill Publishing Company, pp. 250-277.

Schildt, Herbert, 1987, *Artificial Intelligence Using C*, Osborne McGraw-Hill.

Sederberg, Thomas W., Parry, Scott R., 1986, "Free-Form Deformation of Solid Geometric Models", *SIGGRAPH*, vol. 20, no. 4, pp. 151- 160.

Sharpe, D.T., 1974, *Psychology of Color and Design*, Nelson Hall Company, Chicago, U.S.A.

Steadman, J. P., 1983, *Architectural Morphology, An introduction to the geometry of building plans*, Pion Limited, London, pp. 9-20.

Stiny, G., Gips, J., 1978, *Algorithmic Aesthetics*, University of California Press Ltd.

Stiny, G., 1980, "Introduction to shape and shape grammars", *Environment and Planning B*, volume 7, pp. 343-351.

Stiny, G., Mitchell, W. J., 1980, "The grammar of paradise: on the generation of Mughul gardens", *Environment and Planning B*, volume 7, pp. 209-226.

Tjalve, Eskild, 1979, *A short course in Industrial Design*, Newne - Butterworths, Boston, Chapter 4.

Ulrich, K., Seering, W., 1988, "Function Sharing in Mechanical Design", *Proceedings of the Seventh National Conference on Artificial Intelligence*, American Association for Artificial Intelligence, Vol. 1, pp. 342-346.

Wolovich, William A., 1986, *Robotics: Basic Analysis and Design*, Holt, Rinehart and Winston, p. 110.

Woodson, W., 1981, *Human Factors Design Handbook*, McGraw-Hill.

APPENDIX A Elements in the Component Library

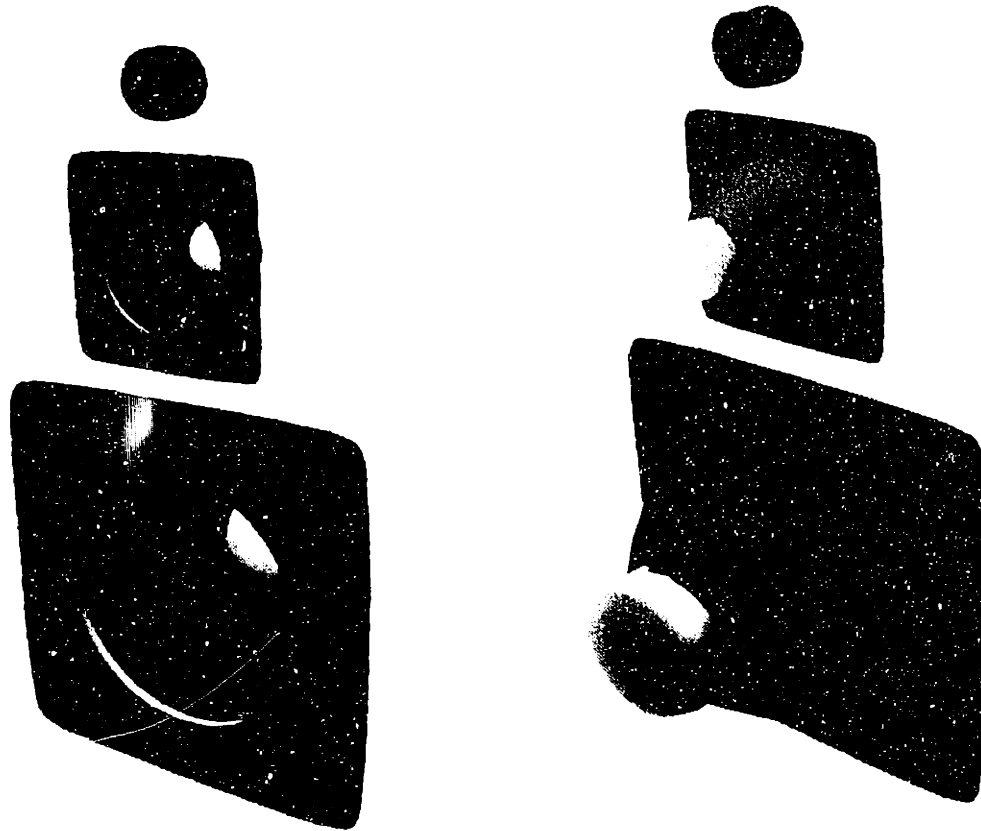


Figure A1 Acoustic components: large speaker; small speaker; and miniature speaker or microphone



Figure A2 Keypad membranes: alphanumeric; numeric; 3 button; and 1 button

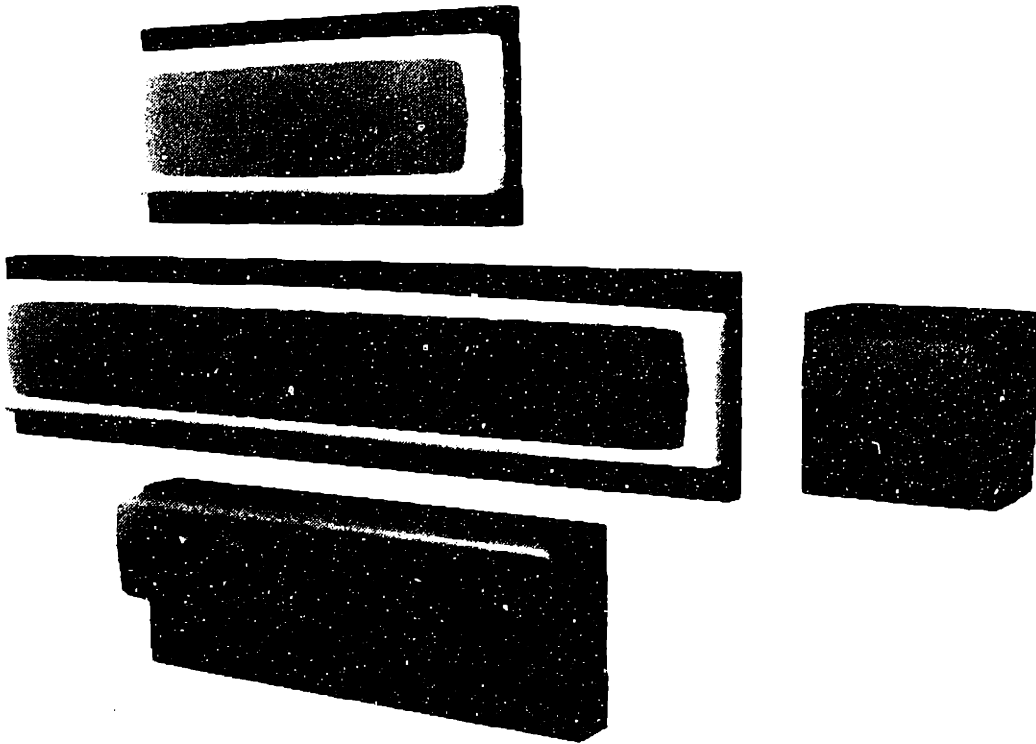


Figure A3 Visual displays: radio tuner; 2 digit LED; 12 and 30 character LCD

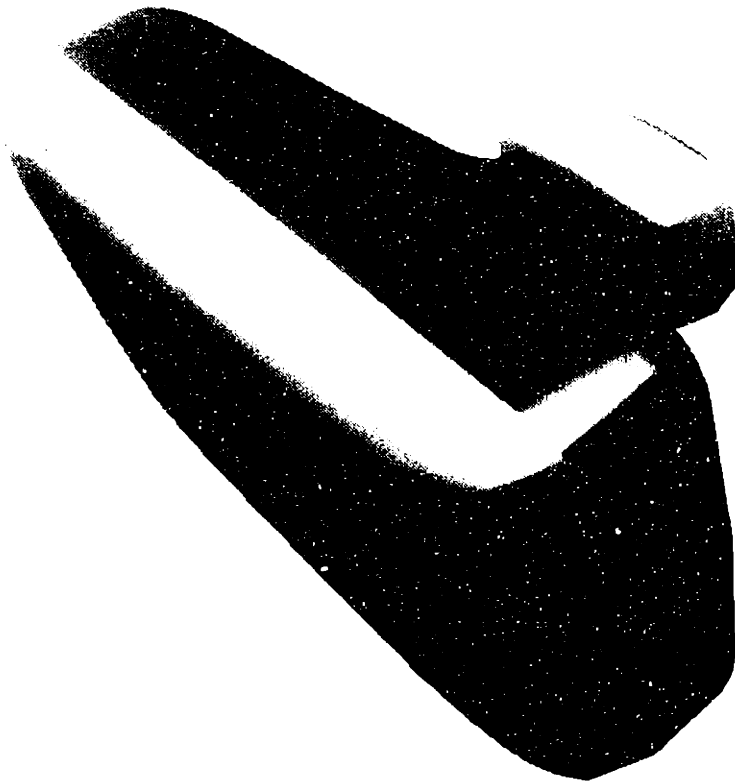


Figure A4 Visual display: 6 x 6 cathode ray tube

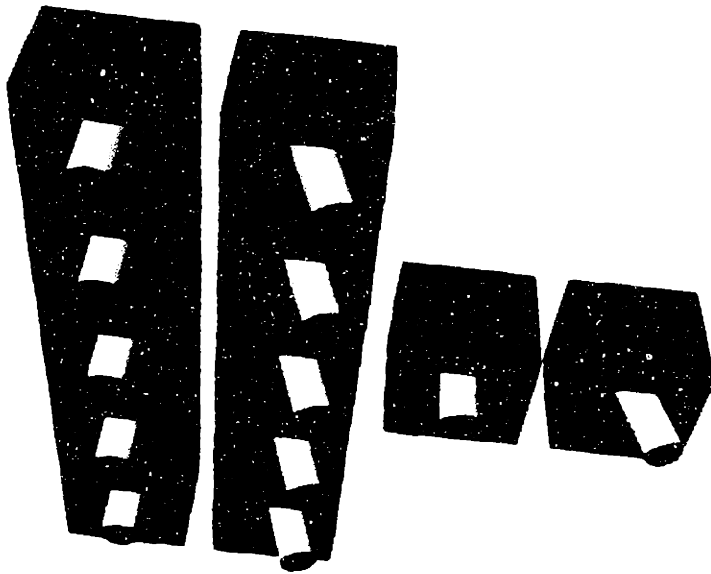


Figure A5 Switches: 5 button press; 5 button toggle; 1 and 1 button press

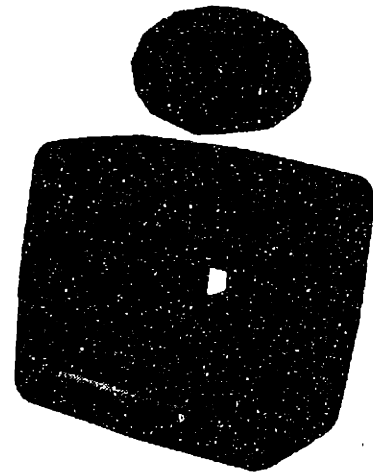


Figure A6 Jacks: power supply and microphone

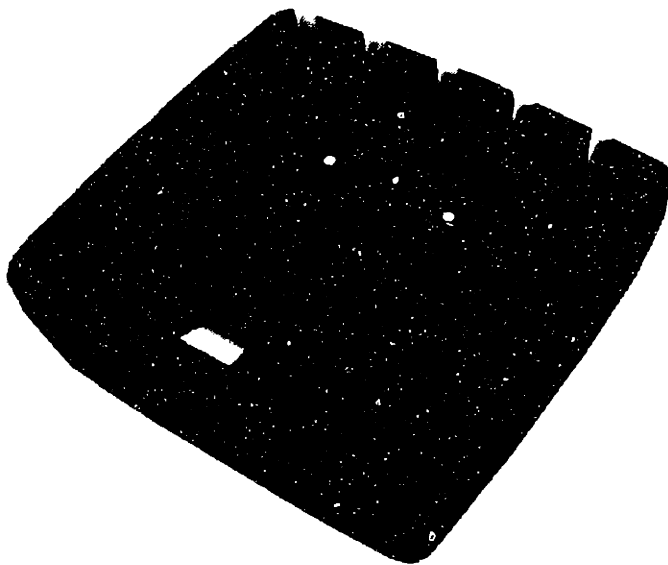


Figure A7 Mechanism: cassette



Figure A8 Jacks: telephone and headphone plug

APPENDIX B Prototypes in the Surface Style Library

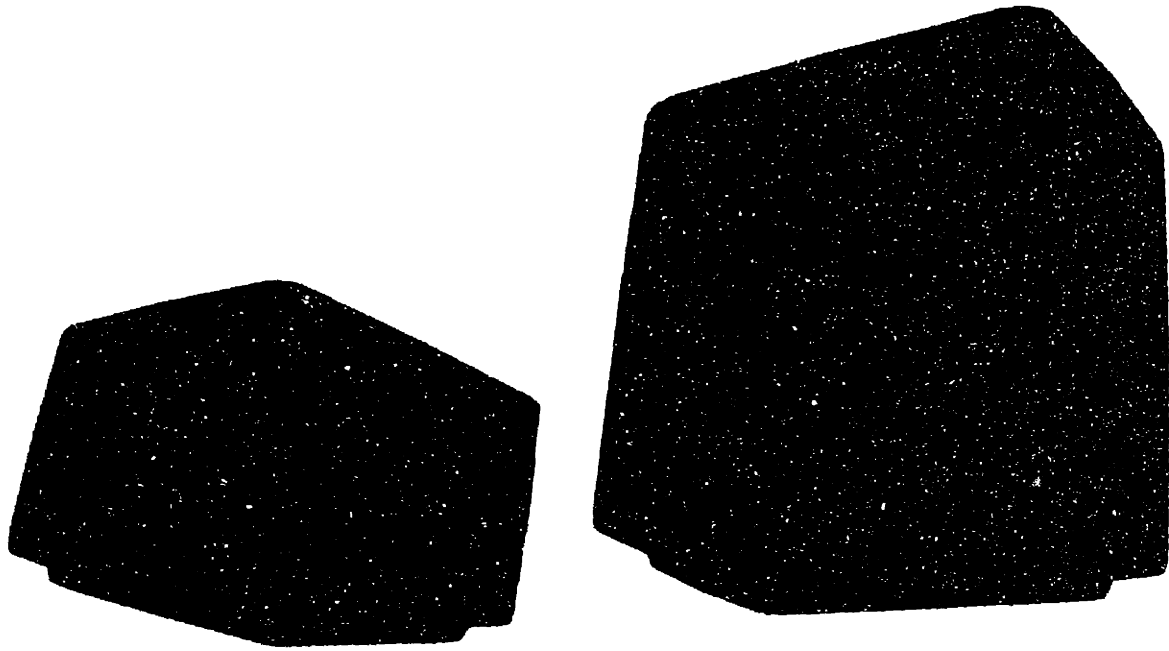


Figure B1 Braun-like desk top prototypes: horizontal and vertical parting line

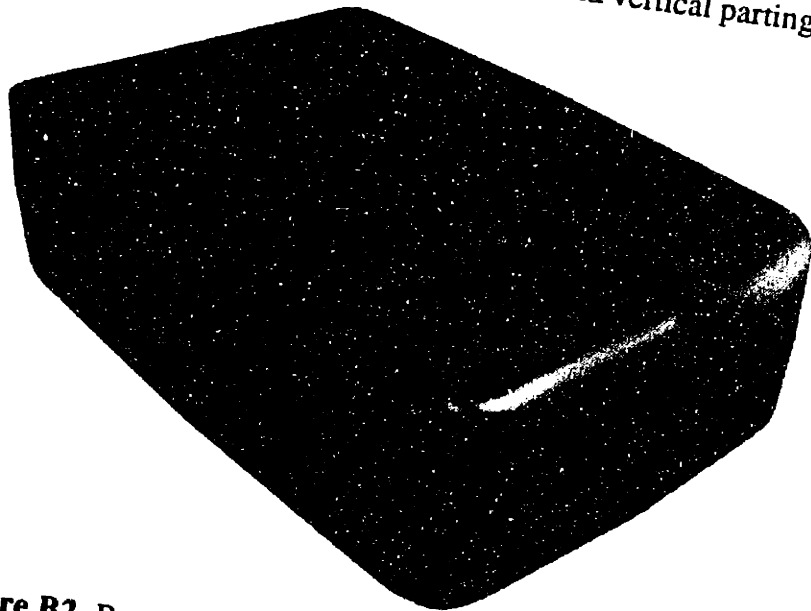


Figure B2 Braun-like hand held prototype:
horizontal and vertical parting line prototypes are the same

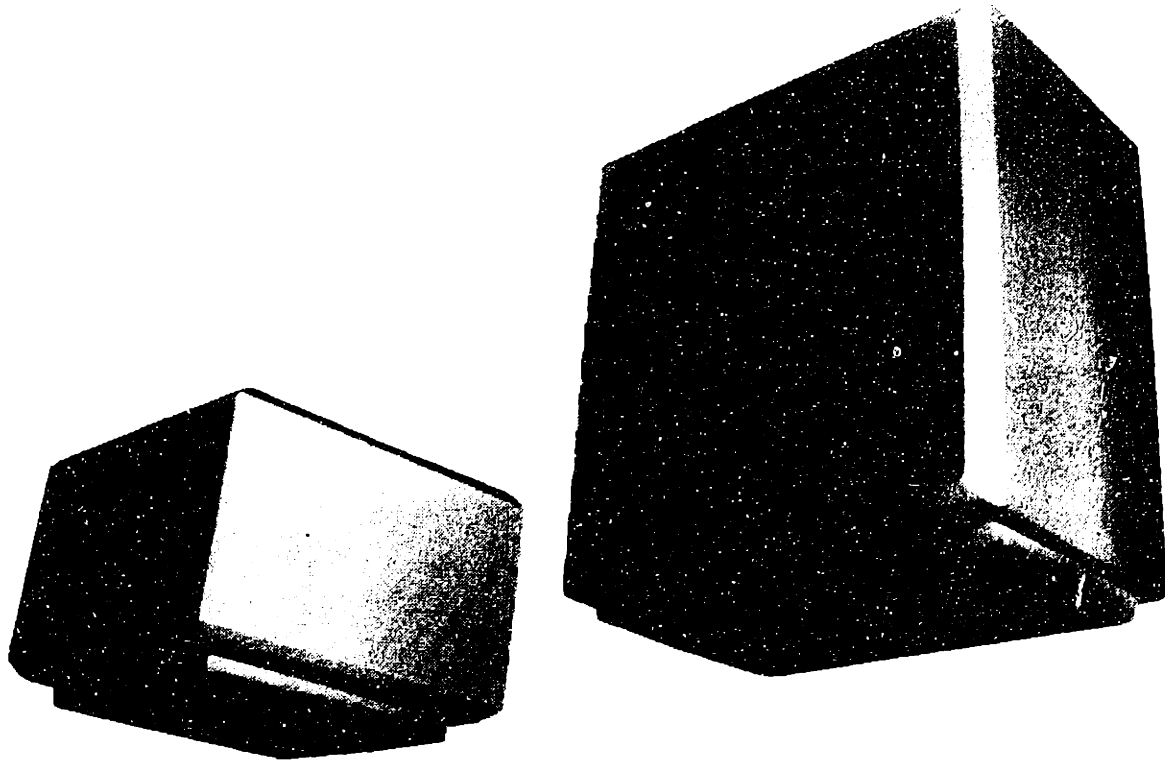


Figure B3 High-Tech desk top prototypes: horizontal and vertical parting line

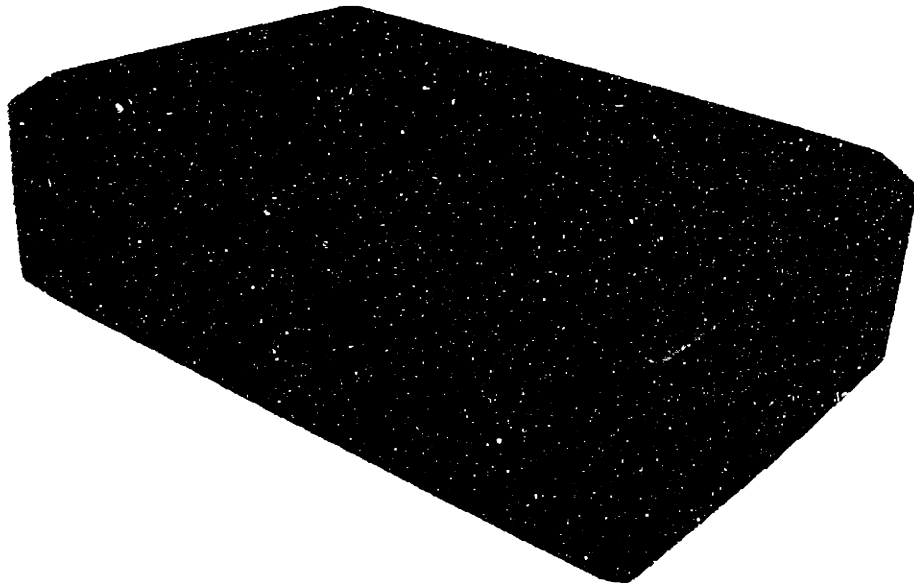


Figure B4 High-Tech hand held prototype:
horizontal and vertical parting line prototypes are the same

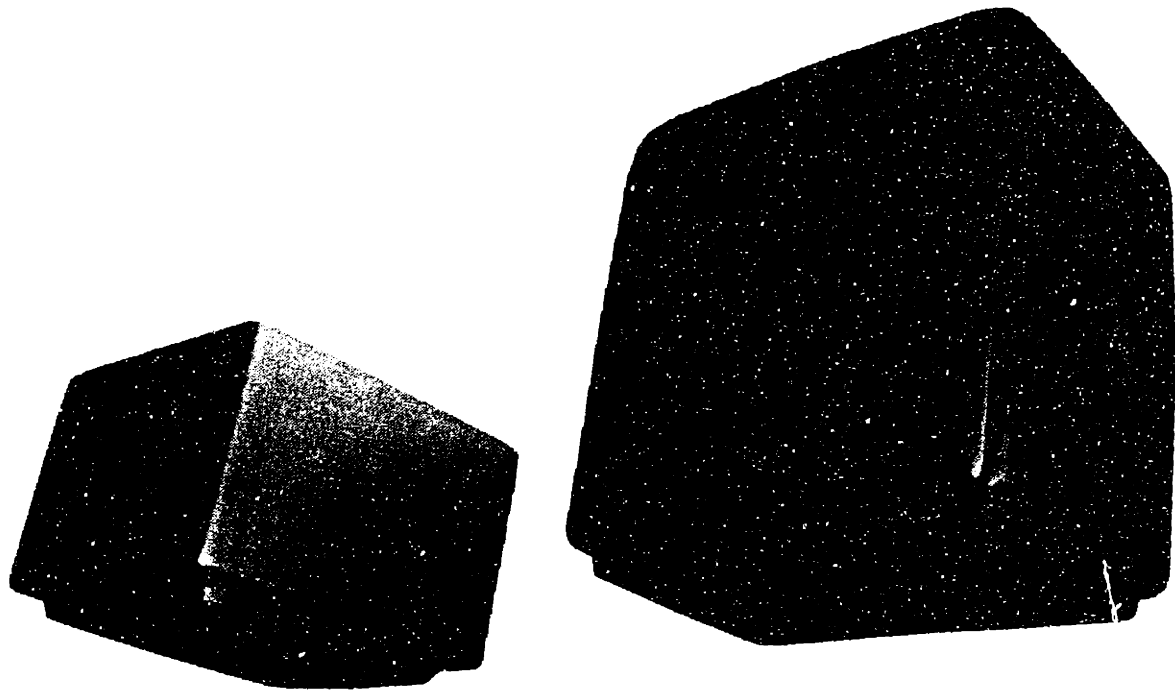


Figure B5 Art Deco desk top prototypes: horizontal and vertical parting line

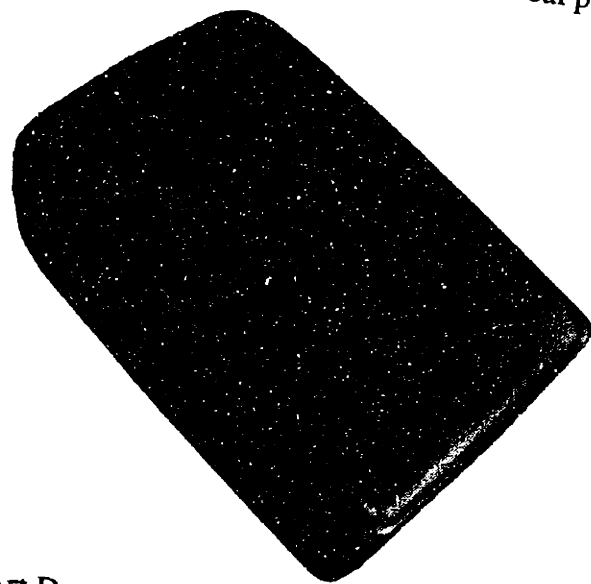


Figure B6 Art Deco hand held prototype:
horizontal and vertical parting line prototypes are the same

APPENDIX C Prototypes in the Detail Library

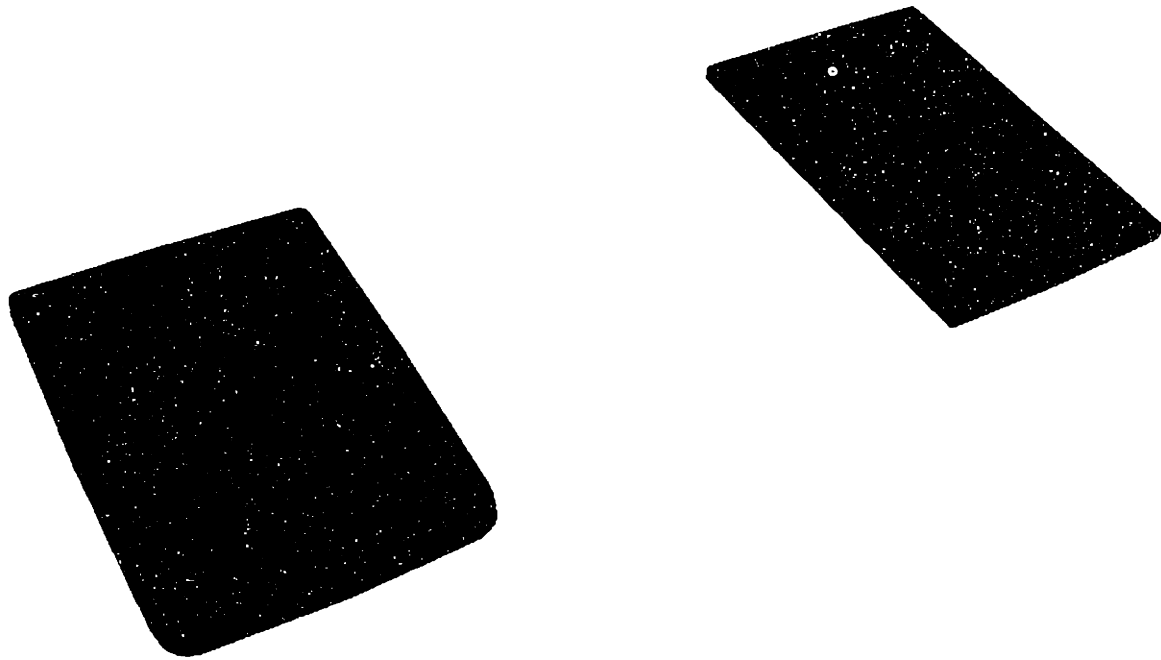


Figure C1 Panel inserts: Braun-like; Art Deco; High-Tech

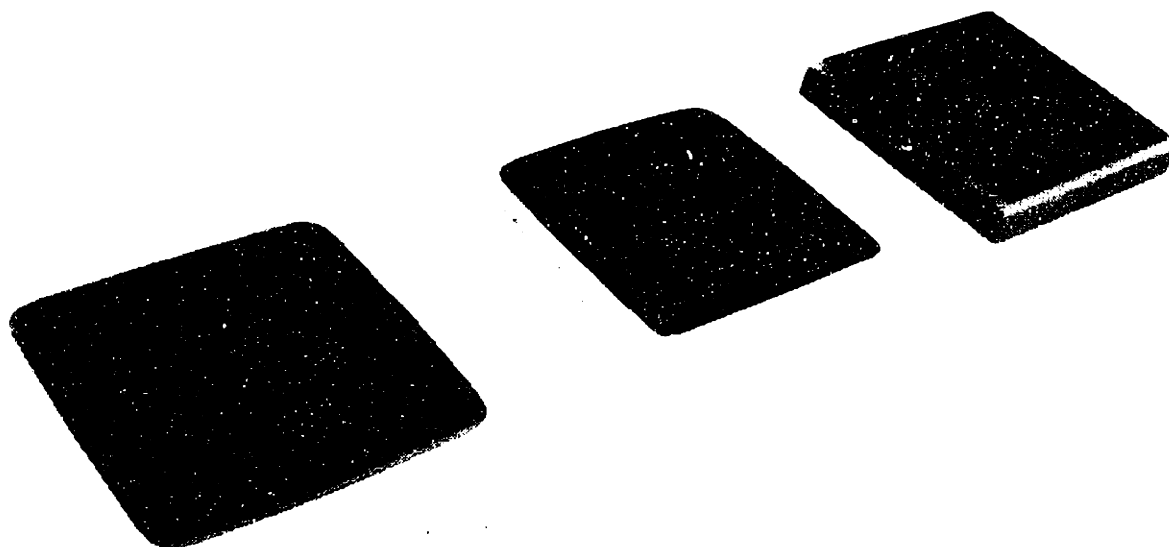


Figure C2 Speaker grill used for visual displays: Braun-like; Art Deco; High-Tech



Figure C3 Typical mask details used for visual displays: Braun-like; Art Deco; High-Tech

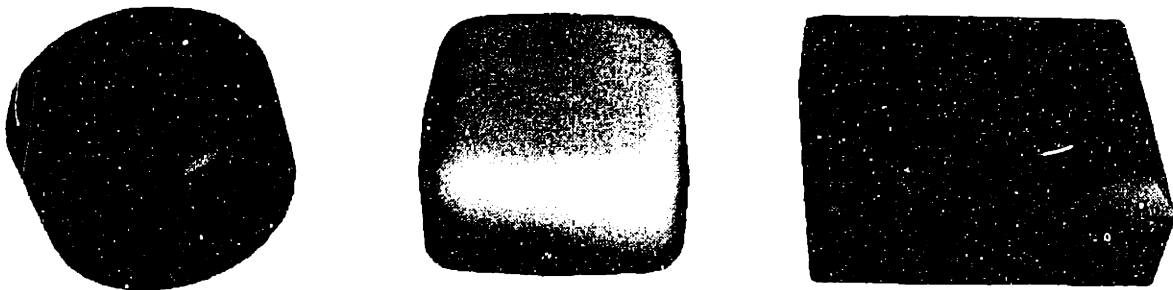


Figure C4 Buttons used for keypads and switches: Braun-like; Art Deco; High-Tech



Figure C5 Toggle switches: Braun-like; Art Deco; High-Tech

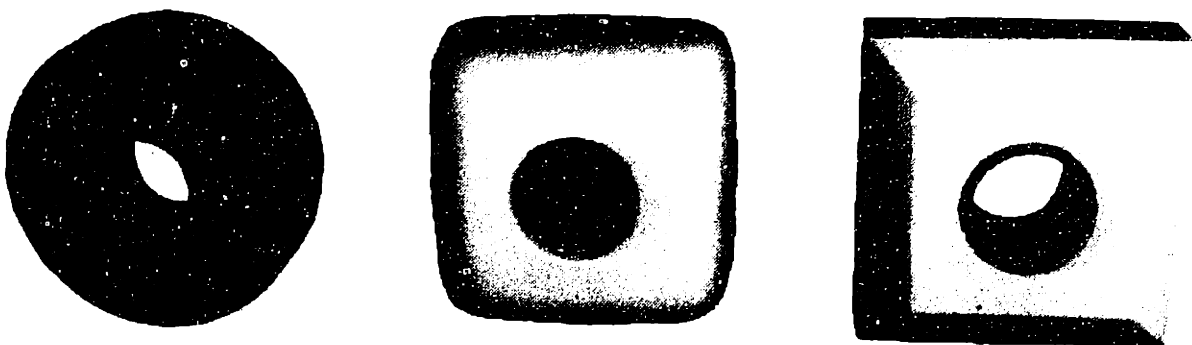


Figure C6 Jack covers for speaker and headphone jacks: Braun-like; Art Deco; High-Tech

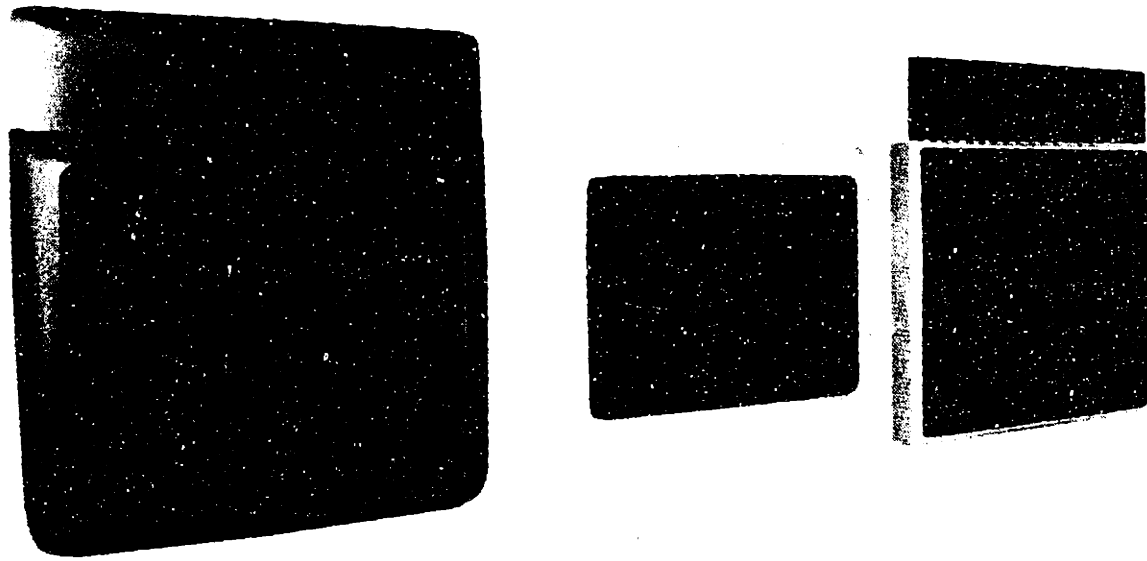


Figure C7 Cassette Door: Braun-like; Art Deco; High-Tech