Machine Learning for Design, Week 3

Prototyping with Generative Models

Four chairs from one prompt



seed 1



seed 2



seed 3



seed 4

Prompt: “a new side chair for a museum cafe, inspired by 18th and 19th century museum chairs, product photo on a plain grey backdrop”; FLUX.1 [schnell], one image per seed

What can an image from a model show about a design?

Today

1. What prototypes prototype
2. How an image model works
3. Sketching with a model
4. Discussion
5. Workshop: sketch to image
6. Partner interviews

1. What prototypes prototype

Today's reading: Houde and Hill, “What Do Prototypes Prototype?,” Handbook of Human-Computer Interaction, 1997



What do Prototypes Prototype?

Stephanie Houde and Charles Hill
Apple Computer, Inc.
Cupertino, CA, USA
s.houde@ix.netcom.com, hillc@ix.netcom.com

1. INTRODUCTION

Prototypes are widely recognized to be a core means of exploring and expressing designs for interactive computer artifacts. It is common practice to build prototypes in order to represent different states of an evolving design, and to explore options. However, since interactive systems are complex, it may be difficult or impossible to create prototypes of a whole design in the formative stages of a project. Choosing the right kind of more focused prototype to build is an art in itself, and communicating its limited purposes to its various audiences is a critical aspect of its use.

The ways that we talk, and even think about prototypes, can get in the way of their effective use. Current terminology for describing prototypes centers on attributes of prototypes themselves, such as what tool was used to create them, and how refined-looking or -behaving they are. Such terms can be distracting. Tools can be used in many different ways, and detail is not a sure indicator of completeness.

We propose a change in the language used to talk about prototypes, to focus more attention on fundamental questions about the interactive system being designed: What role will the artifact play in a user's life? How should it look and feel? How should it be implemented? The goal of this chapter is to establish a model that describes any prototype in terms of the artifact being designed, rather than the prototype's incidental attributes. By focusing on the purpose of the prototype—that is, on *what it prototypes*—we can make better decisions about the kinds of prototypes to build. With a clear purpose for each prototype, we can better use prototypes to think and communicate about design.

In the first section we describe some current difficulties in communicating about prototypes: the complexity of interactive systems; issues of multidisciplinary teamwork; and the audiences of prototypes. Next, we introduce the model and illustrate it with some initial examples of prototypes from real projects. In the following section we present several more examples to illustrate some further issues. We conclude the chapter with a summary of the main implications of the model for prototyping practice.

2. THE PROBLEM WITH PROTOTYPES

Interactive computer systems are complex. Any artifact can have a rich variety of software, hardware, auditory, visual, and interactive features. For example, a personal digital assistant such as the Apple Newton has an operating system, a hard case with various ports, a graphical user interface and audio feedback. Users experience the combined effect of such interrelated features; and the task of designing—and prototyping—the user experience is therefore complex. Every aspect of the system must be designed (or inherited from a previous system), and many features need to be evaluated in combination with others.

Prototypes provide the means for examining design problems and evaluating solutions. Selecting the focus of a prototype is the art of identifying the most important open design questions. If the artifact is to provide new functionality for users—and thus play a new *role* in their lives—the most important questions may concern exactly what that role should be and what features are needed to support it. If the role is well understood, but the goal

This article is published, in a different format, as Houde, S., and Hill, C., *What Do Prototypes Prototype?*, in Handbook of Human-Computer Interaction (2nd Ed.), M. Helander, T. Landauer, and P. Prabhu (eds.): Elsevier Science B. V: Amsterdam, 1997.

What prototypes prototype

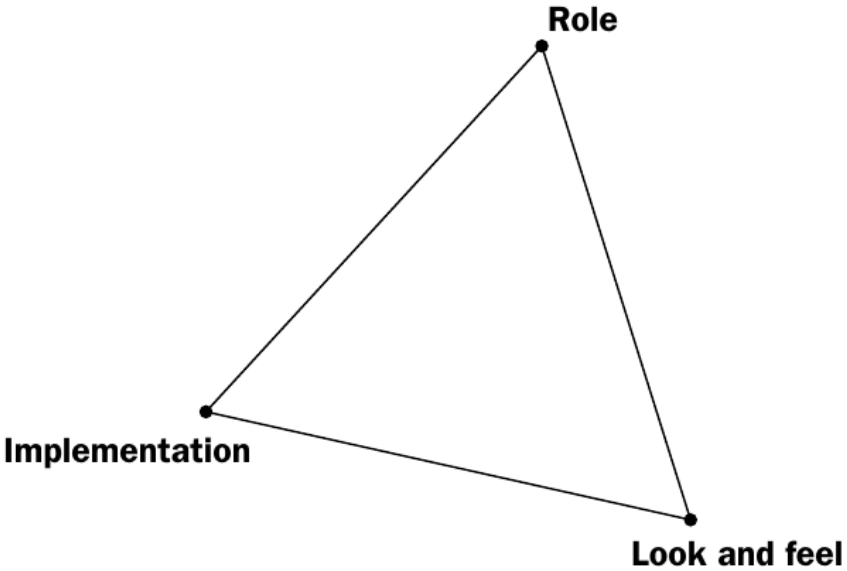


Figure 1. A model of what prototypes prototype.

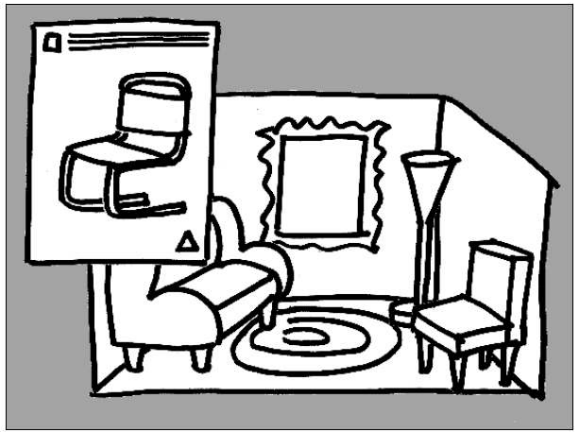
Role: what it does in someone's life

Look and feel: how it looks and feels

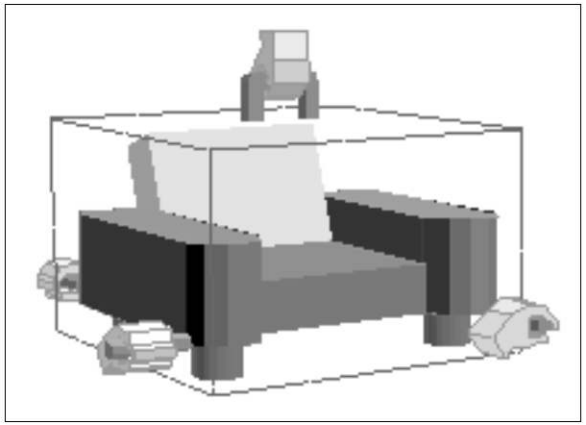
Implementation: how it works

Houde and Hill 1997, Figure 1

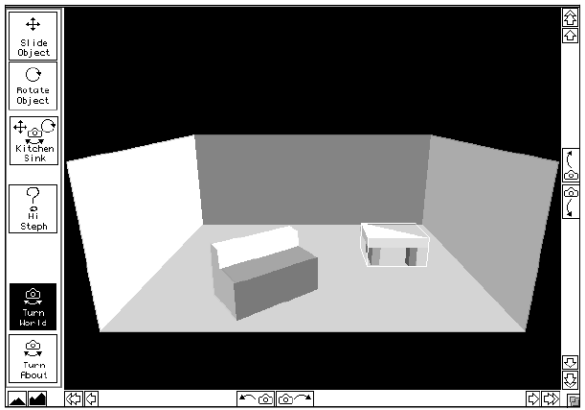
Three prototypes of one system



Role: a slide show of picking furniture from a catalog



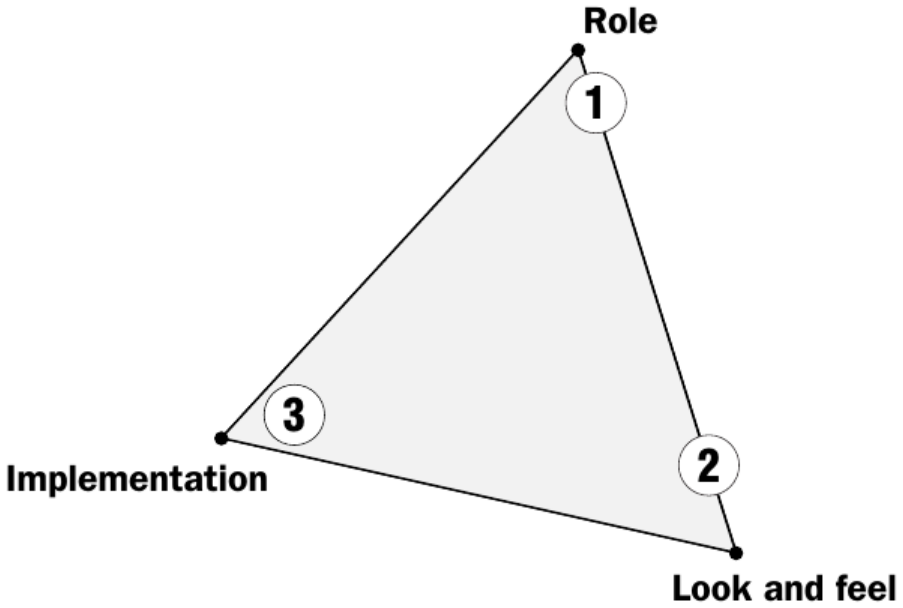
Look and feel: a “handle box” for lifting and turning a chair



Implementation: real-time 3D on a Macintosh IIfx

Houde and Hill 1997, Examples 1-3, a 3D space-planning application (Houde 1990, Chen 1990)

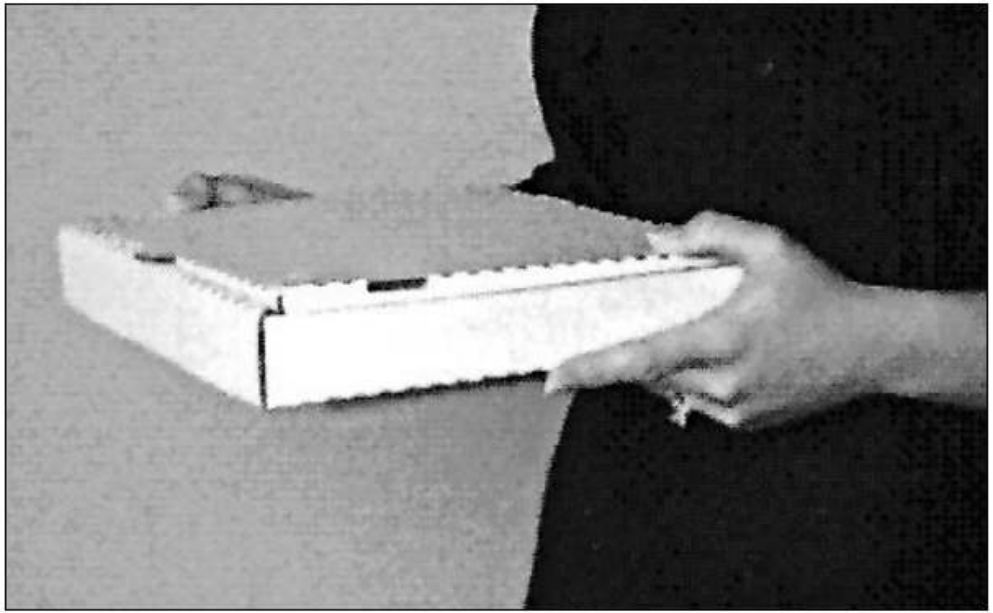
The three prototypes on the triangle



Answers from each fed
into one integrated design.

Houde and Hill 1997, Figure 2

A pizza box as a prototype



Weighted like the planned computer and carried by an architect on a site visit

Houde and Hill 1997, Example 10 (Apple Design Project, 1992)

“Finished-looking (or -behaving) prototypes are often thought to indicate that the design they represent is near completion.”



Houde and Hill 1997, p. 369; image: seed 3 from slide 2

The four chairs on the triangle

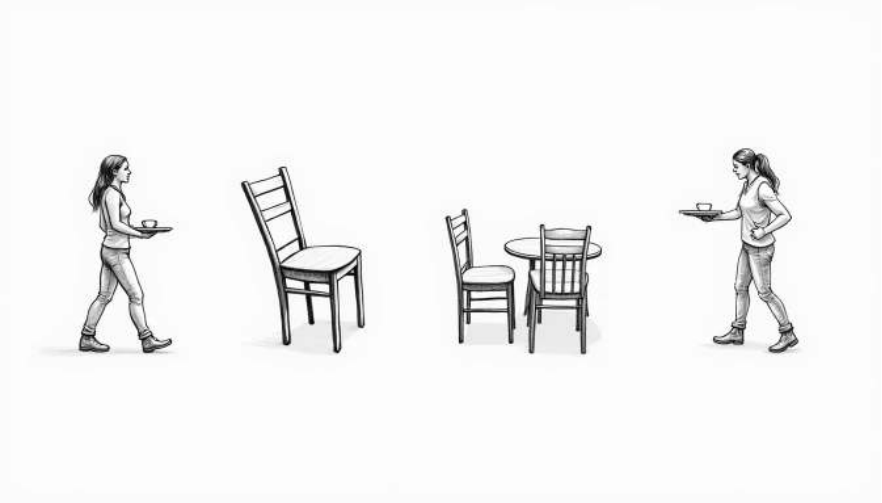


Look and feel: **yes**

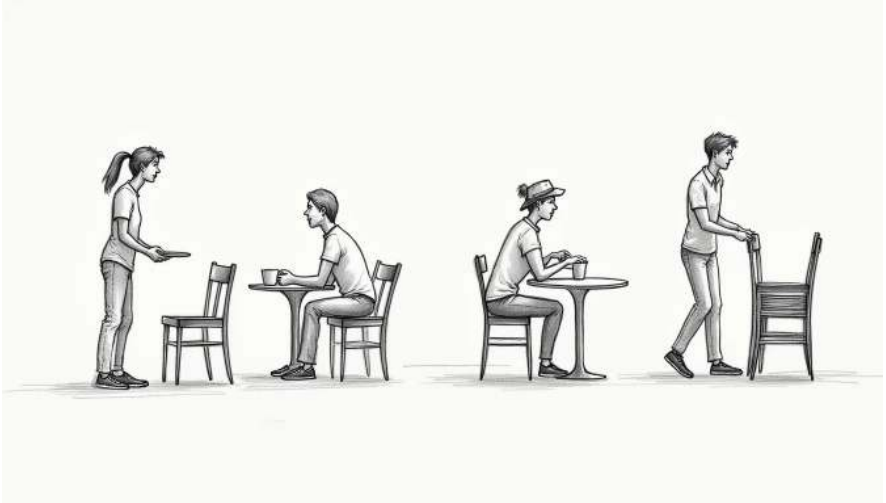
Role: **no**

Implementation: **no**

A storyboard from a model



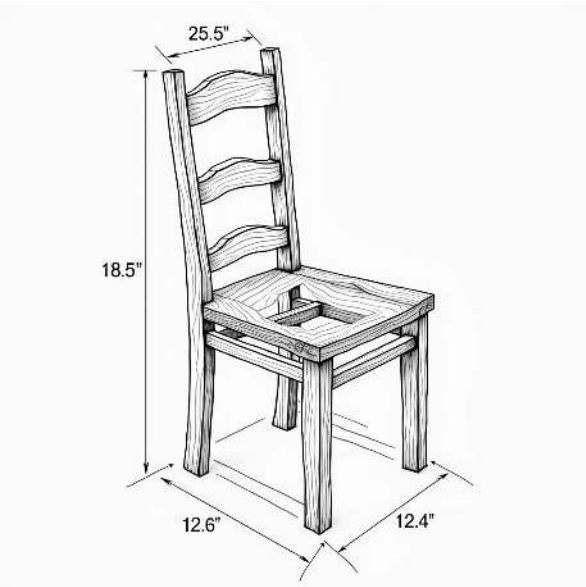
seed 1



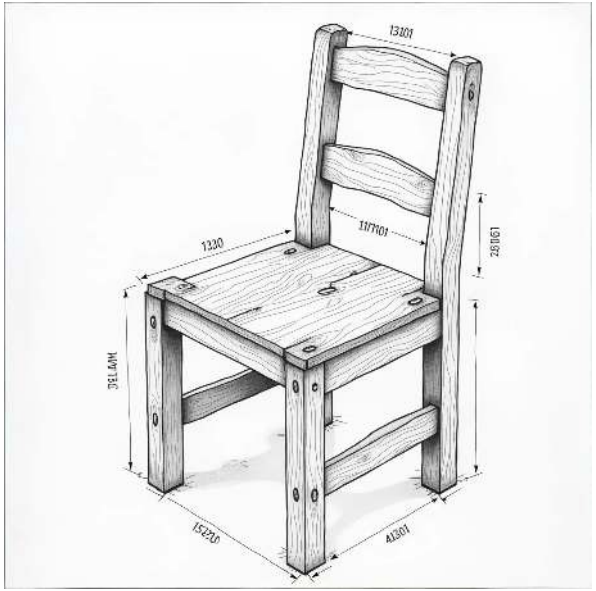
seed 2

Prompt: “a storyboard with four pencil-sketch panels: a museum visitor carries a tray, pulls out a simple oak side chair, sits at a small round cafe table, and a staff member stacks the chairs at closing time”; FLUX.1 [dev]

A technical drawing from a model



seed 1: inches



seed 2: numbers
that do not read

Asked for every
joint, in millimeters

Prompt: “an exploded view technical drawing of a simple oak side chair, showing every joint, with dimensions in millimetres, black lines on white paper”; FLUX.1 [dev], seeds 1 and 2

“Make this chair stackable”



before



after

A second seat on the same frame

Edit instruction “make this chair stackable” on seed 3 from slide 2; FLUX.1 Kontext [pro], seed 1

2. How an image model works

Today's reading: Rombach et al., “High-Resolution Image Synthesis with Latent Diffusion Models,” CVPR 2022



This CVPR paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

High-Resolution Image Synthesis with Latent Diffusion Models

Robin Rombach[1] * Andreas Blattmann[1] * Dominik Lorenz[1] Patrick Esser[2] Björn Ommer[1]
[1]Ludwig Maximilian University of Munich & IWR, Heidelberg University, Germany [2]Runway ML
https://github.com/CompVis/latent-diffusion

Abstract

By decomposing the image formation process into a sequential application of denoising autoencoders, diffusion models (DMs) achieve state-of-the-art synthesis results on image data and beyond. Additionally, their formulation allows for a guiding mechanism to control the image generation process without retraining. However, since these models typically operate directly in pixel space, optimization of powerful DMs often consumes hundreds of GPU days and inference is expensive due to sequential evaluations. To enable DM training on limited computational resources while retaining their quality and flexibility, we apply them in the latent space of powerful pretrained autoencoders. In contrast to previous work, training diffusion models on such a representation allows for the first time to reach a near-optimal point between complexity reduction and detail preservation, greatly boosting visual fidelity. By introducing cross-attention layers into the model architecture, we turn diffusion models into powerful and flexible generators for general conditioning inputs such as text or bounding boxes and high-resolution synthesis becomes possible in a convolutional manner. Our latent diffusion models (LDMs) achieve new state of the art scores for image inpainting and class-conditional image synthesis and highly competitive performance on various tasks, including unconditional image generation, text-to-image synthesis, and super-resolution, while significantly reducing computational requirements compared to pixel-based DMs.

1. Introduction

Image synthesis is one of the computer vision fields with the most spectacular recent development, but also among those with the greatest computational demands. Especially high-resolution synthesis of complex, natural scenes is presently dominated by scaling up likelihood-based models, potentially containing billions of parameters in autoregressive (AR) transformers [64, 65]. In contrast, the promising results of GANs [3, 26, 39] have been revealed to be mostly confined to data with comparably limited variability as their adversarial learning procedure does not easily scale to modeling complex, multi-modal distributions. Recently, diffusion models [79], which are built from a hierarchy of denoising autoencoders, have shown to achieve impressive

*The first two authors contributed equally to this work.

Figure 1. Boosting the upper bound on achievable quality with less agressive downsampling. Since diffusion models offer excellent inductive biases for spatial data, we do not need the heavy spatial downsampling of related generative models in latent space, but can still greatly reduce the dimensionality of the data via suitable autoencoding models, see Sec. 3. Images are from the DIV2K [1] validation set, evaluated at 512^2 px. We denote the spatial downsampling factor by f. Reconstruction FIDs [28] and PSNR are calculated on ImageNet-val. [12]; see also Tab. 8.

results in image synthesis [20, 82] and beyond [7, 44, 47, 56], and define the state-of-the-art in class-conditional image synthesis [15, 30] and super-resolution [70]. Moreover, even unconditional DMs can readily be applied to tasks such as inpainting and colorization [82] or stroke-based synthesis [53], in contrast to other types of generative models [19, 45, 67]. Being likelihood-based models, they do not exhibit mode-collapse and training instabilities as GANs and, by heavily exploiting parameter sharing, they can model highly complex distributions of natural images without involving billions of parameters as in AR models [65].

Democratizing High-Resolution Image Synthesis DMs belong to the class of likelihood-based models, whose mode-covering behavior makes them prone to spend excessive amounts of capacity (and thus compute resources) on modeling imperceptible details of the data [16, 71]. Although the reweighted variational objective [29] aims to address this by undersampling the initial denoising steps, DMs are still computationally demanding, since training and evaluating such a model requires repeated function evaluations (and gradient computations) in the high-dimensional space of RGB images. As an example, training the most powerful DMs often takes hundreds of GPU days (e.g. 150 - 1000 V100 days in [15]) and repeated evaluations on a noisy version of the input space render also inference expensive,

Denoising steps

What the model holds after each step



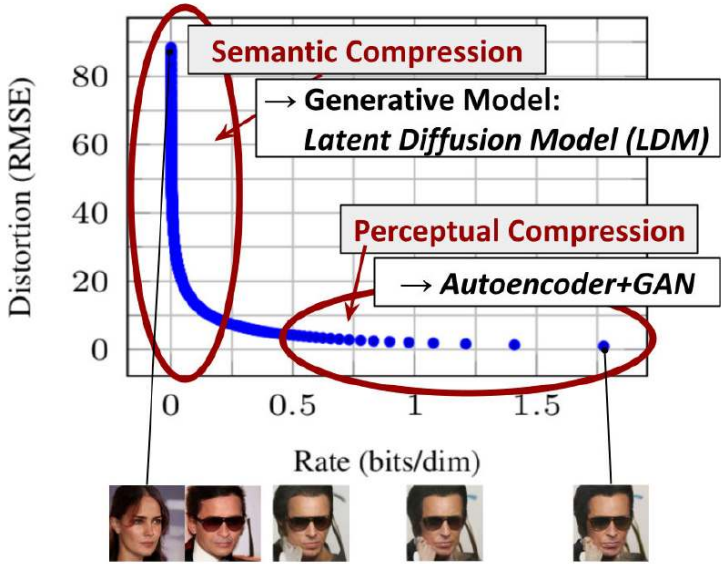
Its guess of the finished chair at that step



step 0 | 1 | 4 | 7 | 14 | 21 | 28

FLUX.1 [dev] on a laptop, 28 steps in 30 seconds, seed 1. Prompt: “a side chair for a museum cafe, oak, product photo on a plain grey backdrop”. Bottom row: the latent at that step minus the model's prediction times the noise level.

Latent space

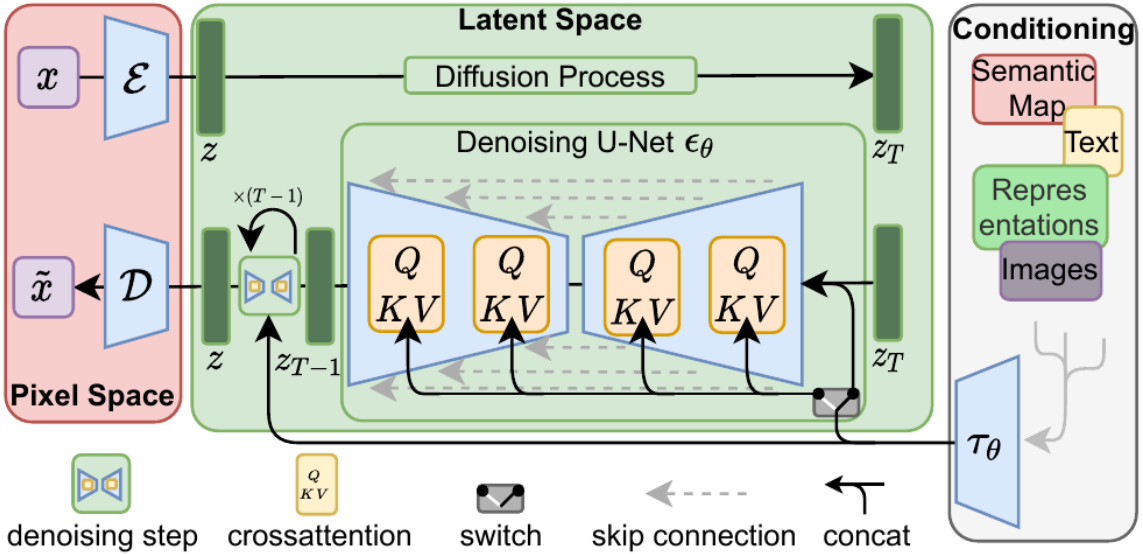


Most bits of an image
are details we cannot see

Our chair in FLUX.1: **96 × 96 × 16** numbers, against **768 × 768 × 3** pixel values, about **12** times fewer

Rombach et al., CVPR 2022, Fig. 2. Our numbers: the FLUX.1 autoencoder at 768 × 768 pixels

The parts of the model



Left: the autoencoder,
E in and **D** out

Middle: the denoising network, run once per step

Right: the prompt, read through cross-attention at every step

A sketch can be stacked onto the latent (concat)

Rombach et al., CVPR 2022, Fig. 3. FLUX.1 uses a transformer in place of the U-Net.

3. Sketching with a model

Today's reading: Schön, “Designing as Reflective Conversation with the Materials of a Design Situation,” Research in Engineering Design, 1992



Res Eng Des (1992) 3:131–147

Research in Engineering Design
Theory, Applications, and
Concurrent Engineering
© 1992 Springer-Verlag New York Inc.

Designing as Reflective Conversation with the Materials of a Design Situation

Donald A. Schon
Department of Urban Studies and Planning, Massachusetts Institute of Technology, Cambridge, Massachusetts, USA

Abstract. What are the prospects for applying the methods of Artificial Intelligence to architectural designers' knowing-in-action? David Marr (Marr, 1982) has advanced the idea of a computational theory of vision, which requires defining the information processing tasks carried out in vision. I ask, by analogy: what are the information processing tasks carried out in design? In order to answer this question, I propose, one ought to study design phenomenology. I illustrate several such studies, based on observations of a design studio, the performance of a design exercise, and the playing of a design game. In order to simulate the transactions with a design situation illustrated in these studies, the computer would have to carry out processes that begin prior to the presentation of what are normally defined as "design inputs." Such processes involve the construction of "design worlds," and they include: the simplest unit of design experimentation, the designer's seeing-moving-seeing; constructing figures from marks on a page; appreciating design qualities; setting design intentions and problems; recognizing the unintended consequences of a move experiments; storing and deploying prototypes; and communicating across divergent design worlds. I conclude that the practitioners of Artificial Intelligence in design would do better to aim at producing design assistants rather than knowledge systems phenomenologically equivalent to those of designers.

1 Introduction

I shall begin with a set of propositions:

Design research, in its artificial intelligence (AI) version, is an attempt to capture design knowledge by embodying it in procedures expressable in a computer program.

Design knowledge is knowing-in-action, revealed in and by actual designing. It is mainly tacit, in several senses of the word: designers know more than they can say, tend to give inaccurate descriptions of what they know, and can best (or only) gain access to their knowing-in-action by putting themselves into the mode of doing—as, to take an example of another, perhaps more familiar kind of skill, a touch-typist, who cannot say offhand just where all the letters are located on the keyboard, can begin to type, even on an imaginary keyboard, and thereby find the "T" just underneath the second finger of the left hand, the "L" just underneath the fourth finger of the right hand, and so on.

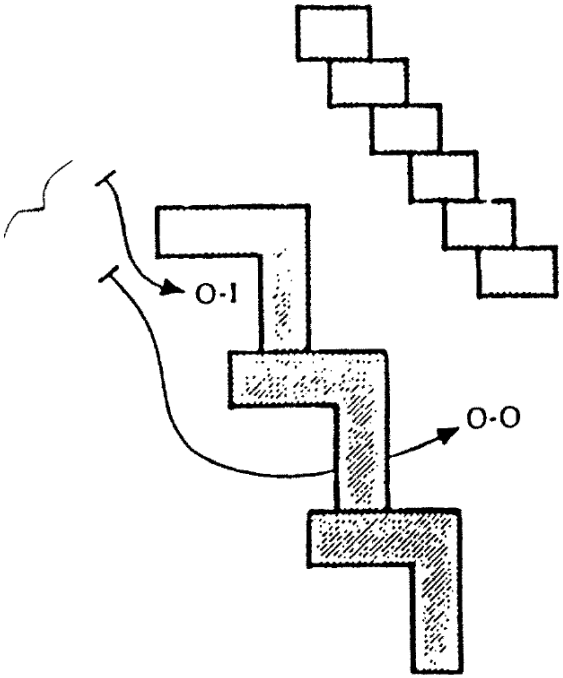
Symbolic, procedural representations of tacit design knowledge are bound to be incomplete or inadequate in relation to the actual phenomena of designing, as I shall try to show. But whether this matters depends on the purpose of the exercise—whether, that is, we seek,

1. To achieve a design output, given some input, as well as or better than designers ordinarily do it, but without particular reference to the *ways* in which they do it. This is the Turing test, more or less, and I shall call it "functional equivalence."
2. To reproduce *how* we actually go about designing; this I shall call "phenomenological equivalence."
3. To assist designers in their designing.
4. To provide an environment for research aimed at understanding how designers design.

The most ambitious purpose would be to build an AI-design version of David Marr's computational theory of vision (Marr, 1982). For Marr, an information-processing approach to vision meant defining the information-processing tasks carried out in vision, making "explicit statements about what is being computed and why" (Marr, p. 19). His formulation of the goal of the visual computation was basically to get from images on the retina to useful descriptions of the shapes and organization of objects in space. This, he thought, required specifi-

Offprint requests: Department of Urban Studies and Planning, Massachusetts Institute of Technology, 77 Massachusetts Avenue, Cambridge, MA 02139, USA

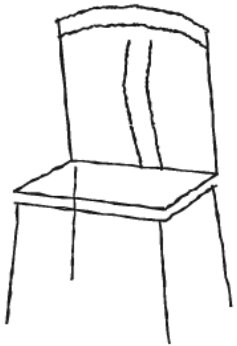
Petra's drawing



“I had six of these classroom units but they were too small in scale to do much with. So I changed them to this more significant layout (the L-shapes).”

Schön 1992, Fig. 2 and p. 133

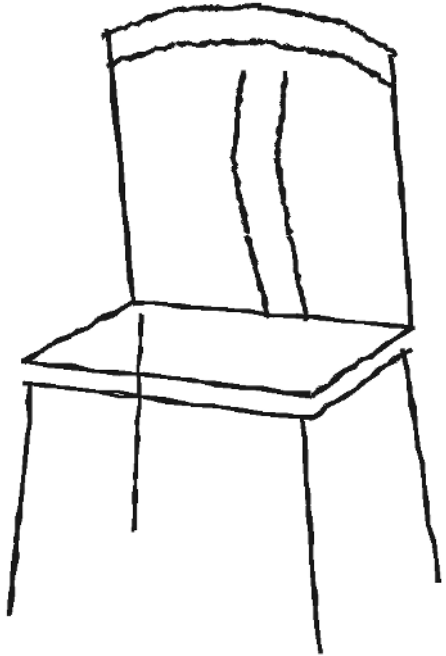
“This process of seeing-drawing-seeing is one kind of example of what I mean by designing as a reflective conversation with the materials of a situation.”





Schön 1992, p. 133; right: our sketch and one image from it (slide 22)

Sketch to image



A rough sketch, drawn with a few lines of code



Prompt: “a side chair for a museum cafe, oak, product photo on a plain grey backdrop”; FLUX.1 Canny [dev] with the sketch as the control image, seeds 1 to 4

Same sketch, different words



“bent plywood”



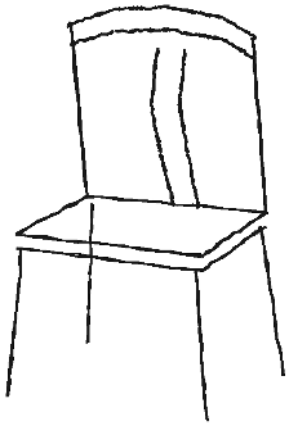
“painted green
with a cane seat”



“walnut with an
upholstered seat”

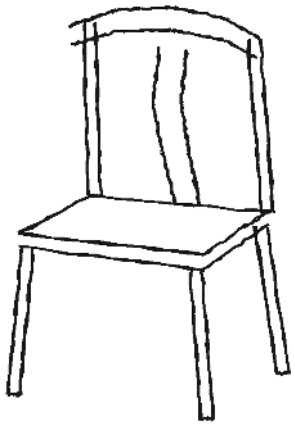
Prompt: “a side chair for a museum cafe, [words], product photo on a plain grey backdrop”; FLUX.1 Canny [dev], seed 1

Redrawing the legs





Legs as single lines





Legs as outlined bars

Same prompt (oak) and seed 1 for both; FLUX.1 Canny [dev]

Changing the legs with words



first image



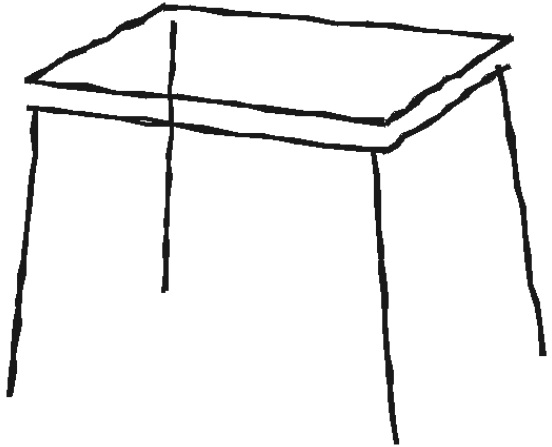
“make the legs and back posts solid oak instead of black metal”



sketch redrawn

Middle: FLUX.1 Kontext [pro] edit of the first image, seed 1. Right: FLUX.1 Canny [dev] from the redrawn sketch, seed 1

A sketch with no back



seat and legs only



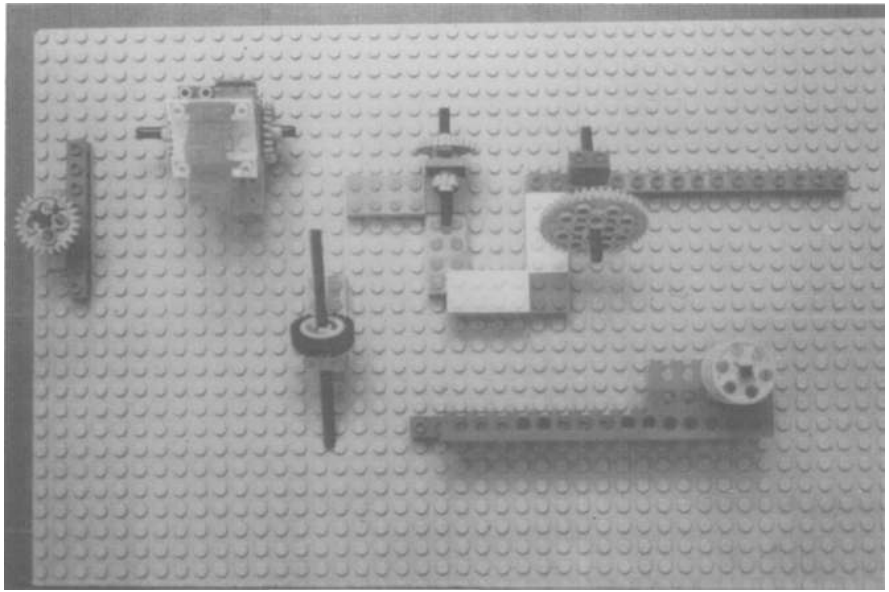
seed 1



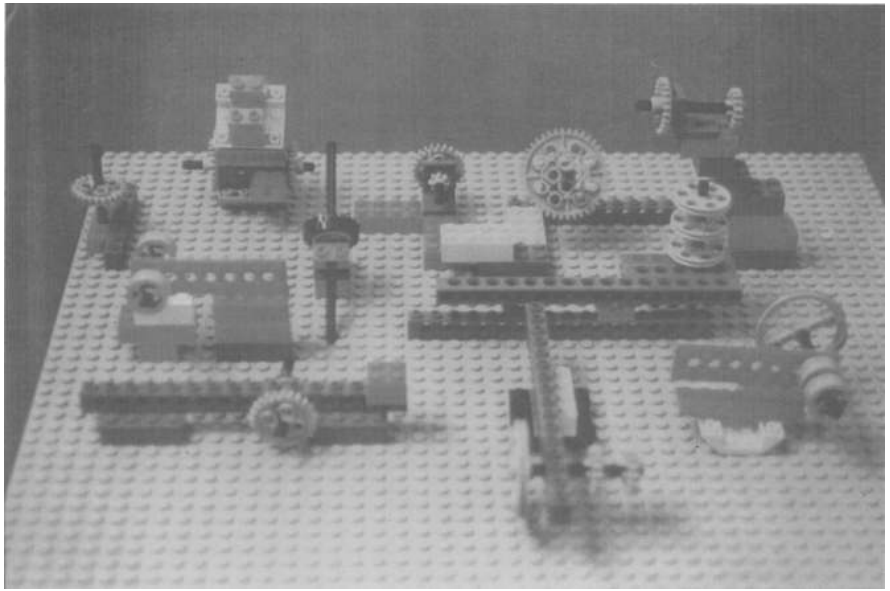
seed 2

Prompt: “a side chair for a museum cafe, oak, product photo on a plain grey backdrop”; FLUX.1 Canny [dev]

Schön's Silent Game



What Fred built (A): “odd angles”



What Turid built (B): “added on wheels”

Schön 1992, Figs. 9 and 10, pp. 142-144

Variations on one chair



seed 3 from slide 2







three new images from it

FLUX.1 Redux [dev], seeds 1 to 3, with no prompt

Discussion

1. Which corner of the triangle can a generated image reach?
2. What did a model show you that you did not draw?
3. When is a rough sketch better than a finished render?

Led by this week's discussion leads, with questions from the reading posts

Workshop: sketch to image

1. Sketch one design on paper and take a photo
2. Turn it into images with the notebook
3. Change one thing in the sketch and run it again
4. Ask for the same change in words, and compare

Due before week 4: both ways side by side, and a few sentences on which got closer to what you wanted

Partner interviews

Due this week

Next: Week 4

Data and Embeddings

Read before the studio:

- Radford et al., “Learning Transferable Visual Models From Natural Language Supervision,” ICML 2021
- Gebru et al., “Datasheets for Datasets,” CACM 2021
- Jiang et al., “AI Art and its Impact on Artists,” AIES 2023