Chapter 3

Cognitive Design Principles for Automated Generation of Visualizations

Barbara Tversky
Stanford University

Maneesh Agrawala
University of California, Berkeley

Julie Heiser
Adobe systems

Paul Lee
NASA-Ames

Pat Hanrahan, Doantam Phan
Stanford University

Chris Stolte
Tableau Software

Marie-Paule Daniel
LIMSI-CNRS, France

Before there were written languages, there were visualizations, painted in caves, inscribed in stone, or carved on wood. Visualizations of things that are actually visible, such as maps, building plans, people, and wildlife, are ancient and widespread. Visualizations of things that are metaphorically visual, like graphs of economic data or charts of organizations, are relatively new inventions, first appearing in Europe in the late 18th century (Beninger & Robyn, 1978). Like written language, visualizations are cognitive tools, designed to augment the capacity of the human mind (see Tversky, 2001, 2005, for further discussion). Then, as now, they serve myriad purposes: to record information; to lighten the burden of working memory; to convey information to others; to promote discovery, inference, and insight; and to facilitate collaboration. Primary among the roles of vizualization is communication. Some visualizations, notably maps, have undergone years of informal user testing as communication tools. The consequent refinements provide suggestions for good design, and the process of refinement provides suggestions for uncovering design principles. Other visualizations are the newest products of the latest computer tools, often in need of refinement.

Now, visualizations seem to be everywhere, in automobiles, newspapers, airports, textbooks, and instructions. As users know all too well, many of these visualizations are frustrating; they are complex and cluttered, often with extraneous and distracting decorative details. Like all communication, to be effective, visualizations should schematize effectively, that is, they should extract, emphasize, and even distort the information that is important to the task and eliminate the information that is not. Take maps as an example. An aerial photograph, despite its photorealism, is not an effective road map. It portrays detail that is not only irrelevant to the road structure, but hides it. An effective map for driving emphasizes roads, intersections, and other features that aid navigation. In fact, road maps exaggerate the size of roads, so that they can be seen in the maps. Maps serve more purposes than providing driving directions. A good map for hiking will display other features such as trails and topography. A tourist map may deliberately mix perspective, to show an overview of the streets to allow people to find their ways, and frontal views of tourist sites, in order to allow people to recognize them. In order to facilitate perception and comprehension, maps may violate certain spatial relations, for example, exact distance, size, angle, and perspective.

In the best cases, visualizations have developed in a community of users who produce and comprehend them, refining and fine-tuning them to suit the circumstances through casual user testing in the wild. This informal process can be systematized in the laboratory. Participants can be asked to produce visualizations; characteristics of those visualizations can be extracted. These characteristics can be systematically varied and tested for comprehension in other participants. What is important and effective in visualizations depends on cognition, on how humans represent the task at hand, and how they perceive and represent the information presented in visualizations.

Creating effective visualizations, as found in instructions, textbooks, and other media, requires collaboration between graphic designers and domain experts, as well as testing the target audience. The domain experts know the content to be communicated; the designers know techniques for conveying content. In practice however, tightly intertwined collaboration between designers and domain experts is not always possible. Even when such collaboration is possible, it is not ideal, as designers may be educated in design, but not in the discipline, and domain experts may be educated in the discipline, but not in general design principles. In addition, designing is a time-consuming, labor-intensive process that cannot keep up with the increasing demand. Try to imagine, for example, the number of maps that are downloaded daily from Web sites. Currently, computers save labor, but primarily by replacing only low-level tools such as pens, brushes, ink, paint, and paper. To keep up with demand, it is essential that computers provide higher level design tools that can make it easier to quickly produce effective visualizations

We have undertaken a novel use for computers, to instantiate cognitive design principles in algorithms that automate the process of generating effective visualizations. Our approach combines research in cognitive and computer science in three iterating steps:

1. Revealing the mental representations people have for a given domain and the visual devices they use to convey it, yielding domain cognitive design principles.
2. Developing algorithms that create effective visualizations based on cognitive design principles.
3. Testing the visualizations to insure that they adequately convey the desired information.

In the section that follows, we consider the issues involved in this three-step approach and illustrate the approach in two domains in which we have worked: route maps and assembly instructions.

A MUTLIFACETED APPROACH

General Cognitive Design Principles

In summarizing a large number of studies comparing static and animated visualizations for teaching a broad range of concepts, we suggested that effective visualizations conform to two general principles (Tversky, Morrison, & Betrancourt, 2002). According to the *Principle of Congruence*, the structure and content of a visualization should correspond to the structure and content of the desired mental representation. According to the *Principle of Apprehension*, the structure and content of a visualization should be readily and accurately perceived and comprehended. To illustrate the depth of these principles, consider animations, which have become increasingly popular as tools for creating them become available.

Many animated diagrams, such as those showing how the heart works or how to operate equipment (see http://www.interactivephysics.com/simulations.html for examples of animated diagrams) at first appear to fulfill the congruence requirement in that they use change over time to convey change over time. Dozens of experiments, however, have failed to show benefits of animations over equivalent still diagrams in conveying information, from animations illustrating how a bicycle pump works to those illustrating how computer algorithms work. This is surprising, given the premises that animations use change in time to convey change in time, and the conclusion arouses controversy. Some of the failures of animation may be because the animations violate the Apprehension Principle. They are too complex or too rapid to be accurately perceived and conceived and, unlike static graphics, they cannot be reinspected at the viewers' own pace. However, animations also seem to violate the Congruence Principle. Although events in the world are continuous, they are typically understood as a sequence of discrete steps (e. g., Zacks, Tversky, & Iyer, 2001). The steps are the joints, the transitions from action to action. The joints of events performed by hands are often objects or object parts; those performed by feet are turns at landmarks (Tversky, Zacks, & Lee, 2004). The joints of events do not come at regular temporal intervals. If people conceive of animated events as sequences of discrete steps, then it may be more effective to visualize events in steps rather than requiring the user to do the segmentation. A sequence of stills, for example, may actually provide a more compatible cognitive match than an animation (Tversky, 2005). A sequence of stills allows viewers to directly compare the state of the system at each important step.

Individualizing Visualizations

The utility of many visualizations depends on their adaptability. For example, with maps, different schematizations of the same environment are desirable depending on the task and the user. Hikers, bikers, drivers, travelers, and surveyors all need different information just as those unfamiliar with an environment need different information from those familiar with it. Hand-designed maps can take such needs into account. But creating effective individualized maps by hand for every user in every situation would be too expensive and time-consuming to be practical. Even within the domain of route maps for drivers, the number of possible routes that drivers may want is inconceivably large.

Our vision is to create individualized visualizations automatically, using computer algorithms that instantiate cognitive design principles. To do this, we require methods for uncovering cognitive design principles and methods for incorporating these principles into computer algorithms. We describe both, using examples from two domains representative of large classes of visualizations, maps, and assembly instructions. Maps are one of the most ancient and pervasive of visualizations, whether drawn on paper or carved in wood or incised in stone (e.g., Brown, 1979). Assembly instructions are representative of a large class of visualizations that includes instructions on how to put something together and how to operate a complex system—as well as how complex systems, from hearts to corporate structures, function. Underlying such visualizations are the parts of the system and their spatial, temporal, or functional relations. Visualizations of systems, too, are ancient; frescoes in Egyptian tombs show how crops are grown and harvested. Within each domain, we have chosen to explore and develop examples that are likely to be familiar to the general public—for maps, route maps and for systems, assembly instructions.

Revealing Cognitive Design Principles

To make visualizations that are congruent with the desired mental representations requires techniques that reveal those internal mental representations. Cognitive psychology has a large bag of tricks for externalizing the internal. Reaction times are commonly used to this end. The reasoning typically goes as follows: If the mental representation has a certain format, then responses for retrieval tasks congruent with that format should be faster and more accurate than responses for formats that are not congruent with the format, as they require more onerous and time-consuming mental transformations. For example, maps that are not oriented in the direction of travel must be mentally transformed to correspond with the navigator's current orientation. Therefore people are generally slower and less accurate when using such maps. Similarly, certain patterns of error, grouping, and description follow from certain types of mental representations elucidating them. Returning to maps, people draw streets that are not parallel as parallel, suggesting that this is how they think about them (e. g., Tversky, 1981).

For complex mental representations, more open-ended techniques may be more revealing. One that we have adopted and describe here is to ask participants to construct descriptions and depictions for a given domain.

This captures the natural way that communication occurs, and provides us with a rich set of data. Analyzing the structure that is common to both descriptions and depictions provides the commonalities and differences between them. The common structure reveals the underlying representation. The features particular to descriptions and depictions provide insights into the design of visualizations, as visualizations are typically accompanied by language. These insights do not fully determine the visualizations. Guidelines must be drawn from other sources as well, and then tested by users in order to qualify as design principles. Comprehension usually goes beyond production, from babies learning to speak to adults using diagrams, so production sets a lower limit. Comprehension tasks, then, complement and supplement production tasks.

Creating Computer Algorithms

Creating effective visualizations entails numerous design decisions, as illustrated by a few examples. For a visualization of a route, what landmarks should be included and where should they be included? How much can angle and distance be distorted without confusing the user? For a visualization of a process that extends in time, how should the process be segmented into steps? How should the steps be ordered? For each step, what is the best view or perspective to show? Which details should be included, and which omitted? Which details need to be distorted, or shown as insets? Which extra-pictorial features need to be added, features such as lines, arrows, and highlighting? The cognitive research will inform these decisions, but it does not completely determine the algorithm. The cognitive principles may be in the form of trade-offs, or simply insufficient, so some aspects of algorithm generation may still rely on the educated sensibilities of the designer. User testing can help overcome these shortcomings, but it may never be possible to test all the possible design variants. Our approach is to build tools that are based on the cognitive design principles, but also allow users to override the automated decisions as necessary. Many of the design issues that arise are not unique to the particular examples we have chosen, so that their solutions will have generality to other domains.

ROUTE MAPS

Revealing Cognitive Design Principles for Route Maps

The map someone sketches to show a friend how to get to a party doesn't usually resemble a map from the United States Geological Survey or Rand

TABLE 3.1
Examples of Route Directions

DW 9

From Roble parking lot
R onto Santa Theresa
L onto Lagunita (the first stop sign)
L onto Mayfield
L onto Campus drive East
R onto Bowdoin
L onto Stanford Ave.
R onto El Camino
Go down few miles. It's on the right.

BD 10

Go down street toward main campus (where most of the buildings are as opposed to where the fields are) make a right on the first real street (not an entrance to a dorm or anything else). Then make a left on the 2nd street you come to. There should be some buildings on your right (Flo Mo) and parking lot on your left. The street will make a sharp right. Stay on it. That puts you on Mayfield road. The first intersection after the turn will be at Campus drive. Turn left and stay on campus drive until you come to Galvez Street. Turn Right. Go down until you get to El Camino. Turn right (south) and Taco Bell is a few miles down on the right.

BD 3

Go out St. Theresa
Turn Rt.
Follow Campus Dr. way around to Galvez
Turn left on Galvez.
Turn right on El camino.
Go till you see Taco Bell on your Right

Note. Adapted From Tversky and Lee (1998).

McNally. Nonetheless, such sketch maps have been used for hundreds of years, presumably with success. Sketch maps differ from the efforts of geographers in several ways. To find out how, Tversky and Lee (1998) approached students near a campus dorm, asking them if they knew the way to a popular fast food place. If they did, they were asked to either sketch a map or write down directions to the restaurant. Typical instructions appear in Table 3.1, and typical sketch maps in Fig. 3.1.

The maps shared a number of characteristics. They had an infrastructure of lines formed into paths and turns. The paths included (in this case,

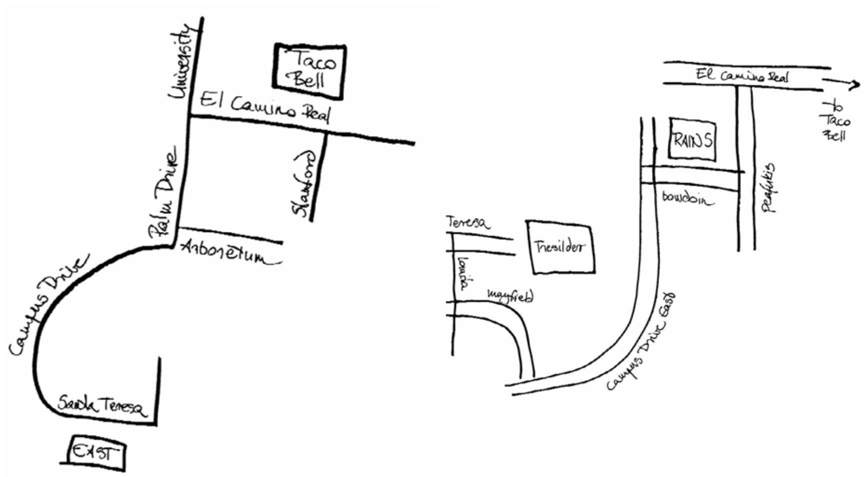

Fig. 3.1.
Two example sketch maps. Adapted from Tversky and Lee (1988).

roads) were primarily those that comprised the route. Other paths intersecting the route were included when they were useful for keeping the traveler on track, for example, intersections just before or after a turn. Paths were simplified, primarily to straight and curved lines. Turns were also simplified, to approximately 90 degrees. Distances were altered; long distances with little action were shortened and short distances with many turns were enlarged.

Note that these simplifications and alterations are actually distortions that violate the metric properties of the environment. However the topology of the sketched routes—the points of intersection between the roads—corresponded to the true topology of the routes in the real environment. Landmarks were included when they were important for turns or for keeping on track; they were typically expressed as names of streets or blobs representing structures and usually labeled. The text route descriptions shared many of the same properties. Exact angles of turns and distances were typically absent. The shape of paths was dichotomized just as in the depictions; for straight paths, informants wrote, "go down" and for curved ones, they wrote, "follow around." Despite these simplifications, omissions, and distortions, such schematized depictions and descriptions are usually sufficient to arrive at the destination because the environment provides the missing information about angles of turns, shapes of roads, and distances. What is essential is the sequence of paths and turns. Visualizations are used in contexts, and the contexts can provide missing information and disambiguate. Thus, the sequence of paths and nodes is the skeletal mental representation underlying both depictions and verbal descriptions of routes.

Of course, a highway map could be used for the same purpose. But a highway map, even one with the route marked on it, has disadvantages. It is cluttered with irrelevant information that makes finding the relevant information more difficult. It has a single scale, so large portions of the map convey no information, and at the same time, discerning many small turns may be difficult. Because a highway map doesn't extract the information needed for a mental representation of a route, it demands considerable time-consuming processing before it can be useful.

Applying Cognitive Design Principles to Computer Algorithms for Route Maps

Two basic design principles follow from the cognitive analysis:

1. People think of routes as sequences of paths and turns at landmarks and therefore the topology of the route (i.e. the turning points) must be depicted accurately.
2. People don't accurately apprehend or represent distances or angles, and therefore such geometric information can be simplified to increase emphasis on the turning points.

We have developed a system called LineDrive that automatically designs route maps for any given origin and destination based on these principles (Agrawala & Stolte, 2001). LineDrive is responsible for choosing graphic attributes, such as, position, orientation, size, extra, for each of the graphic elements in the map, including roads, labels, cross-streets and landmarks. The space of possible map designs encompasses all possible choices of graphic attributes for each of the graphic elements. LineDrive uses search-based optimization over this large multidimensional space to find the map that best adheres to the design principles. The design principles are instantiated as layout constraints within the search-based optimization framework.

Because LineDrive maps are based on a cognitive model of how people think about routes, they are far easier to follow while driving than standard highway maps (see Fig. 3.2). LineDrive has been commercially deployed on large Internet map service sites (see www.mapblast.com) and the maps been received enthusiastically by a large community of users.

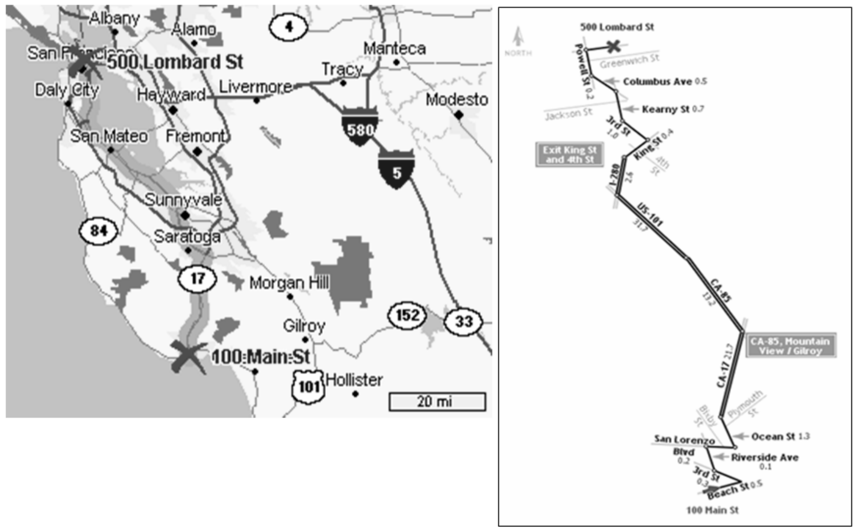

Fig. 3.2.
Highway map (left) and LineDrive map (right) for the same route. Adapted from Agrawala, (2002).

ASSEMBLY INSTRUCTIONS

Maps have been used to communicate spatial information within communities over the millennia, much like spoken language. As for spoken and written language, this provides a natural user-testing laboratory where some construct maps and others use them, with greater or lesser success—an ongoing process that refines and improves. Instructions for assembly or operation have not undergone that refinement, and are indeed the frequent recipient of groans, complaints, and slogans. An informal, but wide-reaching survey of instructions for assembling or operating the sorts of things people bring into their homes—furniture, cameras, cell phones, computers, and the now-proverbial VCR—reveals a number of common difficulties (some of this collection appears in Mijksenaar & Westendorp, 1999) . The typical visualization is an exploded view of the parts with lines and arrows. Such diagrams are usually at one scale and from one perspective, both of which may not be appropriate for all steps of assembly. The diagrams are frequently cluttered with so many parts and connections that it is difficult to discern particular components. They all too frequently show the entire assembly or operation at once, so that the sequence of assembly is not given. They often use extra-pictorial features such as arrows in multiple ways, with insufficient context to disambiguate meaning. As we shall see, even the instructions produced by student novices correct many of these problems. One notable exception to these common shortcomings of instruction is the widely admired instructions for Lego. Lego instructions are step-by-step; they also change perspective and scale when needed to show how to attach components.

Revealing Cognitive Design Principles for Assembly of Objects

The cognitive structures underlying assembly are multiple: A mental model of the object to be assembled, a mental model of the actions required for assembly, and a model for ordering the actions. People think of objects as a hierarchy of parts (Tversky & Hemenway, 1984). The parts segmented are those that are perceptually salient and functionally significant. In most cases, perceptual salience, that is, contour distinctiveness, and functional significance correlate, as in the wheels of an automobile or the handle of a pump or the legs of a chair. The correspondence between perceptual salience and functional significance promotes inferences from structure to function, especially when the form of the part suggests function, as in wheels, handles, and legs. Thus, objects, though sometimes seamless, are nevertheless perceived as consisting of distinct parts segmented by appearance and by function or behavior. The same holds for actions, such as making a bed or assembling an object. Though typically continuous in time, actions are thought of as a sequence as of discrete steps, distinct in both perceived action and conceived function. Goal-directed action sequences such as assembly are also conceived of hierarchically, with the higher level segmented by actions on separate objects or significant object parts and the lower level segmented by finely articulated actions on the same object or object part (Zacks, Tversky, & Iyer, 2001).

To reveal the mental representations underlying assembly and to reveal graphic preferences at the same time, we followed the same general strategy for uncovering cognitive design principles as we had for route maps. Heiser, and Tversky (in preparation) asked students to assemble a TV cart using the picture of the assembled cart on the package as a guide (see Fig. 3.3).

After assembling the cart, the students were asked to construct instructions for assembling it under one of four conditions: (1) Use sketches and language to create instructions so someone else can easily assemble the TV cart; (2) use sketches and language, but confine yourself to short, concise instructions; (3) use only language; and (4) use only sketches. This is

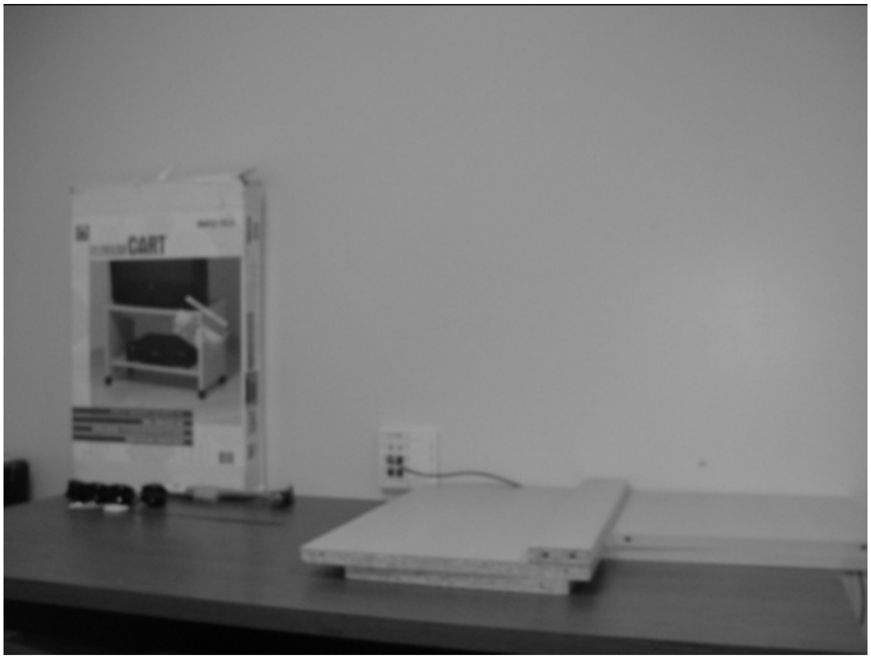

Fig. 3.3.
Participants used the photo of the fully assembled TV cart on the box (to the left) to assemble the parts (to the right. Adapted from Heiser & Tversky in preparation).

the Instruction Production Study. As expected, the steps corresponded to the major object parts, yielding five steps. Of the many possible sequences, participants primarily used two, corresponding to mechanical ease of assembly. The students had been divided by a median split into high and low spatial ability on the basis of spatial tests of mental rotation (Vandenburg & Kuse, 1978) and perspective-taking (Money, & Alexander 1966). There were vast differences in the sketches produced by high-and low ability participants, which are described in the following section. In a second study, the Instruction Rating Study, a subset of instructions from the production study were selected to span a range of sketch manners and techniques. These were given to a new group of participants, as before, split into high and low ability, who first assembled the TV cart using one of the sets of instructions and then rated the instructions for quality. As noted, there were large differences in the sketches produced by high-and low-spatial ability participants. Interestingly, the more sophisticated

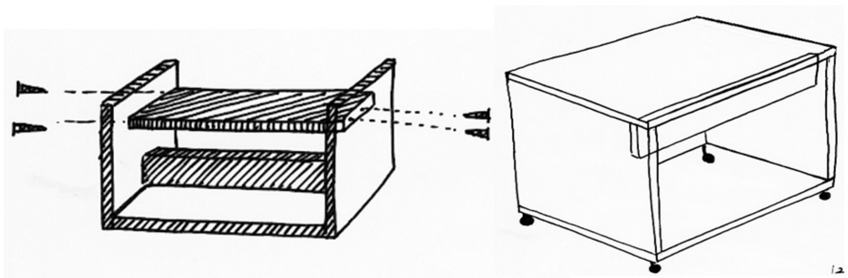

Fig. 3.4.
Examples of an action sketch and a structural sketch. The action sketch is easier to follow because it shows the assembly action that must be performed. In this case, the assembly action is screwing in the shelf.

techniques used in the sketches produced by those with high ability were exactly those preferred by participants of all ability levels in the rating study.

What were the diagrammatic techniques that the high-ability participants used and that received high ratings?

- *Action sketches***.** Those high in ability produced more *action* sketches than *structural sketches,* whereas low-ability participants produced relatively more structural sketches (see Fig.3.4) As the labels imply, an action sketch shows the parts as well as how they are assembled whereas a structural sketch shows only the spatial relations among the parts. For assembly of objects, the action sketches are generally superior to structural sketches because they contain all the information in structural sketches and, in addition, show the actions required to assemble, without introducing too much additional visual.
- *Step-by-step sketches*. The high-ability participants produced more step-by-step sketches, reflecting the hierarchical organization of assembly (see Fig. 3.5).
- *3-D perspective*. The sketches of high-ability participants showed the assembly from a 3-D perspective that made the assembly operations visible (Fig. 3.5).
- *Morphograms*. Highly rated sketches produced by high-ability participants used extra-pictorial devices that we have termed *morphograms* (Tversky, 2003), notably, guide lines and arrows, to show points, direction, and mode of attachment and sequence of steps. *Morphograms* are a class of simple geometric elements used in diagrams, such as lines, arrows, crosses, and boxes, that are readily interpreted partly through

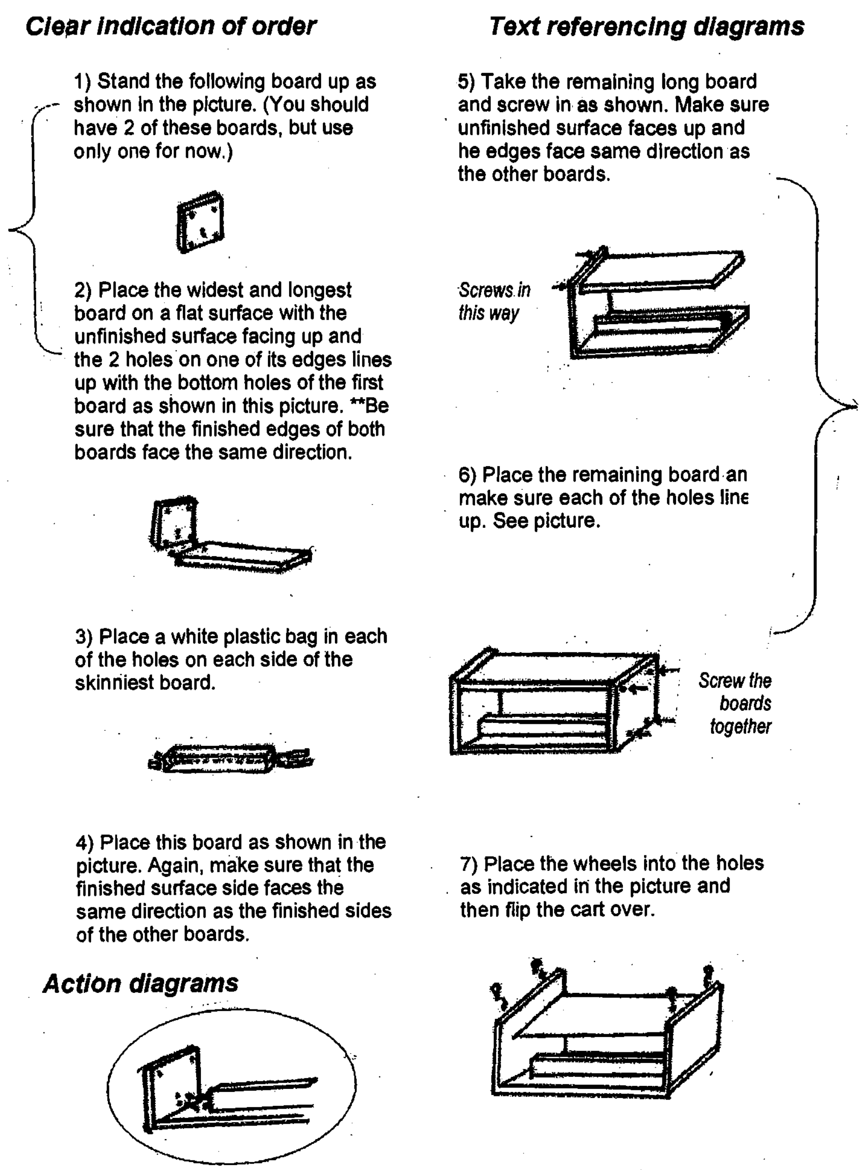

Fig. 3.5.
Instructions produced by one high ability participant, with some of the features characterizing good instructions highlighted.

their Gestalt or mathematical characteristics and partly through context (Tversky, Zacks, Lee, & Heiser, 2000).

- *Effective integration of text with diagrams*. In highly rated sketches, the text referred to the diagrams, elucidating each assembly step. They also supplied caveats and other information not readily available from the diagrams.

Just as in our study of route instructions, there were striking parallels between the depictions and the text descriptions. Nearly half of the statements described actions, such as "Attach other side panel to the open side." Another quarter described the components to be assembled, for example, "The top piece is rectangular." The remaining statements were commentaries and caveats, statements that began "remember that" or "make sure that." Under the concise condition, the proportion of action statements increased and the proportion of descriptions of components and commentaries decreased, suggesting that users believed that the assembly actions, the sequence of steps, were the critical information to be conveyed. In addition, the descriptions were hierarchical; that is, the higher level statements were retained in the concise descriptions at the expense of the lower level statements.

A third study, the Instruction Evaluation Study, tested whether the instructions that received high ratings were more effective. A new group of participants, again split by spatial ability, used instructions varying in rated quality to assemble the TV stand. Those high in ability assembled the stand faster and with fewer errors. For them, the quality of instructions made no difference. For those low in spatial ability, the instructions rated highly led to better performance than those with low ratings, indicating that people's intuitions about this type of instruction are calibrated. Recall that the participants in these studies were students in a highly selective university, so that the low-ability students are probably more representative of the population at large than the high-ability students. These studies are ongoing, the analyses are by no means complete, and we expect further insights to emerge as we continue.

These experiments provide cognitive guidelines for the construction of computer algorithms for assembly. They inform how we can segment a set of assembly actions into steps and substeps, based on the structure of the objects and actions. They also inform how to produce diagrammatic visualizations: We should use step-by-step action diagrams using 3-D views that show how to make the connections, enriched by lines and arrows. But they do not yet answer all of the design issues.

Applying Cognitive Principles to Computer Algorithms for Assembly of Objects

From the results of these experiments some design principles are already clear:

1. People prefer step-by-step instructions in which each step shows how one major part is attached rather than a single diagram showing all the assembly steps at once.
2. Each new part added to the assembly must be clearly visible in each step of the instructions.
3. The instructions must show how the new parts attach to the other parts in the assembly.
4. Action sketches separate the new parts from the old parts and use arrows and guidelines to show how the new parts should attach. Thus, action sketches explicitly show the operations required to perform each attachment and are preferable to structural sketches, which only show the parts included in each step.
5. In each step, the object may be oriented in either its most common real-world orientation (i.e. legs of table should point down and table top should appear horizontal) or in the orientation optimal for that assembly step. Some assemblies may become unstable or unbalanced as parts are attached. In such cases, the instructions may orient the object so that it would remain stable against a ground plane

We have developed an automated assembly instruction design system based on these principles (Agrawala et al., 2003). As input, our system requires a geometric model of the assembly with each part in its final assembled configuration. Users can optionally specify additional semantic information about the model by labeling the parts. In this manner, users can specify symmetries between parts and divide parts into fasteners and fastened parts.

Our system divides the task of generating assembly instructions into two phases; *assembly planning* and *diagram production*. In the assembly planning phase, the system analyzes the geometry of the model to compute the sequence of steps required to build the object. Assembly planning algorithms have been well studied in robotics (Romney, Godard, Goldwasser, & Ramkumar, 1995; Wilson, 1992). These algorithms first compute all possible disassembly sequences for the model by analyzing which parts block other parts. This analysis yields a directed acyclic graph encoding the removability constraints on each part, and every valid topological sort of this removability graph produces a geometrically valid assembly sequence for the object.

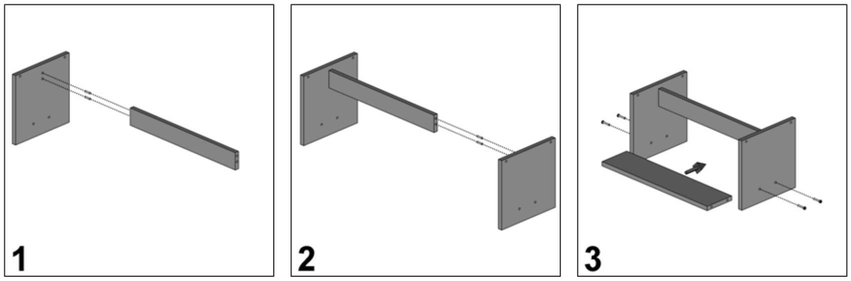

Fig. 3.6.
Instructions for assembling the TV stand as generated by our system. Only the first three steps are shown.

The goal of the prior robotics algorithms was to plan a sequence of instructions that a machine tool could use to build the object. Each valid assembly sequence is evaluated for the particular machine tool it will be implemented on and the "best" sequence is chosen as the final assembly plan. Because these robotics techniques are designed to produce an assembly plan for a robotic machine tool, they never need to produce a visual representation of the instructions and therefore do not consider the requirements of a human builder. As a result, the assembly sequence produced by such systems can be difficult for humans to follow. Our system computes the set of geometrically valid assembly sequences using the removability analysis approach. However, unlike the prior systems, we evaluate the resulting sequences based on the design principles just outlined to produce step-by-step assembly sequences that are well designed for humans. For example, visibility in the instructions affects decisions about assembly order.

Once we have computed the assembly plan, we must generate a diagram showing each step in the plan. We can generate structural sketches by simply showing the set of parts added in each step in their final assembled positions. Our system ensures that all of the parts added in each step and their points of attachment will be visible in each step. This approach yields instructions that are similar to those included with Lego, and we are able to produce assembly sequences that are easy for humans to use and follow. The assembly planning algorithm also determines the set of directions in which the new parts can be moved in to attach to the previous parts. We have recently extended our system to use this information to produce action sketches (see Fig.3.6). The new parts are placed a short distance away from the previous parts and guidelines and arrows are added to indicate the motions of the parts as well as the points of attachment.

Testing the Computer-Generated Instructions

We compared the instructions generated by the computer to those that came in the box with the TV cart, and to the best hand-drawn instructions (Heiser, Phan, Agrawala, Tversky, & Hanrahan, 2004). The instructions in the box consisted of a menu of parts, an exploded diagram of the TV cart with guidelines and arrows indicating attachment, and an enlargement showing attachment specifics. Because the TV cart has relatively few parts and attachment operations, these are not bad instructions. Nevertheless, the computer-generated instructions won hands-down. Ten participants assembled the TV cart using each set of instructions. Those using the computer-generated instructions made significantly fewer errors than those using either of the other sets of instructions while assembling 30% faster. They also reported greater satisfaction with the computer-generated instructions on a number of measures.

STRUCTURE OF VISUALIZATIONS

Route maps and assembly instructions belong to a larger class of visualizations that are used in many contexts and for many purposes, including practical, educational, and aesthetic. The design principles developed for route maps and assembly instructions have broader applications, to other visualizations, as well as providing guidelines for establishing design principles in other visualization domains. In particular, many biological, physical, and conceptual systems have parts in a spatial, temporal, or abstract structure. The design principles developed for routes and assembly suggest that visualizations of these systems should clearly delineate the parts and their relations. Many of the systems have action or change; the design principles developed for assembly suggest that the changes can be conveyed through the use of extra-pictorial features, especially arrows and guidelines.

The program for generating more specific design principles generalizes to other domains. The program entails first revealing the cognitive structures underlying understanding. Asking knowledgeable participants to produce descriptions and visualizations of the systems is one way to reveal cognitive structures. The structure common to both descriptions and depictions represents the cognitive structure underlying understanding. The descriptions and depictions also suggest verbal and pictorial devices for conveying the concepts. Together, these can be used to develop cognitive design principles. The second step is to instantiate the design principles into algorithms that generate visualizations for a large class of instances. Then the effectiveness of the design principles is tested by comparing learning with the computer-generated visualizations to learning with standard visualizations. This program is applicable to myriad visualizations.

Visualizations use space to convey meanings that are spatial or metaphorically spatial. For the most part, proximity in graphic space is used to convey proximity in real or metaphoric space, preserving information at the level of category, order, interval, or ratio.

Icons and Figures of Depiction

Visualizations use elements as well as space to convey meaning. The simplest and most direct kind of element is an icon, where the element bears resemblance to the thing it represents. These are as old as ideographic languages, where schematic animals and edibles represented their real-world counterparts, and as new as the latest computer or Olympics icons. But many useful concepts cannot be readily depicted. Figures of depiction have been spontaneously adopted, again, since ancient times. Synechdoche, where a part represents a whole, is common, as in the horns or head of a sheep to stand for sheep. Similarly, metonomy, where an entity associated with a concept stands for the concept, as in a crown for a king, or scales for justice, or scissors for delete. The same devices, of course, appear in figures of speech. Icons that are related to the things they represent by figures of depiction also appeared in ancient scripts and appear in contemporary machinery. The advantage of icons and figures of depiction is that their meanings are readily understood and remembered.

Morphograms

There is another kind of element that is prevalent across a wide range of graphics and that is readily understood in context (Tversky, 2003). Lines, crosses, arrows, and blobs are simple, schematic geometric figures that are an integral component of many kinds of graphics, maps, graphs, and mechanical diagrams for examples. Their meanings are related to the their geometric or Gestalt properties, but are context-dependent. Lines, for example, connect, they serve as paths from one point to another, suggesting a relationship between the points. Crosses are intersections of lines. And arrows are asymmetric lines, suggesting an asymmetric relationship. Blobs are two-dimensional, suggesting an area. Their amorphous shape suggests that shape is irrelevant. Like words in language, morphograms can be combined in various ways to create varying meanings. Like words in language, there are constraints on how they can be combined. Finally, like words such as *relation, intersection,* and *field*—words corresponding to lines, crosses, and blobs, morphograms are rich in possible senses, which context specifies.

Visualizations as Cognitive Tools

Visualizations are a cognitive tool, designed to augment human cognitive capacities (Tversky, 2001). They aid memory by off-loading it from limited working memory or fallible long-term memory. By off-loading, visualizations relieve working memory of its memory functions, leaving more capacity for processing. Visualizations, for example, doing calculations on paper, can also augment information processing. Visualizations can be explored and altered, promoting inference and insight. They are public, and can be examined and revised collectively, insuring common ground.

Our work illustrates one important and practical function of visualizations: to convey information. Properly designed visualizations can serve people in numerous ways, from helping them to find their destinations, to clarifying how to perform procedures, to informing them how systems work. Proper design means design compatible with human perception and cognition. It is prohibitively expensive to design effective visualizations for each and every person and each and every need. Creating effective visualizations is an ideal task for computers. Computers, then, need to be educated in cognitive design principles. We have presented a program for doing exactly that, along with two successful cases, route maps, and assembly instructions.

ACKNOWLEDGMENTS

We are grateful for the encouragement and support of the following grants: Office of Naval Research, Grants Number NOOO14-PP-1-O649, N000140110717, and N000140210534 to Stanford University.

REFERENCES

Agrawala, M. (2002). *Visualizing route maps*. Doctoral dissertation, Stanford University, Stanford, CA.

Agrawala, M., & Stolte, C. (2001). Rendering effective route maps: Improving usability through generalization. *Proceedings of SIGGRAPH 2001,* 241–250.

Agrawala, M., Phan, D., Heiser, J., Haymaker, J., Klingner, J., Hanrahan, P., & Tversky, B. (2003). Designing effective step-by-step assembly instructions. *Proceedings of. SIGGRAPH 2003,* 828–837.

Beniger, J. R., & Robyn, D. L. (1978). Quantitative graphics in statistics. *The American Statistician, 32,* 1–11.

Brown, L. (1979). *The story of maps*. New York: Dover.

Heiser, J., Phan, D., Agrawala, M., Tversky, B., & Hanrahan, P. (2004). Identification and validation of cognitive design principles for automated generation of assembly instructions. *Proceedings of Advanced Visual Interfaces 2004,* 311–319.

Heiser, J., & Tuersky, B. (in preparation). Designing effective assembly diagrams.

Mijksenaar, P., & Westendorp, P. (1999). *Open here: The art of instructional design.* London: Thames and Hudson.

Money, J., & Alexander, D. (1966). Turner's syndrome: Further demonstrations of the presence of specific cognitional deficiencies. *Journal of Medical Genetics*, *3*, 47–48.

Romney, B., Godard, C., Goldwasser, M., & Ramkumar, G. (1995). An efficient system for geometric assembly sequence generation and evaluation. *Proceedings of the ASME International Computers in Engineering Conference,* 699–712.

Tversky, B. (1981). Distortions in memory for maps. *Cognitive Psychology, 13,* 407–433.

Tversky, B. (2001). Spatial schemas in depictions. In M. Gattis (Ed.), *Spatial schemas and abstract thought* (pp. 79–111). Cambridge, MA: MIT Press.

Tversky, B. (2003). Some ways graphics communicate. In K. Nyiri (Ed.), *Mobile communication: Essays on cognition and community* (pp. 143–156). Wien: Passagen Verlag.

Tversky, B. (2005). Functional significance of visuospatial representations. In P. Shah & A. Miyake (Eds.), *Handbook of higher-level visuospatial thinking* (pp. 1–34). Cambridge, UK: Cambridge University Press.

Tversky, B., & Hemenway, K. (1984). Objects, parts, and categories. *Journal of Experimental Psychology: General, 113,* 169–193.

Tversky, B., & Lee, P. U. (1998). How space structures language. In C. Freksa, C. Habel, & K. F. Wender (Eds.), *Spatial cognition: An interdisciplinary approach to representation and processing of spatial knowledge* (pp. 157–175). Berlin: Springer-Verlag.

Tversky, B., Morrison, J. B., & Betrancourt, M. (2002). Animation: Can it facilitate? *International Journal of Human Computer Systems. 57,* 247–262.

Tversky, B., Zacks, J. M., & Lee, P. (2004). Events by hand and feet. *Spatial Cognition and Computation, 4,* 5–14.

Tversky, B., Zacks, J., Lee, P. U., & Heiser, J. (2000). Lines, blobs, crosses, and arrows: Diagrammatic communication with schematic figures. In M. Anderson, P. Cheng, & V. Haarslev (Eds.), *Theory and application of diagrams* (pp. 221–230). Berlin: Springer-Verlag.

Vandenberg, S. G., & Kuse, A. R. (1978). Mental rotations: A group test of three-dimensional spatial visualization. *Perceptual Motor Skills, 47,* 599–604.

Wilson, R. (1992). *On geometric assembly planning*. doctoral dissertation, Stanford University, Stanford, CA.

Zacks, J., Tversky, B., & Iyer, G. (2001). Perceiving, remembering and communicating structure in events. *Journal of Experimental Psychology: General, 136,* 29–58.