Chapter 6

Rules of Thumb

6.1 The Big Picture

This chapter contains **rules of thumb**: advice and guidelines. Each of them has a catchy title in hopes that you'll remember it as a slogan. Figure 6.1 lists these eight rules of thumb.

6.2 Why and When to Follow Rules of Thumb?

These rules of thumb are my current attempt to synthesize the current state of knowledge into a more unified whole. In some cases I refer to empirical studies, in others I make arguments based on my own experience, and some have been proposed in previous work. They are not set in stone; indeed, they are deeply incomplete. The characterization of what idioms are appropriate for which task and data abstractions is still an ongoing research frontier, and there are many open questions.

6.3 No Unjustified 3D

Many people have the intuition that if two dimensions are good, three dimensions must be better—after all, we live in a three-dimensional world. However, there are many difficulties in visually encoding information with the third spatial dimension, depth, which has important differences from the two planar dimensions.

In brief, 3D vis is easy to justify when the user's task involves shape understanding of inherently three-dimensional structures. In this case, which frequently occurs with inherently spatial data, the benefits of 3D absolutely outweigh the costs, and designers can use the many interaction idioms designed to mitigate those costs.

In all other contexts, the use of 3D needs to be carefully justified. In most cases, rather than choosing a visual encoding using three dimensions of spatial position, a better answer is to visually encode using only two dimensions of spatial position. Often an appropriate 2D encoding follows from a different choice of data abstraction, where the original dataset is transformed by computing derived data.

The cues that convey depth information to our visual system include occlusion, perspective distortion, shadows and lighting, familiar size, stereoscopic disparity, and others. This section discusses the costs of these **depth cues** in a visual encoding context and the challenges of text legibility given current display technology. It then discusses situations where the benefits of showing depth information could outweigh these costs and the need for justification that the situation has been correctly analyzed.

6.3.1 The Power of the Plane

A crucial point when interpreting the channel rankings in Figure 5.6 is that the spatial position channels apply only to planar spatial position, not arbitrary 3D position.

Vertical and horizontal position are combined into the shared category of *planar* because the differences between the up–down and side-to-side axes are relatively subtle. We do perceive height differences along the up–down axis as more important than horizontal position differences, no doubt due to the physical effects of gravity in real life. While the vertical spatial channel thus has a slight priority over the horizontal one, the aspect ratio of standard displays gives more horizontal pixels than vertical ones, so information density considerations sometimes override this concern. For the perceived importance of items ordered within the axes, reading conventions probably dominate. Most Western languages go from left to right and from top to bottom, but Arabic and Hebrew are read from right to left, and some Asian languages are read vertically.

6.3.2 The Disparity of Depth

The psychophysical power law exponents for accuracy shown in Figure 5.7 are different for depth position judgements in 3D than for planar position judgements in 2D. Our highly accurate length perception capability, with the linear n value of 1.0, only holds for planar spatial position. For depth judgements of visual distance, n was measured as 0.67 [Stevens 57]; that exponent is even worse than the value of 0.7 for area judgements. This phenomenon is

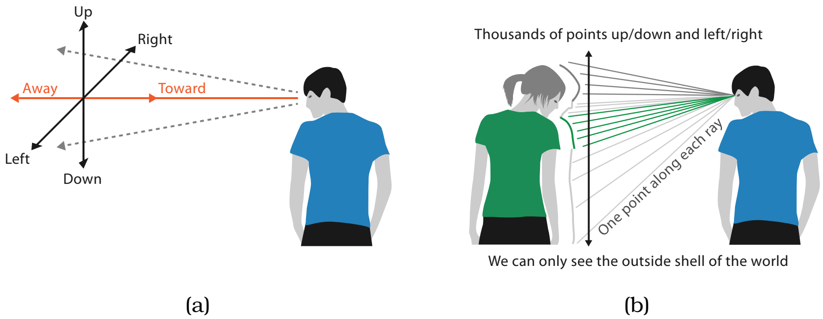

Up
Right
Away
Toward
Left
Down
Thousands of points up/down and left/right
One point along each ray
We can only see the outside shell of the world
(a)
(b)

Figure 6.2. Seeing planar position versus depth. (a) The sideways and up–down axes are fundamentally different from the toward–away depth axis. (b) Along the depth axis we can see only one point for each ray, as opposed to millions of rays for the other two axes. After [Ware 08, page 44].

not surprising when considered mathematically, because as shown in Figure 6.2 the length of a line that extends into the scene is scaled nonlinearly in depth, whereas a line that traverses the picture plane horizontally or vertically is scaled linearly, so distances and angles are distorted [St. John et al. 01].

Considered perceptually, the inaccuracy of depth judgements is also not surprising; the common intuition that we experience the world in 3D is misleading. We do not really live in 3D, or even 2.5D: to quote Colin Ware, we *see* in 2.05D [Ware 08]. That is, most of the visual information that we have is about a two-dimensional *image plane*, as defined below, whereas the information that we have about a third depth dimension is only a tiny additional fraction beyond it. The number of 0.05 is chosen somewhat arbitrarily to represent this tiny fraction.

Consider what we see when we look out at the world along a ray from some fixed viewpoint, as in Figure 6.2(a). There is a major difference between the toward–away depth axis and the other two axes, sideways and up–down. There are millions of rays that we can see along these two axes by simply moving our eyes, to get information about the nearest opaque object. This information is like a two-dimensional picture, often called the **image plane**. In contrast, we can only get information at one point along the depth axis for each ray away from us toward the world, as in Figure 6.2(b). This phenomenon is called **line-of-sight ambiguity** [St. John et al. 01]. In order to get more information about what is hidden behind the closest objects shown in the image plane, we would need to move our viewpoint or the objects. At best we could change the viewpoint by simply moving our head, but in many cases we would need to move our body to a very different position.

6.3.3 Occlusion Hides Information

The most powerful depth cue is **occlusion**, where some objects cannot be seen because they are hidden behind others. The visible objects are interpreted as being closer than the occluded ones. The occlusion relationships between objects change as we move around; this **motion parallax** allows us to build up an understanding of the relative distances between objects in the world.

When people look at realistic scenes made from familiar objects, the use of motion parallax typically does not impose cognitive load or require conscious attention. In synthetic scenes, navigation controls that allow the user to change the 3D viewpoint interactively invoke the same perceptual mechanisms to provide motion parallax. In sufficiently complex scenes where a single fixed viewpoint does not provide enough information about scene structure, interactive navigation capability is critical for understanding 3D structure. In this case, the cost is time: interactive navigation takes longer than inspecting a single image.

The overarching problem with occlusion in the context of visual encoding is that presumably important information is hidden, and discovering it via navigation has a time cost. In realistic environments, there is rarely a need to inspect all hidden surfaces. However, in a vis context, the occluded detail might be critical. It is especially likely to be important when using spatial position as a visual channel for abstract, nonspatial data.

Moreover, if the objects have unpredictable and unfamiliar shapes, understanding the three-dimensional structure of the scene can be very challenging. In this case there can be appreciable cognitive load because people must use internal memory to remember the shape from previous viewpoints, and internally synthesize an understanding of the structure. This case is common when using the spatial position channels for visual encoding. Figure 6.3 illustrates the challenges of understanding the topological structure of a node–link graph laid out in 3D, as an example of the unfamiliar structure that arises from visually encoding an abstract dataset. Synthesizing an understanding of the structure of the linkages hidden from the starting viewpoint shown here is likely to take a considerable amount of time. While sophisticated interaction idioms have been proposed to help users do this synthesis more quickly

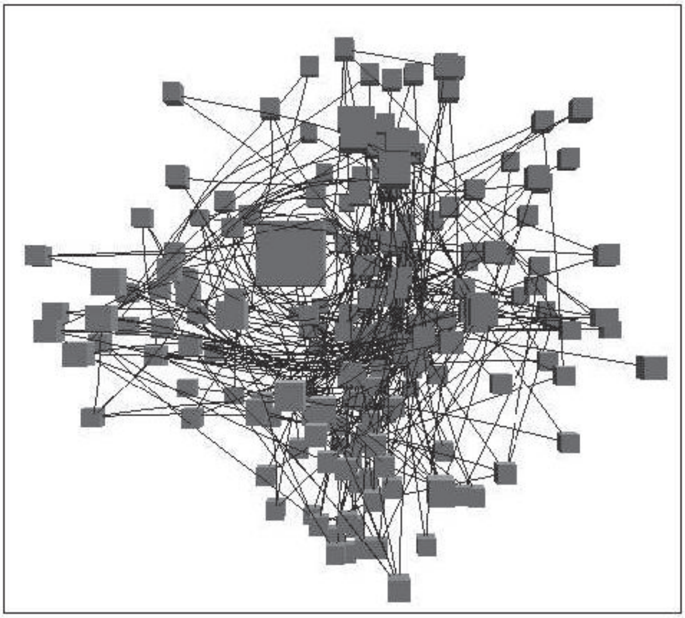


Figure 6.3. Resolving the 3D structure of the occluded parts of the scene is possible with interactive navigation, but that takes time and imposes cognitive load, even when sophisticated interaction idioms are used, as in this example of a node–link graph laid out in 3D space. From [Carpendale et al. 96, Figure 21].

than with simple realistic navigation, thus lowering the time cost, vis designers should always consider whether the benefits of 3D are worth the costs.

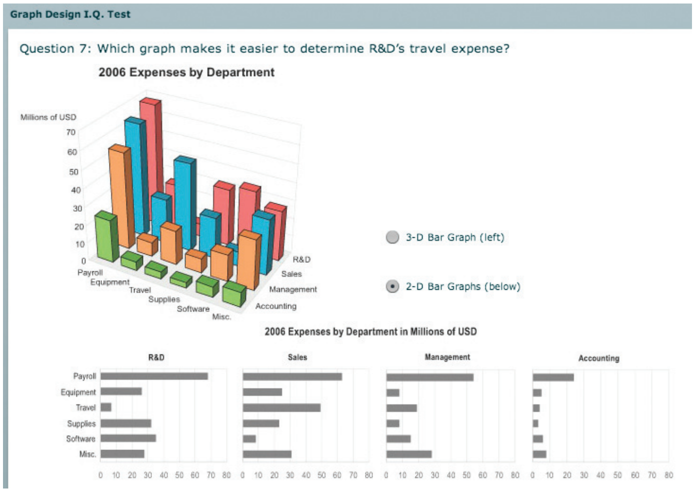
6.3.4 Perspective Distortion Dangers

The phenomenon of **perspective distortion** is that distant objects appear smaller and change their planar position on the image plane. Imagine a photograph looking along railroad tracks: although they are of course parallel, they appear to draw together as they recede into the distance. Although the tracks have the same width in reality, measuring with a ruler on the photograph itself would show that in the picture the width of the nearby track is much greater than that of the distant track.*

One of the major breakthroughs of Western art was the Renaissance understanding of the mathematics of perspective to create very realistic images, so many people think of perspective as a good

► The disparity in our perception of depth from our perception of planar spatial position is discussed in Section 6.3.2.

★ The phenomenon of *perspective distortion* is also known as **foreshortening**.



Graph Design I.Q. Test

Question 7: Which graph makes it easier to determine R&D's travel expense?

○ 3-D Bar Graph (left)

● 2-D Bar Graphs (below)

Figure 6.4. 3D bar charts are more difficult than 2D bar charts because of both perspective distortion and occlusion. From [Few 07, Question 7].

thing. However, in the context of visually encoding abstract data, perspective is a very bad thing! Perspective distortion is one of the main dangers of depth because the power of the plane is lost; it completely interferes with visual encodings that use the planar spatial position channels and the size channel. For example, it is more difficult to judge bar heights in a 3D bar chart than in multiple horizontally aligned 2D bar charts, as shown in Figure 6.4. Foreshortening makes direct comparison of bar heights difficult.

Figure 6.5 shows another example where size coding in multiple dimensions is used for bars that recede into the distance in 3D on a ground plane. The result of the perspective distortion is that the bar sizes cannot be directly compared as a simple perceptual operation.

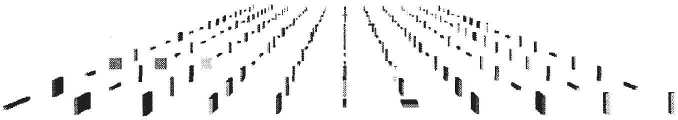


Figure 6.5. With perspective distortion, the power of the planar spatial position channel is lost, as is the size channel. From [Mukherjea et al. 96, Figure 1].

6.3.5 Other Depth Cues

In realistic scenes, one of the depth cues is the size of familiar objects. We roughly know the size of a car, so when we see one at a distance we can estimate the size of a nearby unfamiliar object. If all objects in the scene are visually encoded representations of abstract information, we do not have access to this strong depth cue.

The depth cues of shadows and surface shading also communicate depth and three-dimensional structure information. Cast shadows are useful for resolving depth ambiguity because they allow us to infer the height of an object with respect to a ground plane. Shading and self-shadowing show the three-dimensional shape of an object. One problem with using these lighting-based cues when visualizing abstract data is that they create visual clutter that distracts the viewer's attention from the meaningful parts of the scene that represent information. Another problem is that cast shadows, regions of self-shadowing, or highlights could be mistaken by the viewer for true marks that are the substrate for the visual channels showing attribute information. Cast shadows could also cause problems by occluding true marks. The final problem is that surface shading effects interfere with the color channels: highlights can change the hue or saturation, and shadows change the luminance.

Stereoscopic depth is a cue that comes from the disparities between two images made from two camera viewpoints slightly separated in space, just like our two eyes are. In contrast, all of the previous discussion pertained to pictoral cues from a single camera. Although many people assume that stereo vision is the strongest depth cue, it is in fact a relatively weak one compared with the others listed above and contributes little for distant objects. Stereo depth cues are most useful for nearby objects that are at roughly the same depth, providing guidance for manipulating things within arm's reach.

Stereo displays, which deliver a slightly different image for each of our two eyes, do help people better resolve depth. Conveniently, they do not directly interfere with any of the main visual channels. Stereo displays do indeed improve the accuracy of depth perception compared with single-view displays—but even still depth cannot be perceived with the accuracy of planar position. Of course, stereo cannot solve any of the problems associated with perspective distortion.

The relatively subtle depth cue of atmospheric perspective, where the color of distant objects is shifted toward blue, would conflict with color encoding.

6.3.6 Tilted Text Isn't Legibile

Another problem with the use of 3D is dramatically impaired text legibility with most standard graphics packages that use current display technology [Grossman et al. 07]. Text fonts have been very carefully designed for maximum legibility when rendered on the grid of pixels that makes up a 2D display, so that characters as little as nine pixels high are easily readable. Although hardware graphics acceleration is now nearly pervasive, so that text positioned at arbitrary orientations in 3D space can be rendered *quickly*, this text is usually not rendered *well*. As soon as a text label is tilted in any way off of the image plane, it typically becomes blocky and jaggy. The combination of more careful rendering and very high-resolution displays of many hundred of dots per inch may solve this problem in the future, but legibility is a major problem today.

6.3.7 Benefits of 3D: Shape Perception

The great benefit of using 3D comes when the viewer's task fundamentally requires understanding the three-dimensional geometric structure of objects or scenes. In almost all of these cases, a 3D view with interactive navigation controls to set the 3D viewpoint will allow users to construct a useful mental model of dataset structure more quickly than simply using several 2D axis-aligned views. For these tasks, all of the costs of using 3D discussed above are outweighed by the benefit of helping the viewer build a mental model of the 3D geometry.

For example, although people can be trained to comprehend blueprints with a top view and two side views, synthesizing the information contained within these views to understand what a complex object looks like from some arbitrary 3D viewpoint is a difficult problem that incurs significant cognitive and memory load. The 2D blueprint views are better for the task of accurately discriminating the sizes of building elements, which is why they are still heavily used in construction. However, there is considerable experimental evidence that 3D outperforms 2D for shape understanding tasks [St. John et al. 01].

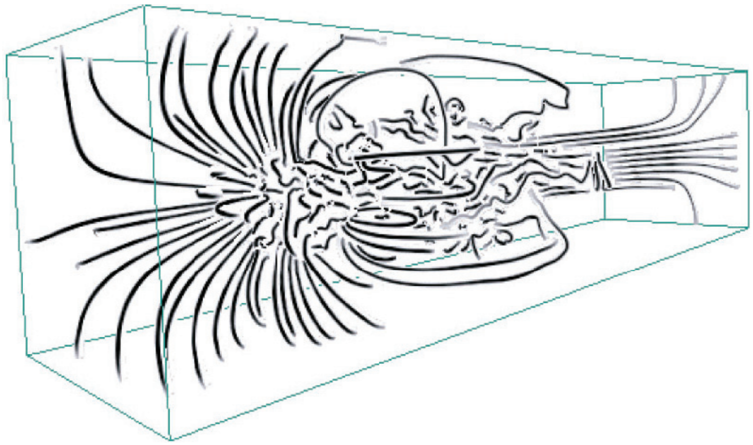


Figure 6.6. The use of 3D is well justified when the central task is shape understanding, as in this example of 3D streamline showing the patterns of fluid flow through a volume. From [Li and Shen 07, Figure 9].

Most tasks that have inherently 3D spatial data after the abstraction stage fall into this category. Some classical examples are fluid flow over an airplane wing, a medical imaging tomography dataset of the human body, or molecular interaction within a living cell. Figure 6.6 shows an example of streamlines in 3D fluid flow [Li and Shen 07], where geometric navigation based on 3D rotation is a good strategy to help users understand the complex shapes quickly.

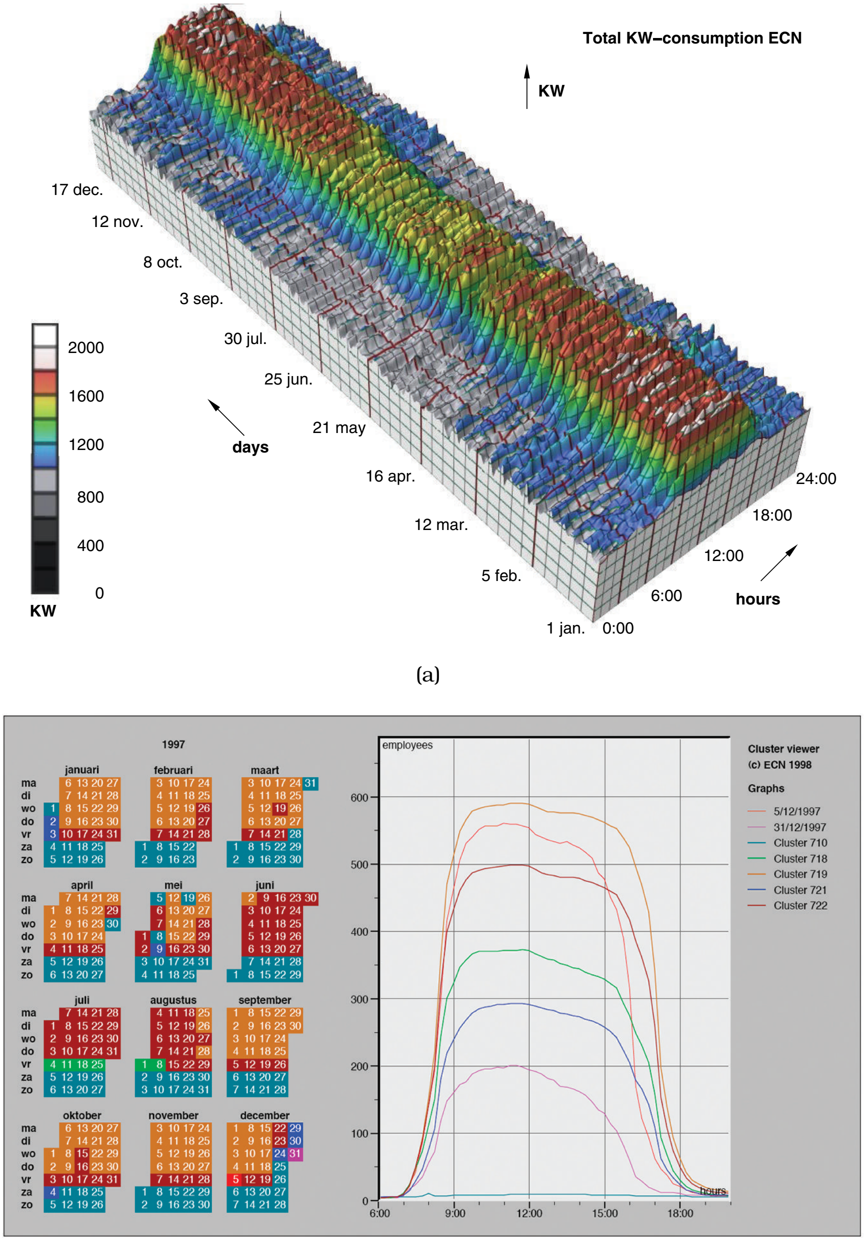
▶ Streamlines are discussed further in Section 8.5, and geometric navigation in Section 11.5.

6.3.8 Justification and Alternatives

The question of whether to use two or three channels for spatial position has now been extensively studied. When computer-based vis began in the late 1980s, there was a lot of enthusiasm for 3D representations. As the field matured, researchers began to better appreciate the costs of 3D approaches when used for abstract datasets [Ware 01]. By now, the use of 3D for abstract data requires careful justification. In many cases, a different choice at the abstraction or visual encoding levels would be more appropriate.

Example: Cluster–Calendar Time-Series Vis

A good example is a system from van Wijk and van Selow designed to browse time-series data [van Wijk and van Selow 99]. The dataset has two



(a)

(b)

Figure 6.7. 3D versus 2D. (a) A 3D representation of this time-series dataset introduces the problems of occlusion and perspective distortion. (b) The linked 2D views of derived aggregate curves and the calendar allow direct comparison and show more fine-grained patterns. From [van Wijk and van Selow 99, Figures 1 and 4].

related sets of measurements: the number of people inside and amount of power used in an office building, with measurements over the course of each day for one full year. The authors compare a straightforward 3D representation with a carefully designed approach using linked 2D views, which avoids the problems of occlusion and perspective distortion. Figure 6.7(a) shows the straightforward 3D representation created directly from the original time-series data, where each cross-section is a 2D time series curve showing power consumption for one day, with one curve for each day of the year along the extruded third axis. Only very large-scale patterns such as the higher consumption during working hours and the seasonal variation between winter and summer are visible.

The final vis designed by the authors uses multiple linked 2D views and a different data abstraction. They created the derived data of a hierarchical clustering of the time-series curves through an iterative process where the most similar curves are merged together into a cluster that can be represented by the average of the curves within it.

▶ Linked views are discussed in Chapter 12.

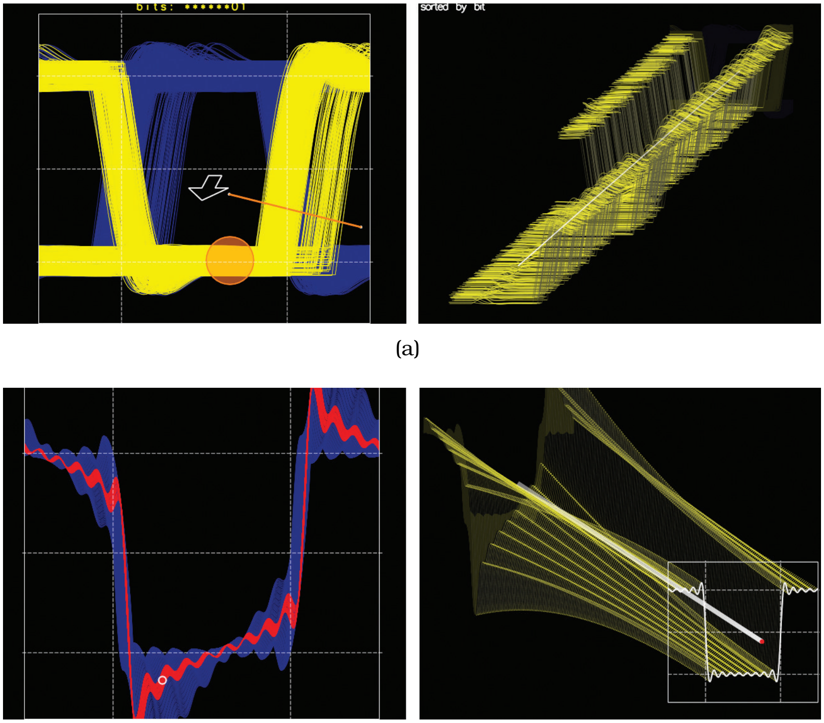
Figure 6.7(b) shows a single aggregate curve for each of the highest-level groups in the clustering in the window on the right of the display. There are few enough of these aggregate curves that they can all be superimposed in the same 2D image without excessive visual clutter. Direct comparison between the curve heights at all times of the day is easy because there is no perspective distortion or occlusion. (The cluster–calendar vis shows the number of people in the building, rather than the power consumption of the 3D extruded vis.)

On the left side of Figure 6.7(b) is a calendar view. Calendars are a very traditional and successful way to show temporal patterns. The views are linked with shared color coding. The same large-scale patterns of seasonal variation between summer and winter that can be seen in 3D are still very visible, but smaller-scale patterns that are difficult or impossible to spot in the 3D view are also revealed. In this Dutch calendar, weeks are vertical strips with the weekend at the bottom. We can identify weekends and holidays as the nearly flat teal curve where nobody is in the building, and normal weekdays as the topmost tan curve with a full house. Summer and Fridays during the winter are the brown curve with one hundred fewer people, and Fridays in the summer are the green curve with nearly half of the employees gone. The blue and magenta curves show days between holiday times where most people also take vacation. The red curve shows the unique Dutch holiday of Santa Claus day, where everybody gets to leave work an hour early.

While unbridled enthusiasm for 3D is no longer common, there are indeed situations where its use is justifiable even for abstract data.

Example: Layer-Oriented Time-Series Vis

Figure 6.8 shows an example that is similar on the surface to the previous one, but in this case 3D is used with care and the design is well justified [Lopez-Hernandez et al. 10]. In this system for visualizing oscilloscope time-series data, the user starts by viewing the data using the traditional eye diagram where the signal is wrapped around in time and shown as many overlapping traces. Users can spread the traces apart using the metaphor of opening a drawer, as shown in Figure 6.8(a). This drawer interface does use 3D, but with many constraints. Layers are orthographically projected and always face the viewer. Navigation complexity is controlled by automatically zooming and framing as the user adjusts the drawer's orientation, as shown in Figure 6.8(b).



(a)

(b)

Figure 6.8. Careful use of 3D. (a) The user can evolve the view from the traditional overlapping eye diagram with the metaphor of opening a drawer. (b) The interaction is carefully designed to avoid the difficulties of unconstrained 3D navigation. From [Lopez-Hernandez et al. 10, Figures 3 and 7].

6.3.9 Empirical Evidence

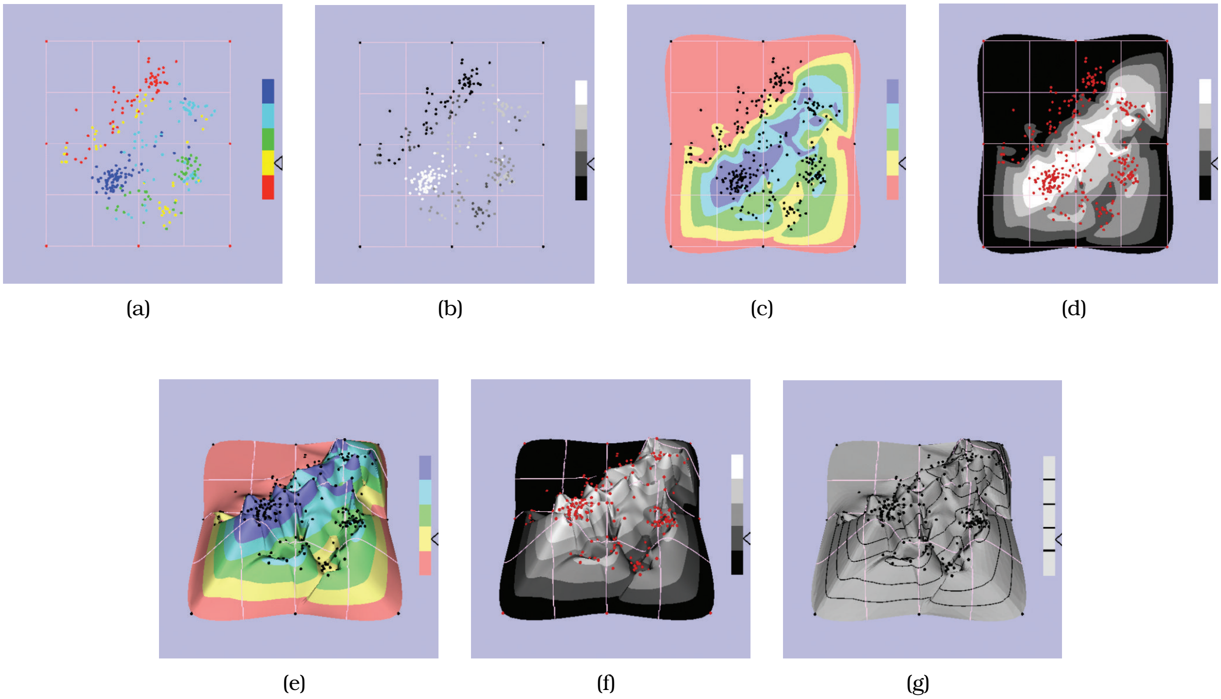
Empirical experiments are critical in understanding user performance, especially because of the well-documented dissociation between stated preference for 3D and actual task performance [Andre and Wickens 95]. Experimental evidence suggests that 3D interfaces are better for shape understanding, whereas 2D are best for relative position tasks: those that require judging the precise distances and angles between objects [St. John et al. 01]. Most tasks involving abstract data do not benefit from 3D; for example, an experiment comparing 3D cone trees to an equivalent 2D tree browser found that the 3D interaction had a significant time cost [Cockburn and McKenzie 00].

Designing controlled experiments that untangle the efficacy of specific interfaces that use 3D can be tricky. Sometimes the goal of the experimenter is simply to compare two alternative interfaces that differ in many ways; in such cases it is dangerous to conclude that if an interface that happens to be 3D outperforms another that happens to be 2D, it is the use of 3D that made the difference. In several cases, earlier study results that were interpreted as showing benefits for 3D were superseded by more careful experimental design that eliminated uncontrolled factors. For example, the 3D Data Mountain interface for organizing web page thumbnail images was designed to exploit human spatial cognition and was shown to outperform the standard 2D Favorites display in Internet Explorer [Robertson et al. 98]. However, this study left open the question of whether the benefit was from the use of 3D or the use of the data mountain visual encoding, namely, a spatial layout allowing immediate access to every item in each pile of information. A later study compared two versions of Data Mountain, one with 3D perspective and one in 2D, and no performance benefit for 3D was found [Cockburn and McKenzie 01].

Another empirical study found no benefits for 3D landscapes created to reflect the density of a 2D point cloud, compared with simply showing the point cloud in 2D [Tory et al. 07]. In the 3D **information landscape** idiom, the density of the points on the plane is computed as a derived attribute and used to construct a surface whose height varies according to this attribute in order to show its value in a form similar to geographic terrain.★ A third alternative to landscapes or points is a **contour plot**, where colored bands show the outlines of specific heights. A contour plot can be used alone as a 2D landscape or can be combined with 3D for a colored landscape. Proponents of this idiom have argued that landscapes

★ Other names for a 3D *landscape* are **height field** and **terrain**.

► Contour plots are discussed in Section 8.4.1.


(a) (b) (c) (d)

(e) (f) (g)

Figure 6.9. Point-based displays were found to outperform information landscapes in an empirical study of visual encodings for dimensionally reduced data. (a) Colored points. (b) Grayscale points. (c) Colored 2D landscape. (d) Grayscale 2D landscape. (e) Colored 3D landscape. (f) Grayscale 3D landscape. (g) Height only. From [Tory et al. 07, Figure 1].

▶ Dimensionality reduction is discussed in Section 13.4.3.

are familiar and engaging, and they have been used in several systems for displaying high-dimensional data after dimensionality reduction was used to reduce to two synthetic dimensions [Davidson et al. 01, Wise et al. 95].

Figure 6.9 shows the seven possibilities tested in the empirical study comparing points to colored and uncolored landscapes. The findings were that points were far superior to landscapes for search and point estimation tasks and in the landscape case 2D landscapes were superior to 3D landscapes [Tory et al. 07]. A follow-up study for a visual memory task yielded similar results [Tory et al. 09].

6.4 No Unjustified 2D

Laying out data in 2D space should also be explicitly justified, compared with the alternative of simply showing the data with a 1D list.

Lists have several strengths. First, they can show the maximal amount of information, such as text labels, in minimal space. In contrast, 2D layouts such as node–link representations of network data require considerably more space to show the same number of labels, so they have notably lower information density.

Second, lists are excellent for lookup tasks when they are ordered appropriately, for example in alphabetical order when the goal is to find a known label. In contrast, finding a specific label in a 2D node–link representation might require the user to hunt around the entire layout, unless a specific search capability is built into the vis tool.

When the task truly requires understanding the topological structure of the network, then the benefits of showing those relationships explicitly outweigh the cost of the space required. However, some tasks are handled well by linear lists, even if the original data has network structure.

6.5 Eyes Beat Memory

Using our eyes to switch between different views that are visible simultaneously has much lower cognitive load than consulting our memory to compare a current view with what was seen before. Many interaction idioms implicitly rely on the internal use of memory and thus impose cognitive load on the viewer. Consider navigation within a single view, where the display changes to show the scene from a different viewpoint. Maintaining a sense of orientation implicitly relies on using internal resources, either by keeping track of past navigation choices (for example, *I zoomed into the nucleus*) or by remembering past views (for example, *earlier all the stock options in the tech sector were in the top corner of the view*). In contrast, having a small overview window with a rectangle within it showing the position and size of the current camera viewport for the main view is a way to show that information through an external representation easily consulted by looking at that region of the screen, so that it can be read off by the perceptual system instead of remembered.

6.5.1 Memory and Attention

Broadly speaking, people have two different categories of memory: long-term memory that can last a lifetime, versus short-term memory that lasts several seconds, also known as **working memory**. While the capacity of long-term memory doesn't have a strict upper limit, human working memory is a very limited resource. When these limits are reached, people experience **cognitive load** and will fail to absorb all of the information that is presented.

Human attention also has severe limits. Conscious search for items is an operation that grows more difficult with the number of items there are to be checked. Vigilance is also a highly limited resource: our ability to perform visual search tasks degrades quickly, with far worse results after several hours than in the first few minutes.

6.5.2 Animation versus Side-by-Side Views

Some animation-based idioms also impose significant cognitive load on the viewer because of implicit memory demands. Animation is an overloaded word that can mean many different things considered through the lens of vis encoding and interaction. I distinguish between these three definitions:

- narrative storytelling, as in popular movies;
- transitions from just one state to another;
- video-style playback of a multiframe sequence: play, pause, stop, rewind, and step forward/back.

Some people have the intuition that because animation is a powerful storytelling medium for popular movies, it should also be suitable in a vis context. However, the situation is quite different. Successful storytelling requires careful and deliberate choreography to ensure that action is only occurring in one place at a time and the viewer's eyes have been guided to ensure that they are looking in the right place. In contrast, a dataset animation might have simultaneous changes in many parts of the view.

Animation is extremely powerful when used for transitions between two dataset configurations because it helps the user maintain context. There is considerable evidence that animated transitions can be more effective than jump cuts, because they help people track changes in object positions or camera viewpoints. These transitions are most useful when only a few things change; if the number of objects that change between frames is large, people will have a hard time tracking everything that occurs. We are blind to changes in regions of the image that are not the focus of our attention.

▶ Change blindness is covered in Section 6.5.3.

Although jump cuts are hard to follow when only seen once, giving the user control of jumping back and forth between just two frames can be effective for detecting whether there is a localized change between two scenes. This *blink comparator* idiom was used by the astronomer who found Pluto.

Finally, I consider animations as sequences of many frames, where the viewer can control the playback using video-style controls of play, pause, stop, rewind, and sometimes single-step forward or backward frame by frame. I distinguish animation from true interactive control, for example, navigation by flying through a scene. With animation the user does not directly control what occurs, only the speed at which the animation is played.

The difficulty of multiframe animations is that making comparisons between frames that do not adjoin relies on internal memory of what previous frames looked like. If changes only occur in one place at a time, the demands on attention and internal memory are small. However, when many things change all over the frame and there are many frames, we have a very difficult time in tracking what happens. Giving people the ability to pause and replay the animation is much better than only seeing it a single time straight through, but that control does not fully solve the problem.

For tasks requiring detailed comparison across many frames, seeing all the frames at once side by side can be more effective than animation. The number of frames must be small enough that the details within each can be discerned, so this approach is typically suitable for dozens but not hundreds of frames with current display resolutions. The action also should be segmented into meaningful chunks, rather than keyframes that are randomly chosen. Many vis idioms that use multiple views exploit this observation, especially small multiples.

▶ Small multiples are covered in Section 12.3.2.

6.5.3 Change Blindness

The human visual system works by querying the world around us using our eyes. Our visual system works so well that most people have the intuition that we have detailed internal memory of the visual field that surrounds us. However, we do not. Our eyes dart around, gathering information just in time for our need to use it, so quickly that we do not typically notice this motion at a conscious level.

The phenomenon of **change blindness** is that we fail to notice even quite drastic changes if our attention is directed elsewhere. For example, experimenters set up a real-world interaction where somebody was engaged by a stranger who asked directions, only to be interrupted by people carrying a door who barged in between them. The experimenters orchestrated a switch during this visual interruption, replacing the questioner with another person. Remarkably, most people did not notice, even when the new questioner was dressed completely differently—or was a different gender than the old one!

Although we are very sensitive to changes at the focus of our attention, we are surprisingly blind to changes when our attention is not engaged. The difficulty of tracking complex and widespread changes across multiframe animations is one of the implications of change blindness for vis.

6.6 Resolution over Immersion

Pixels are precious: if you are faced with a trade-off between resolution and immersion, resolution usually is far more important.

Immersive environments emphasize simulating realistic interaction and perception as closely as possible through technology such as stereo imagery delivered separately to each eye to enhance depth perception, and full six-degree-of-freedom head and position tracking so that the displays respond immediately to the user's physical motion of walking and moving the head around. The most common display technology is head-mounted displays, or small rooms with rear-projection displays on walls, floor, and ceilings. Immersion is most useful when a sense of presence is an important aspect of the intended task. With current display hardware, there is a trade-off between **resolution**, the number of available pixels divided by the display area, and **immersion**, the feeling of presence in virtual reality. The price of immersion is resolution; these displays cannot show as many pixels as state-of-the-art desktop displays of the equivalent area. The number of pixels available on a computer display is a limited resource that is usually the most critical constraint in vis design. Thus, it is extremely rare that immersion is worth the cost in resolution.

► Display resolution constraints are discussed in Section 1.13.

Another price of immersion is the integration of vis with the rest of a user's typical computer-based workflow. Immersive display environments are almost always special-purpose settings that are a different physical location than the user's workspace, requiring them to leave their usual office and go to some other location, whether down the hall or in another building. In most cases users stand rather than sit, so working for extended periods of time is physically taxing compared with sitting at a desk. The most critical problem is that they do not have access to their standard working environment of their own computer system. Without access to the usual input devices of mouse and keyboard, standard applications such as web browsing, email reading, text and spreadsheet editing, and other data analysis packages are completely unusable in most cases, and very awkward at best. In contrast, a vis system that fits into a standard desktop environment allows integration with the usual workflow and fast task switching between vis and other applications.

A compelling example of immersion is the use of virtual reality for phobia desensitization; somebody with a fear of heights would need a sense of presence in the synthetic environment in order to make progress. However, this example is not an application of vis, since the goal is to simulate reality rather than to visually encode information. The most likely case where immersion would be helpful for vis is when the chosen abstraction includes 3D spatial data. Even in this case, the designer should consider whether a sense of presence is worth the penalties of lower resolution and no workflow integration. It is very rare that immersion would be necessary for nonspatial, abstract data. Using 3D for visual encoding of abstract data is the uncommon case that needs careful justification. The use of an immersive display in this case would require even more careful justification.

▶ The need for justifying 3D for abstract data is covered in Section 6.3.

6.7 Overview First, Zoom and Filter, Details on Demand

Ben Shneiderman's influential mantra of **Overview First, Zoom and Filter, Details on Demand** [Shneiderman 96] is a heavily cited design guideline that emphasizes the interplay between the need for overview and the need to see details, and the role of data reduction in general and navigation in particular in supporting both.

A vis idiom that provides an **overview** is intended to give the user a broad awareness of the entire information space. Using the language of the what–why–how analysis framework, it's an idiom with the goal of *summarize*. A common goal in overview design is to show all items in the dataset simultaneously, without any need for navigation to pan or scroll. Overviews help the user find regions where further investigation in more detail might be productive. Overviews are often shown at the beginning of the exploration process, to guide users in choosing where to drill down to inspect in more detail. However, overview usage is not limited to initial reconnaissance; it's very common for users to interleave the use of overviews and detail views by switching back and forth between them many times.

When the dataset is sufficiently large, some form of *reduce* action must be used in order to show everything at once. Overview creation can be understood in terms of both filtering and aggregation. A simple way to create overviews is by zooming out geometrically, so that the entire dataset is visible within the frame. Each object is drawn smaller, with less room to show detail. In this sense, overviews are created by removing all filtering: an overview is created by changing from a zoomed-in view where some items are filtered out, to a zoomed-out view where all items are shown. When the number of items in a dataset is large enough, showing an overview of the entire dataset in a single screen using one mark per item is impossible, even if the mark size is decreased to a single pixel. When the number of items to draw outstrips the number of available pixels, the number of marks to show must be reduced with aggregation. Moreover, even for datasets of medium size, explicitly designing an overview display using a more sophisticated approach than simple geometric zooming can be fruitful. These custom overviews are similar in spirit to semantic zooming, in that the representation of the items is qualitatively different rather than simply being drawn smaller than the full-detail versions. These kinds of overviews often use dynamic aggregation that is implicitly driven by navigation, rather than being explicitly chosen by the user.

► Aggregation is discussed in Section 13.4.

► Geometric and semantic zooming are discussed in Section 11.5.

There is no crisp line dividing an "overview" from an "ordinary" vis idiom, because many idioms provide some form of overview or summary. However, it's often useful to make a relative distinction between a less detailed view that summarizes a lot of data and a more detailed view that shows a smaller number of data items with more information about each one. The former one is clearly the *overview*, the latter one is the *detail view*. It's particularly obvious how to distinguish between these when the idiom design choice of multiple views is being used; the mantra is particularly applicable when the detail view pops up in response to a *select* action by the user, but it's also common for the detail view to be permanently visible side by side with the overview. There two other major families of idioms that support overviewing. One is to use a single view that dynamically changes over time by providing support for *reduce* actions such as zooming and filtering; then that single view sometimes acts as an overview and sometimes as a detail view. The third choice is to embed both detailed focus and overview context information together within a single view.

▶ Chapter 12 covers multiple views.

▶ Chapter 13 covers approaches to data reduction.

▶ Chapter 14 covers focus+context idioms.

This mantra is most helpful when dealing with datasets of moderate size. When dealing with enormous datasets, creating a useful overview for top-down exploration may not be feasible. In slogan form, an alternative approach is **Search, Show Context, Expand on Demand** [van Ham and Perer 09], where search results provide the starting point for browsing of local neighborhoods.

6.8 Responsiveness Is Required

The **latency** of interaction, namely, how much time it takes for the system to respond to the user action, matters immensely for interaction design. Our reaction to latency does not simply occur on a continuum, where our irritation level gradually rises as things take longer and longer. Human reaction to phenomena is best modeled in terms of a series of discrete categories, with a different time constant associated with each one. A system will feel responsive if these latency classes are taken into account by providing feedback to the user within the relevant time scale. The three categories most relevant for vis designers are shown in Table 6.1.

The perceptual processing time constant of one-tenth of a second is relevant for operations such as screen updates. The immediate response time constant of one second is relevant for operations such as visual feedback showing what item that user has selected with a mouse click, or the length of time for an animated transition from one layout to another. The brief task time constant of ten

Time Constant	Value (in seconds)
perceptual processing	0.1
immediate response	1
brief tasks	10

Table 6.1. Human response to interaction latency changes dramatically at these time thresholds. After [Card et al. 91, Table 3].

seconds is relevant for breaking down complex tasks into simpler pieces; a good granularity for the smallest pieces is this brief task time.

6.8.1 Visual Feedback

From the user's point of view, the latency of an interaction is the time between their action and some feedback from the system indicating that the operation has completed. In a vis system, that feedback would most naturally be some visual indication of state change within the system itself, rather than cumbersome approaches such as printing out status indications at the console or a popup dialog box confirmation that would interfere with the flow of exploration.

The most obvious principle is that the user should indeed have some sort of confirmation that the action has completed, rather than being left dangling wondering whether the action is still in progress, or whether the action never started in the first place (for example, because they missed the target and clicked on the background rather than the intended object). Thus, feedback such as highlighting a selected item is a good way to confirm that the desired operation has completed successfully. In navigation, feedback would naturally come when the user sees the new frame is drawn from the changed viewpoint. Visual feedback should typically take place within the immediate response latency class: around one second.

Another principle is that if an action could take significantly longer than a user would naturally expect, some kind of progress indicator should be shown to the user. A good rule of thumb for significantly longer is crossing from one latency class into another, as shown in Table 6.1.

6.8.2 Latency and Interaction Design

Successful interaction design for a vis system depends on having a good match between the latencies of the low-level interaction mechanism, the visual feedback mechanism, the system update time, and the cognitive load of operation itself.

For example, consider the operation of seeing more details for an item and the latency difference between three different low-level interaction mechanisms for doing so. Clicking on the item is slowest, because the user must move the mouse toward the target location, stop the motion in the right place, and press down on the mouse. Mouseover hover, where the cursor is placed over the object for some short period of dwell time but no click is required, may or may not be faster depending on the dwell time. Mouseover actions with no dwell time requirement, where the action is triggered by the cursor simply crossing the object, are of course the fastest because the second step is also eliminated and only the first step needs to take place.

For visual feedback, consider three different mechanisms for showing the information. One is showing the information on a fixed detail pane at the side of the screen. In order to see the information, the user's eyes need to move from the current cursor location to the side of the screen, so this operation has relatively high latency for making use of the visual feedback. On the other hand, from a visual encoding point of view, an advantage is that a lot of detail information can be shown without occluding anything else in the main display. A second feedback mechanism is a popup window at the current cursor location, which is faster to use since there is no need to move the eyes away from tracking the cursor. Since placing information directly in the view might occlude other objects, there is a visual encoding cost to this choice. A third mechanism is a visual highlight change directly in the view, for instance by highlighting all neighbors within the graph that are one hop from the graph node under the cursor through a color change.

System update time is another latency to consider. With tiny datasets stored completely locally, update time will be negligible for any of these options. With larger datasets, the time to redraw the entire view could be considerable unless the rendering★ framework has been designed to deliver frames at a guaranteed rate. Similarly, scalable rendering frameworks can support fast update for changing a few items or a small part of the display without redrawing the entire screen, but most graphics systems do not offer this functionality by default. Thus, designing systems to guarantee immediate response to user actions can require significant algorithmic attention. With distributed datasets, obtaining details may require a round trip from the client to the server, possibly taking several seconds on a congested network.

★ The term **rendering** is used in computer graphics for drawing an image.

When systems are designed so that all of these latencies are well matched, the user interacts fluidly and can stay focused on high-level goals such as building an internal mental model of the dataset. When there is a mismatch, the user is jarred out of a state of flow [Csikszentmihalyi 91] by being forced to wait for the system.

6.8.3 Interactivity Costs

Interactivity has both power and cost. The benefit of interaction is that people can explore a larger information space than can be understood in a single static image. However, a cost to interaction is that it requires human time and attention. If the user must exhaustively check every possibility, use of the vis system may degenerate into human-powered search. Automatically detecting features of interest to explicitly bring to the user's attention via the visual encoding is a useful goal for the vis designer. However, if the task at hand could be completely solved by automatic means, there would be no need for a vis in the first place. Thus, there is always a trade-off between finding automatable aspects and relying on the human in the loop to detect patterns.

6.9 Get It Right in Black and White

Maureen Stone has advocated the slogan **Get It Right in Black and White** as a design guideline for effective use of color [Stone 10]. That is, ensure that the most crucial aspects of visual representation are legible even if the image is transformed from full color to black and white. Do so by literally checking your work in black and white, either with image processing or by simply printing out a screenshot on a black and white printer. This slogan suggests encoding the most important attribute with the luminance channel to ensure adequate luminance contrast and considering the hue and saturation channels as secondary sources of information.

► The principles of using color to visually encode data are discussed in Section 10.2. Figure 12.13 shows an example of explicitly checking luminance contrast between elements on different layers.

6.10 Function First, Form Next

The best vis designs should shine in terms of both form and function; that is, they should be both beautiful and effective. Nevertheless, in this book, I focus on function.

My rationale is that given an effective but ugly design, it's possible to refine the form to make it more beautiful while maintaining the base of effectiveness. Even if the original designer of the vis has no training in graphic design, collaboration is possible with people who do have that background.

In contrast, given a beautiful and ineffective design, you will probably need to toss it out and start from scratch. Thus, I don't advocate a "form first" approach, because progressive refinement is usually not possible. My argument mirrors the claims I made in the first chapter about the size of the vis design space and the fact that most designs are ineffective.

Equally important is the point that I don't advocate "form never": visual beauty does indeed matter, given that vis makes use of human visual perception. Given the choice of two equally effective systems, where one is beautiful and one is ugly, people will prefer the better form. Moreover, good visual form enhances the effectiveness of visual representations.

I don't focus on teaching the principles and practice of graphic design in this book because they are covered well by many other sources. I focus on the principles of vis effectiveness because of the lack of other resources.

6.11 Further Reading

No Unjustified 3D The differences between planar and depth spatial perception and the characteristics of 3D depth cues are discussed at length in both of Ware's books [Ware 08, Ware 13]. An in-depth discussion of the issues of 2D versus 3D [St. John et al. 01] includes references to many previous studies in the human factors and air traffic control literature including the extensive work of Wickens. Several careful experiments overturned previous claims of 3D benefits over 2D [Cockburn and McKenzie 00, Cockburn and McKenzie 01, Cockburn and McKenzie 04].

Memory Ware's textbook is an excellent resource for memory and attention as they relate to vis [Ware 13], with much more detail than I provide here. A recent monograph contains an interesting and thorough discussion of supporting and exploiting spatial memory in user interfaces [Scarr et al. 13].

Animation An influential paper on incorporating the principles of hand-drawn animation into computer graphics discusses the importance of choreography to guide the viewer's eyes during narrative storytelling [Lasseter 87]. A meta-review of animation argues that many seemingly promising study results are confounded by attempts to compare incommensurate situations; the authors find that small multiples are better than animation if equivalent information is shown [Tversky et al. 02] and the segmentation is carefully chosen [Zacks and Tversky 03]. An empirical study found that while trend animation was fast and enjoyable when used for presentation it did lead to errors, and it was significantly slower than both small multiples and trace lines for exploratory analysis [Robertson et al. 08].

Change Blindness A survey paper is a good starting point for the change blindness literature [Simons 00].

Overview, Zoom and Filter, Details on Demand This early and influential mantra about overviews is presented in a very readable paper [Shneiderman 96]. More recently, a synthesis review analyzes the many ways that overviews are used in infovis [Hornbæk and Hertzum 11].

Responsiveness Is Required Card pioneered the discussion of latency classes for vis and human–computer interaction [Card et al. 91]; an excellent book chapter covering these ideas appears in a very accessible book on interface design [Johnson 10, Chapter 12]. The costs of interaction are discussed in a synthesis review [Lam 08] and a proposed framework for interaction [Yi et al. 07].

Get It Right in Black and White A blog post on Get It Right in Black and White is a clear and concise starting point for the topic [Stone 10].

Function First, Form Next A very accessible place to start for basic graphic design guidelines is *The Non-Designer's Design Book* [Williams 08].