

PV251 Visualization

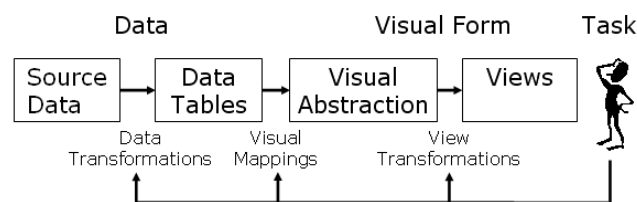
Autumn 2024

Study material

Lecture 4: Basic visualization principles

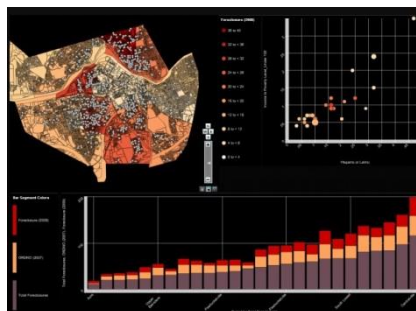
Visualization pipeline

Let us take a closer look at the whole visualization process. All operations (e.g., transformations) performed within the visualization process can be easily mapped to one of the phases of this pipeline.



This image represents the simplest representation of the whole process. There are two key facts to keep in mind:

- 1) Ideally, the user should be able to interact with any part of this pipeline. That is, in individual states, but also in transitions between them (in "nodes" and "edges").
- 2) Each "edge" connecting the individual states is of the "many-to-many" type. This means, for example, that many visualization systems provide the user with several different types of visualization at once (see figure), which are displayed simultaneously on the screen. This, of course, involves multiple mapping and rendering of these different representations simultaneously.



Data processing

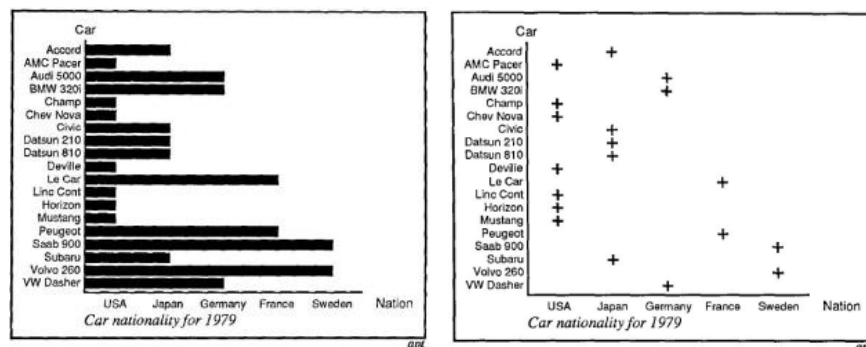
Now let's look at how data is processed in this pipeline - so we'll expand our knowledge of the transformations and processes that change the input data.

Data preprocessing and transformation

We dealt with this area in the last lecture, when we discussed the individual techniques used for data preprocessing. Their goal is to process raw input data into a form that could be used by the selected visualization system. First of all, however, we must make sure that the data is mapped to the basic data types accepted by the computer. Only in the second step can we work with this data, and in this phase come the techniques mentioned in the last lecture, which can deal with missing or incorrect records or too large sets of input data that cannot be processed at once.

Mapping to visualization

If we have the data prepared using the previous phase, it is time to consider the way of their visual representation. This includes mapping a given representation - such as geometry, color, or sound. It is very easy to design a "nonsensical" visualization or one that provides bad information. An example of a visualization that can lead to misinterpretation is shown in the figures.



On the left side, a bar chart is used, where each column passes through all the values on the x-axis and ends at the value of the respective country. The intuitive perception of this representation tells us that each value on the x-axis that the column passes corresponds to a given item on the y-axis. Which is completely wrong in this case, when we want to show the manufacturer of the car brand. For example, according to this representation, Volvo would come from the USA, Japan, Germany, France and Sweden. A much appropriate representation for this is shown in the right image.

The last phase of the visualization pipeline is the mapping of geometric data to the resulting image. This part is solved by interacting with one of the computer graphics APIs. The course of this phase is very dependent on the graphics library used. Examples of such libraries are OpenGL, Processing, Flex, DirectX,...

Visualization metrics

We will now try to mathematically define two basic visualization metrics. These can be applied in all phases of the visualization pipeline. Their importance grows at the moment when we want to measure the transmission of information communicated by a given visualization.

These metrics are **Expressiveness** and **Effectiveness**.

Expressiveness

Expressiveness presents the given information and nothing more. Thus, it does not introduce incorrect information into the visualization. Using expressiveness, we therefore measure the concentration of information in a given visualization.

This metric can be expressed for the given information that we want to display to the user as the ratio M_{exp} of this displayed information to the information that we want to communicate to the user. The value of M_{exp} ranges from 0 upwards. If:

$M_{\text{exp}} = 1$, expressiveness is ideal.

$M_{\text{exp}} < 1$, our visualization is not completely effective because we display less information than originally intended.

$M_{\text{exp}} > 1$, on the contrary, we present more information than we should. This is potentially dangerous because we are displaying incorrect information which, in addition, may interfere with the interpretation of the basic information we sought to communicate.

Effectiveness

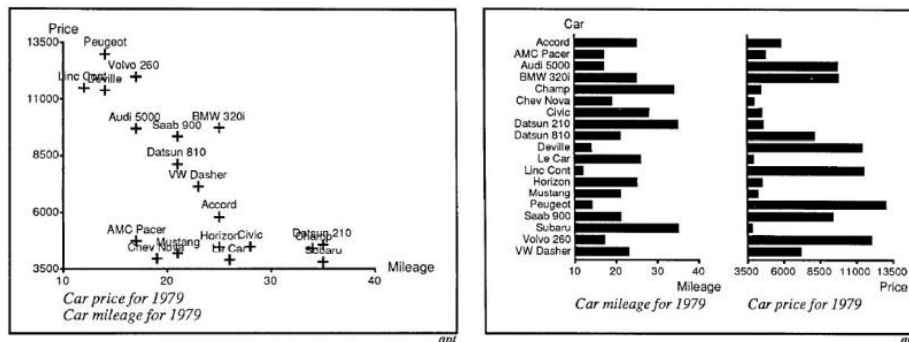
Visualization is effective when it can be interpreted accurately and quickly and when it can be displayed (rendered) "at a reasonable price". Thus, efficiency measures the "price" of a given perception of information.

Efficiency can be defined in a similar way as expressiveness. In this case, we refer to the ratio as M_{eff} . However, in this case, the situation is a bit more complicated. The goal is to define a metric that measures the time required for interpretation for small datasets - because rendering small data is usually very fast. As this interpretation time increases, either due to increasing complexity or size of the data, then the M_{eff} value decreases, with emphasis on the time required for rendering.

M_{eff} is defined as follows:

$$M_{\text{eff}} = 1/(1 + \text{interpret} + \text{render})$$

Again, $0 \leq M_{\text{eff}} \leq 1$. If the M_{eff} value is small, it means that either the interpretation time or the rendering time is large. When the M_{eff} value is close to one, the time required for interpretation and rendering is very short.



As an example of using these two metrics, consider the following situation. Both images show the same information, but each in a different way. The expressiveness of both representations can be considered very similar to identical - both graphs show information about the prices of cars and their mileage and show all the information and just them. In addition, both representations can be rendered very quickly (one of the reasons is a small data set).

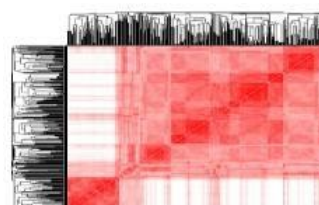
However, M_{eff} is different. The information presented in the graph on the right can be interpreted more accurately and faster for some queries than the one shown in the left graph. For example, for the question "Which car has the least mileage?", The result can be interpreted more quickly from the right graph. But if we extend the question to "Which car, whose price is less than \$ 11,000, has the least mileage?", the right graph is hard to interpret.

Graphic symbols

As a graphic symbol, we denote a visual object that is easily recognizable. Graphic symbols are often part of more complex visualizations, such as various arrows, labels, etc. We will look at how to design graphic symbols and how they are perceived. The area dealing with graphic symbols and signs is referred to as **semiology**.

Any "object" in 2D Euclidean space can be considered a graphical representation consisting of graphical symbols. These can be diagrams, networks, maps, graphs and other commonly used visualizations.

We now focus on the various characteristics of visualization, images, and graphics created from graphic symbols.



In the picture we see two graphic symbols or representations. The left image is very easily recognizable, especially because we probably all know it - we use our previous knowledge and experience. On the other hand, we need much more concentration and attention to understand the picture on the right side.

The process of perceiving and understanding an image takes place in several steps. Understanding the left image is possible thanks to its simplicity and "familiarity" in a single step, when we associate the symbol with its meaning. With the right picture, the situation is more complicated, so the understanding takes place in two steps. In the first step we identify the main and significant elements of the image and in the second step we try to recognize the various relationships between these elements. At this stage, we also try to associate the image with previous knowledge stored in long-term memory. As a result, we define the most interesting information in the picture (for example, clusters, genes, states,...).

Without external - cognitive identification, no graphical representation has meaning. External identification must be easy to read and understand.

The detection of relationships and patterns in a given graphical representation takes place in two basic steps.

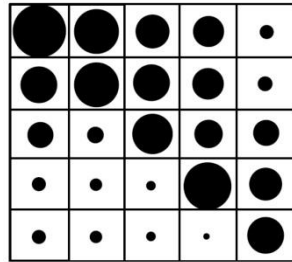
1. Mapping between the relationships of graphic symbols and the data that these symbols represent. In other words, each pattern on the screen must match the pattern in the input data. If this is not the case, it is a disruptive artifact of the selected visual representation.
2. The arrangement of graphic symbols that corresponds to the arrangement on the input data and vice versa. Similar to mapping, if we perceive an arrangement in the image that does not occur in the input data, we again used an incorrect visual representation.

In general, therefore, we must make sure that:

- Similarity between data \Leftrightarrow visual similarity of the corresponding graphic symbols
- Ordering on data \Leftrightarrow visual ordering between the corresponding graphic symbols

Dimensionality of 2D graphics

2D graphics used in visualizations often display three or more dimensions. An example is a picture that shows a matrix with points of different sizes placed in a grid.



Each of the points can be interpreted as a relationship between the position on the x-axis and the y-axis. In addition, the dots vary in size, adding a third dimension to the image. This can be considered the z-axis. Some relationships are obvious at first glance, such as the size of the circles on the diagonal (in addition, the circles at the top of the diagonal are larger than at the bottom). The set of these points represents all the relationships between the x, y and z coordinates that are present in the input data set.



Another example is this picture. At first glance, we see a tree with two branches. We know immediately that it is a tree, without having to look in detail at all the details of the image, such as the leaves. Graphics can contain large amounts of data, which are themselves graphic elements. The only limitation is that these elements must be distinguishable from each other. The human eye is then able to recognize the layout of individual graphic symbols.

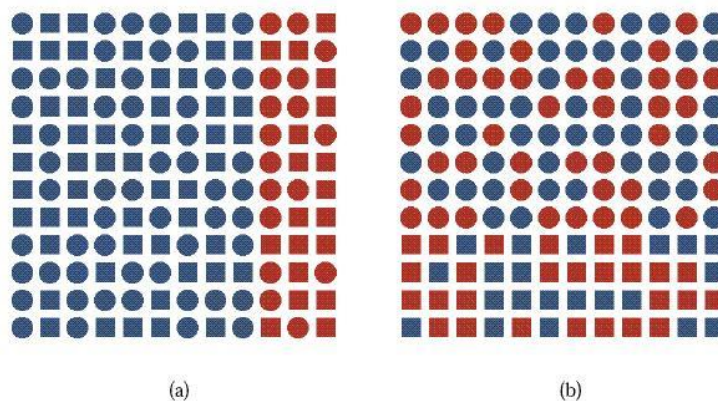
Analysis of graphics

When analyzing graphics, we first perceive a grouping of objects (pre-attentive = subconscious perception). Subconscious processing of visual information takes place by detecting the basic properties of objects in the scene. These properties include, for example, color, length, size, curvature, etc. A huge advantage is that this method of processing helps to quickly focus attention on the target using only a unique visual element.

The table shows a list of known visual elements that we perceive subconsciously, together with a reference to a relevant study describing that.

Feature	Author
line (blob) orientation	Julész & Bergen [1983]; Wolfe [1992]
length	Triesman & Gormican [1988]
width	Julész [1985]
size	Triesman & Gelade [1980]
curvature	Triesman & Gormican [1988]
number	Julész [1985]; Trick & Pylyshyn [1994]
terminators	Julész & Bergen [1983]
intersection	Julész & Bergen [1983]
closure	Enns [1986]; Triesman & Souther [1985]
colour [hue]	Triesman & Gormican [1988]; Nagy & Sanchez [1990]; D'Zmura [1991]
intensity	Beck et al. [1983]; Triesman & Gormican [1988]
flicker	Julész [1971]
direction of motion	Nakayama & Silverman [1986]; Driver & McLeod [1992]
binocular lustre	Wolfe & Franzel [1988]
stereoscopic depth	Nakayama & Silverman [1986]
3-D depth cues	Enns [1990]
lighting direction	Enns [1990]

The figure then shows an example of showing the segregation of a given region by shape and color. The border in the region is perceived better if it is separated by color. The second case of separation by shape already requires more attention.



In the second step, we try to characterize these groupings cognitively. This includes the process of cognition, which includes, but is not limited to, decision-making and reasoning. Finally, we will focus on special cases that are not directly contained in these groups or in their relationships. The whole process can be performed on many levels and using different types of visualization techniques.

Eight visual variables

In order to be able to use graphics to convey information, it is first necessary to understand the individual graphic primitives and their properties. Graphic primitives are often referred to as "**marks**". One way to encode input data for display is to use mapping of different data types to different tags and their attributes. However, the tags themselves will not display the required information - only their spatial arrangement will give them the required sharing.

Placing markers in the screen space adds importance to additional properties of the input data, such as similarity and layout of the data.







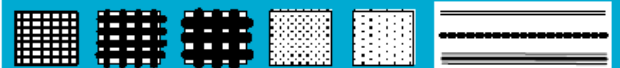
If we have already defined the types and layout of individual tags, we can add additional graphic properties to each of the tags. They can vary in size, color or orientation, for example.

In general, we can distinguish eight ways in which graphical objects can encode information - we refer to it as 8 visual variables:

- Position
- Shape
- Size
- Brightness
- Color
- Orientation
- Texture
- Movement

These variables can be used to maximize the effectiveness of a given visualization method.

In the picture you can see the visual variables defined by Jaques Bertin. His model lacks the eighth variable - motion.

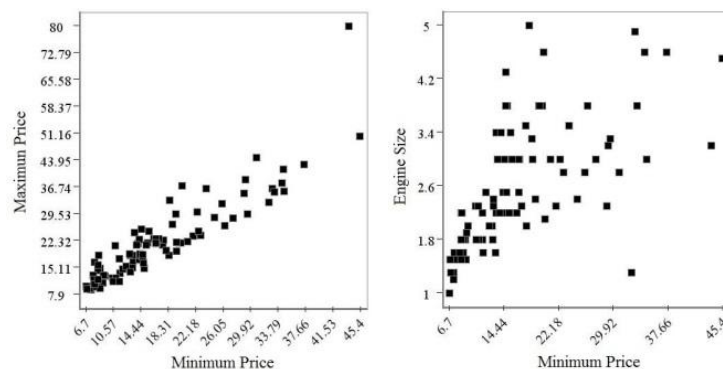
Bertin's Original Visual Variables	
Position changes in the x, y location	
Size change in length, area or repetition	
Shape infinite number of shapes	
Value changes from light to dark	
Colour changes in hue at a given value	
Orientation changes in alignment	
Texture variation in 'grain'	

We will now look at the individual variables in more detail.

Position

The first and most important visual variable is the position - the location of individual graphic elements in the screen space. Position has the greatest impact on the display of information, because the spatial arrangement of graphics is the first step in understanding visualization. It

is true that the larger the distribution of graphic elements on the screen, the more information we are able to convey. The worst case is visualization, in which all data is mapped to graphic symbols placed all in one place - in the result we see only the symbol that was displayed last and we lose the rest of the information. On the contrary, the best scheme maps each graphic symbol to its unique position, where it does not overlap with others. For standard screens with a resolution of 1024 x 768, we get the number of individual pixels of only 786,432. Therefore, we are not able to display a million individual symbols on such a screen. In addition, since more than one pixel is required to reasonably display and understand the information, this maximum number is significantly reduced.



An example of the meaning of position is shown in the figure. The left image shows the minimum vs. maximum prices of cars manufactured in 1993. The distribution of individual points indicates a linear relationship between the minimum and maximum price.

The second figure shows a different set of data - it compares the minimum price and engine size for cars made in 1993. Unlike the first figure, there is no significant relationship between these variables.

Selecting variables to organize data on the screen can answer many questions, such as: Which area does the most data fall into? Do the data copy any of the known layouts? Are there any trends in the data? Can we cluster data in any way?

Another way to discover different relationships, contexts and structures within the displayed data is to use scaling - resizing. In general, the use of scaling is not necessary, but in some cases it must be used.

The first type is so called **linear scale**, that simply increases or decreases the range of values.

The second type is the **logarithmic scale**, which is used to map exponentially changing variables to a more coherent (compact) range.

Although both of these types of scaling are commonly applied to individual variables, it is sometimes necessary to use them in cases of two-dimensional mapping - for example in cartography. In this case, we are talking about projection.

In most cases, it is very advantageous to add additional graphics to the visualization to help describe the space. Examples of such elements are axes. These usually contain ticks that define the intervals of the range of values and text tags describing the values themselves. In addition, each axis contains a label to which data it relates.

Shape

The second visual variable is the shape or mark: points, lines, areas, volumes, and combinations thereof. Tags are graphical primitives representing data. For example, both visualizations relating to automobiles from 1993 in the previous section used a graphical point primitive to display individual values. Any graphic object can be used as a marker, including symbols, letters, and words.

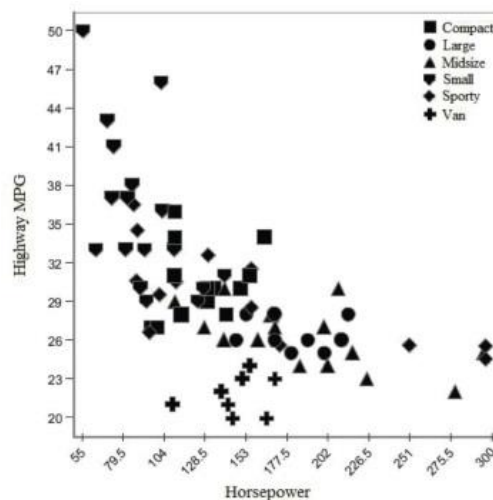
Examples of glyphs are shown in the figure.



When we talk about glyphs, it is important not to mention their size, brightness component, orientation, etc., because these are other visual variables, which we will discuss below.

When using glyphs, it is important to consider the distinction between the glyphs used. Hundreds to thousands of individual glyphs can sometimes be observed in one visualization. It is therefore necessary to choose brands that are not very similar.

As an example, let's take the use of the glyphs we showed in the previous image. This visualization uses different shapes of glyphs to distinguish different types of cars in a graph comparing consumption and engine power. It is possible to observe individual clusters.



Size

The previous two variables formed the basis of all visualizations. Without them, it would be impossible to create a visualization. The remaining variables affect the way individual representations are displayed.

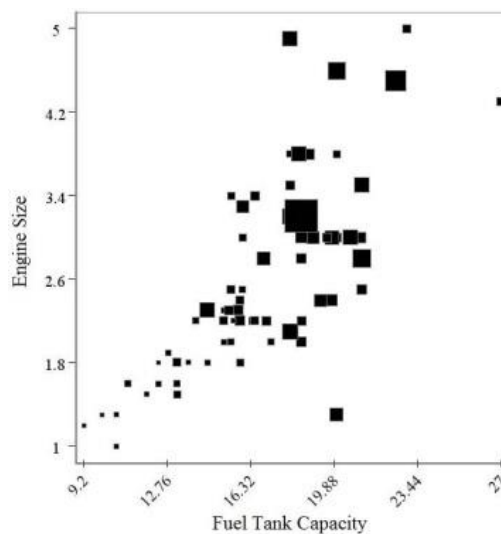
The third visual variable is size. It determines how small or large the displayed marks will be (see picture).



Size can be easily mapped to interval and continuous data variables.

Size is especially useful for datasets with low cardinality, because it is very difficult to distinguish glyphs that differ very little in size.

An important aspect is the choice of glyph type. For points, lines and curves, the use of size is very suitable. This can be seen in the figure, which again shows a dataset representing car models from 1993. We now show the size of the engine relative to the size of the tank. The size of the tags corresponds to the size of the maximum price for the given combination.



If the graphic contains tags in the form of areas, mapping additional information in the form of size is inappropriate.

Scale is also important when comparing the data values according to the size of the corresponding glyphs. The following example shows the difference between cases when we have the representation with no unified scale shared by the data items (left) and situations when the dataset shares a common scale (middle and right). The latter two cases are showing the framed representation (the frame delineates the possible range of values) and the representation where the basis of the bars is aligned with the x-axis. Both

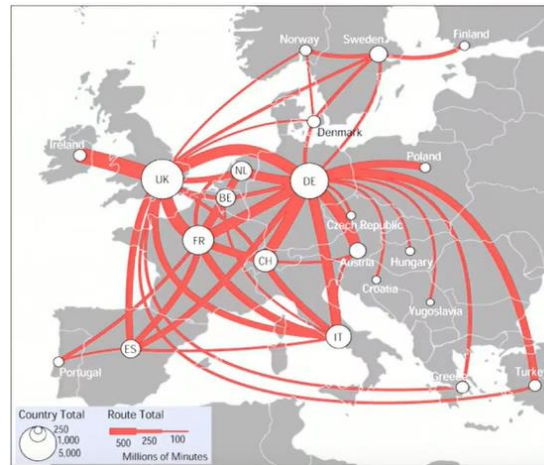
representations significantly increase the accuracy in assessing the difference between the values.

• position along common scale



- frame increases accuracy [Cleveland 84]

When the size is encoded to the thickness of a line, we can use only a very limited number of thickness threshold values to still assure that the differences are distinguishable.



Be careful about using perspective projection as this substantially influences the perception of size differences!



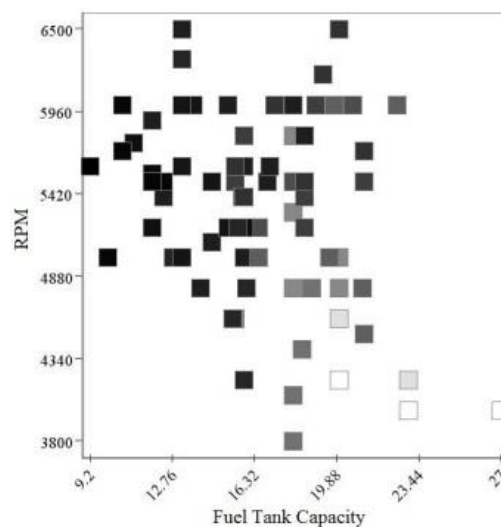
Brightness

The fourth visual variable is brightness or luminance. It is the second variable that is used to modify glyphs to present additional data variables or properties.

Due to the limited possibilities of human perception, which cannot distinguish all existing brightness levels, this variable can be used to display continuous data variables or to accurately distinguish between glyphs drawn in sampled shades of brightness. It is further recommended that the brightness scale be linear, which maximizes the perceptibility of the differences.



An example of the use of the luminance component in the already known case of comparing cars from 1993 is shown in the figure. Here, the luminance component is used to display the width of the car. The darker the point, the wider the car.



Color

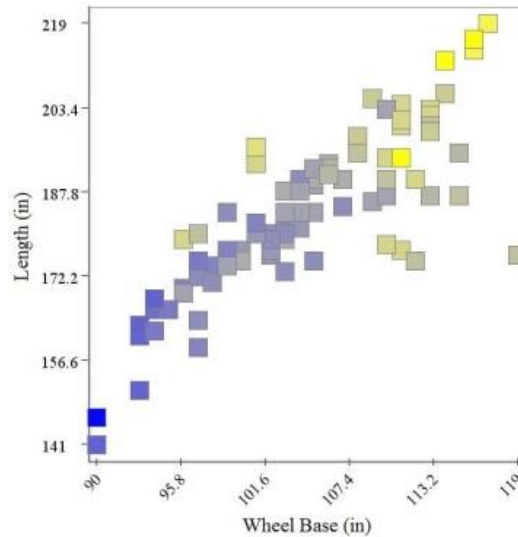
We have dealt with color and its perception before, so now we will only add some information. The difference between brightness and color is that color is defined by two parameters - hue and saturation.

The image showing the color scale is taken from Microsoft programs, where the hue is plotted on the horizontal axis and the saturation on the vertical. The hue is represented by all the colors of the visual spectrum. Saturation is defined by the degree of hue relative to the gray color and affects the "purity" of the color.

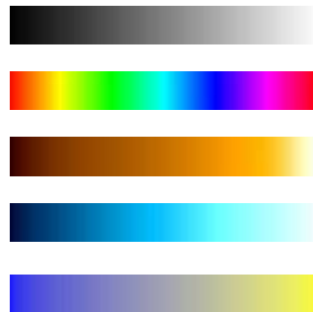
Using color to display information requires mapping data values to individual colors. For this purpose, it is first necessary to define a so-called colormap (color map), which specifies the relationship between the range of values and individual color values. Color maps are suitable for visualization of discrete and continuous data variables.



The picture is an example of a color map and its use for our case with cars. Here the color is used to display the length of the car. This length is also associated with the y-axis. Blue indicates shorter cars, yellow the longer ones.

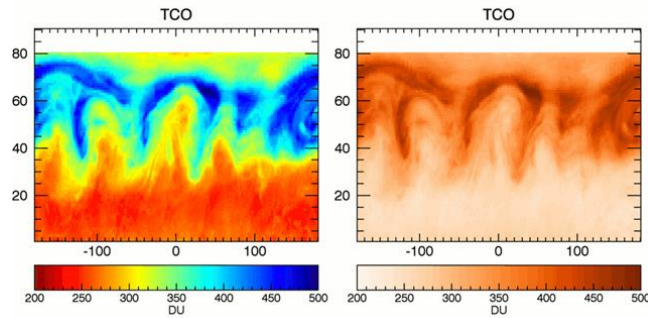


The picture shows the common and popular types of color maps.



Problems with the rainbow scale

The main issue of using the rainbow color scheme is the non-uniformity in the distribution of the individual colors and in perception. Using this scheme can lead to creating sharp artificial boundaries between colors (particularly involving yellow) that are not necessarily present in the underlying data.



The following map describes the amount of evapotranspiration (loss of rain water through evaporation) by county for the 48 contiguous U.S. states.

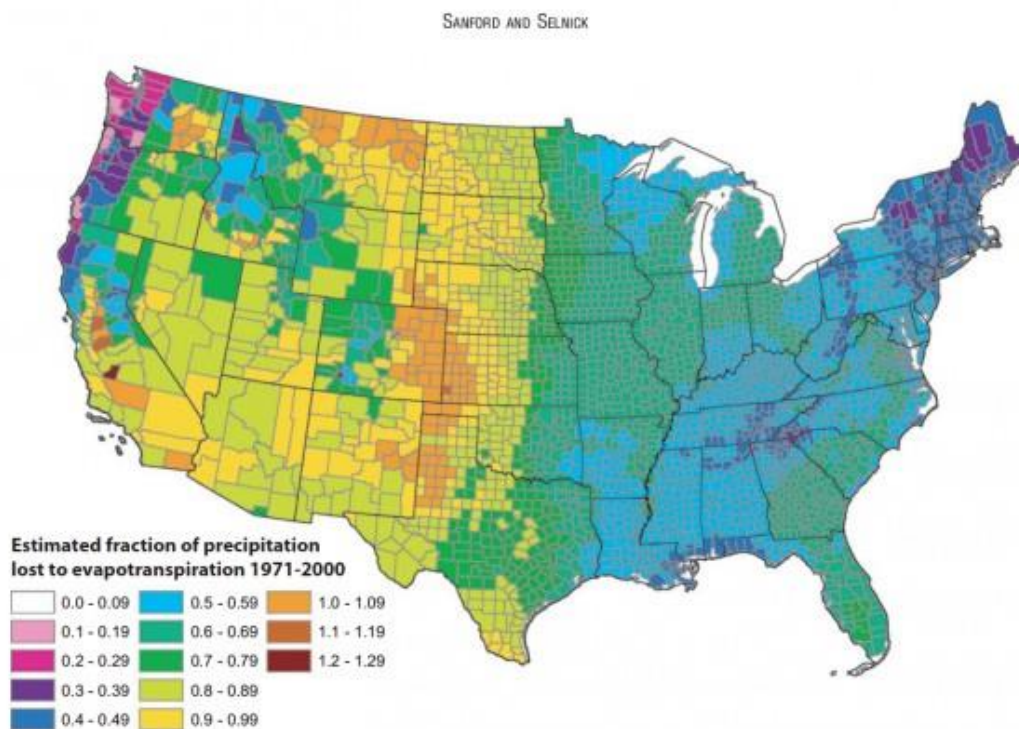
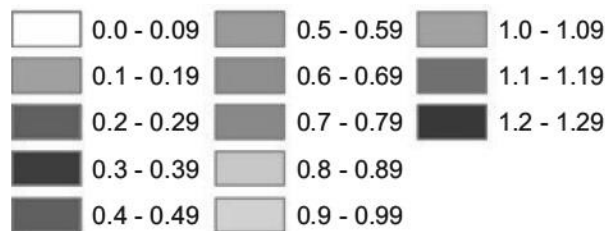


FIGURE 13. Estimated Mean Annual Ratio of Actual Evapotranspiration (ET) to Precipitation (P) for the Conterminous U.S. for the Period 1971-2000. Estimates are based on the regression equation in Table 1 that includes land cover. Calculations of ET/P were made first at the 800-m resolution of the PRISM climate data. The mean values for the counties (shown) were then calculated by averaging the 800-m values within each county. Areas with fractions >1 are agricultural counties that either import surface water or mine deep groundwater.

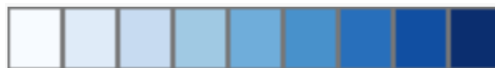
Do you see the how the country is divided down the middle? The Eastern half seems to be all dark green and blue, while the Western half is all light greens, yellow and orange. Surely, there is a huge difference between the two. So where is the problem? As it turns out, the values change smoothly, but the colors do not. There are two problems here: abrupt changes in luminance (perceived brightness of a color) as well as switching between different hues.



The combination of smoothly varying and abruptly changing luminance makes it appear as if there were clearly defined regions on the map. A version of the legend that only shows luminance, without hue, makes this a bit more obvious.



Given the issues, why are the rainbow colormap and its variants so popular? I think the answer is quite simple: it's attractive. Using a single hue to show the data would be reasonably effective, but much less interesting to look at (see figure below). What is more, if you're looking to read off the individual values, the smooth ramp is actually worse because you can't look for a particular hue anymore. The cost is that you create lots of artifacts in the map, though.



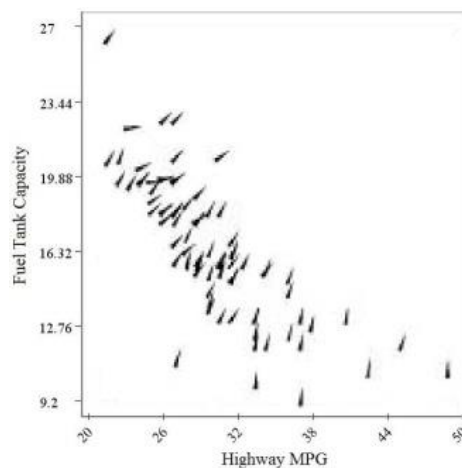
Orientation

The sixth visual variable is orientation. Orientation is very much associated with subconscious perception. This widget property describes how the tag is rotated depending on the data variable. Obviously, orientation cannot be used for all shapes of marks - for example, a circle is not subject to change during rotation. The most suitable markers for displaying orientation are those that have their "main axis". The orientation then changes around this axis. Such markers can then display a range of different orientations, as in the example in the figure. The mark is represented by an elongated triangle, for which the main axis is obvious.



The figure shows the possibility of using this glyph only up to 90°, however this mark is usable for the whole range of 360°. The interpretation is that the lowest data values are mapped to a mark pointing upwards (leftmost) and the increasing data values rotate the glyph clockwise.

The graph again shows the relationship between consumption and tank capacity. The additional data variable displayed using our triangular glyphs is the average price.



Texture

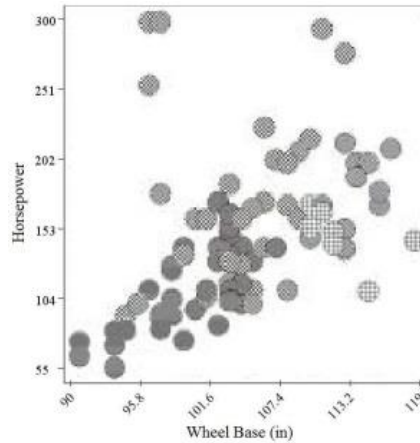
The penultimate visual variable is a texture. It can be thought of as a combination of several other visual variables, including glyphs (texels), color (associated with each pixel in the texture area), and orientation (mediated by changes in color).

Some simple textures with a linear appearance are formed by dashed and dotted lines. It is possible to easily distinguish between them, because there are only a limited number of such textures.



Texture is most often associated with polygons, regions, and surfaces. In 3D, a texture is usually considered an attribute of geometry. Similarly, a texture can be associated with the color of a graphic entity, with regular and irregular color variations of varying extent and layout.

An example of using texture for our case is mapping a car type to different textures.



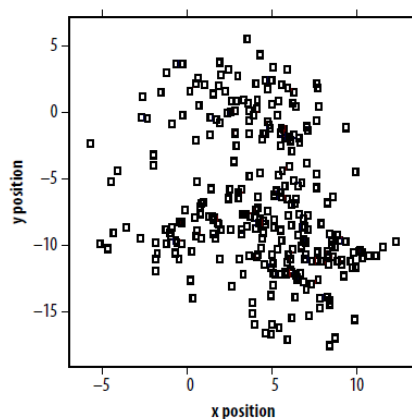
Movement

The last visual variable is movement. In fact, movement may be associated with any of the previously mentioned visual variables. One of the common uses of motion is to change the speed at which a change is made (for example, a change of position or flashing).

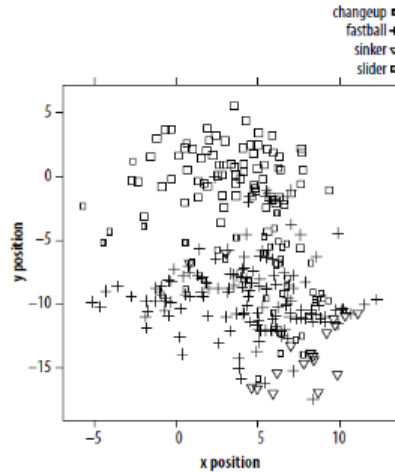
An important aspect of movement is its direction - in the axes, diagonally or arbitrarily. For other visual variables, this is a change: larger/smaller, lighter/darker, larger angle/smaller angle, etc.

Example of usage

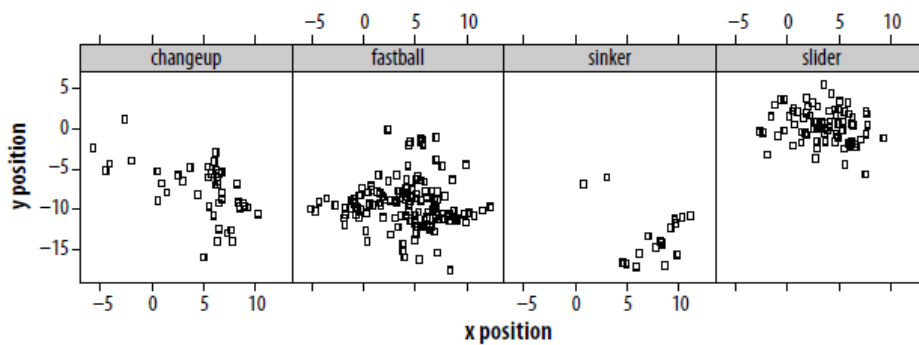
We will show the meaning of individual visual variables and their influence on the presented information about the input data on the example of mapping 287 shots by baseball player Oscar Villarreal in 2008. If we use only one graphic symbol of the same properties (color, shape,...), we can present in 2D only information on where the individual shots were going.



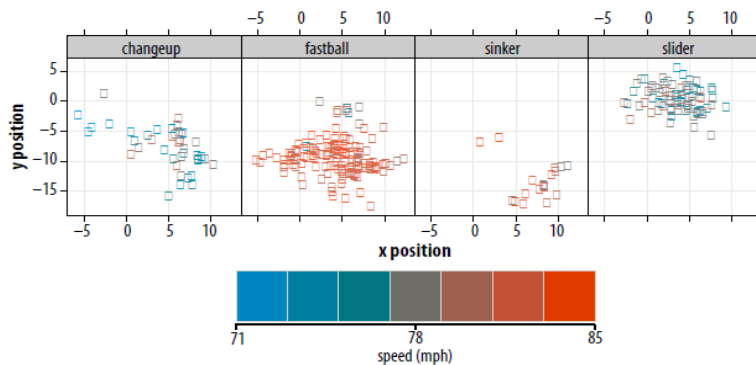
By adding the display of individual items using different types of glyphs, we get another dimension of data into the final visualization, namely the type of individual shots.



To increase clarity, it is possible to divide the input graph into subgraphs based on the type of shots, where each represents only the data corresponding to the given type of shot. Each of the graphs has the same coordinate system.



By simply adding color information to the graphs, we can add additional information to the representation, here the speed of the balls fired.

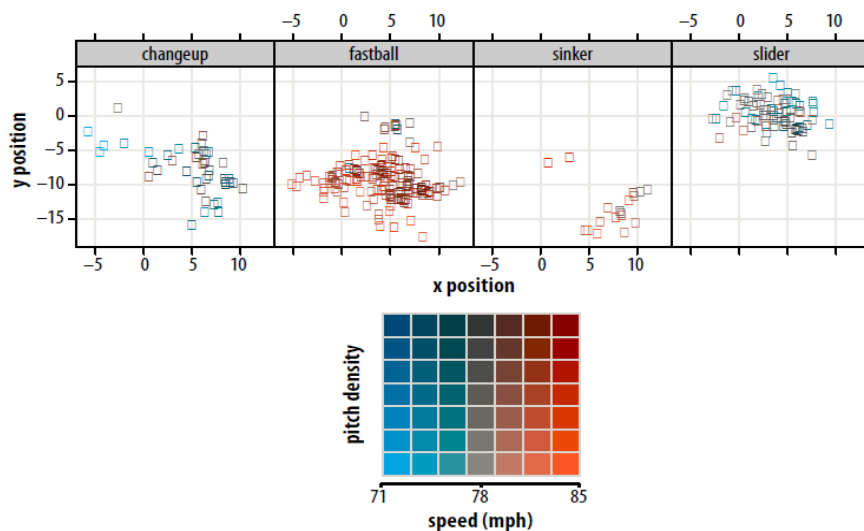


Color is an important element of visualization techniques, especially because differences in color can be detected in as little as 200 milliseconds. Which is even before we even realize we're focusing on visualization. This is called the so-called pre-attentive concept.

Color can be represented in three dimensions. An example is the RGB space, where we are able to determine to some extent whether a given color is more or less red, green and blue. However, this representation is not commonly used. The main two reasons are:

- The fact that a relatively large part of the population is completely color-blind or has some partial impairment of color vision.
- Individual colors have different ranges of color shades that we are able to perceive. An example is yellow and blue, where in yellow we are able to perceive differences between a much smaller number of shades than in blue.

In our example with baseball shots, we will add the latest information about the density of shots at a given location on the field. The density is mapped to the second dimension of the color scale used and corresponds to the hue of the color.



Short history of defining visual representations

Although we have already shown many examples of the use of visualization techniques, we have not yet described a comprehensive apparatus for describing our graphic representations. Visualization is an entire scientific discipline that consistently formalizes general visualizations. Researchers in this field have come and are coming up with their ideas in the form of specifications of various models. Robertson was the first to name the need for a formal model as the basis of any visualization system. In this section we will show an example of formalization of the visualization area and a list of existing formalizations.

Bertin (1967) Semiology of Graphics

In 1967 Jacques Bertin, probably the most important figure in the theory of visualization, published the book *Sémiologie Graphique*. This was the first rigorous attempt to define

graphics and its application to display information. Here, Bertin presents the basics of coding information through graphical representation as so-called semiotics - the science of markup systems.

His first key finding is the strict separation of content (the information we want to encode) from the "transmission medium" (the characteristics of the graphics system used).

In order to fully understand the sign system, it is first necessary to understand all the basic elements that define this system. Based on this knowledge, Bertin compiled the so-called graphic dictionary.

Marks	Points, lines, and areas
Positional	Two planar dimensions
Retinal	Size, value, texture, color, orientation, and shape

Bertin's graphical dictionary is shown in the table. Its goal is to identify some very specific graphic primitives that correspond to perceptual (perceptual) attributes - properties of graphics that are perceptually identifiable.

Bertin further defines the term "plane" by which he considers a clean sheet of white paper of standard size. He further notes that the plane is such a well-known object that its properties do not need to be described and proved, but the most common things are often understood only to a limited extent. The plane is marked by so-called implants - it forms the basic elements of planar geometry (points, lines, surfaces). These three elements are placed in two-dimensional space using imposition, which divides the graphics into four groups: diagrams, networks, maps, and symbols. Visualization is thus defined by implantation and imposition.

The third and last component is a graphical representation using so-called retinal variables - graphical deviations designed for visual perception. These variables affect implants. Experimental psychology defines 6 types of retinal variables:

Size (height, volume or number)

- Value (saturation)
- Texture (fineness or roughness)
- Color (brightness)
- Orientation (angular shift)
- Shape

Bertin further defined several levels of organization specifying how these variables can be combined.

Other systems:

- Mackinlay (1986) APT

<http://assassin.cs.rpi.edu/~cutler/classes/visualization/F10/papers/p110-mackinlay.pdf>

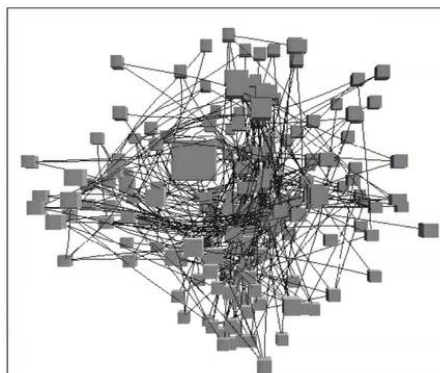
- Bergeron and Grinstein (1989) Visualization Reference Model
- Wehrend and Lewis (1990)
- Robertson (1990) Natural Scene Paradigm
- Roth (1991) Visage and SAGE
- Casner (1991) BOZ
- Beshers and Feiner (1992) AutoVisual
- Senay and Ignatius (1994) VISTA
- Hibbard (1994) Lattice Model
- Golovchinsky (1995) AVE
- Card, Mackinlay and Shneiderman (1999) Spatial Substrate
- Kamps (1999) EAVE
- Wilkinson (1999) Grammar of Graphics
- Hoffman (2000) Table Visualizations

History recap

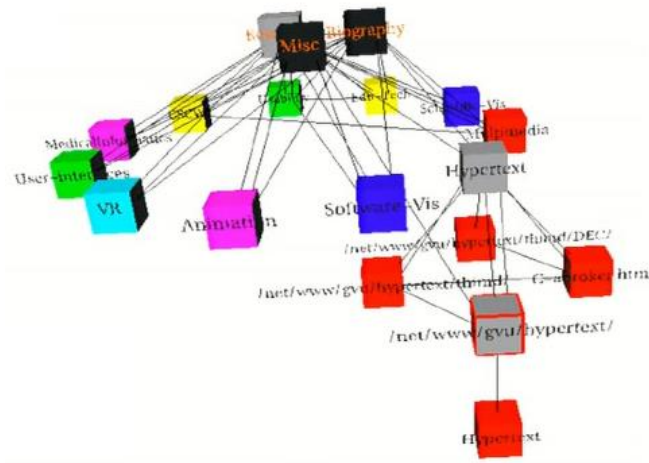
There are a number of graphical models that formalize graphical objects used to display information. Primarily, these models characterize graphics using three components: glyphs, positions, and perceptual variables. Research into these components was initiated by Bertin and still continues. Provides insight into the basics and theory of visualization. Most recent research agrees that Wilkinson has created the most thorough formalization of statistical graphics, which covers most of the basic elements of existing visualization techniques.

Few more remarks regarding the depth perception

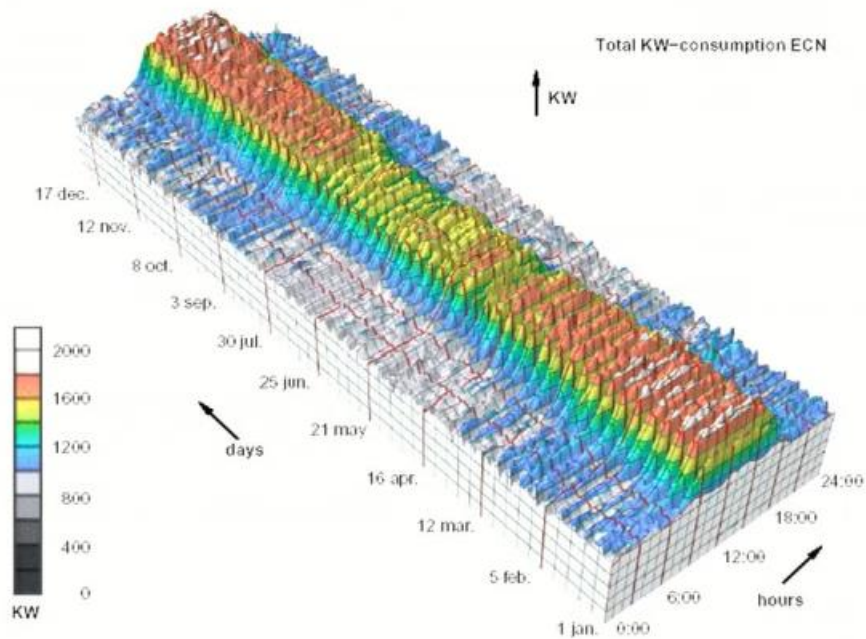
Except for the already mentioned problem with perspective projection, there are some more issues with having a scene in 3D and projecting it to a 2D plane. One of the most typical problems is **occlusion** when some objects can be hidden behind others. This also leads to issues with interaction – direct selection of objects can be performed only on those that are visible from the given viewpoint.



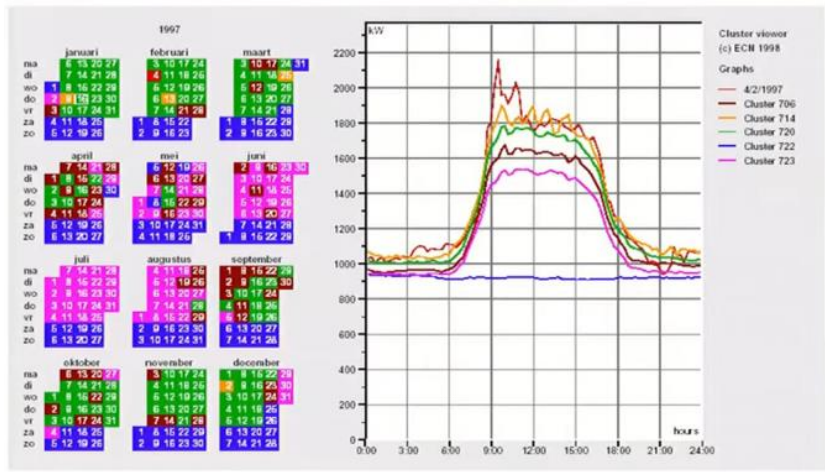
Another issue is the legibility of textual labels – we need to assure that they are always facing the viewport and are not intersecting, too small, etc.



An example of a 3D representation is the following set of time series curves. The vertical axis plots the power consumed in the building, two remaining axes show days and hours. In this representation, it is very hard to compare between distant times and days, thanks to the occlusion.



Possible solution for exploration of this dataset is to create a calendar view where we select a day in the calendar and the data for the given day is plotted in a simple line chart. One may argue that we are losing the overview, such as in which days there was the peak in the power consumption. But this can be encoded, for example, by colored background of the days in the calendar view.



To summarize, when it makes sense to use 3D representation? Definitely if we have true 3D input data. On the other hand, for abstracted representations, this is generally not recommended and there has to be a serious reason for that.