Colour Parametrization in a Multiparametric Image Interface

Natasha Wiebe

An essay presented to the University of Waterloo
in fulfillment for the degree of
Masters of Mathematics
in
Statistics

Waterloo, Ontario, Canada, 1999

Abstract

Multiple views of the same area are used in many applications such as remote sensing and medical diagnostics. To compare the same locus in different views, one can either: analyze views side-by-side, layer a plot with several dimensions using parameters of colour, or use highlighting and linking techniques. This latter approach is discussed and implemented in Quail (a QUantitative Analysis in common Lisp).

Two colour systems, RGB (red, green, blue) and HSV (hue, saturation, lightness), are linked to human colour perception: the former to cone cell types, and the latter to light properties. RGB is used in colour monitors; HSV is intuitive to humans. This implementation converts HSV to RGB. Colour choice, in terms of HSV, is used here to represent image values and in highlighting image values. The choice of what colour parameter to use as the dimension is crucial; lightness and saturation are ordinal, and hue is categorical. This interface gives three choices to represent image values with: by lightness, by hue, or by a combination of the two. This latter choice has the advantage in that it is ordinal and provides a wider range of colours.

1. Introduction

In medical diagnostics and remote sensing, to name only two areas, people analyze multiple views of the same area. Magnetic Resonance Imaging (MRI), Positron Emission Tomography (PET), and Computed Tomography (CT) all use forms of electromagnetic energy, such as X-ray or otherwise to form images. In remote sensing, Multispectral Scanners (MSS) and Thematic Mappers (TM) operate within the infra-red portion of the electromagnetic spectrum to produce images. Synthetic Aperture Radar (SAR) operates in the microwave portion of the spectrum. While many kinds of images are formed using electromagnetic energy, images are also created from sources other than electromagnetic energy. Geographers use aerial photographs, ground surveys, water quality assessments and other similar measures to create maps. Groundwork often serves as reference data for satellite images. Multiple views of the same space are compared and analyzed.

Images, such as the ones described above, are comprised of discrete picture elements. These picture elements, called pixels, are the smallest possible units and cannot be divided nor broken up. The bit resolution of an image is the amount of information that each pixel contains. Hence, images have at least three dimensions: location has two dimensions and the pixel value is the third.

It is natural to use colour as the third dimension on a plot of an image since the electromagnetic energy that we can see, we call colour. Other three-dimensional plots use point symbols that vary in size, shape, and symbol to represent the third dimension. However, a point symbol itself is an image. It, too, uses colour to depict its appearance. Thus, the smallest elements of an image are colours.

Nevertheless, multiple views use more than three dimensions. Some plots presenting more than three dimensions include Andrew plots [see Andrews 1972], grand tours [see Asimov 1985], grid matrices of scatterplots, and other plots that exploit the design of point symbols such as Chernoff faces [see Chernoff 1973].

As plots grow in dimensionality, they become more difficult to understand. Use of graphics in data analysis is intended to simplify the analysis. An alternative to oversaturating one plot with too much information is to have many plots that are linked together through the cases of the data set. This technique has become standard practice. If plots are linked, when point symbols in one plot are highlighted, all the corresponding point symbols (i.e., same case data points) in all the other linked plots are also highlighted. In this way, relationships, patterns, and outliers are easily investigated.

Analyzing multiple views can be done in several ways. Multiple views can be analyzed side by side. Alternatively, by parametrizing and layering colour one can map image values where colour is used not uni-dimensionally but multi-dimensionally. Or, one can use linking and highlighting methods.

The following report, in general, discusses the use of colour in a multiparametric image interface. **Section 2** looks at: colour perception in humans, specifically the functions of rod and cone cells, two colour systems RGB (red, green, blue) and HSV (hue, saturation, value) that fall naturally from how humans perceive colour, and the conversion algorithms between these two systems. **Section 3** investigates using parameters of colour as dimensioning devices in image plots. Highlighting techniques and tools are discussed in **Section 4**. And in **Section 5**, an implementation of these preceding ideas is summarized.

The implementation is in Quail, a statistics environment extension to Common Lisp. Quail stands for QUantitative Analysis In common Lisp [see Oldford 1998].

2. Colour

Colour is light. And light is electromagnetic energy. The wavelengths in the human visual spectrum are interpreted by our eye as hue, ranging from ultraviolet reds, violets, indigos, blues, greens, yellows, oranges, to infra-reds [see Figure 1]. Ultraviolet radiation and infra-red rays border the visual spectrum.

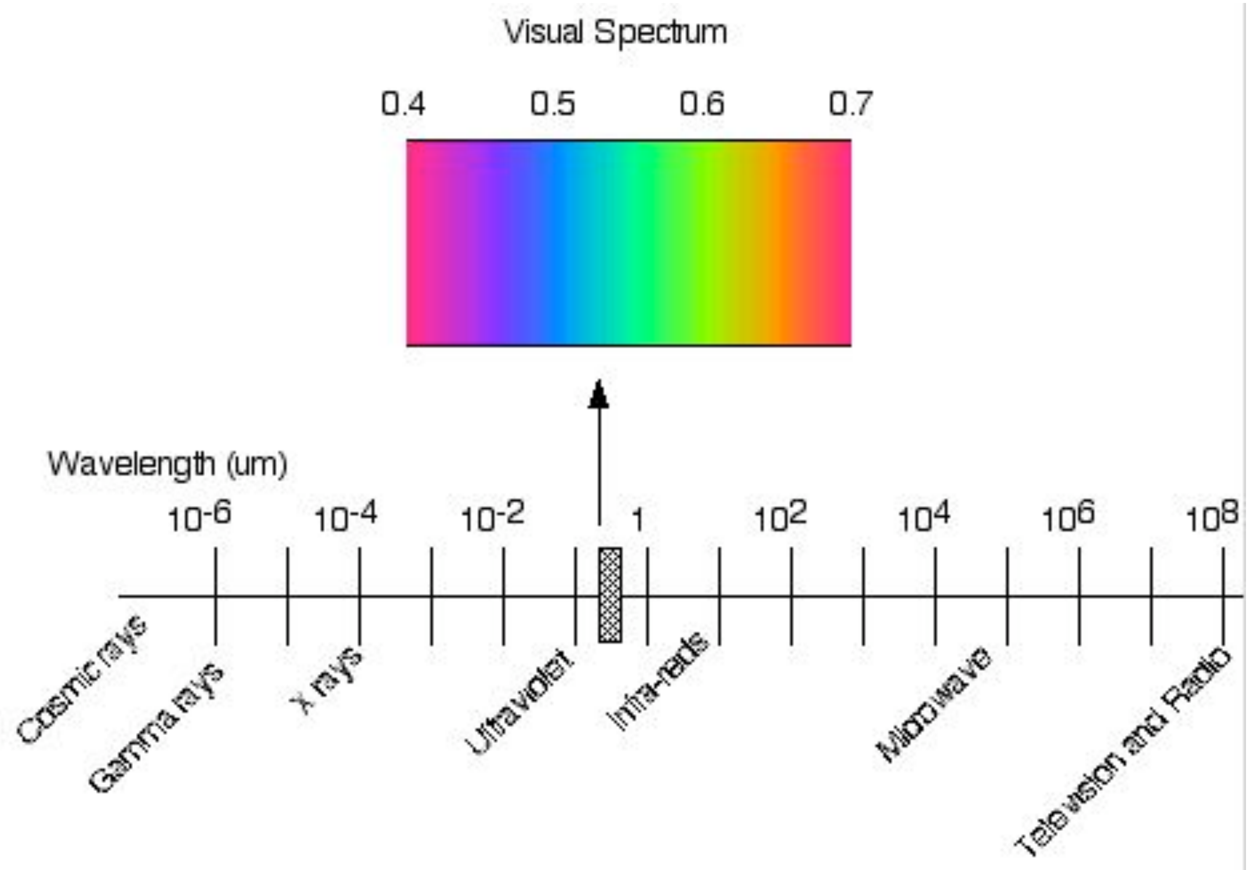


Figure 1. The Electromagnetic Spectrum

People only see the electromagnetic energy from the visual spectrum of colours shown in Figure 1. Combinations of these wavelengths produce the full range of colours that we see. Light can be parametrized in terms of the amount of light (the amplitude) and the size of the wavelength interpreted as hue. Lightness, the amount of light, is also called intensity, value, and brightness.

2.1 Colour Perception

These visual images are projected onto the retina at the back of the eye. The retina is comprised of layers of cells. The first layer of cells filter light through to the optic nerve which is the eye's gateway to the brain. This layer contains two cell types called cones and rods. Cones and rods operate at different ranges of lightness: cones are responsible for day vision; rods are responsible for night vision.

Rods only distinguish two states, white and black. The grays we see at night are only speckles of white and black. Compared to cones, rods are highly sensitive to the amount of light, so they distinguish many degrees of lightness very well.

In contrast, cones only function within a small range of moderate lightness and so are chiefly responsible for spatial details. We cannot see hue at night and we do not see hue very well in intense light. It is argued by Rubin and Richards (1982) that hue provides information about the material of objects and so contributes to the perception of contrast. We contrast, classify, and group objects more by hue than by lightness.

There are three types of cone cells. They perceive short, medium, and longer overlapping ranges of wavelengths [see Figure 2].

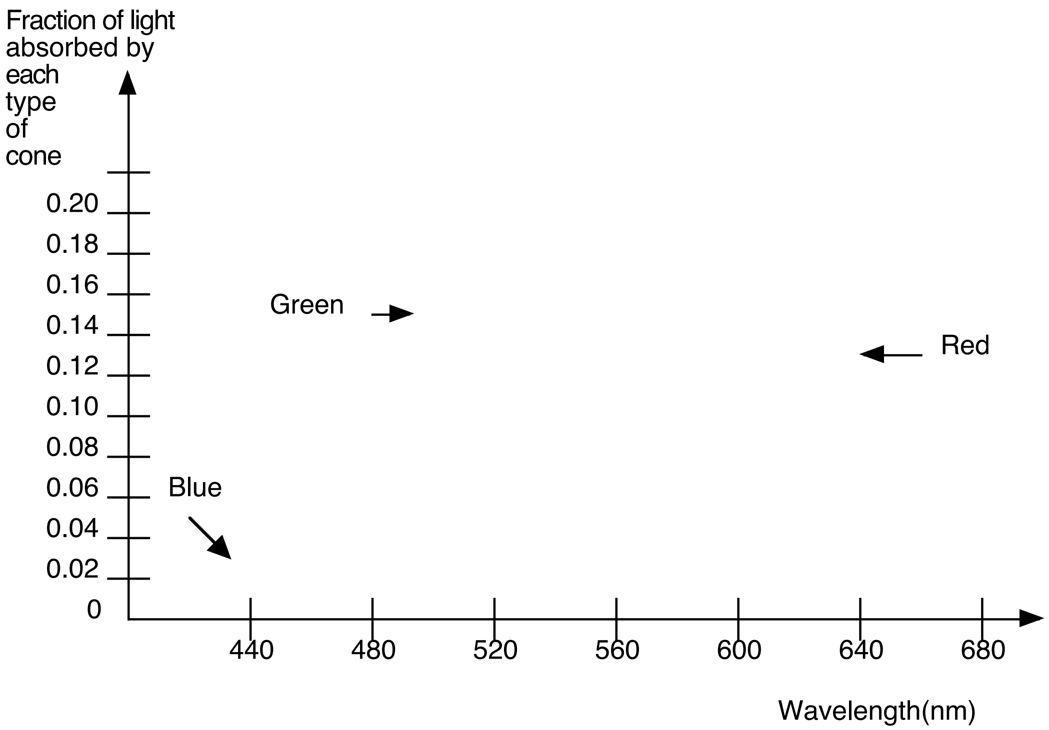


Figure 2. Wavelength Ranges

Colour blindness occurs in people whose cone peak sensitivities are slightly different than what the average person would have. The cone cell types are labelled blue, green, and red, based on data from dichromats, persons who lack one cone cell type. When a dicromat is missing a cone type, it is the colour they cannot see that is used to label the cone type. For example, the dicromat missing the red cone cell type cannot see red.

On their own, each type of cone cannot distinguish a hue. There have been two competing theories on colour perception: Trichromacy Theory and Opponents Colour Theory. The Zone Theory combines the two: trichromacy theory operates in zone one, and opponents colour theory operates in zone two [see Wyszecki 1982].

Trichromacy theory simply states that it is the ratio of responses of each cone type that is responsible for the specification of hue.

Opponents colour theory can be described as follows. There are four psychologically unique hues: red, yellow, green, and blue. Each of these hues can be described without the influence of any other psychologically unique hue. For instance, red and green light are perceived as a yellow light. However, there exists a yellow light with equal amounts of red and green such that the yellow cannot be described as a reddish-yellow or a greenish-yellow. Combinations of any two neighbouring (psychologically unique) colours in varying amounts produce all the colours between them on the visual spectrum. One can get oranges from reddish-yellows and limes from greenish-yellows and so on. However, combinations of bluish-yellows and reddish-greens do not exist as they are not neighbouring colours. Red and green are regarded as opposite as are blue and yellow.

There are two channels that combine to produce hue: a red-green channel and a blue-yellow channel. The amount of red, green, and blue content are the responses of the cone cell types. Note, the combination of the red and green types produce the amount of yellow type. The channels work as follows: one colour cancels out equal amounts of the other colour. When blue and yellow contents are equal, and red and green contents are equal, an achromatic light is produced. Otherwise the two channels combine to produce a chromatic light, never having a reddish-green or vice versa, or a bluish-yellow or vice versa.

Colour, according to this latest theory on chromatic perception and most of the other theories that have preceded it, is inherently three dimensional for humans.

According to zone theory, trichromacy theory is true in so far that the three cone types resonante according to the wavelengths in the operative light. This occurs in zone one in the retina of the eye. In the second zone in the brain, opponents colour theory operates. The three responses are manipulated to produce a singular hue perception.

2.2 Colour Systems

Trichromatic colour systems produce the entire visual spectrum of colours with the smallest number of parameters. We consider two such systems here: RGB (red, green, blue), and HSV (hue, saturation, value).

2.2.1 RGB

RGB is one very important trichromatic colour system because of its use in digital colour systems such as computer and television monitors. Maxwell and Helmholtz (Boynton 1979) found that mixing varying amounts of red, green, and blue could produce any colour in lighting. All systems that produce light are additive which means when the maximum of all three colours are added equally they produce white. The absence of all three colours looks black.

The RGB colour system can be modelled as a cube [see Figure 3]. The eight corners are red, magenta, blue, cyan, green, yellow, white, and black. At the black point, red, green, and blue are equal to zero. As you move away from the black point along the axes, the colour becomes redder, greener, or bluer according to the axis. When red, green, and blue reach their maximum values, they produce white. Full redness and greenness equals yellow. Blue and red make magenta and blue and green make cyan. Achromatic colours are created with equal amounts of red, green, and blue; these lie along the achromatic axis from black to white.

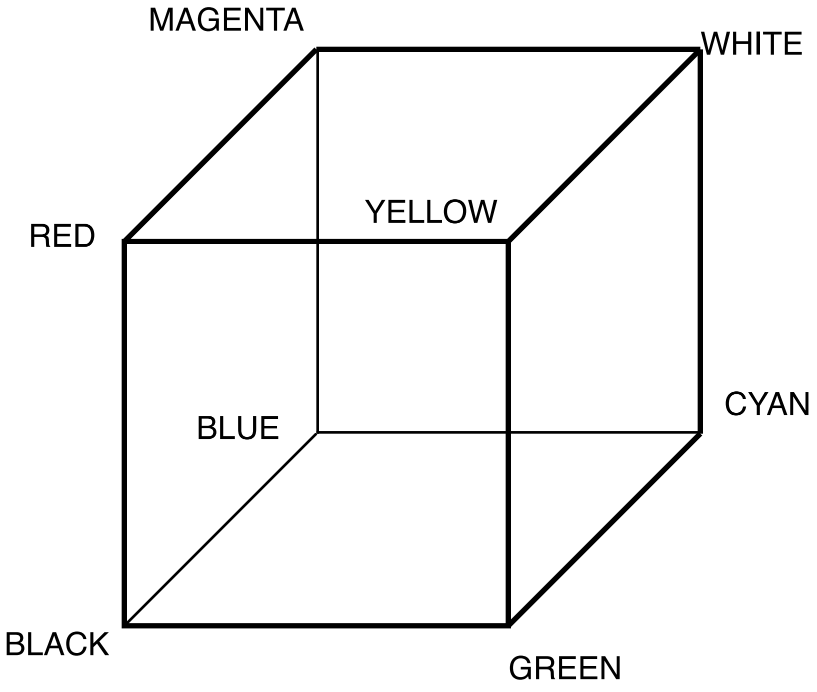


FIGURE 3. RGB Colour Cube

It is not easy to describe colour in terms of RGB. However, another far more intuitive trichromatic colour system is HSV: hue, saturation, and value. This system is relatively simple to use to describe a colour choice.

2.2.2 HSV

HSV is most easily described as a cylinder [see Figure 4].

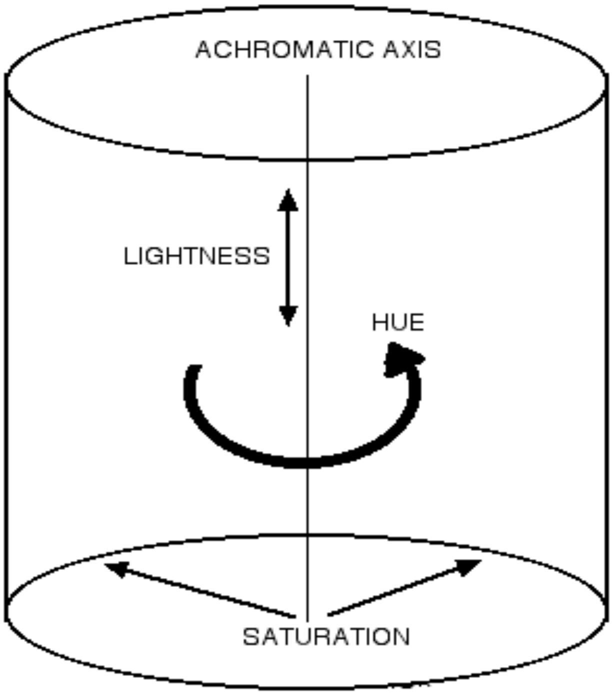


Figure 4. HSV Cylinder

Hue and saturation are the polar coordinates of the cylinder --- saturation is the radius and hue is the angle. Although hue is not a circular variable, it can be made circular if one end of the visual spectrum is bent to the other end of the visual spectrum. To the human eye, infra-red blends into ultraviolet red. The height of the cylinder is lightness. The axis which runs through the centre of the cylinder is called the achromatic axis. Only grays exist on this axis, with white at one end and black at the other end. Along this axis, hue is undefined, saturation equals zero, and only height has magnitude. Moving radially from the axis to the surface of the cylinder increases the saturation of the hue from a pure gray to a completely saturated hue.

2.3 Conversion From RGB to HSV

Colour monitors use the RGB system. They use three colour guns: red, green, and blue to emit light. Since users prefer to specify colour in terms of the HSV colour system, it is necessary to convert from RGB to HSV and vice versa for computer graphics.

The algorithms for mapping RGB to HSV and vice versa are not one-to-one. Fully saturated colours do not have the same perceived lightness value, hence there are variations of HSV models. For example, a pure yellow hue always looks brighter than a pure blue hue. This HSV colour system takes on different shapes algorithmically. Three of these shapes, which I have implemented following Levkowitz (1997), are the HSL-Triangle, the HSV-Hexcone, and the HLS-Double-Hexcone. Their differences are based solely on how lightness is derived from the RGB cube.

2.3.1 HSL-Triangle

The HSL (hue, saturation, lightness) triangle model defines lightness as (R+G+B)/3. Visually, if the RGB colour cube is set on its achromatic or gray axis, the triangle model divides the cube into three sections perpendicular to this axis [see Figure 5]. When the cube is sliced perpendicular to the gray axis, the slices are triangular or hexagonal planes. Slices in the top and bottom third are triangular; slices in the middle third are hexagonal. Each slice represents a constant lightness surface.

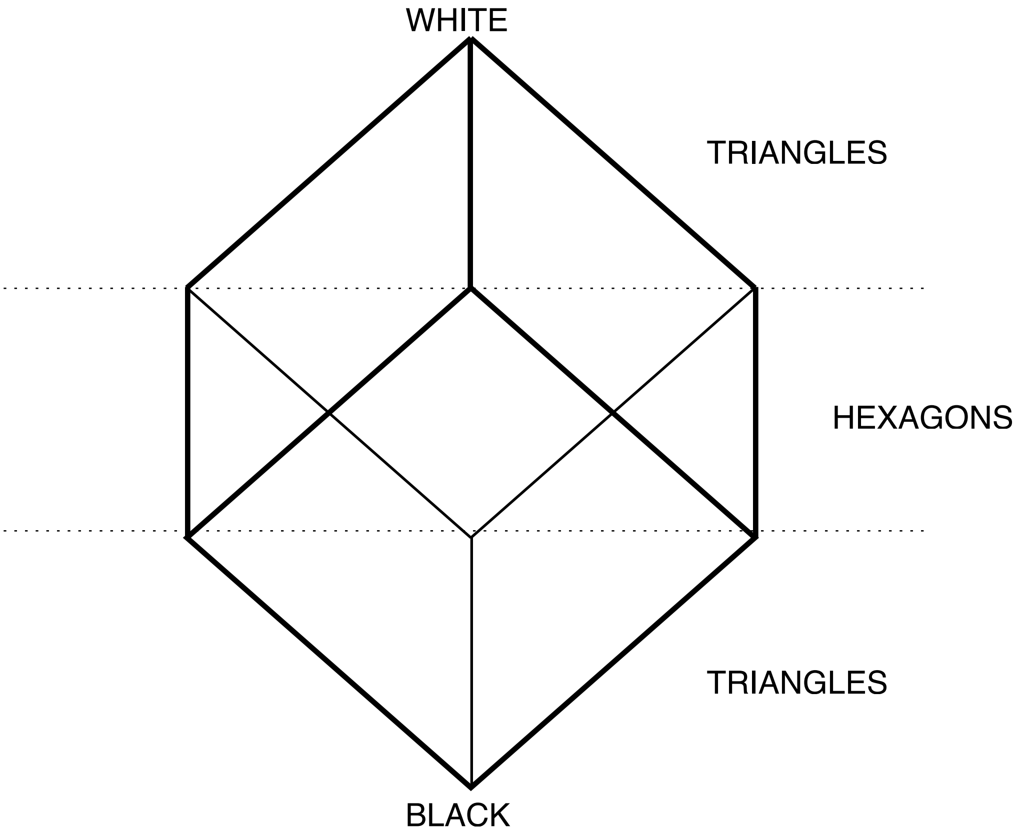


FIGURE 5. HSL-Triangle

2.3.2 HSV-Hexcone

The HSV (hue, saturation, value) hexcone gives another lightness projection. Here, lightness is defined as the maximum value of red, green, or blue. These constant lightness surfaces are what is visible of the subcube from the top lightest point of the achromatic axis for all possible subcubes of the colour cube. The subcubes all include the black point. The largest subcube is the entire colour cube. When the top visible surface of these subcubes are projected onto a plane perpendicular to the achromatic axis, they become hexagonal discs. When the discs are layered according to increasing area, they form a hexagonal cone (a hexcone) [see Figure 6]. Black, the smallest hexagonal disc, is the point of the hexcone. The rim of the hexcone contains all the pure hues. The lightest red of this model is pure red, unlike the white of the HLS triangle or the HSV cylinder.

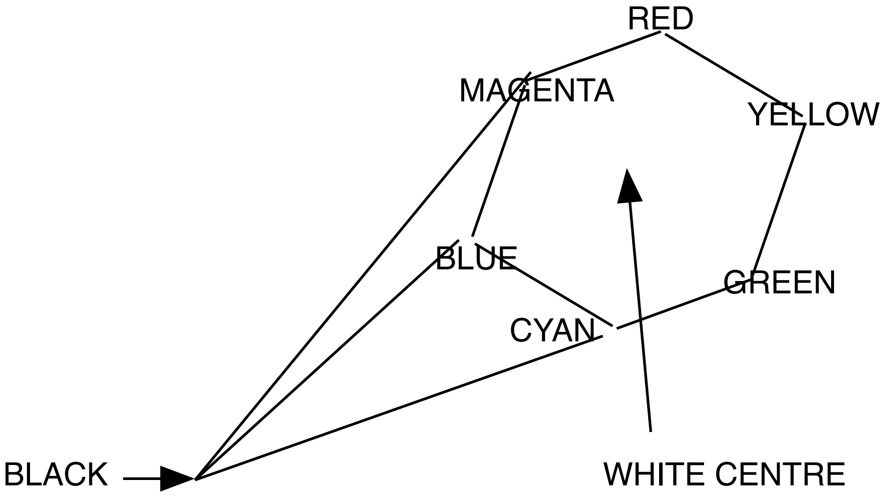


FIGURE 6. HSV-Hexcone

2.3.3 HLS-Double-Hexcone

The HLS (hue, lightness, and saturation) double-hexcone model is similar to the HSV hexcone although the projection is different. The lightness equation is (max(R,G,B) +min(R,G,B))/2. Here there are two sets of subcubes. One set includes subcubes grown from the black point and the other set includes subcubes shrunk from the white point. If the lightness is less than the maximum value of R, G, or B divided by two, then the subcube includes the black point otherwise the subcube includes the white point. Thus, the largest subcube that includes the white point and the largest subcube that includes the black point are both the entire colour cube. These subcubes are different from the subcubes of the HSV hexcone model such that the side length is twice that of the lightness value. Consequently, the lightness value of these constant lightness surfaces are the centres of these subcubes which fall along the achromatic axis. The surfaces that are projected onto a plane to form the double-hexcone are the six planes that connect the centre to six sets of radially-neighouring primary and secondary colour corners. When these subcubes are projected onto planes, they yield hexagonal discs. When they are layered, they form two hexcones back-to-back, with the black point and white point on either end [see Figure 7]. Here, the lightest red is white.

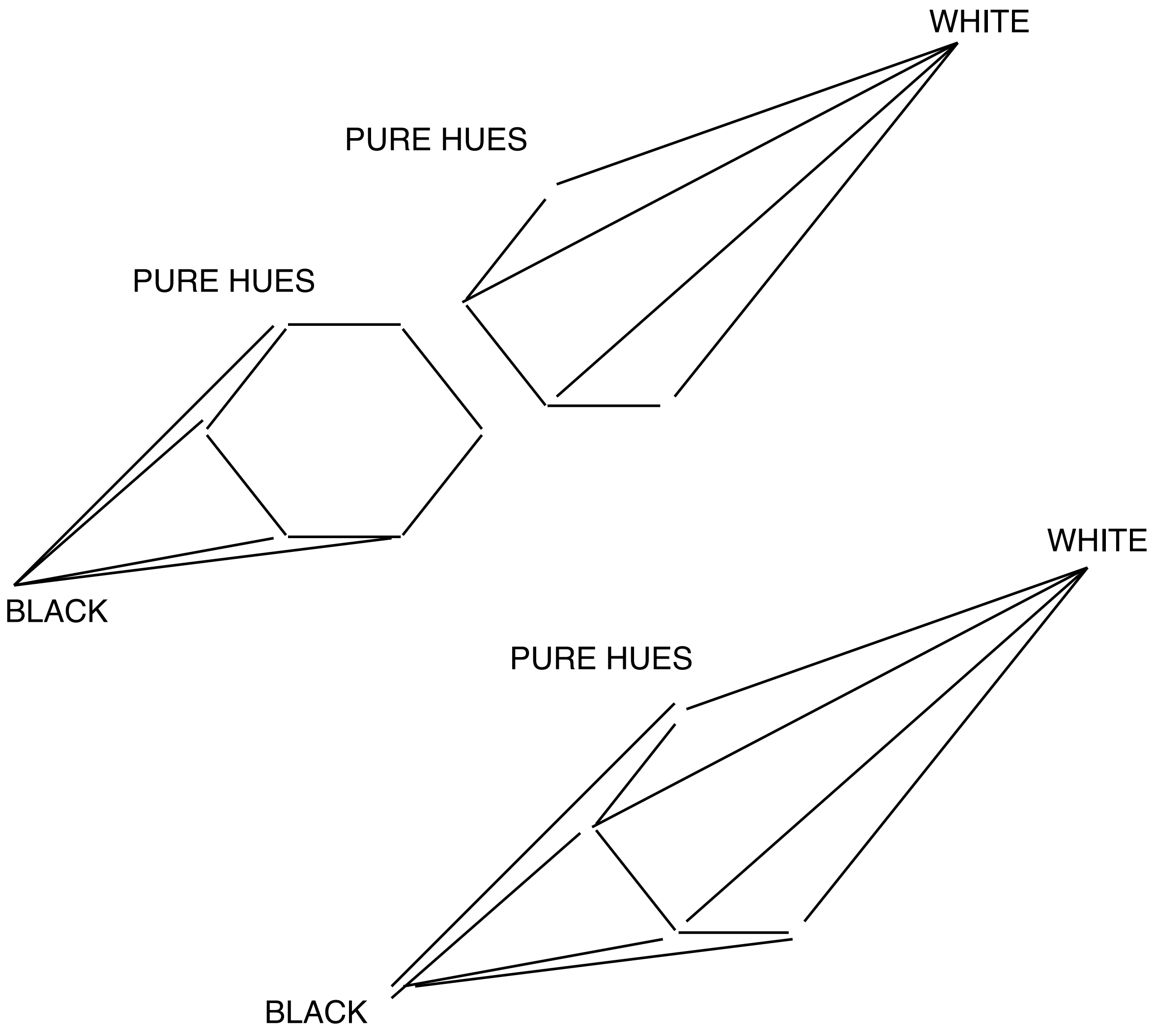


FIGURE 7. HLS-Double-Hexcone

2.3.4 Saturation

Hue and saturation are calculated independently of lightness and are the same for all three models. Since we have either hexagons or triangles, the size of which varies with lightness, and where we want a common scale 0 to 1 for saturation, saturation is determined to be the relative distance from the centre in any plane perpendicular to the achromatic axis. The numerator is the magnitude from the achromatic axis to the colour location [see A in Figure 8]. The denominator is the “radius”, that is, the magnitude from the achromatic axis to the surface of the cube (in the HSL triangle model), hexcone, or double-hexcone on the particular hexagonal or triangular slice [see B in Figure 8]. All points that lie along concentric hexagones or triangles within the hexagonal or triangular slice have the same saturation level.

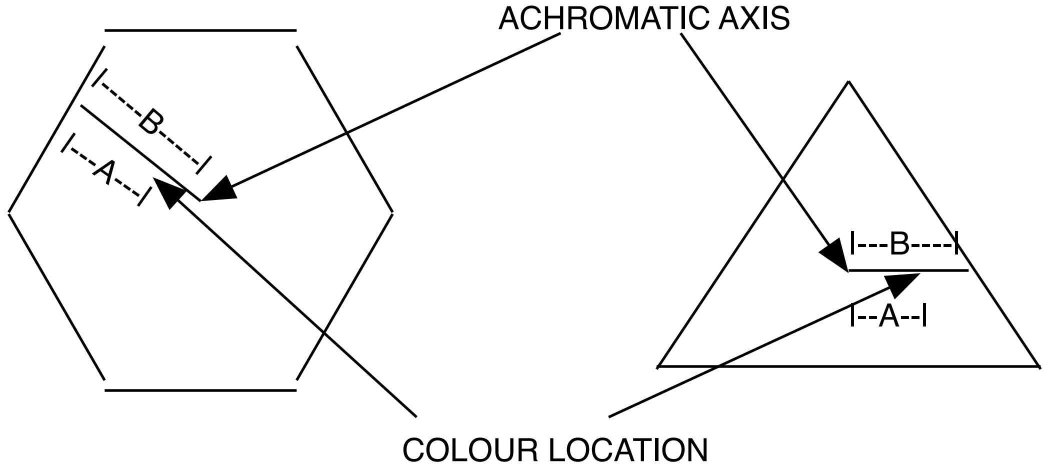


FIGURE 8. A Hexgonal Slice and a Triangular Slice

2.3.5 Hue

Hue is calculated in the following way. The full spectrum of hues divides the hexagons and triangles into six sectors according to the relative sizes of the values of red, green, and blue. For instance, a colour will be in sector zero if red is the maximum value, green is the middle value, and blue is the minimum value. A colour will be in sector five if the red is the maximum value, *blue* is the middle value, and green is the minimum value.

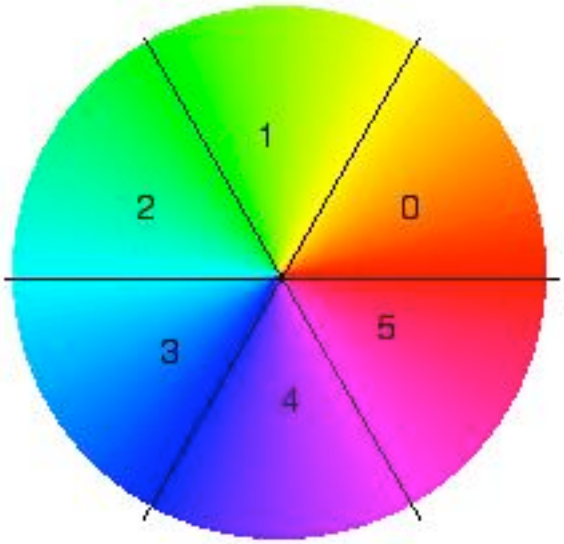


Figure 9. Hue Sectors

Figure 9 shows a circle divided into six sectors. The hexgons and triangles would be similarly divided. The sector number, however, does not specify the exact hue angle. The hue fraction describes how far into the sector the colour is located. Max, mid, and min correspond to the maximum, the median, and the minimum values of the red, green, and blue content of a particular colour.

hue-fraction = (mid-min) / (max-min) for even sectors

hue-fraction = (max-mid) / (max-min) for odd sectors

The hue angle is measured from pure red axis.

hue = (sector-number * 60^{o}) + hue-fraction

2.3.6 Uniform Colour Space

Using colour in graphics to represent values requires uniformity of colour space: the perceived distance between two colours must be equidistant at any two points. The Commision Internationale de l'Eclairage (CIE) system using the opponents color theory tried to do just that using empirical studies [see CIE 1971]. Levkowitz's intention was to seek out a uniform colour space from within the HSV family. Recall how the maximum, median, and minimum of the contents of red, green, and blue are weighted differently for each lightness calculation of the three models.

Model	Minimum	Median	Maximum
HSL-Triangle	1/3	1/3	1/3
HSV-Hexcone	0	0	1
HLS-Double-Hexcone	1/2	0	1/2

His algorithm allows for any possible HSV model that can be contrived from these three colour models by different weightings of the three levels of RGB in the lightness calculation. Call this the HSV space. He found that, although different weightings came closer to a perceptually uniform colour space, none actually succeeded. My implementation of Levkowitz's algorithm gives users the choice of these three models rather than his entire HSV space. It is easier for a user to pick one of these three models because they are more easily visualized from the colour cube.

3. Dimensioning with Colour

The HSV models are continuous and largely intuitive. They are not intuitively ordered with respect to hue. Saturation and lightness as uni-dimensions are naturally ordered; moving from dark to light and from gray to a pure hue is ordinal. Hues, although ordered by wavelength, our perception does not seem to order them. That is, they do not map naturally to ordered data. Red is perceived to be neither less than nor greater than say green. With some careful thought, some of us would be able to sort the spectrum into red, orange, yellow, green, blue, indigo, and violet or we might have just memorized the acronym ROY G BIV that Isaac Newton provided.

If our third variable is categorical, hue is a natural choice. As Rubin and Richards (1982) stated about the identification of material, the evolutionary purpose of hues may be a categorical concept. However, most images use a continuous variable not a discrete variable. The following paragraphs discuss the use of lightness and hue to represent the third variable of an image. Saturation can also be used, but it has neither the powerful ordinal quality of lightness nor the wideness of hue range.

Gray images, or other single hue images [see Figure 10] which vary by lightness are limited by the range of discernible image values. Varying the image values over hue [see Figure 11], however, allows for a much wider range. If each colour gun has 2^8

different values, then there can only be 2^8 different gray values.

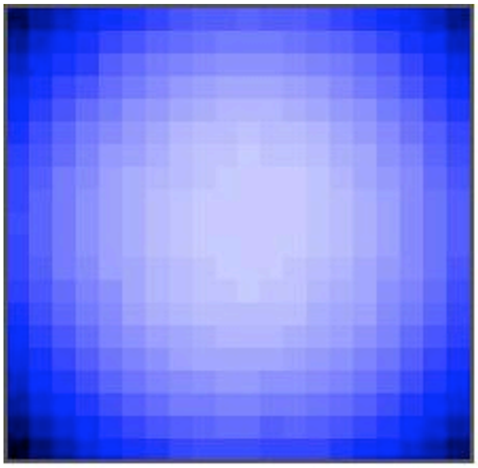


Figure 10. Image Values
Ranging by Lightness

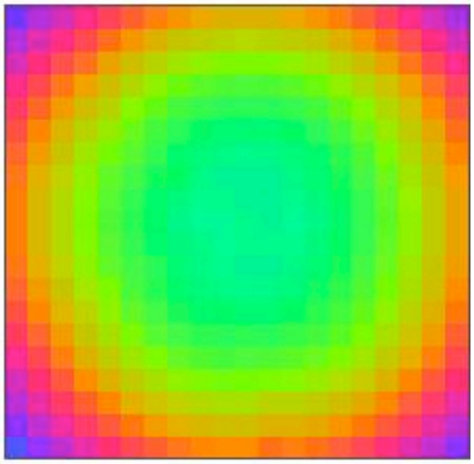


Figure 11. Image Values
Ranging By Hue

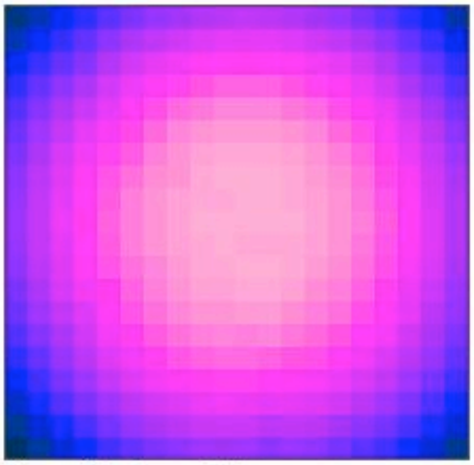


Figure 12. Image Values
Ranging by Hue and Lightness

If the lightness value is kept constant and if the lightness value is moderate, then the values of red, green, and blue can be permuted many more ways than 2^8 giving a much wider range. Recall that the high and low values of lightness impede discernment of hue. In any case, ordering by hue would be a difficult task for a user to do. Since hue is not intuitively ordinal, she or he would continually refer to the legend.

We can see, however, how small ranges of hue blend into one another. There is a natural order for magenta, purple, and blue, but interestingly no sense of one being bigger than the other. However, any three neighbouring colours from the visual spectrum are easy to order. For instance, thermal images order hues yellow, orange, and red quite naturally---yellow representing a cooler temperature and red representing a very hot temperature. Firstly, in this example, there is a sense of one being bigger than the other. This is because yellows are usually lighter than red so lightness brings magnitude. Secondly, there appear to be many psychological links between hue and physical concepts. Visible infra red is the closest visible electromagnetic energy to the infra-red portion of the spectrum which radiates thermal energies. Hot temperatures are therefore linked to redness.

While hues may extend the range of available distinct colours, unfortunately, the lack of ordering suggests that it is advantageous to use fewer hues or small ranges. Lightness can be used to emphasize the direction from one hue into another. [See Figure 12 for an image that combines lightness and hue variation into one dimension.] Since lightness is ordered, it will dominate any ordering of hue. Together, hue and lightness form a plane. Taking the north-east diagonal from the origin maps low lightness and low hue values to low image values and high lightness and hue values to high image values. For instance, if yellow represents the cooler hue with lower image values, making that end the lightest would enhance the idea that yellow represents lower temperature values. This technique achieves benefits from both scales: the ordinal dominance of the lightness scale, and the heightened contrast effect of the hue scale. This technique is also beneficial for people with colour blindness, since, while they cannot perceive the hue changes, they can still recognize the lightness changes.

Instead of using both lightness and hue as one dimension, one could use lightness and hue as a third and fourth dimension, respectively. Layering more and more dimensions onto one plot, however, makes it harder and harder to read [see Trombo 1981 for a discussion on colouring bivariate maps]. One can also add saturation as a fifth dimension, but as Levkowitz (1997) found, saturation is often confused with lightness in multiparametrized colour images.

Another limitation to using hues and lightness is that, although, they can be both ordinal and nominal, we cannot readily see by how much they are different. Using length and area, we can estimate that different point symbols are about half the size or twice the size, but when we use colour, both the background and the size of the coloured areas confound our assessments [Bertin 1983]. Nevertheless, by linking images to one dimensional dotplots, one can easily decipher any information contained within the image to the exact quantity.

4. Highlighting

Highlighting points on a scatterplot is best done with colour. Brighter colours draw more attention than changes in texture or shape. But with a colourful image, highlighting has many more difficulties. The following section describes some of these difficulties, a user-pick-hue-light-range function which incorporates control to overcome these difficulties, and four highlighting tools.

Choosing a single highlighting colour for an image would obscure the image. Choosing a different hue and maintaining the lightness and saturation values rather than a whole new colour, however, would allow one to see the highlighting and maintain the representation of the third variable of the image [see Figure 13].
Human beings, reportedly, have the best hue discrimination in the magenta range [MacAdam 1943]. Thus, the magenta colour is a natural choice for highlighting as well as colouring an image. However, if the hue of your image is defined then gray would be the obvious choice since it would clearly differentiate itself from the "hue-ful" (trade marked) image. Pastel colours would also serve; the higher lightness values would initially draw most of the viewer's attention.

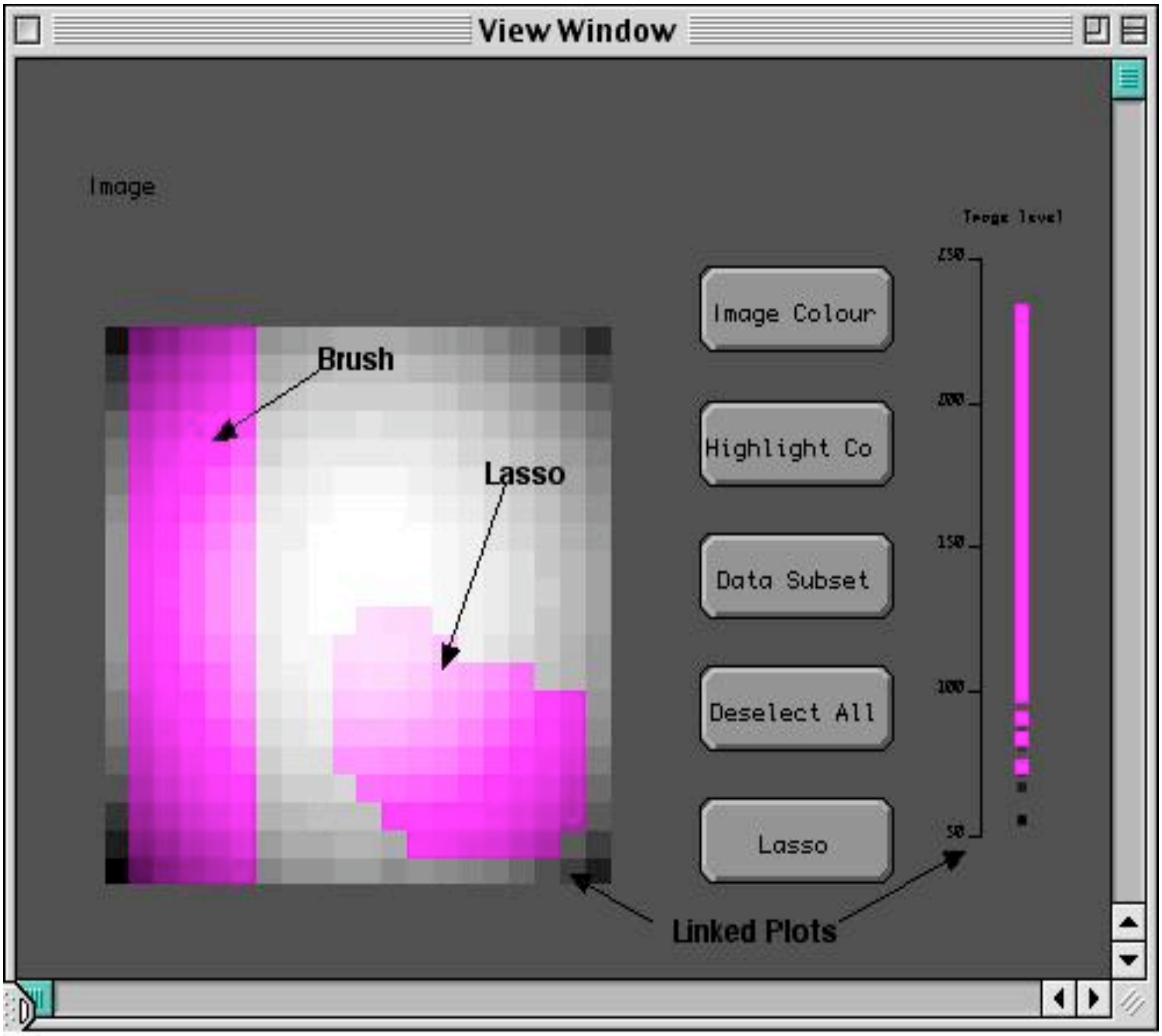


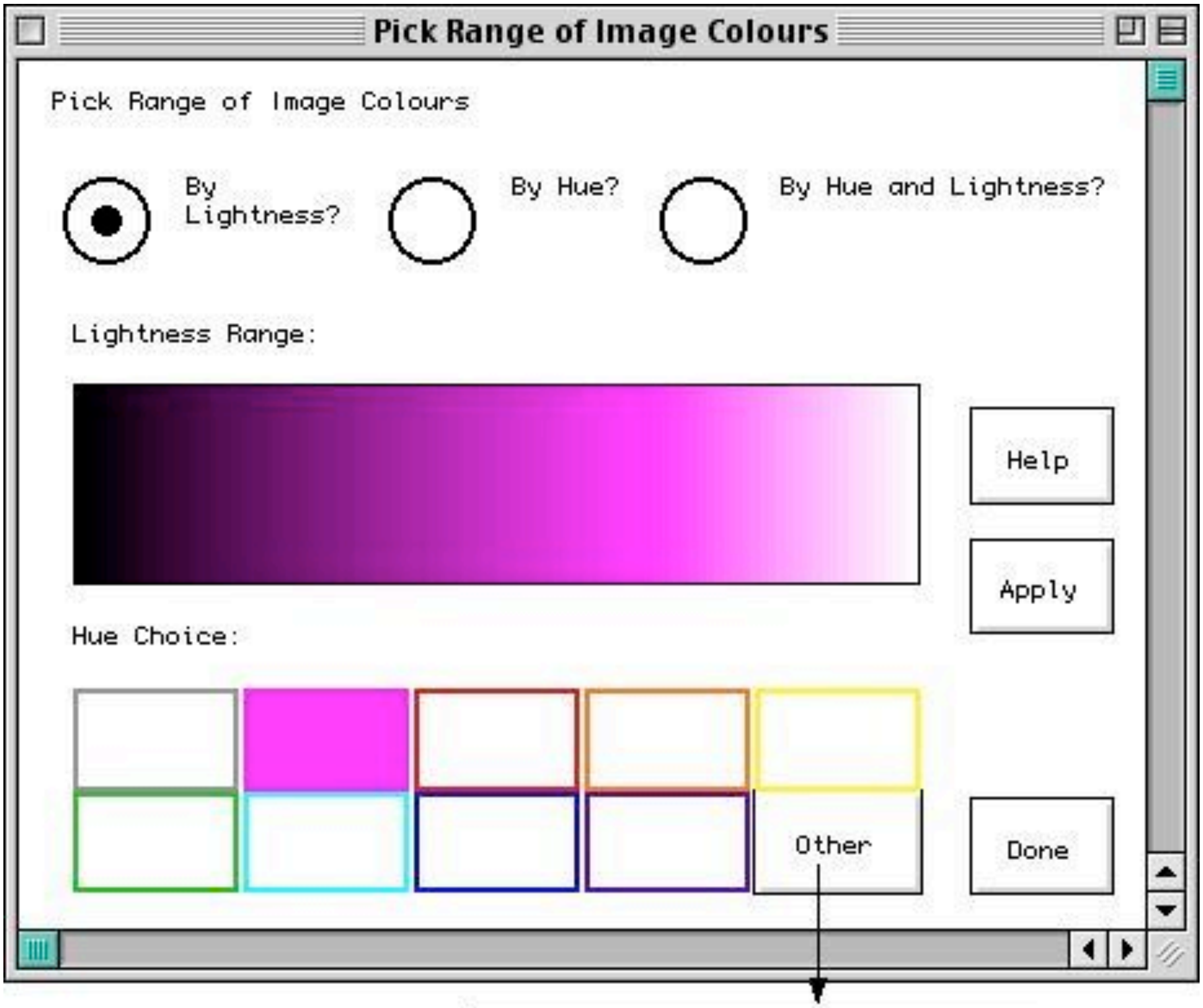
Figure 13. Highlighting

Another difficulty with highlighting is when the range of colour in the image includes white and black. For example, simply changing the hue would change a black coloured point symbol to the darkest colour for that hue, which would also be black. And depending on what model you are using to convert from HSV to RGB, the lightest magenta would either be white or the pure magenta colour. So both the choice of HSV model and the restriction of the range of lightness affect how point symbols would be highlighted on an image. So, if lightness is used as the third dimension, then the highlighting range should be bound away from the end points.

4.1 Implementation User-pick-hue-light-range

I have implemented a function which allows the user to select how colour and highlighting colour should be mapped onto the third dimension of an image. This function is also used to pick the range of colours for the third dimension of the image. The user is given three choices: by lightness, by hue, or by both hue and lightness.

Lightness: the horizontal slider allows the user to select any range of lightness. The hue chips allow only one of seven discrete choices. The “other” hue chip allows the user to select one hue from a seemingly continuous range of hues on the hue wheel. After any hue selection, the lightness slider immediately takes on the chosen hue. [See Figure 14.]



Pick Range of Image Colours

Pick Range of Image Colours

● By Lightness? ○ By Hue? ○ By Hue and Lightness?

Lightness Range:

Hue Choice:

Hue Wheel

Click on hue:

Figure 14. By Lightness?

Hue: the horizontal slider allows the user to select any range of hues. The vertical slider allows the user to select any one level of lightness. As the lightness slider moves up and down, the hue slider simultaneously takes on the level of lightness. [See Figure 15.]

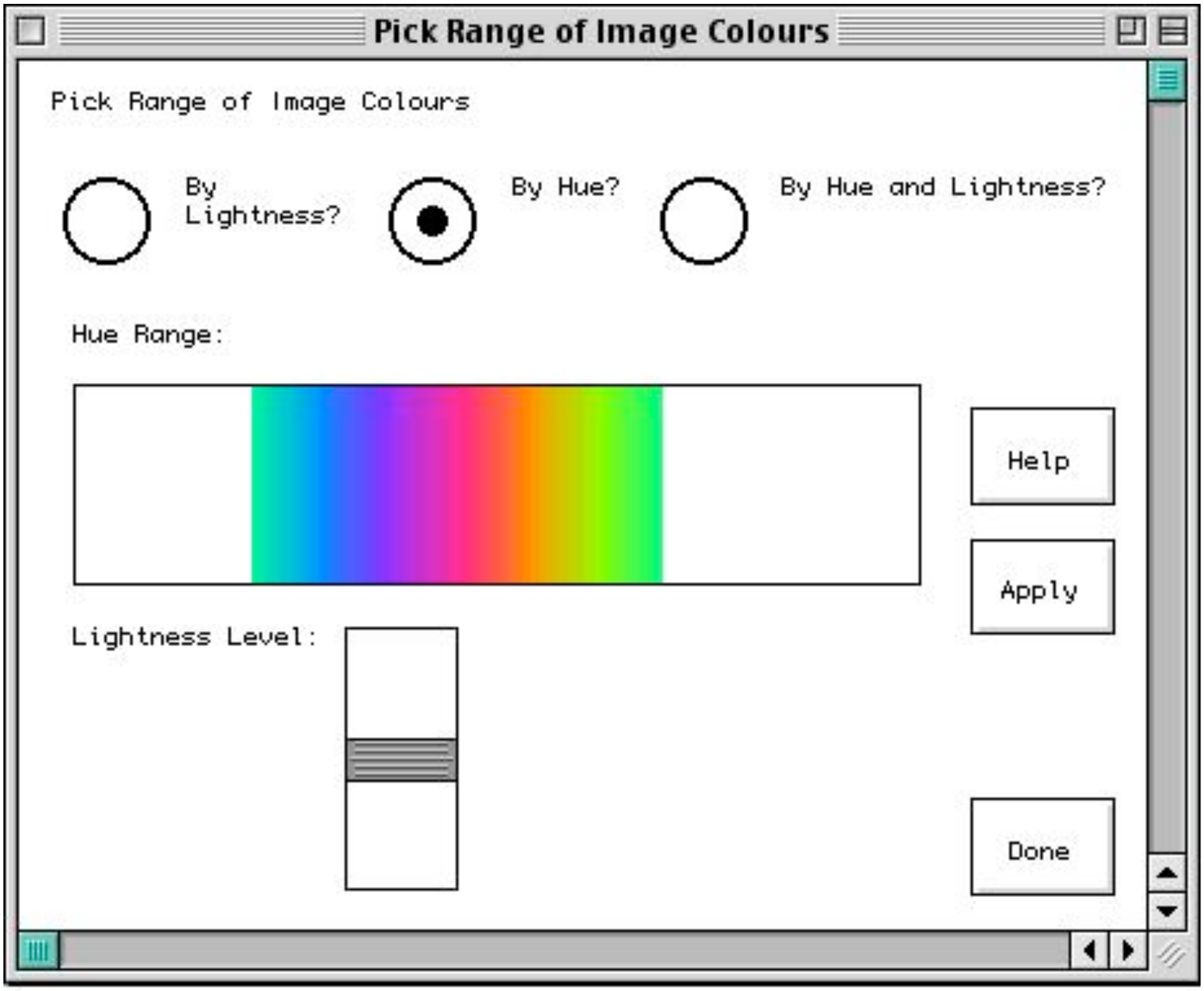
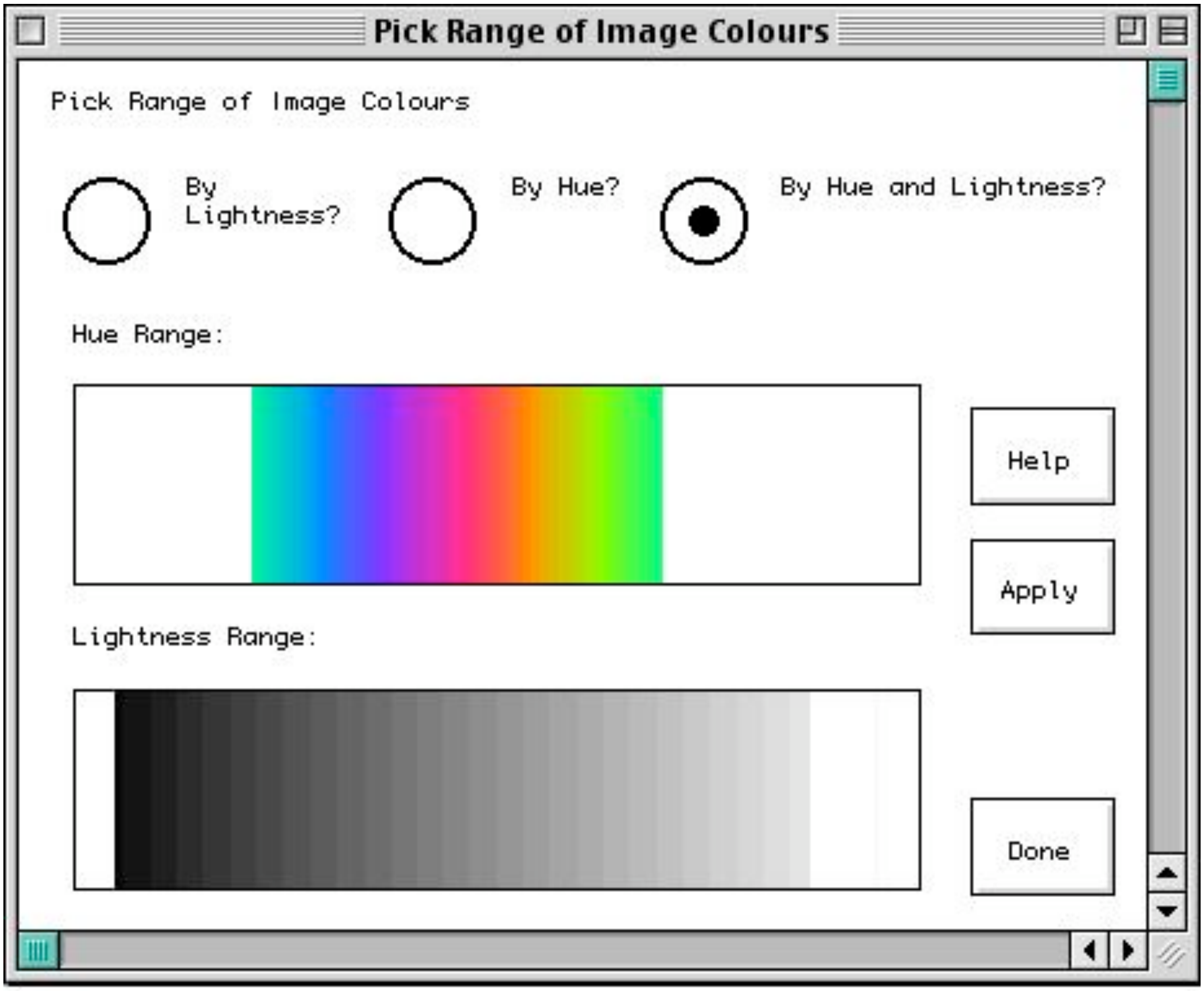


Figure 15. By Hue?

Both: there are two sliders, one for choosing a lightness range and one for choosing a hue range. [See Figure 16.]


Pick Range of Image Colours

Pick Range of Image Colours

○ By Lightness?
○ By Hue?
● By Hue and Lightness?

Figure 16. By Hue and Lightness?

When the “Both” selection is chosen, it is difficult to envision how the two choices, the ranges of hue and lightness, will manifest themselves until the choice is actually applied. Other limitations include the following:

- The saturation default is always set to one which gives the most vibrant possible colours. There is no current provision to allow the user to modify this. It might be especially useful to modfiy saturation for fixed hue and lightness.
- The model choice is always defaulted to the double hexcone model. The user should be able to pick which HSV model they would prefer.
- There is no choice to flip the range sliders. Darker lightness levels and the ultraviolet end of the spectrum always map to low image values. There should be a switch that flips the ranges of any of the sliders.

4.2 Highlighting Tools

In the image plot implementation, there are three different methods for highlighting: brushing, lassoing, and thresholding. The user can also change the colours of the highlighted areas, distinguishing these areas from the rest of the image plot.

4.2.1 Brushing

Brushes have a clearly defined shape, often rectangular. As the brush drags over the points on a graph, the brushed points are highlighted. The brush can either have lasting effects -- the points remain highlighted -- or transient effects -- the points are highlighted only when they are under the brush. If multiple plots are linked, then the corresponding points in the other plots are also highlighted. Buja et al. [1996] list three purposes for brushing: conditioning, sectioning, and data base querying. Conditioning investigates whether variables are dependent on other variables. Sectioning investigates the potential of reducing the dimensionality of the data. Database querying involves activities such as cluster searching and investigating influential outliers. [See Becker et al. 1988 for further discussions on brushing.]

Brushing irregularly shaped regions on an image is more difficult. You could make the brush very small and then drag the brush over every point symbol you want selected. This is effectively painting.

4.2.2 Lasso

A lasso device selects point symbols by enclosing them. The user clicks and drags the mouse, looping the area they want highlighted. As the point symbols are dragged over, each is highlighted. Once enclosed all the point symbols within the lasso will also be highlighted.

The algorithm highlights point symbols that have at least one point symbol highlighted above and below it, and to left and right of it. The algorithm is:

```
lasso-region = smallest rectangle that includes all the point symbols inside the lasso
loop for point-symbol in lasso-region
     if there is a “lassoed” point symbol above point-symbol
    and
     if there is a “lassoed” point symbol below point-symbol
    and
     if there is a “lassoed” point symbol to the left of point-symbol
    and
     if there is a “lassoed” point symbol to the right of point-symbol
     then highlight point-symbol
```

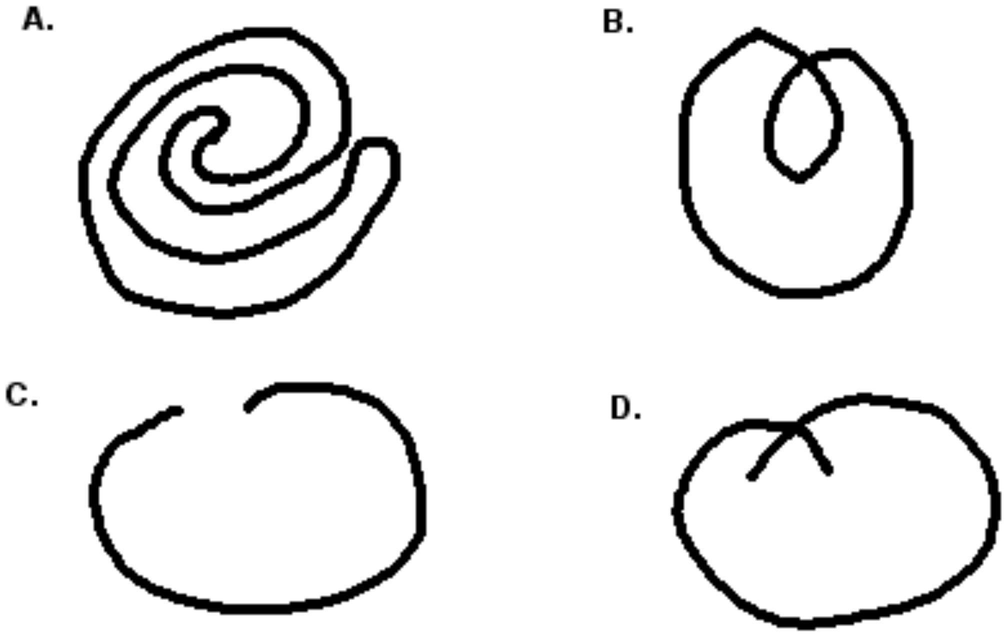


Figure 17. Irregular Lassos

This algorithm would fail for many irregular shapes, such as the three in Figure 17. Lasso A folds around itself and lasso B crosses over itself. Lasso C and D do not close themselves. If we highlighted only point symbols that have an odd number of point symbols highlighted above and below it, and to left and right of it, then lasso A and B would highlight point symbols within itself correctly, but lassos C and D would still fail.

In addition, if the lasso is dragged too quickly over some point symbols many of them would fail to highlight, especially with a high-resolution image. The lasso function should find those broken spaces in the lasso and highlight the fewest number of point symbols that would close the lasso. This includes the two ends of the lasso, where the user may not have closed the lasso. This would improve lasso C and D.

4.2.3 Thresholding

Traditionally, thresholding divides an image on a chosen intensity value. A portion of the point symbols in the image would be one colour and the remaining portion another colour. Here, the user can highlight a portion of the image and leave the other portion unchanged. The user would choose a dividing image value and either the minimum or maximum value for the range of highlighting.

In this implementation, thresholding works via two plots: the image plot and a uni-variate dotplot. By brushing sections of the dotplot, one can choose the range of image values they want selected in the image. The capabilities of thresholding are naturally extended using a brush and a dotplot in this way. Instead of dividing the image by only one value, you can select as many different ranges as you want [see Figure 18]. For instance, if you suspected that all water bodies were within a certain range, you could select that range of image values and have it highlighted.

4.2.4 Colour Changes

In addition, once you have highlighted a group of image points, you can change their image colours [see Figure 18]. The range of colour depends on whether the colour of the image points range by lightness, by hue, or by some combination of the two. The user has the option of choosing a hue --- the selected-hue. If the image colours range by lightness, then the permanent colour values also range by lightness but they would display the selected-hue. If the image colours range by hue, then the selected-hue would be employed as the mid-hue used for the range of hue in the colour change. If the image colours ranged by a combination of hue and lightness, then a combination of hue and lightness values would be used in the colour change.

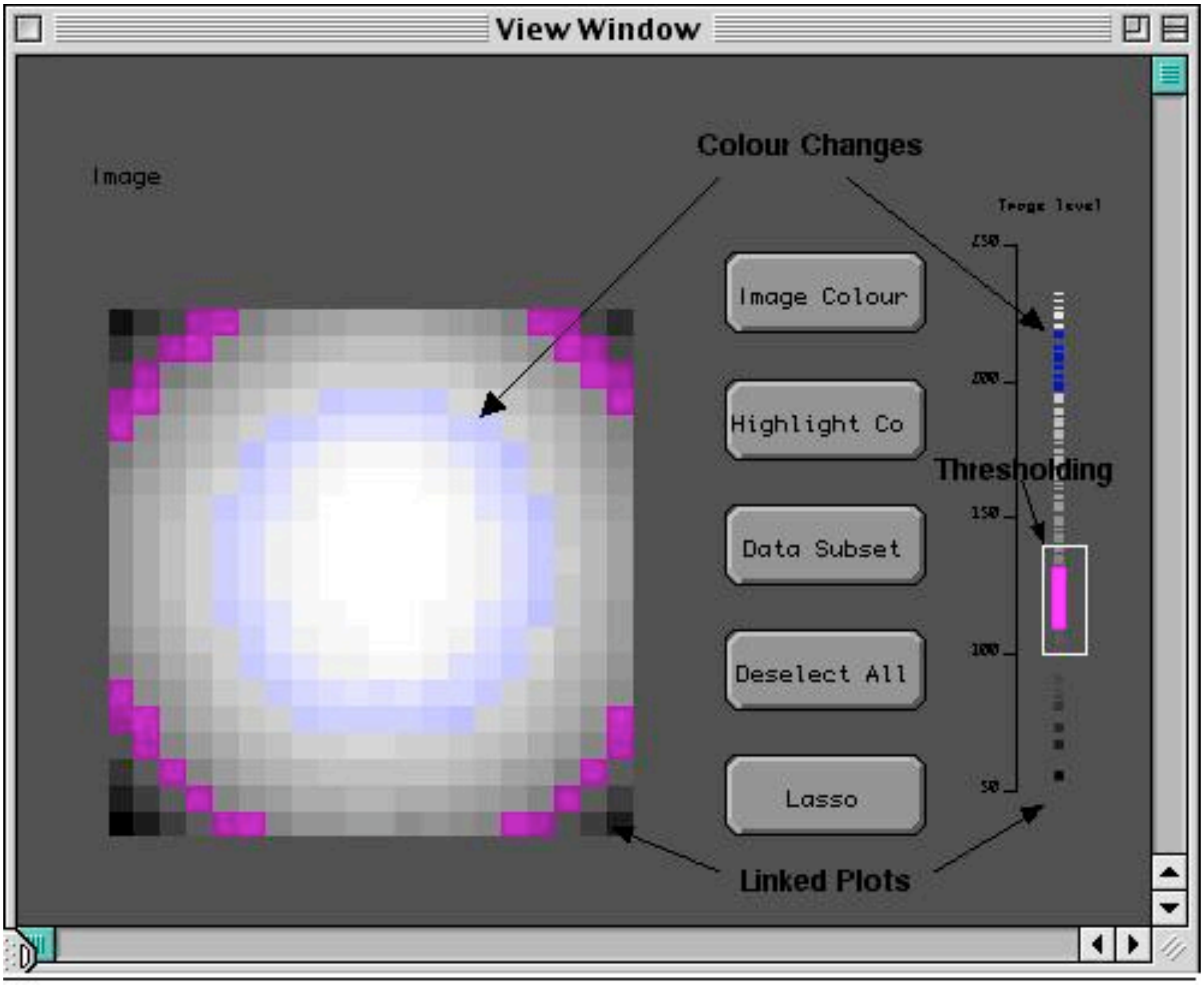


Figure 18. Thresholding and Colour Changes

5. Image Implementation

This section reviews some of the remaining issues involved in the implementation of this image interface.

Quail is both a windowed and command line environment. There are fold-down menus and typed commands to be issued in a window called the Listener. Graphics are displayed in view windows. Data objects are displayed within view windows in viewports. Viewports are rectangular regions within the view window.

5.1 2d-Image

An image is similar to a two-dimensional point cloud in Quail except for its denseness. Each coordinate has an existing point within the plot. The image class inherits most of the properties that a two-dimensional point cloud would have [see Figure 19, the class tree of a 2d-image and a 2d-point-cloud], such as: a coordinates cache, data set attributes, functions and transformations on both location variables, brushing capabilities, and unique information on each case such as colour, size, and shape. The image class also has unique properties such as: the number of dimensions, a structural image matrix of the cases, and a highlight colour and colour map to translate image values into colour. The colour maps consists of: a single or two-valued range of lightness and hue values, a saturation value, the minimum and maximum image values, and the HSV model used in the conversion from HSV to RGB. These maps are used to translate the image values into colours in the selected colour range. Number of

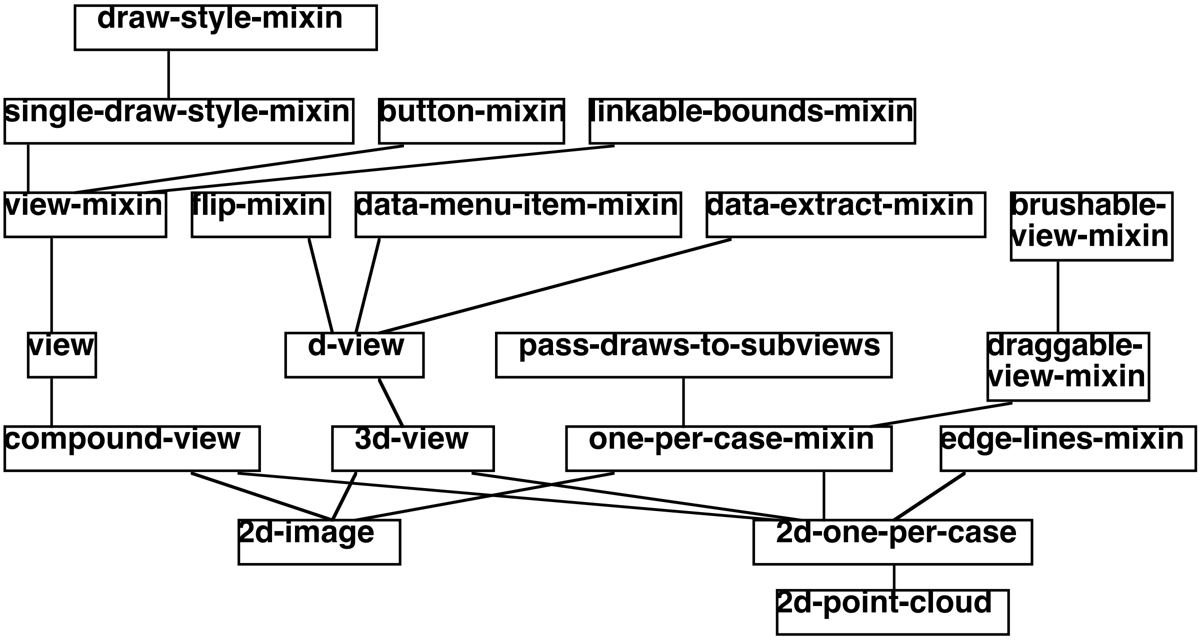


FIGURE 19. Class Structure of a 2d-image and a 2d-point-cloud

dimensions and image matrix are meant for functions that utilize the spatial characteristics of plots, such as for highlighting regionally. For instance, the image matrix organizes the pointers of the imagels [see Imagel section] into an X-Y grid, otherwise, the imagels are accessed from a collective list in no particular order. The lasso function can access rectangular regions from the image plot directly.

5.2 Imagel

The implemented image consists not of pixels nor of point symbols but of what I call imagels. Recall, pixels are discrete units that the screen of the monitor displays. Point symbols in Quail represent cases in plots such as scatterplots and dotplots. Point symbols always inhabit a square viewport. Imagels are discrete image elements. They are not pixels because they can encompass more than a single pixel. They are not point symbols because imagels are limited to the shape that they take and because they are not necessarily square. Imagels acquire the same aspect as the viewport of the entire image. When the view window is made smaller or larger, wider or narrower, the aspect of the imagel changes accordingly. Imagels are restricted to being rectangular. In the future, they might also be triangles or trapezoids. Point symbols can be text, stars, circles, crosses, diamonds, etc. Thus, with the square viewport restriction on the point symbol and the symbol restriction on the imagel, point symbols are neither superclasses nor subclasses of the imagel [see Figure 20, the class structure of imagels and point symbols].

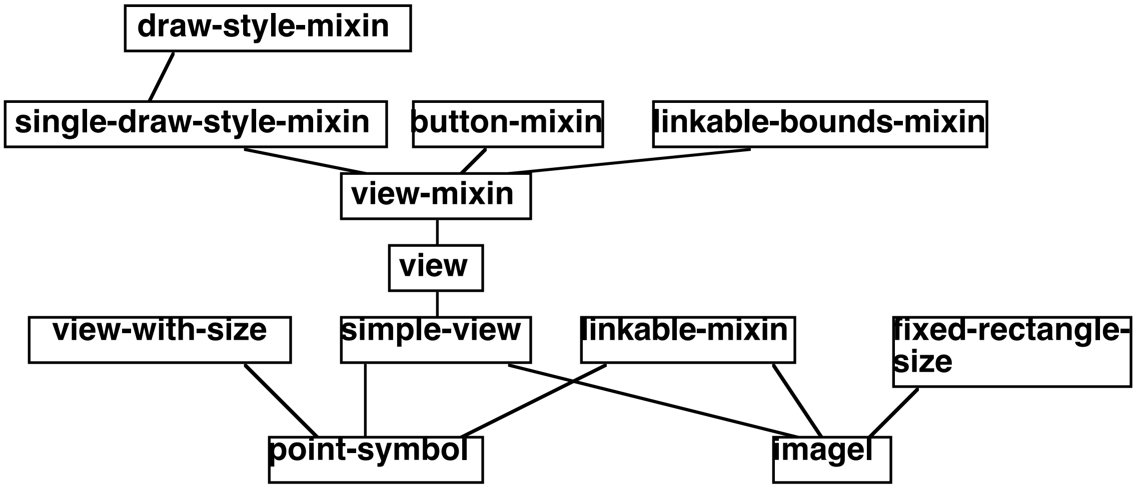


FIGURE 20. Class Structure of an Imagel and a Point-Symbol

Many of the properties acquired at the time of creating the 2d-image object are transferred to the imagels including the colour, the highlight colour, and the size (both width and height) which is dependent on the size of the viewport.

5.3 Image Interface

Within view windows, control is issued through menus, buttons, and sliders. In the 2d-image plot, there are three venues to control explorations: the imagel menus, the image menu, and the image plot buttons. Thus, there are menus for the image and menus for the imagels as well as graphic buttons for the image display.

5.3.1 Image Menu

The image menu [see Figure 21] has the following features:

1. Brush - Highlights rectangular regions of the image. The brush is dragged over regions. The brush has a resizing option and an angling option.

2. Colour - Changes the colour of the highlighted region namely by hue [see Colour Changes section], or, if no imagels are selected, it changes the colour of the entire image.

2. Toggle highlight - Toggles between highlighting the entire image and de-highlighting the entire image.

3. Invisible? - Makes the selected imagels invisible, or, if no imagels are selected, it makes the entire image invisible.

4. All Visible? - Makes the entire image visible.

5. Variables - Gives the user the option of changing the X, Y and Z (i.e., the image values) variable to some other variable from the same data set. Also gives the user the option of transforming the X,Y, and Z variable by taking the square, the identity, the square root, the log, or the inverse of the X or Y variable.

6. Flip-x - Reverses the X axis.

7. Flip-y - Reverses the Y axis.

All image menu items are options available to all two-dimensional point clouds.

5.3.2 Imagel Menu

The menu [see Figure 22] options for imagels are as follows:

1. Highlight? - Highlights a single imagel.
2. Colour - Changes the colour of a single imagel namely by hue [see Colour Changes section].
3. Invisible? - Toggles the selected imagel between invisibility and visibility.
4. Width - Narrows or widens the width of the imagel.
5. Height - Shortens or lengthens the height of the imagel.

The imagel menu is similar to the point symbol menu except that the point symbol menu has an size option in place of a width and height option.

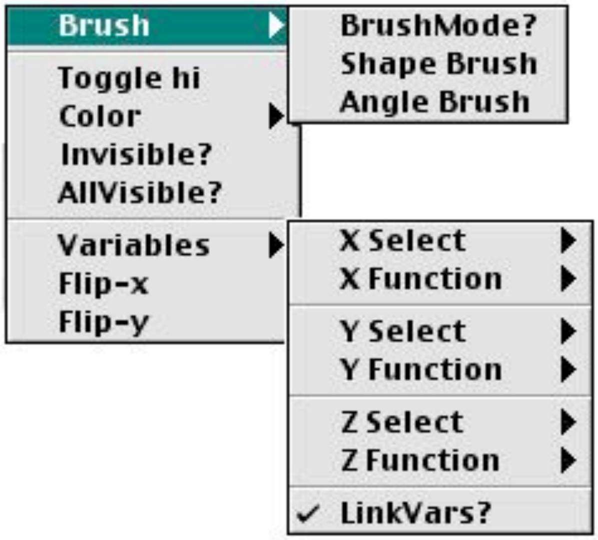


Figure 21. Image Menu

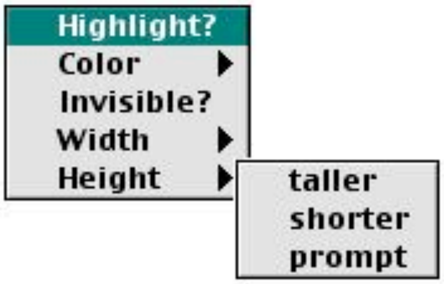


Figure 22. ImageI Menu

Figure 23. Image Plot Buttons

5.3.3 Image-Plot Buttons

The image display has the following buttons [see Figure 23]:

1. Image Colours - Allows the user to choose the range of colours according to lightness or hue or a combination of lightness and hue.

2. Highlight Colours - Allows the user to choose the range of highlight colours according to lightness or hue or a combination of lightness and hue.

3. Data Subset - Creates a data-set object with only the selected imagels. This object can then be viewed separately without the entire data set, although the cases will remain linked with the cases from the entire data set.

4. Deselect All - Deselects all highlighted imagels.

5. Lasso - With the mouse, the user can select simple closed shapes to be highlighted [see Figure 13].

5.4 Possible Extensions

In addition to correcting the limitations described earlier, the possibility of adding more dimensioning capabilities to a 2d-image is a potential area for improvement. Brushing and linking are preferable to adding another layer of information onto the plot when exploring data sets. With static displays or print there can be no dynamic interaction between the user and the graph. Brushing and linking, therefore, are not options, however, using colour for more than one dimension, should still be an option. The user can use both hues and lightness and perhaps also saturation as a dimension. The user can also represent data by lightness level and by two different hues. Where hues overlap, the user could use an additive colour mix such as where reds and blues make purples. Ultimately, the user must decide what is and what is not easy to comprehend.

Other image packages come with smoothing and enhancement options. Smoothing enhances visibility. It is sometimes easier to see what is there if the picture is less sensitive to small image value variations. Conversely, sometimes it is easier to find what you are looking for if you make the image more sensitive to changes in image values, such as for applications in edge detection. Restricting and enlarging the range of image colours can meet these two demands in a small way, but often you would like the neighbouring image values to have an impact on the image values. Kernalling addresses the variability issues of smoothing and enhancement thoroughly.

6. Summary

Colour perception can be linked to two colour systems: RGB and HSV. RGB depicts how we physically see colour. Cones cells respond at different ranges of wavelength. recording the amount of light at particular wavelengths. So amounts of red, green, and blue (the named cone cell types) combine to produce colour. Computer monitors use the RGB system as do most people-made sources emitting light.

The HSV systems depicts how we understand colour. Humans can intuitively define colour in terms of hue, saturation, and lightness. The multiparametric image interface in Quail converts between the two systems to create a user-friendly interface.

Image plots in Quail parametrize colour either by lightness, by hue, or by a combination of the two to represent the image values. Lightness has an ordinal quality. Hue is not ordinal for large ranges, however, we propose that hue is ordinal for small ranges.

Colour is also used to highlight aspects of the image. We recommend that if the user plots image values by lightness that highlighting be done by just changing the hue of the imagel colours. If the image values range by hues, then either use grays or pastel colours, that is, lighten the imagel colours.

References

Andrews DF. Plots of High-Dimensional Data. Biometrics 1972; 28:125-136

Asimov D. The Grand Tour: A Tool for Viewing Multidimensional Data. SIAM Journal of Scientific and Statistical Computing 1985; 6:128-143

Becker RA, Cleveland WS. “Brushing Scatterplots” Dynamic Graphics for Statistics. 1988. Wadsworth & Brooks: Belmont, California

Bertin J. Semiology of Graphics. 1983. The University of Wisconsin Press, Ltd: Madison, Wisconsin

Boynton RM. Human Color Vision. 1979. Holt, Rinehort and Winston: Toronto

Buja A, Cook D, Swayne DF. Interactive High-Dimensional Data Visualization. Journal of Computational and Graphical Statistics 1996; 5:78-99

Chernoff H. Using Faces to Represent Points in K-Dimensional Space Graphically. Journal of the American Statistical Association. 1973; 63:361-368

CIE. Recommendations on Uniform Color Spaces, Color-Difference Equations, Psychometric Color Terms, Supplement No. 2 of CIE Publication No.15 (E-1.3.1). 1971. Bureau Central de la CIE: Paris

Levkowitz H. Color Theory and Modeling for Computer Graphics, Visualization, and Multimedia Applications. 1997. Kluwer Academic Publishers: Boston

MacAdam DL. Specification of Small Chromaticity Differences. Journal of the Optical Society of America 1943; 33:18-26

Oldford RW. “The Quail Project: A Current Overview” Computer Science and Statistics, Proceedings of the 30th Interface Symposium. 1998. Springer-Verlag: New York.

Rubin JM. Richards WA. Color Vision and Image Intensities: When Are Changes Material? Biological Cybernetics 1982; 45:215-226

Trombo BE. A Theory for Coloring Bivariate Statistical Maps. The American Statistician. 1981; 35:220-226

Wyszecki G, Stiles WS. Color Science: Concepts and Methods, Quantitative Data and Forumulae, 2nd edition. 1982. John Wiley & Sons, Inc: Toronto