Image Display in Teaching Image Processing
Part I: Monochrome Images

H. J. Trussell

Dept. of Electrical and Computer Engineering
North Carolina State University
Raleigh, NC 27695-7911

M. J. Vrhel

Color Savvy Systems Ltd.
305 South Main Street
Springboro, OH 45066

Abstract

The concentration of most image processing courses is on the various algorithms used to alter an input image or data to produce a desired output image. Little effort is expended on teaching the appropriate display methods that are required to accurately demonstrate the effectiveness of the processing algorithms under investigation. This paper reviews the fundamentals of image display that every student should know before displaying the homework and project results that are assigned in those courses.

1 Introduction

Every student who takes an image processing course looks forward to processing an image. To generate an image from numerical data is a large step up from generating graphs as has been done for many previous courses. With the aid of MATLAB and similar software packages, display of image data is easy. It is so easy, in fact, that most students and many instructors forget to take the time to do it right. This paper reviews some of the fundamental rules for "getting it right" for monochrome images. A subsequent paper will address the more complicated case of color images.

Let us consider some of the basic questions that should be asked in order to display image data appropriately.

1. Is the two-dimensional data an image in the pictorial sense? If it is a 2-D function, what are the attributes that need to be displayed for the observer?
2. Does the image have common visual meaning? For example, tomographic reconstructions never existed as visual images, so the proper display method is not obvious.
3. Is there is set of images that are to be compared?
4. Is there an original image that is to be reproduced? If so, what is the physical basis for the image? Reflection? Density? Radiance?

The answers to these questions will give the image processor a good start on determining the appropriate method to display the images. First let us consider the differences between displaying 1-D and 2-D data and the various methods of displaying 2-D data.

2 Mathematical Functions

One-dimensional functions are easily displayed on a graph where the scaling is obvious. The value of points along the axes are usually displayed adjacent to the points. The observer need only examine the numbers that label the axes to determine the scale of the graph and get a mental picture of the function.

With two-dimensional scalar-valued functions, the display is more complicated since there are more degrees of freedom in the function than available on the two-dimensional display. There are many ways of displaying such functions. There are advantages and disadvantages to all of them. The point here is to emphasize the scaling of the displays and the representation of the value of the function.

The three most common two-dimensional representations are the isometric plot, the contour plot and the gray scale plot. All are supported by MATLAB [1]. The user should choose the right display for the information to be conveyed. Let us consider each of the three display modalities. As a simple example, consider the two-dimensional functional form

$$f(m,n) = sinc(\frac{m^2}{a^2} + \frac{n^2}{b^2})$$

where, for the following plots, $a = 1$ and $b = 2$. This form is chosen to show both positive and negative values, something that occurs even when processing nonnegative signals such as images.

2.1 Isometric Plots

The isometric or surface plots give the appearance of a three-dimensional drawing. The surface can be represented as a wire mesh or as a shaded solid, as in Fig. 1. In both cases, portions of the function will be obscured by other portions. Therefore, this form works well with functions where the unseen part can be predicted, as with symmetric or very smooth functions. Typical examples include point spread functions and filters in the space or frequency domains. An advantage of the surface plot is that it gives some indication of the values of the function since a scale

0-7803-6293-4/00/$10.00

is readily displayed on the axes. However, it is difficult to read values accurately, or even approximately, from the graph. It is rarely effective for the display of images.

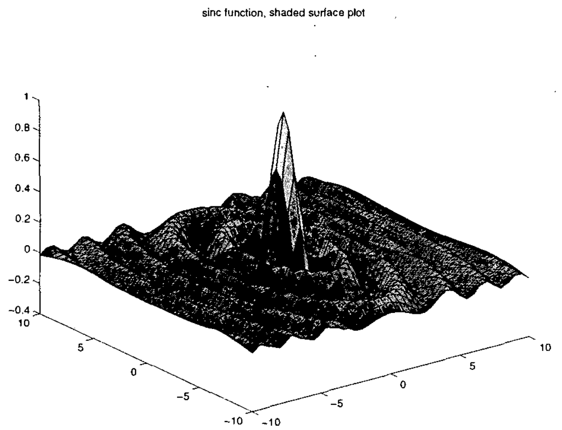

Figure 1: Shaded Surface Plot

2.2 Contour Plots

Contour plots are analogous to the contour or topographic maps used to describe geographical locations. All points that have a specific value are connected to form a continuous line. For a continuous function, the lines must form closed loops. This type of plot is useful in locating the relative position of maxima or minima in images or two-dimensional functions. It is used primarily in spectrum analysis and pattern recognition applications. It is difficult to read values from the contour plot and takes some effort to determine whether the functional trend is up or down. The filled contour plot, shown in Fig. 2, helps in this last task. The relative gradient of the function can be qualitatively observed by the closeness of the contour lines. The steep gradient near the origin can be seen in Fig. 2. Again, it is not possible to obtain very accurate values from the graph.

2.3 Gray-scale Graphs

Most monochrome images are displayed using the gray scale plot where the value of a pixel is represented by its relative lightness. Since in most cases, high values are displayed as light and low values are displayed as dark, it is easy to determine functional trends or relative shading. It is almost impossible to determine exact values. For images, which are nonnegative functions, the display is natural; but for functions that have negative values, this type of display can be quite artificial.

In order to use this type of display with images or functions, the representation must be scaled to fit in the range of displayable gray levels. This is most often done using a min/max scaling, where the values are linearly mapped such that the minimum value appears as black and the maximum value appears as white. This method was used for the sinc function shown in Fig. 3. A scale of gray level values can be added to aid the observer in estimating actual pixel values. The gray scale will be discussed in more detail later. For the display of functions, the min/max scaling can be effective to indicate trends in the behavior. Scaling for images is another matter.

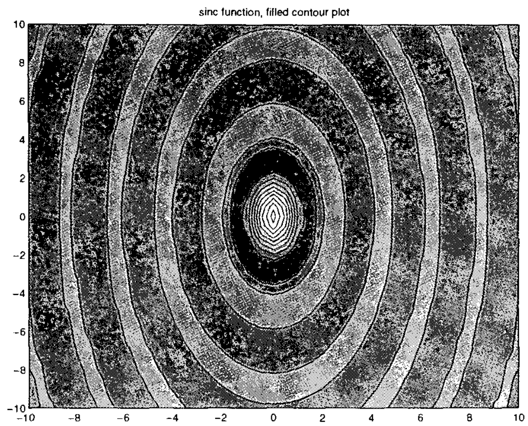

Figure 2: Filled Contour Plot

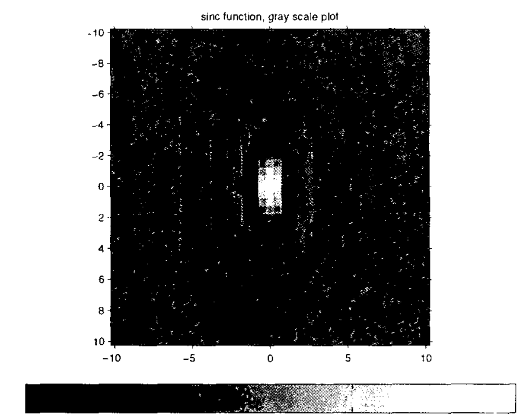

Figure 3: Gray-scale Plot

3 Image Display and Comparisons

Let's consider a monochrome image that has been digitized by some device, e.g. a scanner or camera. The values of the image function represent some physical quantity such as lumens or optical density. Without knowing the physical process of digitization, it is impossible to determine the best or most accurate way to display the image. The proper display of images requires calibration of both the input and output devices. However, for most classwork, original images are obtained as simple matrices of values, usually in the range [0,255]. These images have no reference to a physical quantity. We will address the accurate display of physically meaningful images later. At this point, it is reasonable to give some general rules about the display of monochrome images.

1. For the comparison of a sequence of images (comparative display) , it is *imperative* that all images be displayed using the same scaling.

2. Display a step-wedge, a strip of sequential gray levels from minimum to maximum values, with the image to show how the image gray levels are mapped to brightness or density.

3. Use a graytone mapping that allows a wide range of gray levels to be visually distinguished.

3.1 Comparative Display

It is hard to emphasize the rule of comparative display sufficiently and hard to count all the misleading results that have occurred when it has been ignored. The most common violation of this rule occurs when comparing an original and processed image. The user scales both images independently using min/max scaling. Min/max scaling is done by finding the minimum and maximum of the image to be displayed and mapping those values linearly to the minimum and maximum values of the display device. In many cases, the scaling can produce significant enhancement of low contrast images, which can be mistaken for improvements produced by an algorithm under investigation.

For example, consider an algorithm designed to reduce noise. The noisy image is modeled by

$$\mathbf{g} = \mathbf{f} + \mathbf{n}.$$

Since the noise is both positive and negative, the noisy image, **g**, has a larger range than the clean image, **f**. To obtain an accurate impression of the effect of the noise, both **g** and **f** should be displayed using the same scaling. In considering the effects of processing, it is noted that almost any noise reduction method will reduce the range of the processed image, thus, the smoothed (processed) image undergoes additional contrast enhancement if min/max scaling is used. The result is greater apparent dynamic range and a better looking image.

Examples of the effects of this rule are shown in figures 4 - 6. The original image is shown in Fig. 4. The range of this image is [30,220], on a scale of [0, 255]. The image can be enhanced by using min/max scaling as shown in Fig. 5. If the only reason for the display was to discuss features of the image, this might be a suitable display. However, if the reasons for display include a comparison with processed images, then this display can produce misleading images. Noise is added to the original and shown in Fig. 6. The noise has been created to produce an image with a maximum range of [0,255]. Thus, this image will not change appearance with min/max scaling and represents a valid comparison with the original, Fig. 4. A comparison of Fig. 5 and 6 would give a misleading impression of the effect of the noise. Note the easily observable differences in the average values of very dark and very light regions of the images.

There are several ways to implement the comparative display rule. The most appropriate way will depend on the application. The scaling may be done using the min/max of the collection of all images to be compared. In some cases, it is appropriate to truncate values at the limits of the display, rather than force the entire range into the range of the display. This is particularly true of images containing a few outliers. It may also be advantageous to reduce the display of the image to a particular region of interest, which will usually reduce the range to be reproduced.

3.2 Gray Scale Inclusion

The inclusion of a gray scale allows some idea of the quantitative values associated with the pixels. The comparison of gray scales of processed and unprocessed images gives an indication of the effect of any scalings that have been done for the display. A gray scale has been included in the original image of Fig. 4. This scale shows values in 32 equally spaced intervals. Note that the gray scale that is used for showing the mapping of the image values to the display should not be included in any computation of the minimum and maximum of the the image for display purposes. In this image, the gray scale range is [15,255], which includes values below and above the range of the image. The MATLAB software provides a command (colormap) that generates a gray scale, which denotes the numerical values of the gray levels.

In addition to indicating where the range of the image fits within the range of the display, the inclusion of a gray scale can demonstrate the effect of any mappings used to display the image. Consider the min/max scaling used to display Fig. 5. The figure shows the gray scale of the display mapping, i.e. the gray levels [15,255] on the bottom. This scale agrees with the MATLAB scale. The result of min/max scaling is shown by the top scale. From a comparison of the two scales, it is easy to see how the range of the original has been increased. Note that there are several gray values at the extremes of the scaled version (top) that are indistinguishable after the scaling.

4 Accurate Display of Physical Images

For the case where the image data represents a known physical quantity that is related to the visual system, it is important to display the image as it should appear to an observer. This requires profiling the display device. If the image is obtained from a scanner or camera that stores the data in values that are unique to that device, the profile on the input device must be known. Otherwise, the image data is assumed to represent some meaningful visual quantity such as CIE luminance.

A device profile is basically a set of mappings from the device dependent values used internally to a device independent (DI) image representation space that

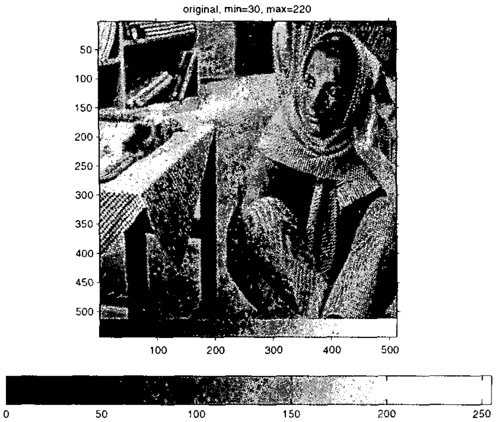

Figure 4: Original

is related to the visual system. For monochrome images, luminance or density are the meaningful quantities. In order to obtain the mappings to these DI spaces, samples must be measured that give values in the DI spaces. This requires the use of colorimeters or densitometers, devices that are not common to most academic image processing labs. It is hoped that this paper will encourage more labs to obtain some of this instrumentation.

There is not sufficient space in this paper to detail the profiling process. A complete description is found in [2]. The profiling process for a printer is outlined here. It is assumed that the user has image data in a DI space and wishes to print the image accurately.

To create the printer profile, a collection of samples are printed, which span the range of control values. The samples are measured to give a collection of ordered pairs $\{(c_i, d_i)\}$, which are used to define a functional mapping, $f(c) = d$ from control values, c to DI values, d. The function is usually determined using least squares fitting methods. Assuming that the image values are in the range of the printer, the control values that are sent to the printer are obtained from $f^{-1}(d) = c$.

If the image contains values that cannot be displayed by the device, then the user must select a gamut mapping algorithm to match the gamut (visual range) of the image to the gamut of the printer. The mapping should convey the intent of the user and any distortions that occur should be noted. As in the case of nonpictorial images, the decisions on proper gamut mapping are subjective

References

[1] MATLAB is a software package produced and supported by The MathWorks, Inc, Natick, MA (www.mathworks.com).

[2] M. J. Vrhel and H. J. Trussell, "Color Device Calibration: a mathematical formulation," IEEE Trans. IP, Dec., 1999.

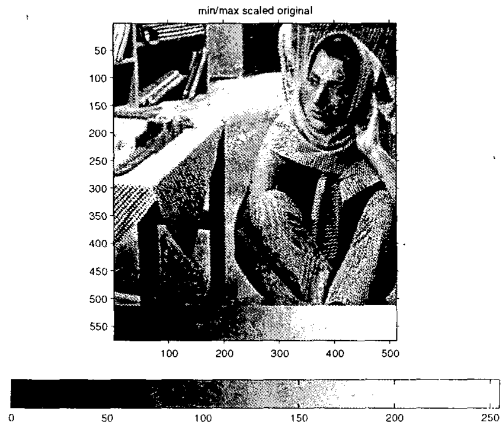

Figure 5: min/max Scaled Original

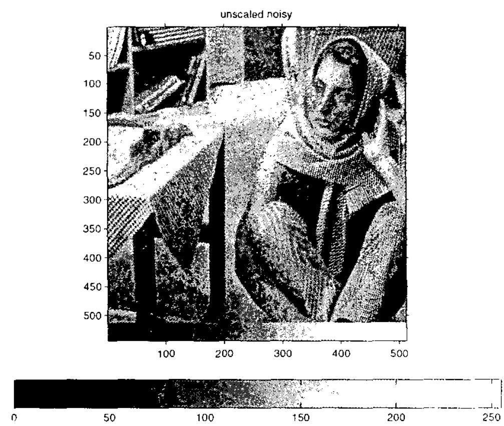

Figure 6: Noisy Original