Virtual Reality Tool for visualising RGB colour space using VRML

Anthony J. Bardos and Stephen J. Sangwine
The University of Reading
Whiteknights, Reading, RG6 6AY, UK
Email:{a.j.bardos/s.j.sangwine}@reading.ac.uk

ABSTRACT

This paper presents a novel Virtual Reality tool for visualising image neighbourhoods. It is currently implemented using RGB colour space, but the technique is not limited to Cartesian spaces. The Visualisation Tool is based on Virtual Reality Modeling Language (VRML). Visualisation using virtual reality techniques is a useful tool to help evaluate the effectiveness of colour filters. It can also be used as an aid to developing new ideas for filters. VRML makes Virtual Reality Visualisation Tools relatively trivial to implement and negates the need for expensive software packages.

1 Introduction

In this paper we present a novel virtual reality tool designed for visualising colour image data at the neighbourhood level. The Visualisation Tool is based on Virtual Reality Modeling Language (VRML). In Section 2 we give some background information on this language, followed by an explanation of how the Visualisation Tool works in Section 3. Section 4 gives a description of the Vector Median Filter which is the test filter used to demonstrate the Visualisation Tool. The capabilities of the Visualisation Tool are demonstrated in Section 5 using the standard colour test image of 'Lena'. Limitations of the Visualisation Tool are considered in Section 6. Finally, Section 7 suggests some further ways in which the Visualisation Tool could be developed.

Measuring the effectiveness of colour image filters can be problematic. The standard methods for evaluating the effectiveness of colour filters include: calculation of the Normalised Mean Square Error (NMSE) between the original and the filtered image; calculation of the Mean Chromaticity Error (MCRE) between the original and the filtered image [1]; and Visual Inspection of the original and the filtered image. The problem with these methods is that they are very general and do not give a specific indication of what happens at the neighbourhood level. They do not give an indication, for example, of whether a replacement pixel in a noise removing filter is the first best selection, the second best selection, or an equivalent selection.

2 VRML

Virtual Reality Modeling Language, VRML, is a layout language similar to HTML, but more powerful. It has been designed to allow the development of 3D Virtual Reality objects and worlds, applications include scientific visualisation, and web pages. The current version is VRML 97. The international standard is ISO/IEC 14772-1:1997 [2].

VRML, in common with HTML has the useful property that it can be easily generated by other applications from given parameters, thus allowing new scenes to be instantly generated and viewed.

3 The Visualisation Tool

The Visualisation Tool works in the following manner:

1. The user selects the part of the image that is to be examined.

2. The Visualisation Tool generates VRML code for the selected part of the image. This code describes a Virtual Reality model of the colour space showing the pixel values in the selected region.

3. When this VRML model is viewed (using a standard plug in to an Internet browser, for example), it is possible to explore visually in three dimensions, the positions of the pixel values in colour space by altering the viewpoint.

The pixel values are represented by small spheres plotted in a $256 \times 256 \times 256$ 'wire-frame' colour cube. The spheres are coloured according to the RGB pixel values they represent. The edges of the 'wire-frame' colour cube are coloured in accordance with the RGB colour cube (Figure 1). The number of spheres plotted is reflected by the size of the neighbourhood to be visualised. A 3×3 neighbourhood results in 9 pixels being plotted. The centre pixel of the neighbourhood can be marked bright blue if required.

If the effects of a particular filter are to be observed, the pixel selected by the filter, is represented by a bright green sphere in the neighbourhood. If the filter is of the type that generates a new pixel value this is shown as a small square instead of a sphere. This allows one to visualise the spatial

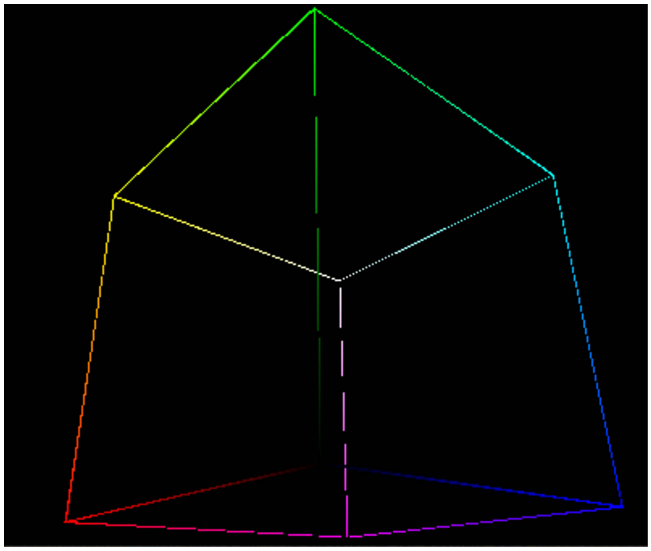

Figure 1: The Wire Frame Colour Cube in which the Colour Neighbourhoods are displayed

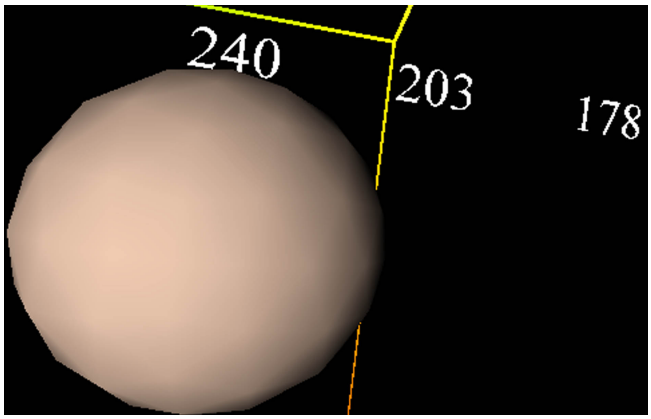

Figure 2: Close up of a colour pixel as represented by the Visualisation Tool. The colour is derived from its RGB values which are shown in text above it.

relationship between the pixel values selected by several different filters.

Addition information such as actual RGB values may be included in the VRML rendering as text messages placed by the spheres, figure 2.

4 The Colour Vector Median Filter

In this paper we have used the Vector Median Filter (VMF) as an example filter to demonstrate the use of the Visualisation Tool. We now give a brief description of how this filter works.

The $Median\ Filter$ is well established in monochrome image processing. It selects the pixel with median intensity from the pixels in a neighbourhood. It has the useful property of removing impulsive noise without blurring edges. Median filtering, however, cannot be directly transposed to colour images by filtering of the separate RGB image components of a colour image. This would take no account of the interdependency of the RGB values and leads to colour shifting rather than impulse removal. This problem is solved by the Vector Median Filter (VMF) [3]. It takes account of the interdependency of the RGB components by treating each pixel as a vector value. It then selects the pixel which is the $vector\ median$ of the input neighbourhood as the output value. Thus the output value will always be one of the input values (as in the case of monochrome median filtering) and preserves local colour characteristics.

The Vector Median Filter may be viewed as a generalisation of the median filter to vector-valued signals and images. The vector median filter may be evaluated as follows, assuming an RGB image and a simple Pythagorean distance metric in RGB space. We assume a neighbourhood such as a 3×3 pixel square, although other shapes and sizes of neighbourhood may be used, just as with conventional median filtering. For each pixel in the chosen neighbourhood the following sum must be computed (1):

$$S_i = \sum_{j=1}^{N} ||x_i - x_j||, i = 1, ...N \tag{1}$$

where x_i is the ith pixel in the neighbourhood, and $N = 9$ for a 3×3 neighbourhood. The pixel, which yields the minimum value of S, is selected as the vector median and used at the corresponding pixel position in the filtered image. In rectangular RGB space the norm $||x_i - x_j||$ is the Pythagorean distance between the two pixels (2).

$$||x_i - x_j|| = \sqrt{(r_i - r_j)^2 + (g_i - g_j)^2 + (b_i - b_j)^2} \tag{2}$$

where r_i is the red component of pixel x_i and so on. Thus S is simply the sum of the distances in RGB space from the pixel to all other pixels in the neighbourhood. The selection of the pixel with minimum S may be readily visualised as finding the pixel nearest the 'centre' of the pixels within the neighbourhood viewed as a cluster in RGB space. This is illustrated by the Visualisation Tool in figure 5.

5 Results

The standard colour test image of 'Lena' is used to demonstrate the capabilities of the Visualisation Tool shown in figure 5. The sample points A and B are indicated by arrows. A is at (x, y) position $(203, 453)$ and B is at (x, y) position $(400, 333)$.

Figure 5 shows how the neighbourhood at sample point A appears in the Visualisation Tool. The blue sphere is the pixel value at the centre of the neighbourhood, the green pixel is the Vector Median and the square is the average of the neighbourhood. A close up of how this neighbourhood would appear on a graphics display is shown in figure 4. The colours appear to be similar, but their differences can be shown by the Visualisation Tool.

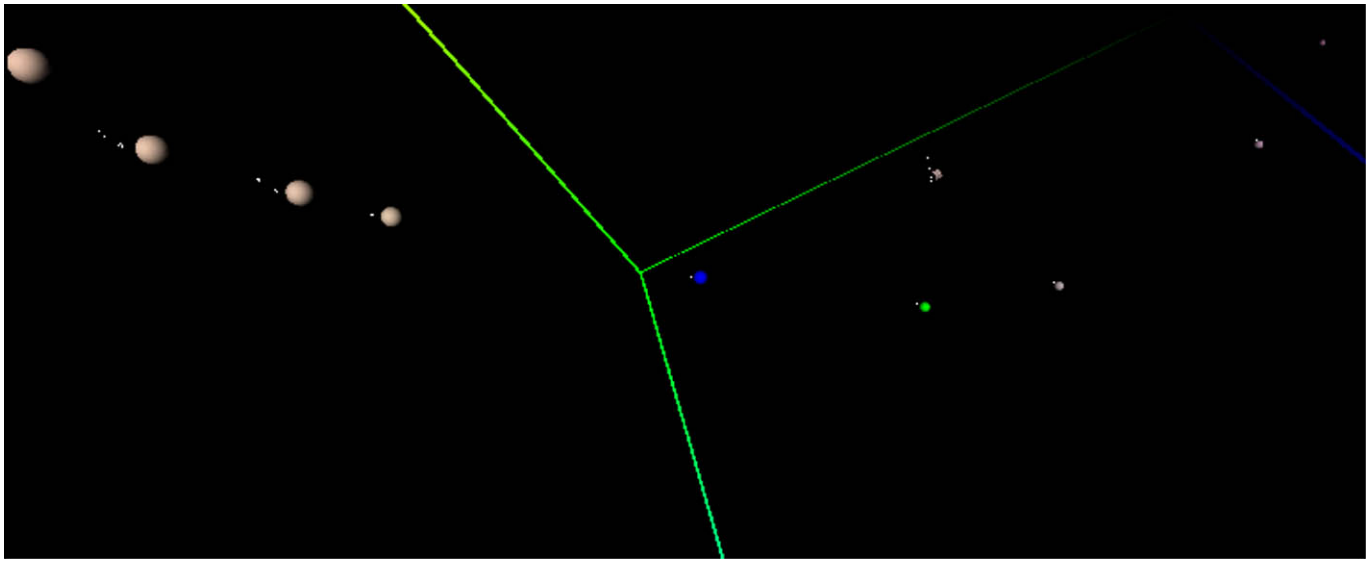

Figure 6: The neighbourhood at (X, Y) postion $(400, 333)$ in the standard test image 'Lena'. This shows that image neighbourhoods do not necessarily form neat cluster shapes, but can be spread out. This suggests that the selection of the Average or Vector Median for these neighbourhoods is problematic.

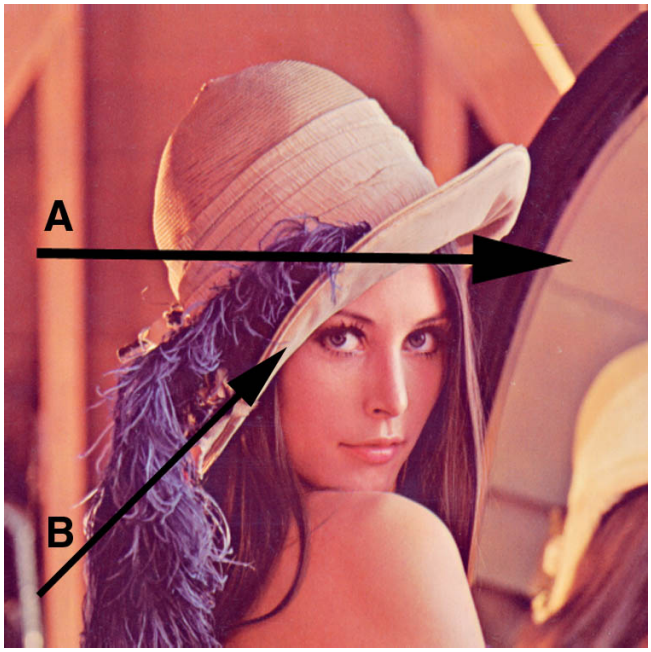

Figure 3: The Standard test image of 'Lena' with arrows pointing to sample point A, (x, y) postion $(203, 453)$, shown in figures 4 and 5, and sample point B at (x, y) position $(400, 333)$, shown in figure 6

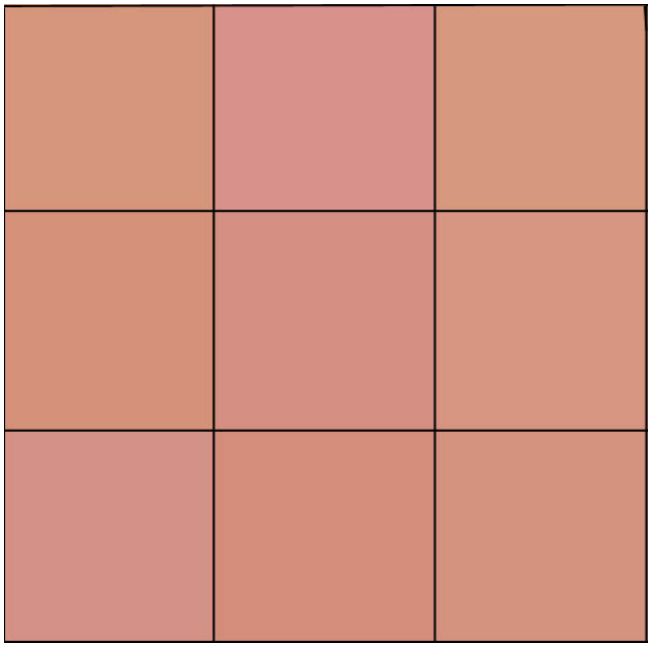

Figure 4: A close up of the 3×3 neighbourhood at sample point A, (x, y) postion $(203, 425)$ in the standard test image 'Lena'. The colours appear to be similar, but their differences can be shown by the Visualisation Tool, figure 5

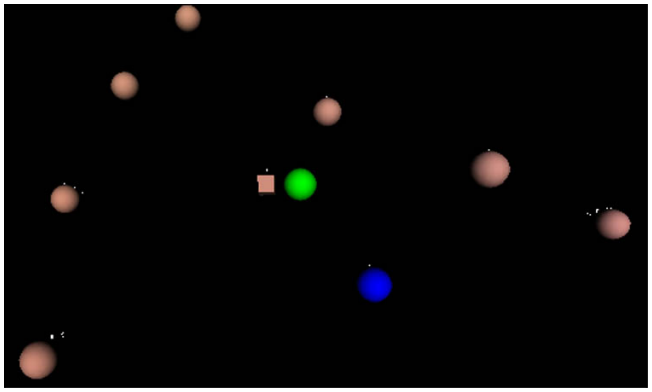

Figure 5: The neighbourhood at sample point A, (x, y) position $(453, 203)$ in the standard test image 'Lena'. The blue sphere represents the center pixel, the green sphere is the VMF and the cube is the average

Using the Visualisation Tool it is possible to navigate through a neighbourhood, zooming in or rotating the scene at will in order to fully examine its characteristics.

Figure 6 shows the neighbourhood at sample point B, (x,y) position $(400, 333)$ it shows that image neighbourhoods do not necessarily form neat cluster shapes, but can be spread out. This suggests that the selection of the Average or Vector Median for these neighbourhoods is problematic.

6 Limitations of the System

There are currently some limitations and difficulties with the system these are all problems with the VRML Viewers. VRML viewers, as currently implemented, can be slow rendering large numbers of points or shapes. It would, for example, be impractical to create a visualisation of the 3D RGB colour histogram, which would be an interesting project. The VRML viewers are not very intuitive and it can be difficult to manoeuvre the view into the desired position. There are some slight differences between how VRML is rendered by different viewers and it is sometimes not possible to see certain characterises of a view with some VRML viewers, for example, some viewers will show the edges of the 'wire frame' colour cube as a single colour and not as a graduation from one colour to another.

7 Improvements to the Visualisation Tool

VRML offers many possibities which the current version of the Visualisation Tool does not exploit. VRML unlike HTML includes scripting and this could, for example, give the Visualisation Tool the ability to allow the removel of a pixel from the scene and then dynamically recalculate the positions of the VMF and Average to see what effect the pixel was having on the VMF and average.

8 Conclusion

This tool can provide a useful visualisation system to look at the effects of colour image filters. However one does have to be tolerant of the inadequacies and inconsistencies of existing VRML viewers.

References

[1] Trahanias, P. E. and Venetsanopoulos, A. N., 1993. *Vector Directional Filters: a new class of multi-channel image processing filters*, IEEE Trans. Image Processing, **2**(4), 528-534.

[2] Information technology – Computer graphics and image processing – The Virtual Reality modeling Language (VRML) – Part 1: Functional specification and UTF-8 encoding. ISO/IEC 14772-1:1997 `http://www.vrml.org/technicalinfo/specifications/vrml97/index.htm`.

[3] J. Astola, P. Haavisto, and Y Neuvo, 1990 *Vector median filters*, Proceedings IEEE, **78**(4), 678-689.