

On viewscreen geometry vs. the pixmap for storing image color data

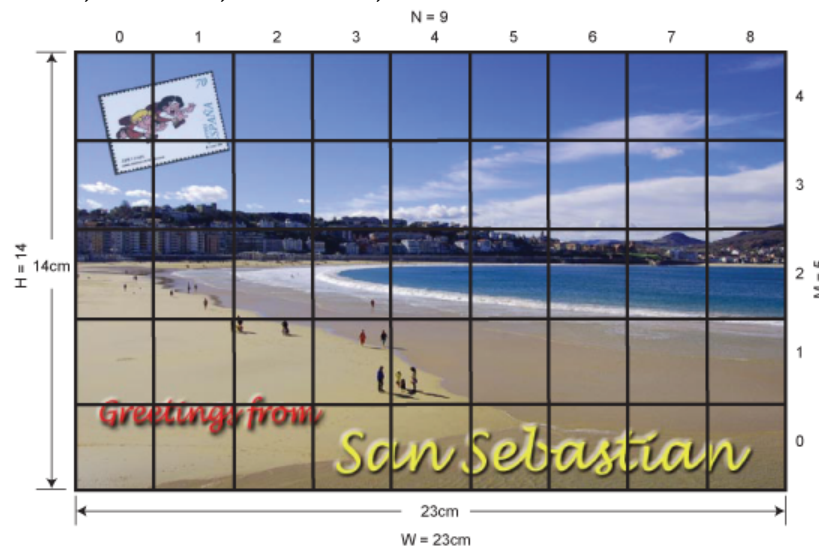
D. House, 12/11/2010

The key to understanding the difference between the viewscreen and the digital image that you create when rendering a scene is that the viewscreen is a virtual object in the virtual space of your scene. It is measured spatially, for example in centimeters. On the other hand, the pixmap that you will be using to store your color image, as you render it is a 2-D array of pixel color values stored in the memory of the computer. It is not a spatial object, but it does have dimensions, which are pixel indices.

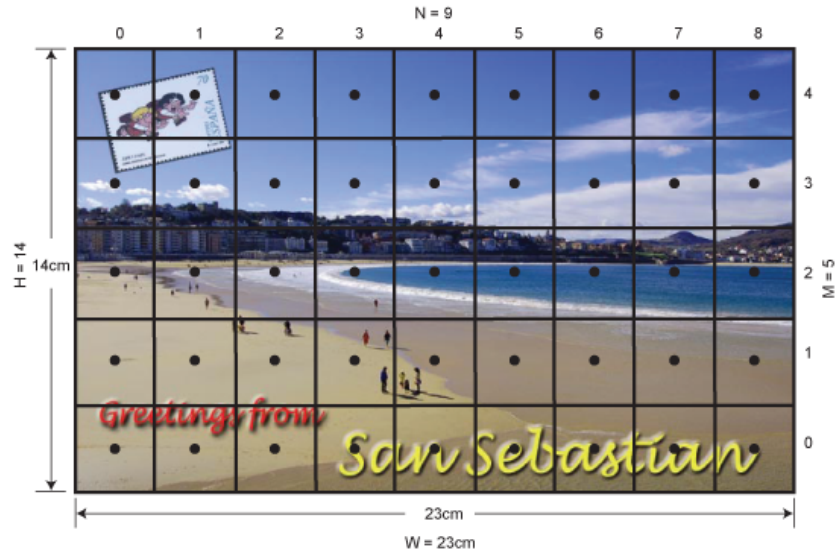
The postcard below is a real object that can be measured. In this case, it is 14cm high and 23 cm wide. Imagine that this is the viewscreen of your virtual camera.



Now, let us superimpose a grid on this postcard. In the example below, I have made the grid with 5 rows numbered by indices 0 through 4, and 9 columns numbered by indices 0 through 8. Referring to my notes raycast.pdf on the page numbered 13, in my postcard example W is 23cm, H is 14cm, M is 5 rows, and N is 9 columns.



Now, in raycasting, the idea is that we construct a ray from the viewpoint of the virtual camera through the center of the region of the viewscreen that corresponds to the pixel that we want to color. We do this for every pixel in the image, and use the ray to determine the color of the pixel. The spatial location of the points on the viewscreen through which we will shoot rays are shown in the diagram below.



For example, if we were to shoot a ray through pixel (3, 5), its spatial location relative to the lower lefthand corner of the postcard would be:

$$x = (5 + 0.5) (23.0/9) = 14.06\text{cm}, \text{ and } y = (3 + 0.5) (14.0 / 5) = 9.80\text{cm}$$

The column index + 0.5 gives how many pixel widths from the lefthand side of the postcard. The row index + 0.5 gives how many pixel heights from the bottom of the image. Multiplying by the width of a single pixel (W/N) or the height of a single pixel (H/M) gives the coordinates of the center of the pixel relative to the bottom lefthand corner of the viewscreen.