

## Image Manipulation Using Mathematica

Wolfram Alpha's programming language can be used for various purposes, including changing and creating images that can later be exported to your file system. This is possible largely through the use of vectors, matrices and arrays to represent the different aspects of the image. For example, given the image below, the dimensions can be given as a vector.

```
In[32]:=i = CurrentImage[]
```



```
Out[32]=
```

```
In[33]:=ImageDimensions[i]
```

```
Out[33]={320, 240}
```

A matrix or a multidimensional array can also be used to represent the colors of each individual pixel in this image:

Out[34]=RGB

In[35]:=MatrixForm /@ ImageData[ImageTake[i, {15, 20}, {10, 12}], "Byte", Interleaving -> False]

$$\text{Out[35]} = \left\{ \begin{pmatrix} 59 & 56 & 52 \\ 57 & 52 & 54 \\ 55 & 56 & 58 \\ 58 & 59 & 61 \\ 54 & 51 & 54 \\ 57 & 57 & 53 \end{pmatrix}, \begin{pmatrix} 92 & 92 & 91 \\ 92 & 91 & 92 \\ 88 & 92 & 94 \\ 91 & 92 & 94 \\ 88 & 86 & 89 \\ 91 & 93 & 89 \end{pmatrix}, \begin{pmatrix} 146 & 143 & 140 \\ 145 & 143 & 144 \\ 142 & 145 & 146 \\ 145 & 146 & 148 \\ 141 & 142 & 144 \\ 144 & 148 & 144 \end{pmatrix}, \begin{pmatrix} 255 & 255 & 255 \\ 255 & 255 & 255 \\ 255 & 255 & 255 \\ 255 & 255 & 255 \\ 255 & 255 & 255 \\ 255 & 255 & 255 \end{pmatrix} \right\}$$

In[36]:=MatrixForm /@ ImageData[ImageTake[i, {15, 20}, {10, 12}], "Byte"]

$$\text{Out[36]} = \left\{ \begin{pmatrix} 59 & 92 & 146 & 255 \\ 56 & 92 & 143 & 255 \\ 52 & 91 & 140 & 255 \end{pmatrix}, \begin{pmatrix} 57 & 92 & 145 & 255 \\ 52 & 91 & 143 & 255 \\ 54 & 92 & 144 & 255 \end{pmatrix}, \begin{pmatrix} 55 & 88 & 142 & 255 \\ 56 & 92 & 145 & 255 \\ 58 & 94 & 146 & 255 \end{pmatrix}, \begin{pmatrix} 58 & 91 & 145 & 255 \\ 59 & 92 & 146 & 255 \\ 61 & 94 & 148 & 255 \end{pmatrix}, \begin{pmatrix} 54 & 88 & 141 & 255 \\ 51 & 86 & 142 & 255 \\ 54 & 89 & 144 & 255 \end{pmatrix}, \begin{pmatrix} 57 & 91 & 144 & 255 \\ 57 & 93 & 148 & 255 \\ 53 & 89 & 144 & 255 \end{pmatrix} \right\}$$

In the first set of matrices above, the rows of each matrix correspond to pixels 15 through 20 from the left edge of the image, and the columns correspond to pixels 10 through 12 from the bottom edge of the image. The last matrix represents the highest value possible for each color channel (Red, Green, and Blue) in each pixel: 255. The other three matrices represent the actual values of each color channel for each pixel. The pixel 15 pixels from the left and 10 pixels from the bottom, for example, has a value of 59 for red coloring, 92 for green, and 146 for blue.

In the second set of matrices, the last column of each matrix show the maximum value possible in each entry, while the other columns represent RGB values. The first matrix shows the values of pixels 10 through 12 from the bottom and 15 pixels from the left, the second shows pixels 10 through 12 from the bottom and 16 pixels from the left, and so on. These two sets of matrices are different ways of representing the colors of each specified pixel in the image.

Mathematica can take this representation further by separating an image according to color channels. In the example below, only red is pulled out of the image.

```
In[37]:=ColorSeparate[ , {"R"}]
```



```
Out[37]←{ }
```



The image data for the same pixels as sampled from the original image can then be compared. The matrix for the red values of each pixel will remain the same as the original image, but to make this a gray scale image using red values, the matrices for blue and green must match. RGB values of (0, 0, 0) refer to pure black,

while (255, 255, 255) would refer to a pure white pixel.

```
In[48]:=MatrixForm /@ ImageData[ImageTake[, {15, 20}, {10, 12}], "Byte",
```

Interleaving → False]

$$\text{Out[48]} = \left\{ \begin{pmatrix} 59 & 56 & 52 \\ 57 & 52 & 54 \\ 55 & 56 & 58 \\ 58 & 59 & 61 \\ 54 & 51 & 54 \\ 57 & 57 & 53 \end{pmatrix}, \begin{pmatrix} 59 & 56 & 52 \\ 57 & 52 & 54 \\ 55 & 56 & 58 \\ 58 & 59 & 61 \\ 54 & 51 & 54 \\ 57 & 57 & 53 \end{pmatrix}, \begin{pmatrix} 59 & 56 & 52 \\ 57 & 52 & 54 \\ 55 & 56 & 58 \\ 58 & 59 & 61 \\ 54 & 51 & 54 \\ 57 & 57 & 53 \end{pmatrix}, \begin{pmatrix} 255 & 255 & 255 \\ 255 & 255 & 255 \\ 255 & 255 & 255 \\ 255 & 255 & 255 \\ 255 & 255 & 255 \\ 255 & 255 & 255 \end{pmatrix} \right\}$$


However, if a digital RGB image like this one needed to be printed, it would need to be converted to CMYK colors, as printers don't use red, blue, and green ink:

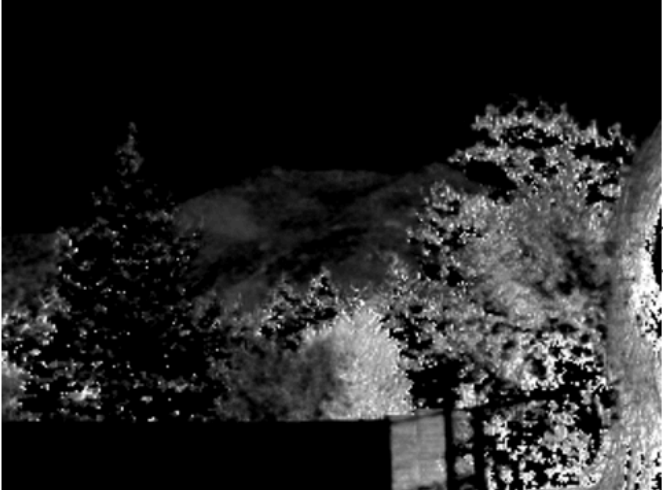
```
In[42]:=ColorConvert[, "CMYKColor"]
```

Out[42]=



The new image looks exactly the same as the original, but can be separated into cyan, magenta, yellow, and black values for a printer to read.

```
In[43]:=ColorSeparate[, {"Y"}]
```

Out[43]=

Images can also be created in Mathematica by initializing a matrix of values and then taking the entries to be RGB or Gray scale values.

```
In[28]:=arr = Array[.1 + .1#1 + .2#2&, {3, 3}]
```

```
Out[28]={{0.4, 0.6, 0.8}, {0.5, 0.7, 0.9}, {0.6, 0.8, 1.}}
```

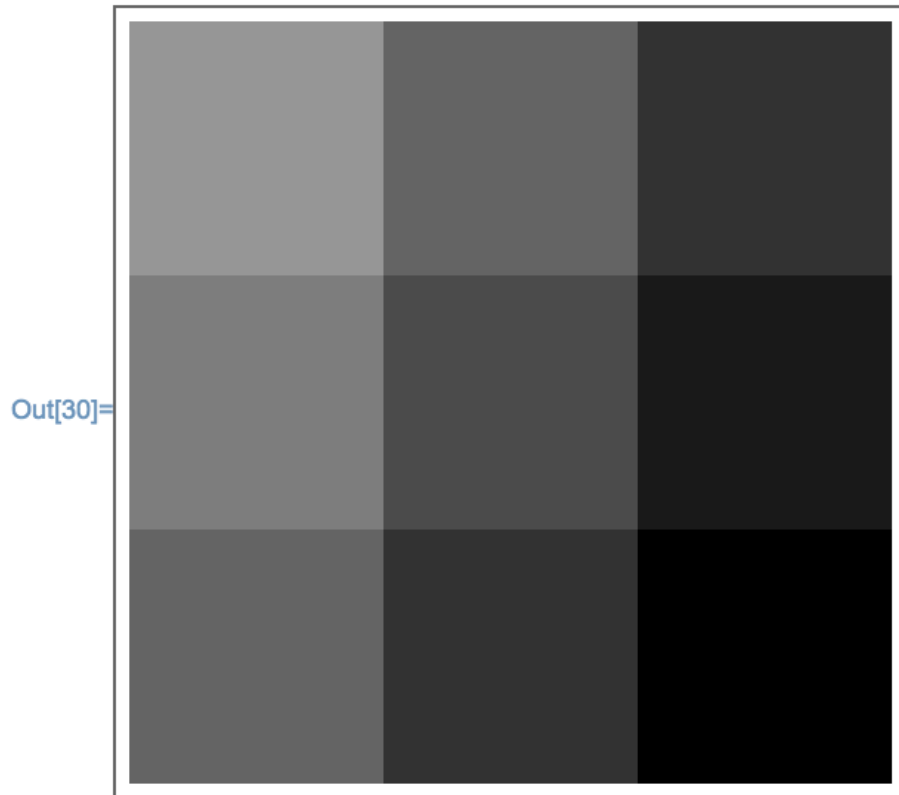
```
In[29]:=MatrixForm[arr]
```

```
Out[29]/MatrixForm=  

$$\begin{pmatrix} 0.4 & 0.6 & 0.8 \\ 0.5 & 0.7 & 0.9 \\ 0.6 & 0.8 & 1. \end{pmatrix}$$

```

```
In[30]:=i = ArrayPlot[arr]
```



```
In[31]:=Image[arr]
```

```
Out[31]=
```



For a 2D array, an array plot takes each row to be a different Gray scale color, while an image takes each column. For a 3D array, the colors will be in RGB by default, or another specified color space with three color channels.

```
In[21]:= arr2 = Array[.1#1 + .1#2 + .2#3&, {3, 3, 3}, 0]
```

```
Out[21]= {{ {0., 0.2, 0.4}, {0.1, 0.3, 0.5}, {0.2, 0.4, 0.6} }, { {0.1, 0.3, 0.5}, {0.2, 0.4, 0.6}, {0.3, 0.5, 0.7} }, { {0.2, 0.4, 0.6}, {0.3, 0.5, 0.7}, {0.4, 0.6, 0.8} } }
```

```
In[22]:= MatrixForm[arr2]
```

```
Out[22]//MatrixForm=
```

$$\begin{pmatrix} \begin{pmatrix} 0. \\ 0.2 \\ 0.4 \end{pmatrix} & \begin{pmatrix} 0.1 \\ 0.3 \\ 0.5 \end{pmatrix} & \begin{pmatrix} 0.2 \\ 0.4 \\ 0.6 \end{pmatrix} \\ \begin{pmatrix} 0.1 \\ 0.3 \\ 0.5 \end{pmatrix} & \begin{pmatrix} 0.2 \\ 0.4 \\ 0.6 \end{pmatrix} & \begin{pmatrix} 0.3 \\ 0.5 \\ 0.7 \end{pmatrix} \\ \begin{pmatrix} 0.2 \\ 0.4 \\ 0.6 \end{pmatrix} & \begin{pmatrix} 0.3 \\ 0.5 \\ 0.7 \end{pmatrix} & \begin{pmatrix} 0.4 \\ 0.6 \\ 0.8 \end{pmatrix} \end{pmatrix}$$

```
In[24]:= Image[arr2, ColorSpace -> "RGB"]
```



```
In[25]:= Image[arr2, ColorSpace -> "HSB"]
```



The ImageData function can then retrieve the RGB values for the images, which are equal to the entries in the arrays used. This is shown as a partitioned matrix, where each partition represents one color.

```
In[29]:= ImageData[Image[arr2]] // MatrixForm
```

```
Out[29]//MatrixForm=
```

$$\begin{pmatrix} \begin{pmatrix} 0. \\ 0.2 \\ 0.4 \end{pmatrix} & \begin{pmatrix} 0.1 \\ 0.3 \\ 0.5 \end{pmatrix} & \begin{pmatrix} 0.2 \\ 0.4 \\ 0.6 \end{pmatrix} \\ \begin{pmatrix} 0.1 \\ 0.3 \\ 0.5 \end{pmatrix} & \begin{pmatrix} 0.2 \\ 0.4 \\ 0.6 \end{pmatrix} & \begin{pmatrix} 0.3 \\ 0.5 \\ 0.7 \end{pmatrix} \\ \begin{pmatrix} 0.2 \\ 0.4 \\ 0.6 \end{pmatrix} & \begin{pmatrix} 0.3 \\ 0.5 \\ 0.7 \end{pmatrix} & \begin{pmatrix} 0.4 \\ 0.6 \\ 0.8 \end{pmatrix} \end{pmatrix}$$

Aside from other applications, Mathematica can create and manipulate images using matrices, vectors and partitioned matrices.

## Works Cited

- "Array." —*Wolfram Language Documentation*. N.p., n.d. Web. 01 May 2016.  
<<https://reference.wolfram.com/language/ref/Array.html>>.
- "Color Processing." —*Wolfram Language Documentation*. N.p., n.d. Web. 31 Mar. 2016.  
<<http://reference.wolfram.com/language/guide/ColorProcessing.html>>.
- "ColorConvert." —*Wolfram Language Documentation*. N.p., n.d. Web. 29 Mar. 2016.
- "ColorSeparate." —*Wolfram Language Documentation*. N.p., n.d. Web. 30 Mar. 2016.
- "EasyBib: The Free Automatic Bibliography Composer." *EasyBib*. N.p., n.d. Web. 30 Mar. 2016. <<http://www.easybib.com/>>.
- "Image Processing." —*Wolfram Language Documentation*. N.p., n.d. Web. 30 Mar. 2016.  
<<https://reference.wolfram.com/language/tutorial/ImageProcessing.html>>.
- "Image Representation." —*Wolfram Language Documentation*. N.p., n.d. Web. 31 Mar. 2016.  
<<https://reference.wolfram.com/language/guide/ImageRepresentation.html>>.
- "ImageData." —*Wolfram Language Documentation*. N.p., n.d. Web. 31 Mar. 2016.  
<<https://reference.wolfram.com/language/ref/ImageData.html>>.
- "ImageLines." —*Wolfram Language Documentation*. N.p., n.d. Web. 31 Mar. 2016.  
<<https://reference.wolfram.com/language/ref/ImageLines.html>>.
- "MatrixPlot." —*Wolfram Language Documentation*. N.p., n.d. Web. 31 Mar. 2016.  
<<https://reference.wolfram.com/language/ref/MatrixPlot.html>>.
- "RGBColor." —*Wolfram Language Documentation*. N.p., n.d. Web. 29 Mar. 2016.



Wolfram, Stephen. *An Elementary Introduction to the Wolfram Language*. N.p.: n.p., n.d.

*Preface: Elementary Introduction to the Wolfram Language*. Web. 30 Mar. 2016.

<<http://www.wolfram.com/language/elementary-introduction/preface.html>>.