# Linear algebra with pictures

In[5]:=
```
NotebookFileName[]
```

Out[5]=
```
C:\Dropbox\Work\myweb\Courses\Math_pages\Math_204\Pictures_of_Delft.nb
```

---

## Preliminaries

### Import a picture

First set the working directory.

This is the code to set the working directory on my office PC. It is commented out below since need to use a different code. First you verify where you saved your file. Save the jpg picture in the same directory.

In[6]:=
```
NotebookDirectory[]
```

Out[6]=
```
C:\Dropbox\Work\myweb\Courses\Math_pages\Math_204\
```

In[7]:=
```
SetDirectory[NotebookDirectory[]]
```

Out[7]=
```
C:\Dropbox\Work\myweb\Courses\Math_pages\Math_204
```

You need to figure set the working directory on your PC in which you have this *Mathematica* file and the picture file.

Now import a sample picture:

In[8]:=
```
DelftPic = Import["578px-Vermeer-view-of-delft.jpg"]
```

Out[8]=

Make the picture smaller.

In[9]:=
```
DelftPic = ImageResize[DelftPic, {144, 120}]
```

Out[9]=

What is DelftPic?

In[10]:=
```
DelftPic
```

Out[10]=

To manipulate graphics we need to see what is inside DelftPic: (it is a lot of numbers and some *Mathematica* commands)

## Understand what is the code inside the picture

The cell below contains a lot of information. Remove the semicolon to see the information.

In[11]:=
```
DelftPic
```

Out[11]=

In[12]:=
```
FullForm[DelftPic]
```

Out[12]//FullForm=
```
Image[ … 1 … ]
```
large output | show less | show more | show all | set size limit...

In[13]:=
```
DelftPicM = ImageData[DelftPic]
```

Out[13]=
```
{{{0.658824, 0.662745, 0.658824}, {0.717647, 0.709804, 0.701961}, … 141 …, {0.6, 0.545098, 0.458824}},
 … 118 …, {… 1 …}}
```
large output | show less | show more | show all | set size limit...

The file DelftPicM contains a lot of numeric information. Mathematica command Image[DelftPic] presents that numerical information as the real image:

In[14]:=
```
Image[DelftPicM]
```

Out[14]=

Just the file name is a lot of numerical information. Let us see how many basic components are there:

In[15]:=
```
Dimensions[DelftPicM]
```

Out[15]=
```
{120, 144, 3}
```

Make the painting black and white.

In[16]:=
```
test = (DelftPicM /. {r_, g_, b_} → 1/3 {r + g + b, r + g + b, r + g + b});
```

In[17]:=
```
test[[1, 1]]
```

Out[17]=
```
{0.660131, 0.660131, 0.660131}
```

In[18]:=
```
Image[test]
```

Out[18]=

Make the painting shade of light blue.

In[19]:=
```
testB = (DelftPicM /. {r_, g_, b_} → (((1 - (r + g + b)/3) {0, 0, 1} + (r + g + b)/3 {1, 1, 1})));
```

In[20]:=
```
Image[testB]
```

Out[20]=

Make the painting black and white, then make it dark, then make it light.

In[21]:=
```
DelftPics =
 GraphicsRow[{Image[DelftPicM], Image[(DelftPicM /. {r_, g_, b_} → 1/3 {r + g + b, r + g + b, r + g + b})],
   Image[(DelftPicM /. {r_, g_, b_} → 0.5 {r, g, b})],
   Image[(DelftPicM /. {r_, g_, b_} → ((1 - 0.5) {r, g, b} + 0.5 {1, 1, 1}))]}, ImageSize → 500]
```

Out[21]=

Make the painting different shades of dark.

In[22]:=
```
DelftPicsD = GraphicsRow[{Image[DelftPicM],
   Image[(DelftPicM /. {r_, g_, b_} → 0.75 {r, g, b})], Image[(DelftPicM /. {r_, g_, b_} → 0.5 {r, g, b})],
   Image[(DelftPicM /. {r_, g_, b_} → 0.25 {r, g, b})]}, ImageSize → 500]
```

Out[22]=

Make the painting different shades of light.

In[23]:=
```
DelftPicsL =
 GraphicsRow[{Image[DelftPicM], Image[(DelftPicM /. {r_, g_, b_} → ((1 - 0.25) {r, g, b} + 0.25 {1, 1, 1}))],
   Image[(DelftPicM /. {r_, g_, b_} → ((1 - 0.5) {r, g, b} + 0.5 {1, 1, 1}))],
   Image[(DelftPicM /. {r_, g_, b_} → ((1 - 0.75) {r, g, b} + 0.75 {1, 1, 1}))]}, ImageSize → 500]
```

Out[23]=

In[24]:=
```
Dimensions[DelftPicM]
```

Out[24]= {120, 144, 3}

In[25]:=
```
Length[DelftPicM[[1]]]
```

Out[25]= 144

In[26]:=
```
DelftPicM[[1]][[1]]
```

Out[26]= {0.658824, 0.662745, 0.658824}

Now see what each individual component is:

In[27]:=
```
Length[DelftPicM[[17]]]
```

Out[27]= 144

In[28]:=
```
240 × 289
```

Out[28]= 69 360

So, this picture consists of 240 rows with 289 pixels in each row, the total of 69360 pixels.

The part below contains the color information about each pixel.

In[29]:=
```
DelftPicM[[1]]
```

Out[29]=
```
{{0.658824, 0.662745, 0.658824}, {0.717647, 0.709804, 0.701961}, {0.729412, 0.709804, 0.647059},
 {0.698039, 0.67451, 0.607843}, {0.698039, 0.682353, 0.627451}, {0.666667, 0.65098, 0.580392},
 {0.654902, 0.627451, 0.552941}, {0.635294, 0.6, 0.533333}, {0.619608, 0.588235, 0.505882},
 {0.611765, 0.584314, 0.501961}, {0.6, 0.572549, 0.498039}, {0.580392, 0.552941, 0.478431},
 {0.576471, 0.533333, 0.454902}, {0.580392, 0.537255, 0.454902}, {0.556863, 0.513725, 0.439216},
 {0.54902, 0.505882, 0.427451}, {0.545098, 0.513725, 0.419608}, {0.521569, 0.490196, 0.407843},
 {0.509804, 0.478431, 0.403922}, {0.513725, 0.478431, 0.411765}, {0.513725, 0.482353, 0.407843},
 {0.509804, 0.478431, 0.403922}, {0.505882, 0.47451, 0.4}, {0.501961, 0.470588, 0.396078},
 {0.501961, 0.466667, 0.4}, {0.501961, 0.466667, 0.4}, {0.501961, 0.466667, 0.4},
 {0.505882, 0.470588, 0.403922}, {0.501961, 0.466667, 0.4}, {0.501961, 0.466667, 0.4},
 {0.501961, 0.466667, 0.4}, {0.501961, 0.466667, 0.4}, {0.501961, 0.470588, 0.396078},
 {0.498039, 0.466667, 0.392157}, {0.498039, 0.466667, 0.392157}, {0.498039, 0.466667, 0.392157},
 {0.498039, 0.454902, 0.384314}, {0.490196, 0.447059, 0.372549}, {0.486275, 0.443137, 0.372549},
 {0.494118, 0.45098, 0.376471}, {0.494118, 0.45098, 0.360784}, {0.478431, 0.435294, 0.352941},
 {0.478431, 0.431373, 0.356863}, {0.462745, 0.423529, 0.352941}, {0.45098, 0.415686, 0.34902},
 {0.447059, 0.411765, 0.345098}, {0.443137, 0.407843, 0.341176}, {0.439216, 0.403922, 0.337255},
 {0.439216, 0.403922, 0.337255}, {0.435294, 0.4, 0.333333}, {0.435294, 0.4, 0.333333},
 {0.427451, 0.392157, 0.32549}, {0.427451, 0.392157, 0.32549}, {0.427451, 0.392157, 0.32549},
 {0.431373, 0.396078, 0.329412}, {0.431373, 0.396078, 0.329412}, {0.419608, 0.392157, 0.321569},
 {0.423529, 0.392157, 0.32549}, {0.427451, 0.392157, 0.32549}, {0.435294, 0.4, 0.333333},
 {0.439216, 0.403922, 0.337255}, {0.45098, 0.415686, 0.34902}, {0.462745, 0.427451, 0.360784},
 {0.47451, 0.439216, 0.372549}, {0.470588, 0.435294, 0.368627}, {0.478431, 0.443137, 0.376471},
 {0.486275, 0.45098, 0.384314}, {0.498039, 0.462745, 0.396078}, {0.509804, 0.462745, 0.396078},
 {0.501961, 0.458824, 0.380392}, {0.501961, 0.458824, 0.364706}, {0.509804, 0.470588, 0.372549},
 {0.545098, 0.490196, 0.407843}, {0.533333, 0.486275, 0.403922}, {0.541176, 0.494118, 0.415686},
 {0.541176, 0.498039, 0.419608}, {0.537255, 0.490196, 0.407843}, {0.537255, 0.490196, 0.407843},
 {0.517647, 0.470588, 0.388235}, {0.541176, 0.490196, 0.407843}, {0.560784, 0.501961, 0.423529},
 {0.533333, 0.482353, 0.4}, {0.521569, 0.47451, 0.392157}, {0.513725, 0.470588, 0.388235},
 {0.498039, 0.454902, 0.388235}, {0.494118, 0.454902, 0.388235}, {0.494118, 0.458824, 0.396078},
 {0.478431, 0.447059, 0.388235}, {0.45098, 0.427451, 0.372549}, {0.427451, 0.403922, 0.345098},
 {0.423529, 0.403922, 0.329412}, {0.415686, 0.392157, 0.317647}, {0.419608, 0.392157, 0.321569},
 {0.423529, 0.396078, 0.329412}, {0.419608, 0.392157, 0.333333}, {0.423529, 0.396078, 0.333333},
 {0.423529, 0.396078, 0.32549}, {0.419608, 0.392157, 0.321569}, {0.423529, 0.396078, 0.333333},
 {0.427451, 0.4, 0.333333}, {0.419608, 0.396078, 0.329412}, {0.419608, 0.396078, 0.333333},
 {0.431373, 0.407843, 0.34902}, {0.443137, 0.419608, 0.364706}, {0.454902, 0.431373, 0.376471},
 {0.470588, 0.443137, 0.388235}, {0.482353, 0.454902, 0.388235}, {0.494118, 0.462745, 0.403922},
 {0.513725, 0.478431, 0.415686}, {0.52549, 0.490196, 0.419608}, {0.537255, 0.509804, 0.431373},
 {0.545098, 0.509804, 0.439216}, {0.556863, 0.521569, 0.458824}, {0.568627, 0.529412, 0.458824},
 {0.588235, 0.545098, 0.47451}, {0.619608, 0.580392, 0.501961}, {0.678431, 0.635294, 0.556863},
 {0.701961, 0.65098, 0.572549}, {0.701961, 0.65098, 0.576471}, {0.713725, 0.670588, 0.6},
 {0.733333, 0.690196, 0.619608}, {0.72549, 0.682353, 0.611765}, {0.721569, 0.678431, 0.607843},
 {0.721569, 0.678431, 0.6}, {0.721569, 0.682353, 0.584314}, {0.717647, 0.67451, 0.584314},
 {0.717647, 0.670588, 0.580392}, {0.72549, 0.670588, 0.572549}, {0.733333, 0.670588, 0.568627},
 {0.733333, 0.670588, 0.568627}, {0.737255, 0.670588, 0.568627}, {0.701961, 0.643137, 0.541176},
 {0.631373, 0.572549, 0.486275}, {0.603922, 0.552941, 0.478431}, {0.592157, 0.54902, 0.454902},
 {0.568627, 0.529412, 0.423529}, {0.568627, 0.517647, 0.423529}, {0.568627, 0.521569, 0.431373},
 {0.588235, 0.541176, 0.454902}, {0.560784, 0.517647, 0.427451}, {0.552941, 0.509804, 0.415686},
 {0.560784, 0.513725, 0.423529}, {0.564706, 0.513725, 0.423529}, {0.6, 0.545098, 0.458824}}
```

How to make a table with two parameters.

In[30]:=
```
Table[{j, k}, {j, 1, 5}, {k, 1, 3}]
```

Out[30]=
```
{{{1, 1}, {1, 2}, {1, 3}}, {{2, 1}, {2, 2}, {2, 3}},
 {{3, 1}, {3, 2}, {3, 3}}, {{4, 1}, {4, 2}, {4, 3}}, {{5, 1}, {5, 2}, {5, 3}}}
```

In[31]:=
```
Table[{j, k}, {j, 1, 5}, {k, 1, 3}][[3, 2]]
```

Out[31]=
```
{3, 2}
```

Here, again, at the end we have some *Mathematica* commands and a lot of numbers preceding the commands.

This is the color of the fourth pixel in the first row.

In[32]:=
```
DelftPicM[[1, 4]]
```

Out[32]=
```
{0.698039, 0.67451, 0.607843}
```

In[33]:=
```
DelftPicM[[251, 3]]
```

Part: Part 251 of
{{{0.658824, 0.662745, 0.658824}, {0.717647, 0.709804, 0.701961}, ≪7≫, {0.611765, 0.584314, 0.501961}, ≪134≫}, {≪1≫}, ≪7≫, {≪1≫}, ≪110≫}
does not exist.

Out[33]=
```
{{{0.658824, 0.662745, 0.658824}, ··· 142 ···, {0.6, 0.545098, 0.458824}}, ··· 118 ···, {··· 1 ···}}[[251, 3]]
```

large output | show less | show more | show all | set size limit...

In[34]:=
```
DelftPicM[[1, 2]]
```

Out[34]=
```
{0.717647, 0.709804, 0.701961}
```

The code below has only the color information for each pixel. Again, to see it remove

In[35]:=
```
DelftPicM[[1, 1]]
```

Out[35]=
```
{0.658824, 0.662745, 0.658824}
```

In[36]:=
```
Last[DelftPicM[[1]]]
```

Out[36]=
```
{0.6, 0.545098, 0.458824}
```

The above list is where the basic information is: a lot of triples, each triple is the color of a pixel. In this list there are 240 rows each containing 289 triples. Thus, this picture has the resolution of 240x289.

In[37]:=
```
Length[DelftPicM]
```

Out[37]=
```
120
```

The above number tells that there are 480 rows in the list.

In[38]:=
```
Length[DelftPicM[[1]]]
```

Out[38]=
```
144
```

## The conclusion

The core information is in the following list

In[39]:=
```
DelftPicM; (* Vermeer's painting in Mathematica's language *)
```

This list consists of 240 lists of 289 triples of numbers between 0 and 1. Those triples define colors for each pixel. There are 240x289 = 69360 pixels.

For example, the pixel at the position {183,222} is colored by the following color:

In[40]:=
```
DelftPicM[[183, 222]]
```

Part: Part 183 of
{{{0.658824, 0.662745, 0.658824}, {0.717647, 0.709804, 0.701961}, ≪7≫, {0.611765, 0.584314, 0.501961}, ≪134≫}, {≪1≫}, ≪7≫, {≪1≫}, ≪110≫}
does not exist.

Out[40]=
```
{{{0.658824, 0.662745, 0.658824}, ··· 142 ···, {0.6, 0.545098, 0.458824}}, ··· 118 ···, {··· 1 ···}}[[
 183, 222]]
```

large output | show less | show more | show all | set size limit...

The new meaning of the Transpose learned in Linear Algebra

In[41]:=
```
Image[Transpose[DelftPicM]]
```

Out[41]=

# Delft in *Mathematica*

## Delft in the coordinate system

Now I show how to use the above list (DelftPicM) (representing Vermeer's painting) in a coordinate system. The command below puts the painting in the coordinate system:

In[42]:=
```
Dimensions[DelftPicM]
```

Out[42]=
```
{120, 144, 3}
```

In[43]:=
```
RGBColor[DelftPicM[[123, 256]]]
```

Part: Part 123 of
{{{0.658824, 0.662745, 0.658824}, {0.717647, 0.709804, 0.701961}, ≪7≫, {0.611765, 0.584314, 0.501961}, ≪134≫}, {≪1≫}, ≪7≫, {≪1≫}, ≪110≫}
does not exist.

Out[43]=
```
RGBColor[{{{0.658824, 0.662745, 0.658824}, ··· 142 ···, {0.6, 0.545098, 0.458824}},
    ··· 118 ···, {··· 1 ···, ··· 142 ···, {··· 1 ···}}}[[123, 256]]]
```

large output | show less | show more | show all | set size limit...

On a slow computer the command below can be slow. It will be faster if you reduce the size of the picture.

In[44]:=
```
Show[Graphics[
  Table[{PointSize[0.015], RGBColor[DelftPicM[[j, k]]], Point[N[{j/100, k/100}]]},
   {j, 1, Dimensions[DelftPicM][[1]]}, {k, 1, Dimensions[DelftPicMc][[2]]}]
 ],
 AspectRatio → Automatic, ImageSize → 200, Frame → True,
 PlotRange → {{0, 2.40}, {0, 2.89}}]
```

Part: Part 2 of {} does not exist.

Table: Iterator {k, 1, Dimensions[DelftPicMc][[2]]} does not have appropriate bounds.

Part: Part 2 of {} does not exist.

Table: Iterator {k, 1, Dimensions[DelftPicMc][[2]]} does not have appropriate bounds.

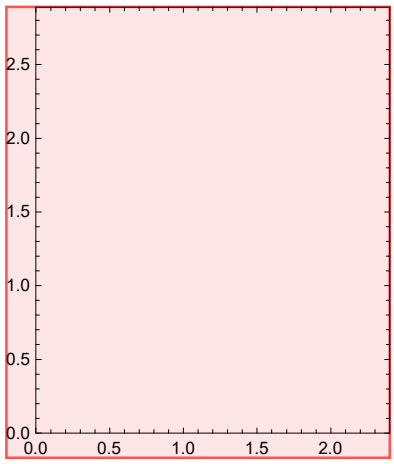
Part: Part 2 of {} does not exist.

General: Further output of Part::partw will be suppressed during this calculation.

Table: Iterator {k, 1, Dimensions[DelftPicMc][[2]]} does not have appropriate bounds.

General: Further output of Table::iterb will be suppressed during this calculation.

Out[44]=



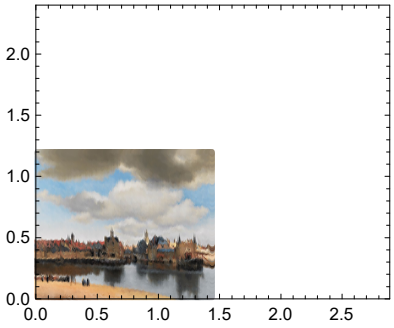

In[45]:=
```
DelftPicMc = Reverse[#] & /@ Transpose[DelftPicM];
```

In[46]:=
```
Dimensions[DelftPicMc]
```

Out[46]=
```
{144, 120, 3}
```

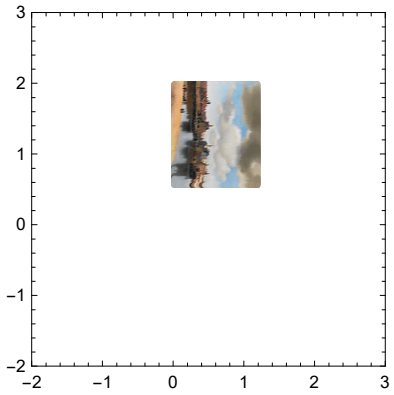
In[47]:=
```
Show[Graphics[
  Table[{PointSize[0.015], RGBColor[DelftPicMc[[j, k]]], Point[{j/100, k/100}]},
   {j, 1, Dimensions[DelftPicMc][[1]]}, {k, 1, Dimensions[DelftPicMc][[2]]}]
 ],
AspectRatio → Automatic, ImageSize → 200, Frame → True,
PlotRange → {{0, 2.89}, {0, 2.4}}]
```

Out[47]=




In[48]:=
```
Show[Graphics[
  Table[{PointSize[0.015], RGBColor[DelftPicMc[[j, k]]], Point[{{0, 1}, {-1, 0}}.{j/100, k/100} + {0, 2}]},
   {j, 1, Dimensions[DelftPicMc][[1]]}, {k, 1, Dimensions[DelftPicMc][[2]]}]
 ],
AspectRatio → Automatic, ImageSize → 200, Frame → True,
PlotRange → {{-2, 3}, {-2, 3}}]
```

Out[48]=




In[49]:=
```
{{0, 1}, {-1, 0}}.{j/100, k/100} + {0, 2}
```

Out[49]=
$$\left\{\frac{k}{100}, 2 - \frac{j}{100}\right\}$$

# Colorizing pictures

In this section, I explore which different colors turn into the same shade of gray. This must have been important when they develop software to colorize old black-and-white pictures. This is of course the first step, they must have developed sophisticated ways of choosing the exact color to colorize. But the shapes that we present give the first guide.

In[50]:=
```
Clear[mU]; mU = Transpose[{{-1/Sqrt[2], 1/Sqrt[2], 0}, {-1/Sqrt[6], -1/Sqrt[6], 2/Sqrt[6]}, {1/Sqrt[3], 1/Sqrt[3], 1/Sqrt[3]}}];
(* orthonormal basis *)
Clear[Int]; (* along the diagonal at a*Sqrt[3] units from the origin,
this is the intersection of the cube surface and the plane,
see below *)Int[a_] := If[a ≤ 1/3, {{3 a, 0, 0}, {0, 3 a, 0}, {0, 0, 3 a}},
  If[a < 2/3, {{3 a - 1, 1, 0}, {1, 3 a - 1, 0}, {1, 0, 3 a - 1}, {3 a - 1, 0, 1}, {0, 3 a - 1, 1}, {0, 1, 3 a - 1}},
   {{1, 3 a - 2, 1}, {1, 1, 3 a - 2}, {3 a - 2, 1, 1}}]];

Clear[DotL]; (* creates 1/st points between points pA and pB  *)

DotL[{pA_, pB_}, st_] := ((1 - #) pA + # pB) & /@ Range[0, 1, st]
```

In[54]:=
```
Transpose[mU].mU
```

Out[54]=
$$\{\{1, 0, 0\}, \{0, 1, 0\}, \{0, 0, 1\}\}$$

In[55]:=
```
Transpose[mU].{0, 0, 1}
```

Out[55]=
$$\left\{0, \sqrt{\frac{2}{3}}, \frac{1}{\sqrt{3}}\right\}$$

In[56]:=
```
Transpose[mU].{0, 1, 0}
```

Out[56]=
$$\left\{\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{6}}, \frac{1}{\sqrt{3}}\right\}$$

In[57]:=
```
Transpose[mU].{1, 0, 0}
```

Out[57]=
$$\left\{-\frac{1}{\sqrt{2}}, -\frac{1}{\sqrt{6}}, \frac{1}{\sqrt{3}}\right\}$$

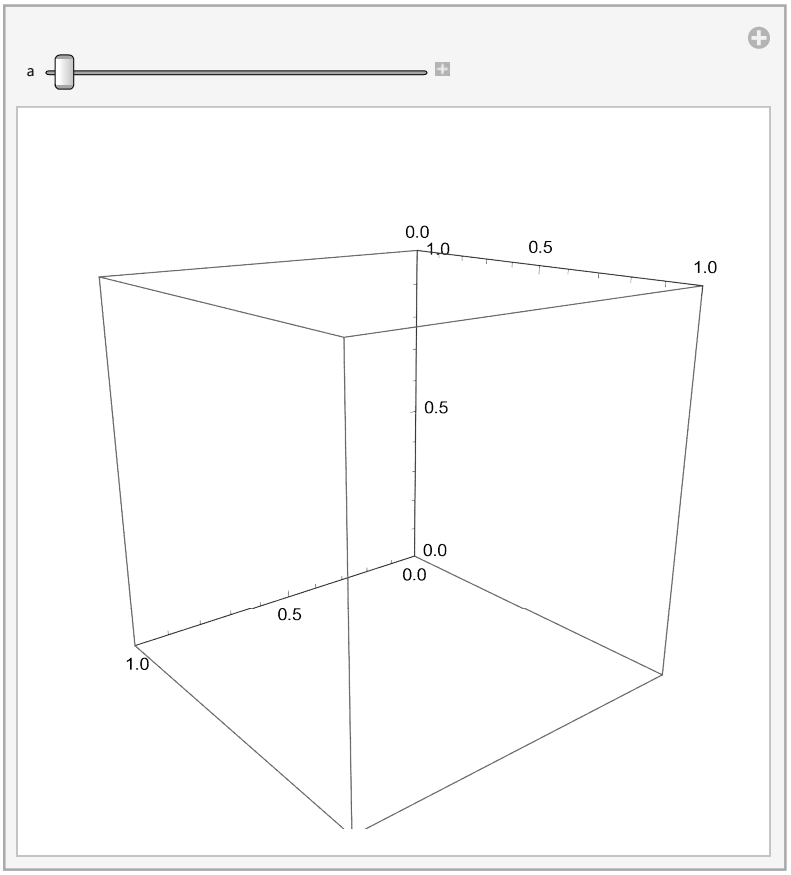
In[58]:=
```
Int[4 / 7]
```

Out[58]=
$$\left\{\left\{\frac{5}{7}, 1, 0\right\}, \left\{1, \frac{5}{7}, 0\right\}, \left\{1, 0, \frac{5}{7}\right\}, \left\{\frac{5}{7}, 0, 1\right\}, \left\{0, \frac{5}{7}, 1\right\}, \left\{0, 1, \frac{5}{7}\right\}\right\}$$

Explore the role of **Int[]**

In[59]:=
```
Manipulate[Graphics3D[{Line[Append[Int[a], First[Int[a]]]]},
  PlotRange → {{0, 1}, {0, 1}, {0, 1}}, Axes → True, ViewPoint → {1.44, 1.8, 0.8}], {a, 0, 1}]
```

Out[59]=




In[60]:=
```
DotL[{{0, 0, 0}, {1, 1, 1}}, 0.1]
```

Out[60]=
```
{{0., 0., 0.}, {0.1, 0.1, 0.1}, {0.2, 0.2, 0.2}, {0.3, 0.3, 0.3}, {0.4, 0.4, 0.4},
 {0.5, 0.5, 0.5}, {0.6, 0.6, 0.6}, {0.7, 0.7, 0.7}, {0.8, 0.8, 0.8}, {0.9, 0.9, 0.9}, {1., 1., 1.}}
```

The command below creates dots along the line of intersection as calculated by Int[aa]

In[61]:=
```
aa = 0.456; Flatten[DotL[#, 0.1] & /@ Partition[Append[Int[aa], First[Int[aa]]], 2, 1], 1]
```

Out[61]=
```
{{0.368, 1., 0.}, {0.4312, 0.9368, 0.}, {0.4944, 0.8736, 0.}, {0.5576, 0.8104, 0.}, {0.6208, 0.7472, 0.},
 {0.684, 0.684, 0.}, {0.7472, 0.6208, 0.}, {0.8104, 0.5576, 0.}, {0.8736, 0.4944, 0.}, {0.9368, 0.4312, 0.},
 {1., 0.368, 0.}, {1., 0.368, 0.}, {1., 0.3312, 0.0368}, {1., 0.2944, 0.0736}, {1., 0.2576, 0.1104},
 {1., 0.2208, 0.1472}, {1., 0.184, 0.184}, {1., 0.1472, 0.2208}, {1., 0.1104, 0.2576}, {1., 0.0736, 0.2944},
 {1., 0.0368, 0.3312}, {1., 0., 0.368}, {1., 0., 0.368}, {0.9368, 0., 0.4312}, {0.8736, 0., 0.4944},
 {0.8104, 0., 0.5576}, {0.7472, 0., 0.6208}, {0.684, 0., 0.684}, {0.6208, 0., 0.7472}, {0.5576, 0., 0.8104},
 {0.4944, 0., 0.8736}, {0.4312, 0., 0.9368}, {0.368, 0., 1.}, {0.368, 0., 1.}, {0.3312, 0.0368, 1.},
 {0.2944, 0.0736, 1.}, {0.2576, 0.1104, 1.}, {0.2208, 0.1472, 1.}, {0.184, 0.184, 1.},
 {0.1472, 0.2208, 1.}, {0.1104, 0.2576, 1.}, {0.0736, 0.2944, 1.}, {0.0368, 0.3312, 1.},
 {0., 0.368, 1.}, {0., 0.368, 1.}, {0., 0.4312, 0.9368}, {0., 0.4944, 0.8736}, {0., 0.5576, 0.8104},
 {0., 0.6208, 0.7472}, {0., 0.684, 0.684}, {0., 0.7472, 0.6208}, {0., 0.8104, 0.5576},
 {0., 0.8736, 0.4944}, {0., 0.9368, 0.4312}, {0., 1., 0.368}, {0., 1., 0.368}, {0.0368, 1., 0.3312},
 {0.0736, 1., 0.2944}, {0.1104, 1., 0.2576}, {0.1472, 1., 0.2208}, {0.184, 1., 0.184},
 {0.2208, 1., 0.1472}, {0.2576, 1., 0.1104}, {0.2944, 1., 0.0736}, {0.3312, 1., 0.0368}, {0.368, 1., 0.}}
```

In[62]:=
```
aa = 0.456; ({aa, aa, aa} + 0.5 (# - {aa, aa, aa})) & /@ Int[aa]
```

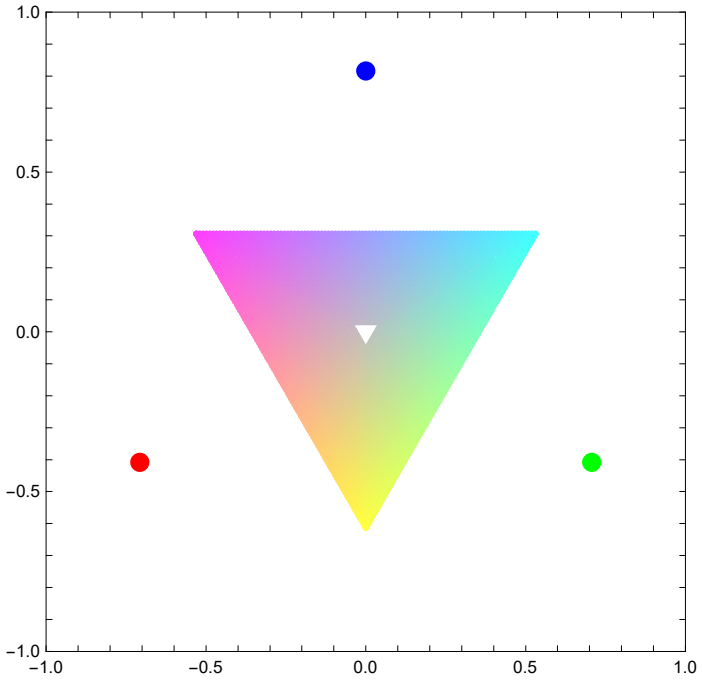
Out[62]=
```
{{0.412, 0.728, 0.228}, {0.728, 0.412, 0.228}, {0.728, 0.228, 0.412},
 {0.412, 0.228, 0.728}, {0.228, 0.412, 0.728}, {0.228, 0.728, 0.412}}
```

In[63]:=
```
aa = 0.456;
Flatten[DotL[#, 0.1] & /@ Partition[Append[({aa, aa, aa} + 0.5 (# - {aa, aa, aa})) & /@ Int[aa],
    First[({aa, aa, aa} + 0.5 (# - {aa, aa, aa})) & /@ Int[aa]]], 2, 1], 1]
```

Out[63]=
```
{{0.412, 0.728, 0.228}, {0.4436, 0.6964, 0.228}, {0.4752, 0.6648, 0.228},
 {0.5068, 0.6332, 0.228}, {0.5384, 0.6016, 0.228}, {0.57, 0.57, 0.228}, {0.6016, 0.5384, 0.228},
 {0.6332, 0.5068, 0.228}, {0.6648, 0.4752, 0.228}, {0.6964, 0.4436, 0.228},
 {0.728, 0.412, 0.228}, {0.728, 0.412, 0.228}, {0.728, 0.3936, 0.2464}, {0.728, 0.3752, 0.2648},
 {0.728, 0.3568, 0.2832}, {0.728, 0.3384, 0.3016}, {0.728, 0.32, 0.32}, {0.728, 0.3016, 0.3384},
 {0.728, 0.2832, 0.3568}, {0.728, 0.2648, 0.3752}, {0.728, 0.2464, 0.3936}, {0.728, 0.228, 0.412},
 {0.728, 0.228, 0.412}, {0.6964, 0.228, 0.4436}, {0.6648, 0.228, 0.4752}, {0.6332, 0.228, 0.5068},
 {0.6016, 0.228, 0.5384}, {0.57, 0.228, 0.57}, {0.5384, 0.228, 0.6016}, {0.5068, 0.228, 0.6332},
 {0.4752, 0.228, 0.6648}, {0.4436, 0.228, 0.6964}, {0.412, 0.228, 0.728}, {0.412, 0.228, 0.728},
 {0.3936, 0.2464, 0.728}, {0.3752, 0.2648, 0.728}, {0.3568, 0.2832, 0.728}, {0.3384, 0.3016, 0.728},
 {0.32, 0.32, 0.728}, {0.3016, 0.3384, 0.728}, {0.2832, 0.3568, 0.728}, {0.2648, 0.3752, 0.728},
 {0.2464, 0.3936, 0.728}, {0.228, 0.412, 0.728}, {0.228, 0.412, 0.728}, {0.228, 0.4436, 0.6964},
 {0.228, 0.4752, 0.6648}, {0.228, 0.5068, 0.6332}, {0.228, 0.5384, 0.6016}, {0.228, 0.57, 0.57},
 {0.228, 0.6016, 0.5384}, {0.228, 0.6332, 0.5068}, {0.228, 0.6648, 0.4752}, {0.228, 0.6964, 0.4436},
 {0.228, 0.728, 0.412}, {0.228, 0.728, 0.412}, {0.2464, 0.728, 0.3936}, {0.2648, 0.728, 0.3752},
 {0.2832, 0.728, 0.3568}, {0.3016, 0.728, 0.3384}, {0.32, 0.728, 0.32}, {0.3384, 0.728, 0.3016},
 {0.3568, 0.728, 0.2832}, {0.3752, 0.728, 0.2648}, {0.3936, 0.728, 0.2464}, {0.412, 0.728, 0.228}}
```

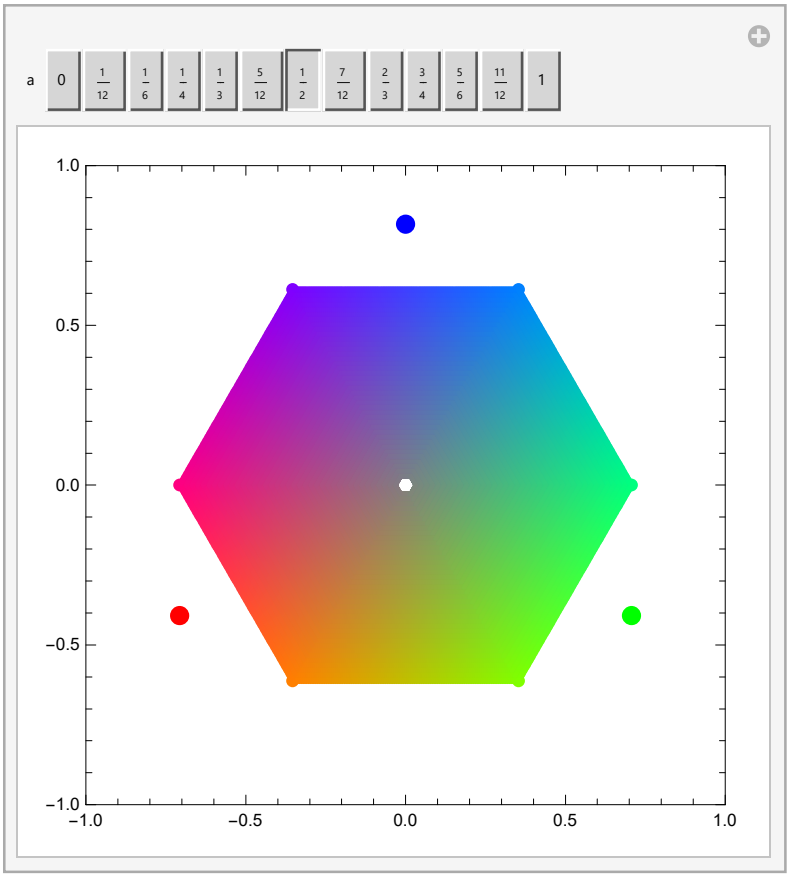
In[64]:=
```
aa = 3 / 4; Graphics[{
  {
   PointSize[0.03],
   {RGBColor[#], Point[Drop[IdentityMatrix[3], -1].Transpose[mU].#]} & /@ {{0, 0, 1}, {0, 1, 0}, {1, 0, 0}
    }},
  {RGBColor[0.5, 0.5, 0.5],
   Thickness[0.002],
   Line[(Drop[IdentityMatrix[3], -1].Transpose[mU].#) & /@ Append[Int[aa], First[Int[aa]]]]
  },
  {
   PointSize[0.01], {RGBColor[#], Point[Drop[IdentityMatrix[3], -1].Transpose[mU].#]} & /@ Int[aa],
   {RGBColor[#], Point[Drop[IdentityMatrix[3], -1].Transpose[mU].#]} & /@
    Flatten[DotL[#, 0.01] & /@ Partition[Append[Int[aa], First[Int[aa]]], 2, 1], 1],

   {RGBColor[#], Point[Drop[IdentityMatrix[3], -1].Transpose[mU].#]} & /@ Flatten[
     Table[(Flatten[DotL[#, 0.01] & /@ Partition[Append[({aa, aa, aa} + r (# - {aa, aa, aa})) & /@ Int[aa],
           First[({aa, aa, aa} + r (# - {aa, aa, aa})) & /@ Int[aa]]], 2, 1], 1]), {r, 0.1, 1, .05}], 1]
  }
}, PlotRange → {{-1, 1}, {-1, 1}}, Frame → True]
```

Out[64]=



In[65]:=
```
Manipulate[
 ttt = Flatten[Table[(Flatten[DotL[#, 0.01] & /@ Partition[Append[({a, a, a} + r (# - {a, a, a})) & /@ Int[a],
         First[({a, a, a} + r (# - {a, a, a})) & /@ Int[a]]], 2, 1], 1]), {r, 0.05, 1, .025}], 1];
 Graphics[{
   {
    PointSize[0.03],
    {RGBColor[#], Point[Drop[IdentityMatrix[3], -1].Transpose[mU].#]} & /@ {{0, 0, 1}, {0, 1, 0}, {1, 0, 0}
     }},
   {RGBColor[0.5, 0.5, 0.5],
    Thickness[0.002],
    Line[(Drop[IdentityMatrix[3], -1].Transpose[mU].#) & /@ Append[Int[a], First[Int[a]]]]
   },
   {
    PointSize[0.02],
    {RGBColor[#], Point[Drop[IdentityMatrix[3], -1].Transpose[mU].#]} & /@ Int[a], PointSize[0.01],
    {RGBColor[#], Point[Drop[IdentityMatrix[3], -1].Transpose[mU].#]} & /@ ttt
   }
  }, PlotRange → {{-1, 1}, {-1, 1}}, Frame → True],
 {{a, 1 / 2}, Range[0, 1, 1/12], Setter}, ControlPlacement → Top]
```

Out[65]=




In[66]:=
```
vp = {1, 2, 1}
```

Out[66]= {1, 2, 1}

In[67]:=
```
vp
```

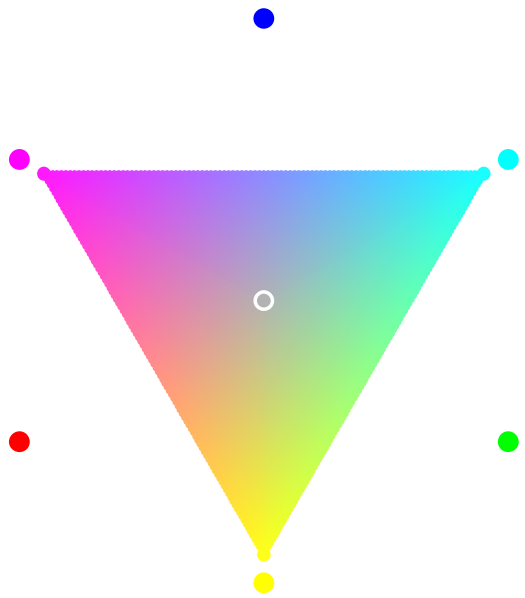
Out[67]= {1, 2, 1}

In[68]:=
```
{1.44, 1.8, 0.8}
```

Out[68]= {1.44, 1.8, 0.8}

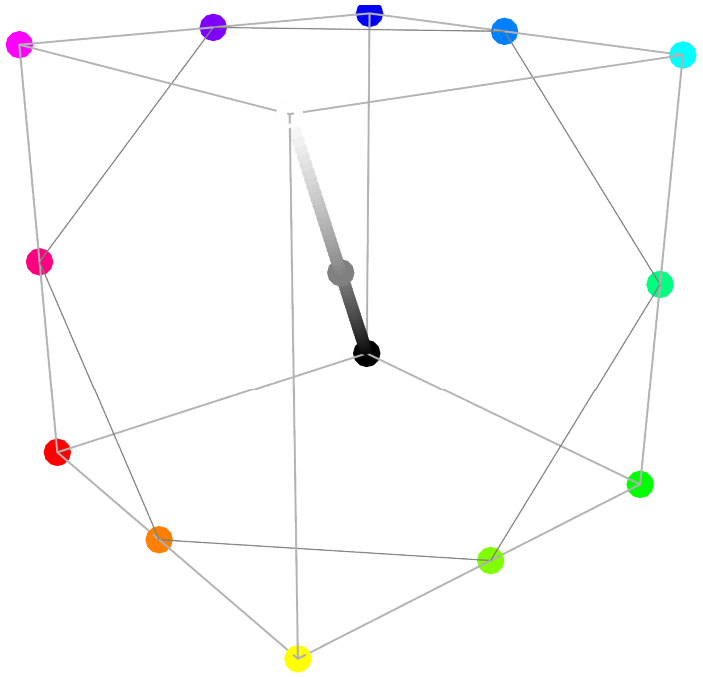
In[69]:=
```
a = 0.7; Graphics[{
  {
   PointSize[0.03],
   {RGBColor[#], Point[Drop[IdentityMatrix[3], -1].Transpose[mU].#]} & /@ {{0, 0, 1}, {0, 1, 0}, {1, 0, 0},
     {0, 1, 1}, {1, 0, 1}, {1, 1, 0}, {0, 0, 0}, {1, 1, 1}},
   {RGBColor[#], Point[Drop[IdentityMatrix[3], -1].Transpose[mU].#]} &[{a, a, a}]
  },
  {RGBColor[0.5, 0.5, 0.5],
   Thickness[0.002],
   Line[(Drop[IdentityMatrix[3], -1].Transpose[mU].#) & /@ Append[Int[a], First[Int[a]]]]
  },
  {
   PointSize[0.02],
   {RGBColor[#], Point[Drop[IdentityMatrix[3], -1].Transpose[mU].#]} & /@ Int[a], PointSize[0.01],
   Table[{RGBColor[#], Point[Drop[IdentityMatrix[3], -1].Transpose[mU].#]} & /@
     (Flatten[DotL[#, 0.01] & /@ Partition[Append[({a, a, a} + r (# - {a, a, a})) & /@ Int[a],
          First[({a, a, a} + r (# - {a, a, a})) & /@ Int[a]]], 2, 1], 1]),
    {r, 0.05, 1, .025}], PointSize[0.03], RGBColor[1, 1, 1], Point[{0, 0}],
   PointSize[0.02], {RGBColor[#], Point[Drop[IdentityMatrix[3], -1].Transpose[mU].#]} &[{a, a, a}]
  }
}, PlotRange → {{-1, 1}, {-1, 1}}, Frame → False]
```

Out[69]=



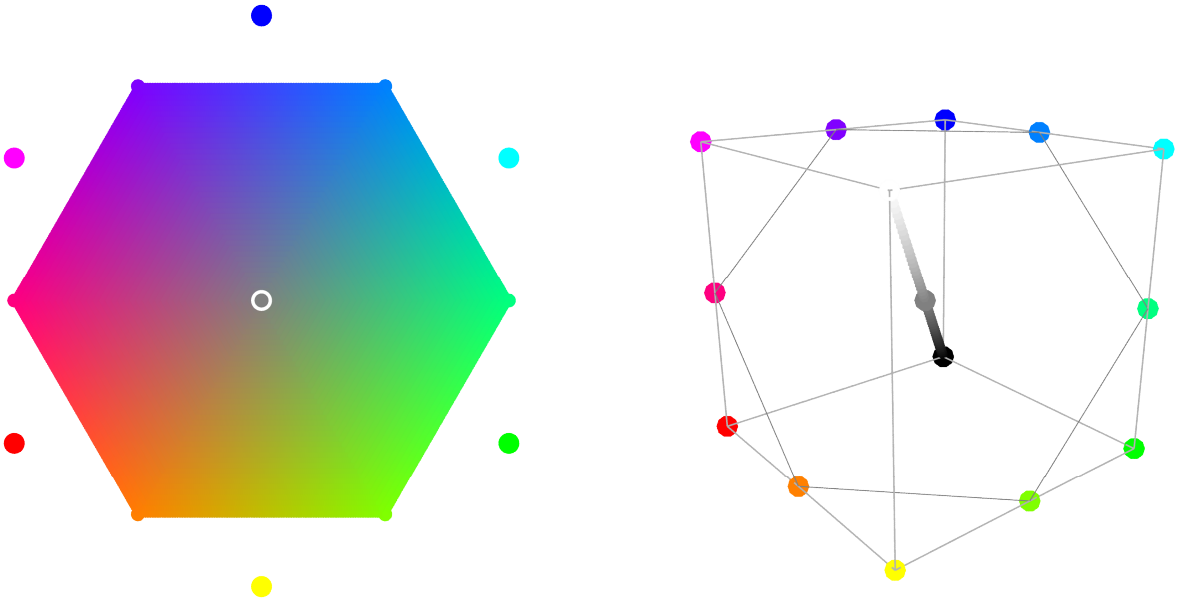
In[70]:=
```
a = 1 / 2; Graphics3D[{
  {RGBColor[0.7, 0.7, 0.7], Thickness[0.003],
   Line[{{0, 0, 0}, {1, 0, 0}, {1, 1, 0}, {0, 1, 0}, {0, 0, 0}}],
   Line[{{0, 0, 1}, {1, 0, 1}, {1, 1, 1}, {0, 1, 1}, {0, 0, 1}}],
   Line[{{0, 0, 0}, {0, 0, 1}}], Line[{{1, 0, 0}, {1, 0, 1}}],
   Line[{{0, 1, 0}, {0, 1, 1}}], Line[{{1, 1, 0}, {1, 1, 1}}]},
  {
   PointSize[0.04], {RGBColor[#], Point[#]} & /@ {{0, 0, 1}, {0, 1, 0}, {1, 0, 0},
     {0, 1, 1}, {1, 0, 1}, {1, 1, 0}, {0, 0, 0}, {1, 1, 1}}, PointSize[0.02],
   {RGBColor[#], Point[#]} & /@ DotL[{{0, 0, 0}, {1, 1, 1}}, 0.02]
  },
  {RGBColor[0.5, 0.5, 0.5], Thickness[0.002], Line[Append[Int[a], First[Int[a]]]]},
  {
   PointSize[0.04], {RGBColor[#], Point[#]} & /@ Int[a], {RGBColor[#], Point[#]} &[{a, a, a}]
  }
 }, PlotRange → {{-0.02, 1.02}, {-0.02, 1.02}, {-0.02, 1.02}},
 Axes → False, Boxed → False, ViewPoint → {1.44, 1.8, 0.8}]
```

Out[70]=



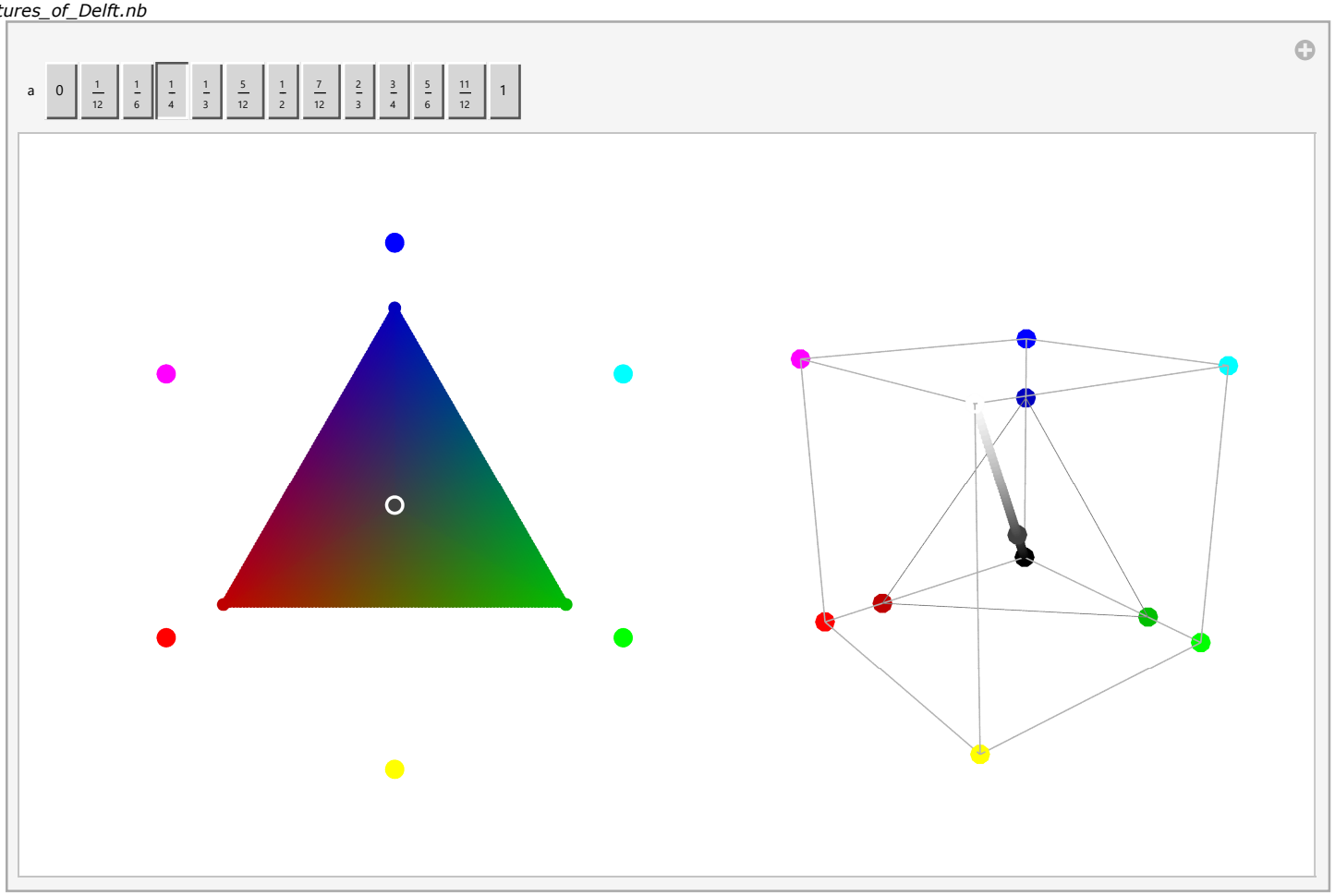
In[71]:=
```
a = 1 / 2; GraphicsRow[{
 Graphics[{
   {
    PointSize[0.03], {RGBColor[#], Point[Drop[IdentityMatrix[3], -1].Transpose[mU].#]} & /@
     {{0, 0, 1}, {0, 1, 0}, {1, 0, 0},
      {0, 1, 1}, {1, 0, 1}, {1, 1, 0}, {0, 0, 0}, {1, 1, 1}},
    {RGBColor[#], Point[Drop[IdentityMatrix[3], -1].Transpose[mU].#]} &[{a, a, a}]
   },
   {RGBColor[0.5, 0.5, 0.5],
    Thickness[0.002],
    Line[(Drop[IdentityMatrix[3], -1].Transpose[mU].#) & /@ Append[Int[a], First[Int[a]]]]
   },
   {
    PointSize[0.02],
    {RGBColor[#], Point[Drop[IdentityMatrix[3], -1].Transpose[mU].#]} & /@ Int[a], PointSize[0.01],
    Table[{RGBColor[#], Point[Drop[IdentityMatrix[3], -1].Transpose[mU].#]} & /@
      (Flatten[DotL[#, 0.01] & /@ Partition[Append[({a, a, a} + r (# - {a, a, a})) & /@ Int[a],
           First[({a, a, a} + r (# - {a, a, a})) & /@ Int[a]]], 2, 1], 1]),
     {r, 0.05, 1, .025}], PointSize[0.03], RGBColor[1, 1, 1], Point[{0, 0}],
    PointSize[0.02], {RGBColor[#], Point[Drop[IdentityMatrix[3], -1].Transpose[mU].#]} &[{a, a, a}]
   }
  }, PlotRange → {{-1, 1}, {-1, 1}}, Frame → False],
 Graphics3D[{
   {RGBColor[0.7, 0.7, 0.7], Thickness[0.003],
    Line[{{0, 0, 0}, {1, 0, 0}, {1, 1, 0}, {0, 1, 0}, {0, 0, 0}}],
    Line[{{0, 0, 1}, {1, 0, 1}, {1, 1, 1}, {0, 1, 1}, {0, 0, 1}}],
    Line[{{0, 0, 0}, {0, 0, 1}}], Line[{{1, 0, 0}, {1, 0, 1}}],
    Line[{{0, 1, 0}, {0, 1, 1}}], Line[{{1, 1, 0}, {1, 1, 1}}]},
   {
    PointSize[0.04], {RGBColor[#], Point[#]} & /@ {{0, 0, 1}, {0, 1, 0}, {1, 0, 0},
      {0, 1, 1}, {1, 0, 1}, {1, 1, 0}, {0, 0, 0}, {1, 1, 1}}, PointSize[0.02],
    {RGBColor[#], Point[#]} & /@ DotL[{{0, 0, 0}, {1, 1, 1}}, 0.02]
   },
   {RGBColor[0.5, 0.5, 0.5], Thickness[0.002], Line[Append[Int[a], First[Int[a]]]]},
   {
    PointSize[0.04], {RGBColor[#], Point[#]} & /@ Int[a], {RGBColor[#], Point[#]} &[{a, a, a}]
   }
  }, PlotRange → {{-0.02, 1.02}, {-0.02, 1.02}, {-0.02, 1.02}},
  Axes → False, Boxed → False, ViewPoint → {1.44, 1.8, 0.8}]
},
ImageSize →
 700]
```

Out[71]=



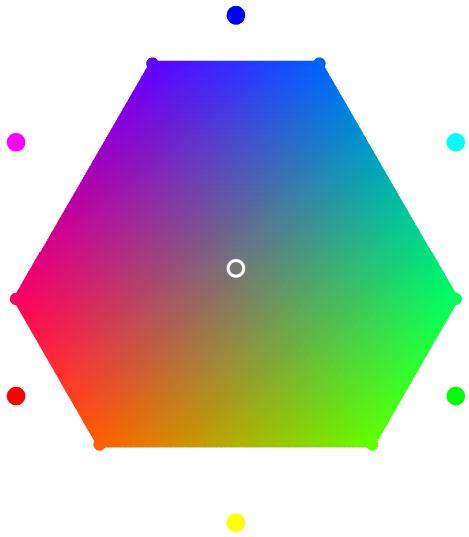
In[72]:=
```
Clear[a]; Manipulate[GraphicsRow[{
  Graphics[{
    {
     PointSize[0.03], {RGBColor[#], Point[Drop[IdentityMatrix[3], -1].Transpose[mU].#]} & /@
      {{0, 0, 1}, {0, 1, 0}, {1, 0, 0},
       {0, 1, 1}, {1, 0, 1}, {1, 1, 0}, {0, 0, 0}, {1, 1, 1}},
     {RGBColor[#], Point[Drop[IdentityMatrix[3], -1].Transpose[mU].#]} &[{a, a, a}]
    },
    {RGBColor[0.5, 0.5, 0.5],
     Thickness[0.002],
     Line[(Drop[IdentityMatrix[3], -1].Transpose[mU].#) & /@ Append[Int[a], First[Int[a]]]]
    },
    {
     PointSize[0.02],
     {RGBColor[#], Point[Drop[IdentityMatrix[3], -1].Transpose[mU].#]} & /@ Int[a], PointSize[0.01],
     Table[{RGBColor[#], Point[Drop[IdentityMatrix[3], -1].Transpose[mU].#]} & /@
       (Flatten[DotL[#, 0.01] & /@ Partition[Append[({a, a, a} + r (# - {a, a, a})) & /@ Int[a],
            First[({a, a, a} + r (# - {a, a, a})) & /@ Int[a]]], 2, 1], 1]),
      {r, 0.05, 1, .025}], PointSize[0.03], RGBColor[1, 1, 1], Point[{0, 0}],
     PointSize[0.02], {RGBColor[#], Point[Drop[IdentityMatrix[3], -1].Transpose[mU].#]} &[{a, a, a}]
    }
   }, PlotRange → {{-1, 1}, {-1, 1}}, Frame → False],
  Graphics3D[{
    {RGBColor[0.7, 0.7, 0.7], Thickness[0.003],
     Line[{{0, 0, 0}, {1, 0, 0}, {1, 1, 0}, {0, 1, 0}, {0, 0, 0}}],
     Line[{{0, 0, 1}, {1, 0, 1}, {1, 1, 1}, {0, 1, 1}, {0, 0, 1}}],
     Line[{{0, 0, 0}, {0, 0, 1}}], Line[{{1, 0, 0}, {1, 0, 1}}],
     Line[{{0, 1, 0}, {0, 1, 1}}], Line[{{1, 1, 0}, {1, 1, 1}}]},
    {
     PointSize[0.04], {RGBColor[#], Point[#]} & /@ {{0, 0, 1}, {0, 1, 0}, {1, 0, 0},
       {0, 1, 1}, {1, 0, 1}, {1, 1, 0}, {0, 0, 0}, {1, 1, 1}}, PointSize[0.02],
     {RGBColor[#], Point[#]} & /@ DotL[{{0, 0, 0}, {1, 1, 1}}, 0.02]
    },
    {RGBColor[0.5, 0.5, 0.5], Thickness[0.002], Line[Append[Int[a], First[Int[a]]]]},
    {
     PointSize[0.04], {RGBColor[#], Point[#]} & /@ Int[a], {RGBColor[#], Point[#]} &[{a, a, a}]
    }
   }, PlotRange → {{-0.02, 1.02}, {-0.02, 1.02}, {-0.02, 1.02}},
   Axes → False, Boxed → False, ViewPoint → {1.44, 1.8, 0.8}]
}, ImageSize → 700], {{a, 1/4}, Range[0, 1, 1/12], Setter}, ControlPlacement → Top]
```

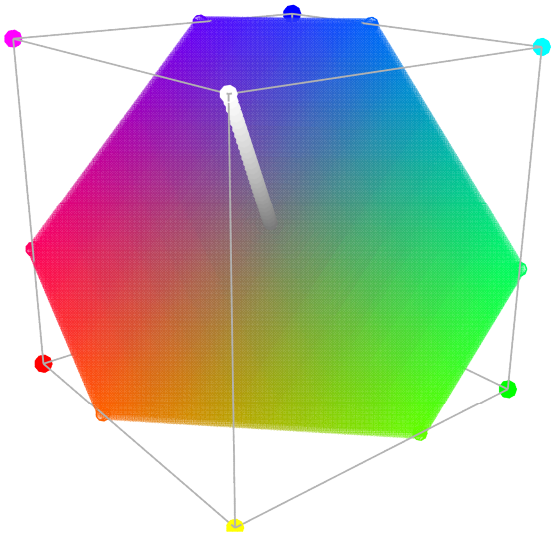
Out[72]=



In[73]:=
```
Timing[testpic = Module[{a, ttt}, a = N[23 / 50];
  ttt = Flatten[Table[(Flatten[DotL[#, 0.01] & /@ Partition[Append[({a, a, a} + r (# - {a, a, a})) & /@ Int[a],
          First[({a, a, a} + r (# - {a, a, a})) & /@ Int[a]]], 2, 1], 1]), {r, 0.05, 1, .025}], 1];
  GraphicsRow[{
    Graphics[{
      {
       PointSize[0.03], {RGBColor[#], Point[Drop[IdentityMatrix[3], -1].Transpose[mU].#]} & /@
        {{0, 0, 1}, {0, 1, 0}, {1, 0, 0},
         {0, 1, 1}, {1, 0, 1}, {1, 1, 0}, {0, 0, 0}, {1, 1, 1}},
       {RGBColor[#], Point[Drop[IdentityMatrix[3], -1].Transpose[mU].#]} &[{a, a, a}]
      },
      {RGBColor[0.5, 0.5, 0.5],
       Thickness[0.002],
       Line[(Drop[IdentityMatrix[3], -1].Transpose[mU].#) & /@ Append[Int[a], First[Int[a]]]]
      },
      {
       PointSize[0.02], {RGBColor[#], Point[Drop[IdentityMatrix[3], -1].Transpose[mU].#]} & /@ Int[a],
       PointSize[0.01], {RGBColor[#], Point[Drop[IdentityMatrix[3], -1].Transpose[mU].#]} & /@ ttt,
       PointSize[0.03], RGBColor[1, 1, 1], Point[{0, 0}],
       PointSize[0.02], {RGBColor[#], Point[Drop[IdentityMatrix[3], -1].Transpose[mU].#]} &[{a, a, a}]
      }
     }, PlotRange → {{-1, 1}, {-1, 1}}, Frame → False, ImageSize → 400],
    Graphics3D[{
      {RGBColor[0.7, 0.7, 0.7], Thickness[0.003],
       Line[{{0, 0, 0}, {1, 0, 0}, {1, 1, 0}, {0, 1, 0}, {0, 0, 0}}],
       Line[{{0, 0, 1}, {1, 0, 1}, {1, 1, 1}, {0, 1, 1}, {0, 0, 1}}],
       Line[{{0, 0, 0}, {0, 0, 1}}], Line[{{1, 0, 0}, {1, 0, 1}}],
       Line[{{0, 1, 0}, {0, 1, 1}}], Line[{{1, 1, 0}, {1, 1, 1}}]},
      {
       PointSize[0.03], {RGBColor[#], Point[#]} & /@ {{0, 0, 1}, {0, 1, 0}, {1, 0, 0},
         {0, 1, 1}, {1, 0, 1}, {1, 1, 0}, {0, 0, 0}, {1, 1, 1}}, PointSize[0.025],
       {RGBColor[#], Point[#]} & /@ DotL[{{0, 0, 0}, {1, 1, 1}}, 0.02]
      },
      {
       PointSize[0.025], {RGBColor[#], Point[#]} & /@ Int[a],
       PointSize[0.02], Opacity[0.35],
       {RGBColor[#], Point[#]} & /@ ttt, {RGBColor[#], PointSize[0.03], Point[#]} &[{a, a, a}]
      }
     }, PlotRange → {{-0.02, 1.02}, {-0.02, 1.02}, {-0.02, 1.02}},
     Axes → False, Boxed → False, ViewPoint → {1.44, 1.8, 0.8}, ImageSize → 400]
   }, ImageSize → 700]]]
```

Out[73]= {1.96875,

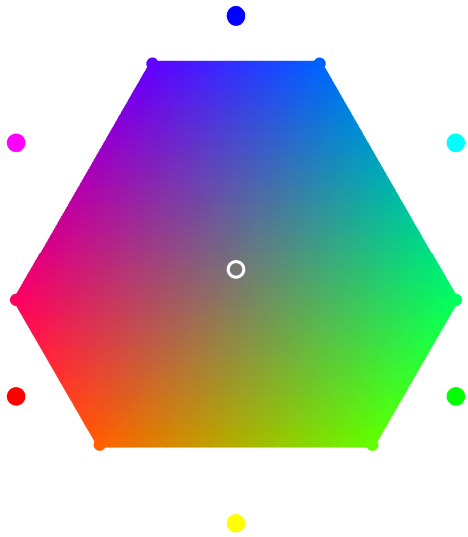


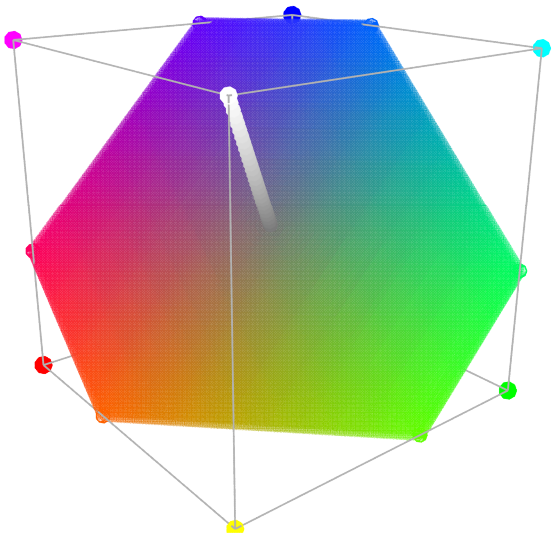


}

In[74]:= **Show[testpic]**

Out[74]=





In[75]:= **Directory[]**

Out[75]= C:\Dropbox\Work\myweb\Courses\Math_pages\Math_204

In[76]:= **(* Timing[Export["testpic5.gif",testpic,"GIF","ImageSize"→700]] *)**

In[77]:= **1 / 75.**

Out[77]= 0.0133333

In[78]:=
```
Options[GraphicsRow]
```

Out[78]=
```
{Alignment → Automatic, AlignmentPoint → Center, AspectRatio → Automatic, Axes → False, AxesLabel → None,
 AxesOrigin → Automatic, AxesStyle → {}, Background → None, BaselinePosition → Automatic, BaseStyle → {},
 ColorOutput → Automatic, ContentSelectable → Automatic, CoordinatesToolOptions → Automatic,
 DefaultBaseStyle → GraphicsGrid, DisplayFunction :→ $DisplayFunction, Dividers → None, Epilog → {},
 FormatType :→ TraditionalForm, Frame → None, FrameLabel → None, FrameStyle → Automatic,
 FrameTicks → Automatic, FrameTicksStyle → {}, GridLines → None, GridLinesStyle → {}, ImageMargins → 0.,
 ImagePadding → All, ImageSize → Automatic, ImageSizeRaw → Automatic, ItemAspectRatio → Automatic,
 LabelStyle → {}, Method → Automatic, PlotLabel → None, PlotRange → All, PlotRangeClipping → False,
 PlotRangePadding → Automatic, PlotRegion → Automatic, PreserveImageOptions → Automatic,
 Prolog → {}, RotateLabel → True, Spacings → Scaled[0.1], Ticks → Automatic, TicksStyle → {}}
```

```
(* Timing[BandWani=Table[Module[{a,ttt},a=t;
     ttt=Flatten[Table[(Flatten[DotL[#,0.01]&/@Partition[Append[({a,a,a}+r(#-{a,a,a}))&/@Int[a],
           First[({a,a,a}+r(#-{a,a,a}))&/@Int[a]]],2,1],1]),{r,0.05,1,.025}],1];
     GraphicsRow[{
       Graphics[{
         {
          PointSize[0.03],
          {RGBColor[#],Point[Drop[IdentityMatrix[3],-1].Transpose[mU].#]}&/@{{0,0,1},{0,1,0},{1,0,0},
            {0,1,1},{1,0,1},{1,1,0},{0,0,0},{1,1,1}},
          {RGBColor[#],Point[Drop[IdentityMatrix[3],-1].Transpose[mU].#]}&[{a,a,a}]
         },
         {RGBColor[0.5,0.5,0.5],
          Thickness[0.002],
          Line[(Drop[IdentityMatrix[3],-1].Transpose[mU].#)&/@Append[Int[a],First[Int[a]]]]
         },
         {
          PointSize[0.02],{RGBColor[#],Point[Drop[IdentityMatrix[3],-1].Transpose[mU].#]}&/@Int[a],
          PointSize[0.01],{RGBColor[#],Point[Drop[IdentityMatrix[3],-1].Transpose[mU].#]}&/@ttt,
          PointSize[0.03],RGBColor[1,1,1],Point[{0,0}],
          PointSize[0.02],{RGBColor[#],Point[Drop[IdentityMatrix[3],-1].Transpose[mU].#]}&[{a,a,a}]
         }
        },PlotRange→{{-1,1},{-1,1}},Frame→False,ImageSize→400],
       Graphics3D[{
         {RGBColor[0.7,0.7,0.7],Thickness[0.003],
          Line[{{0,0,0},{1,0,0},{1,1,0},{0,1,0},{0,0,0}}],
          Line[{{0,0,1},{1,0,1},{1,1,1},{0,1,1},{0,0,1}}],
          Line[{{0,0,0},{0,0,1}}],Line[{{1,0,0},{1,0,1}}],
          Line[{{0,1,0},{0,1,1}}],Line[{{1,1,0},{1,1,1}}]},
         {
          PointSize[0.03],{RGBColor[#],Point[#]}&/@{{0,0,1},{0,1,0},{1,0,0},
            {0,1,1},{1,0,1},{1,1,0},{0,0,0},{1,1,1}},PointSize[0.025],
          {RGBColor[#],Point[#]}&/@DotL[{{0,0,0},{1,1,1}},0.02]
         },
         {
          PointSize[0.025],{RGBColor[#],Point[#]}&/@Int[a],
          PointSize[0.02],Opacity[0.35],
          {RGBColor[#],Point[#]}&/@ttt,{RGBColor[#],PointSize[0.03],Point[#]}&[{a,a,a}]
         }
        },PlotRange→{{-0.02,1.02},{-0.02,1.02},{-0.02,1.02}},
        Axes→False,Boxed→False,ViewPoint→{1.44,1.8,0.8},ImageSize→400]
      },ImageSize→700]],{t,0,1,0.01}];] *)

(* BandWani[[45]] *)
```

In[82]:=

```
(* Export["BandWaniS1.gif",BandWani[[30]],"gif","ImageSize"→700]; *)

(* dds=0.5&/@Range[Length[BandWani]];
 Export["BandWani.gif",BandWani,"gif",
 "AnimationRepetitions"→1,"ImageSize"→700,"DisplayDurations"→dds]; *)
```