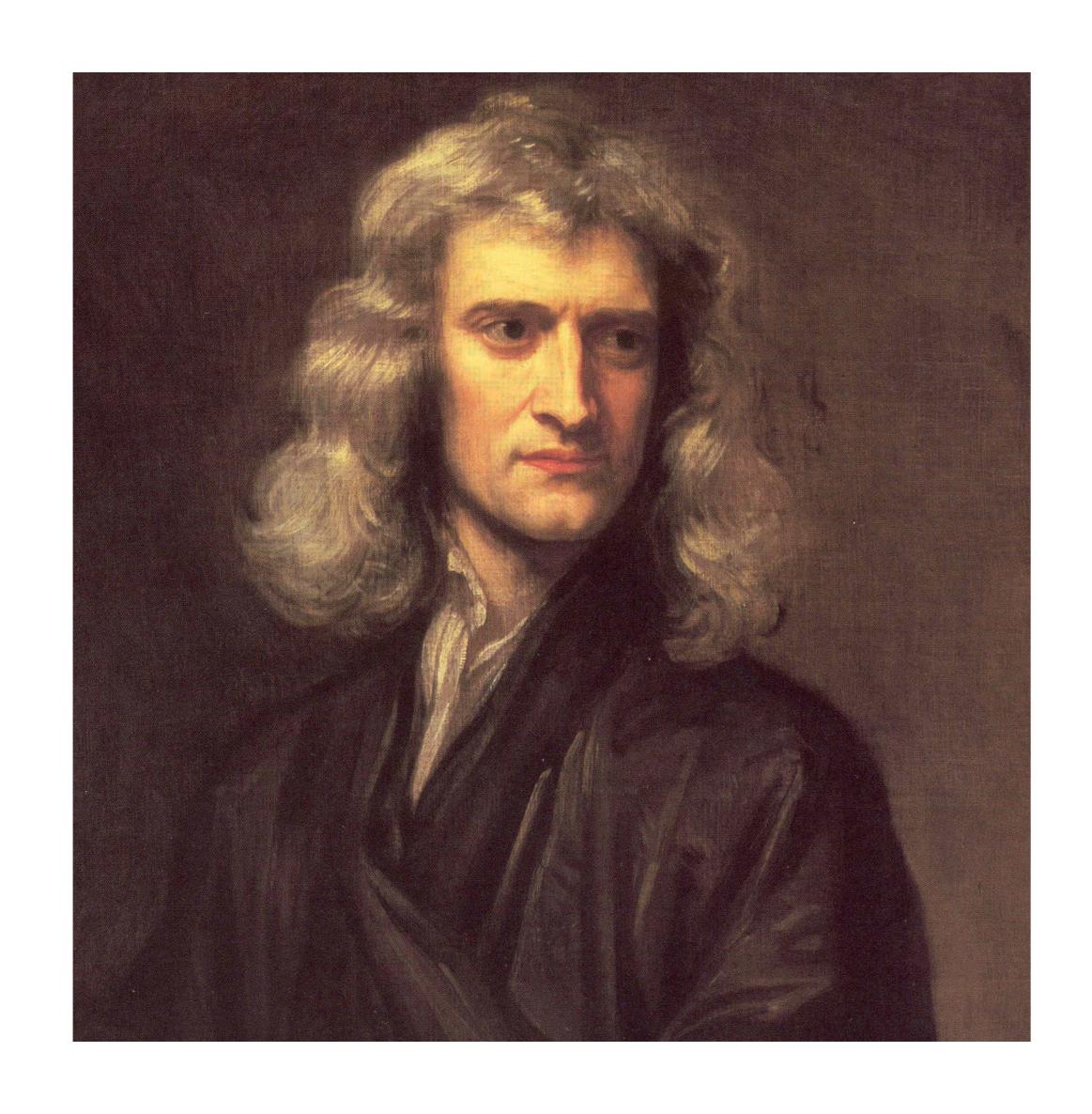
## Color Schemes

Creative Data Mining Ricardo Joss

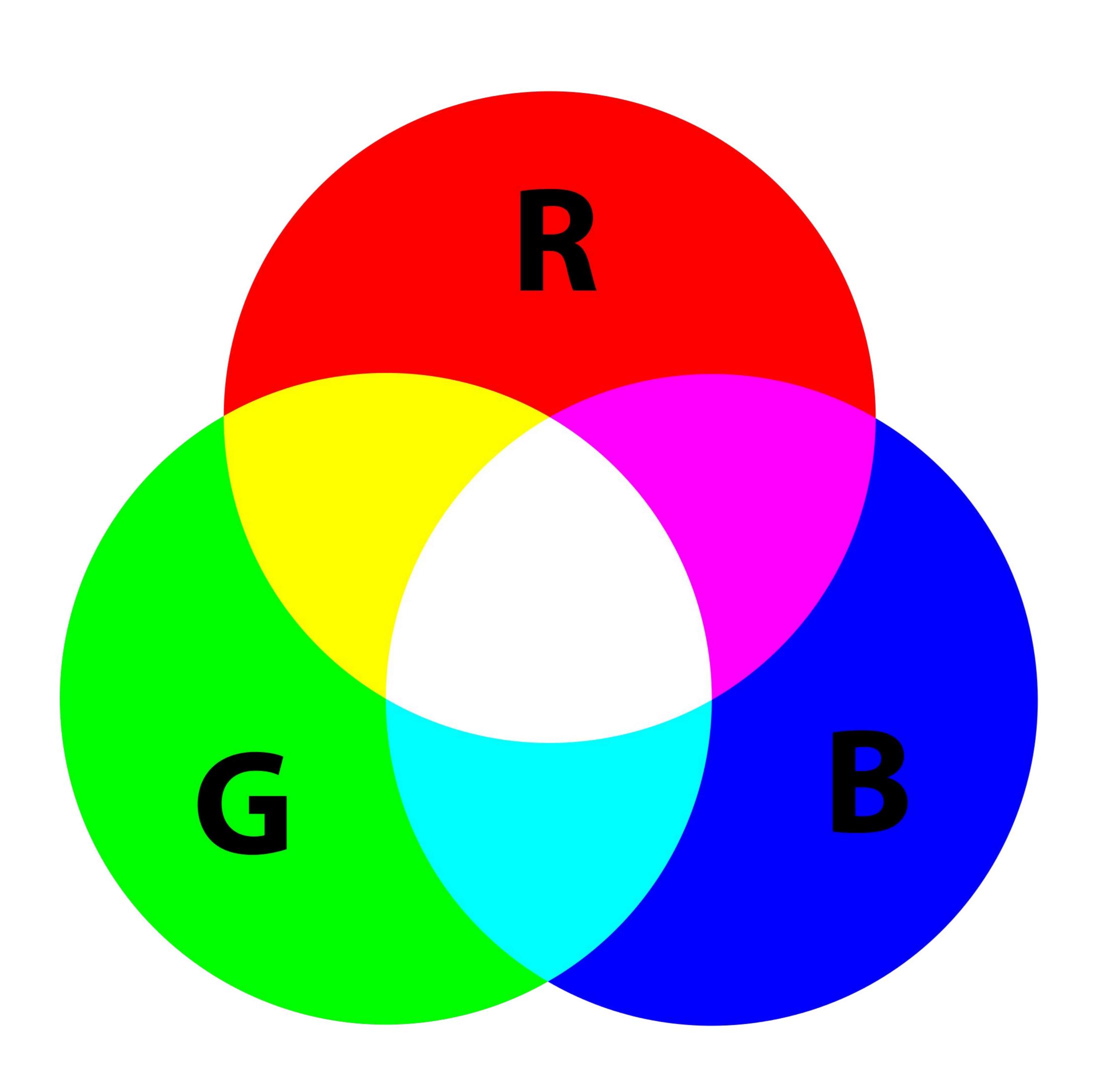
Is there a link of the pure color schemes of a place, to it's perception of...

... beauty / ugliness
... lightness / darkness
... openness / enclosedness
... order / chaos

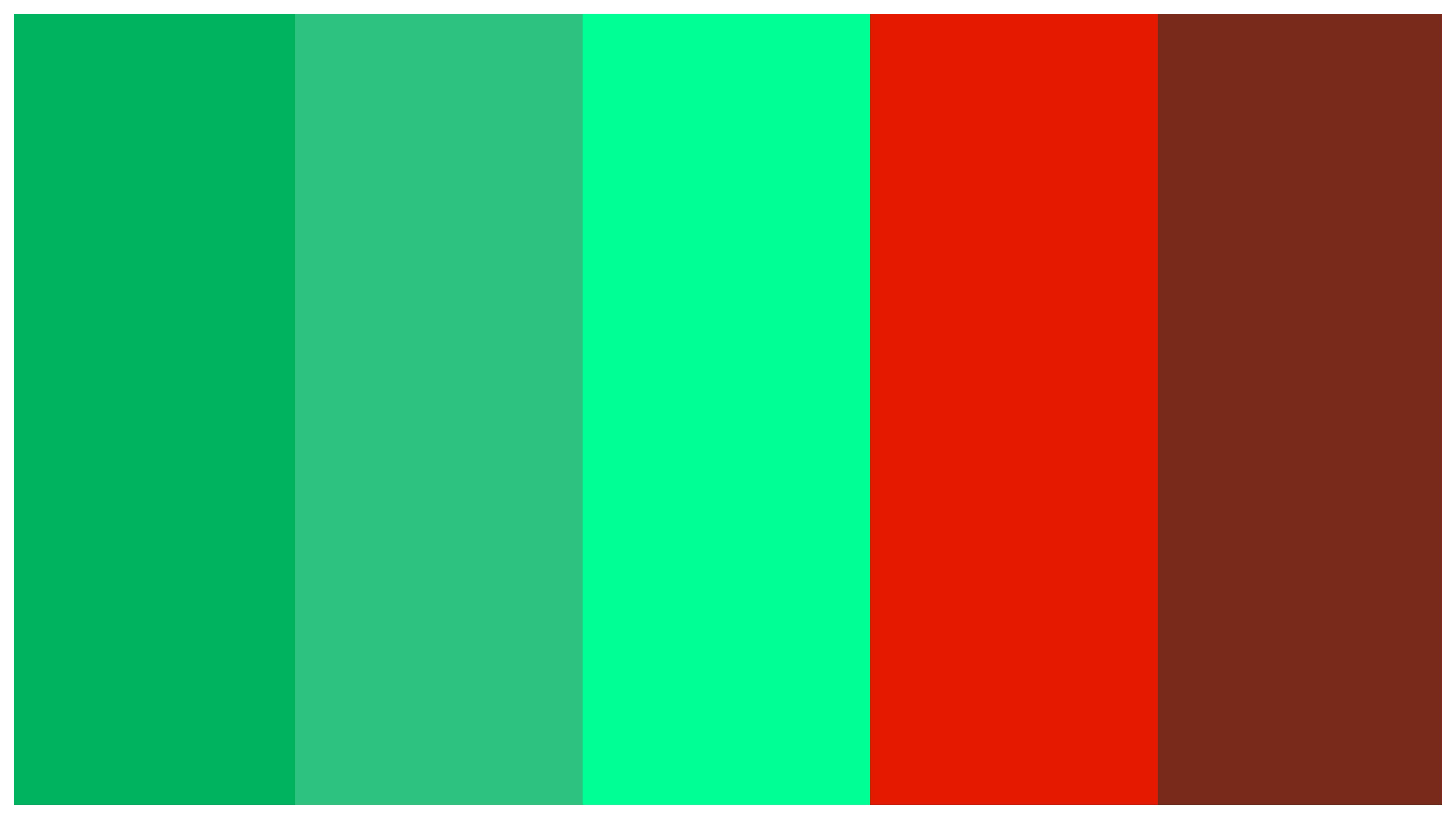


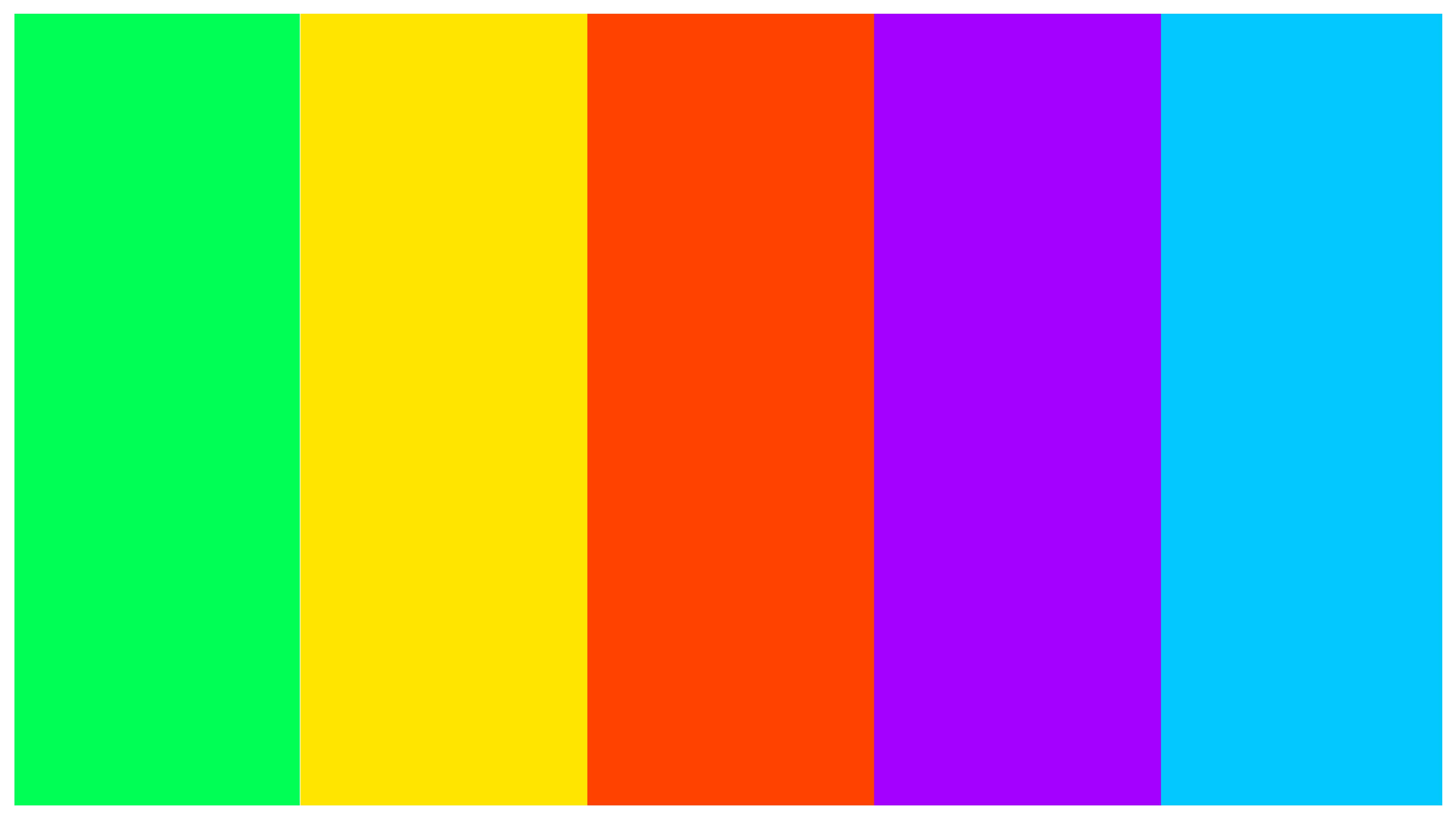


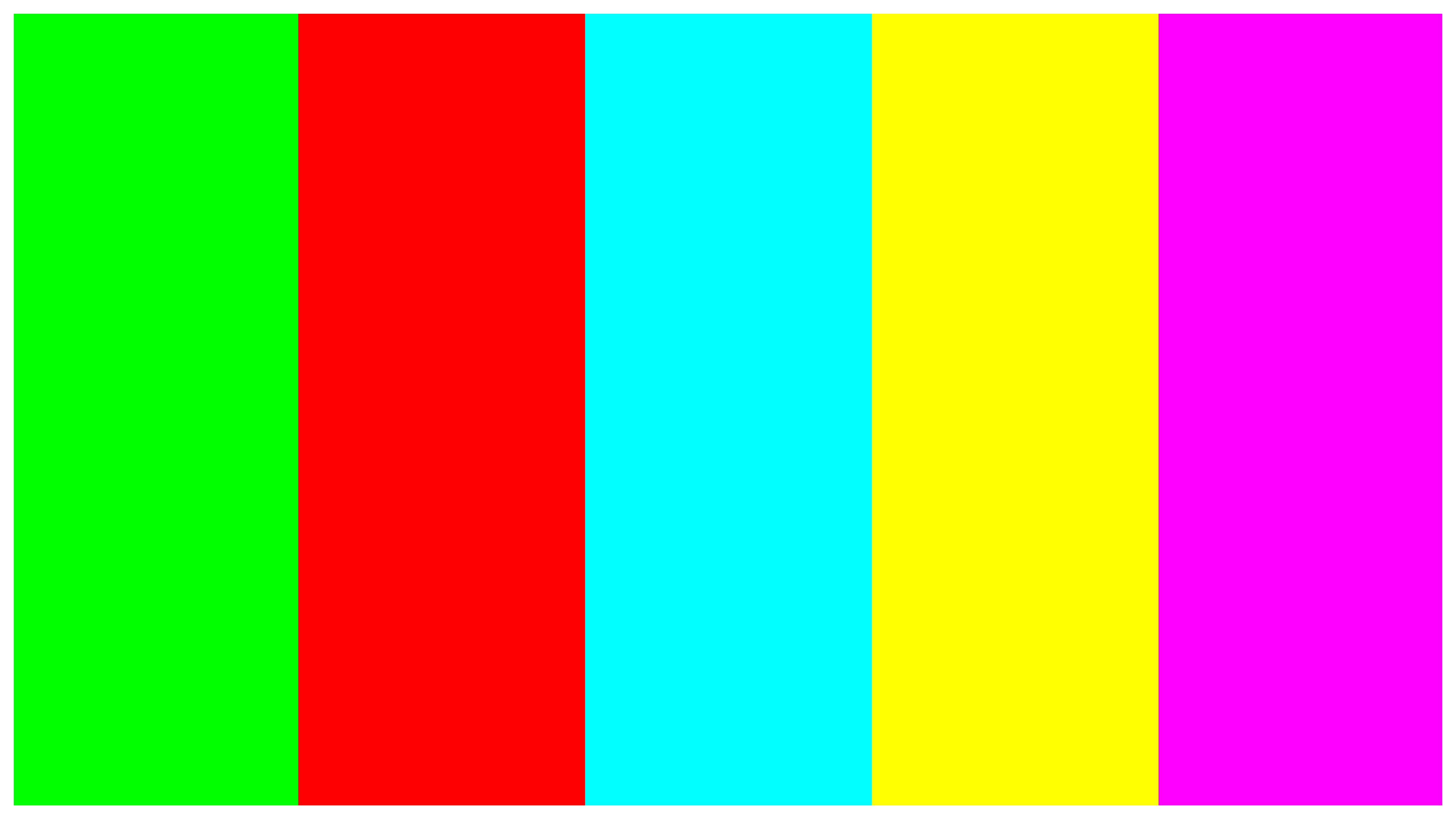






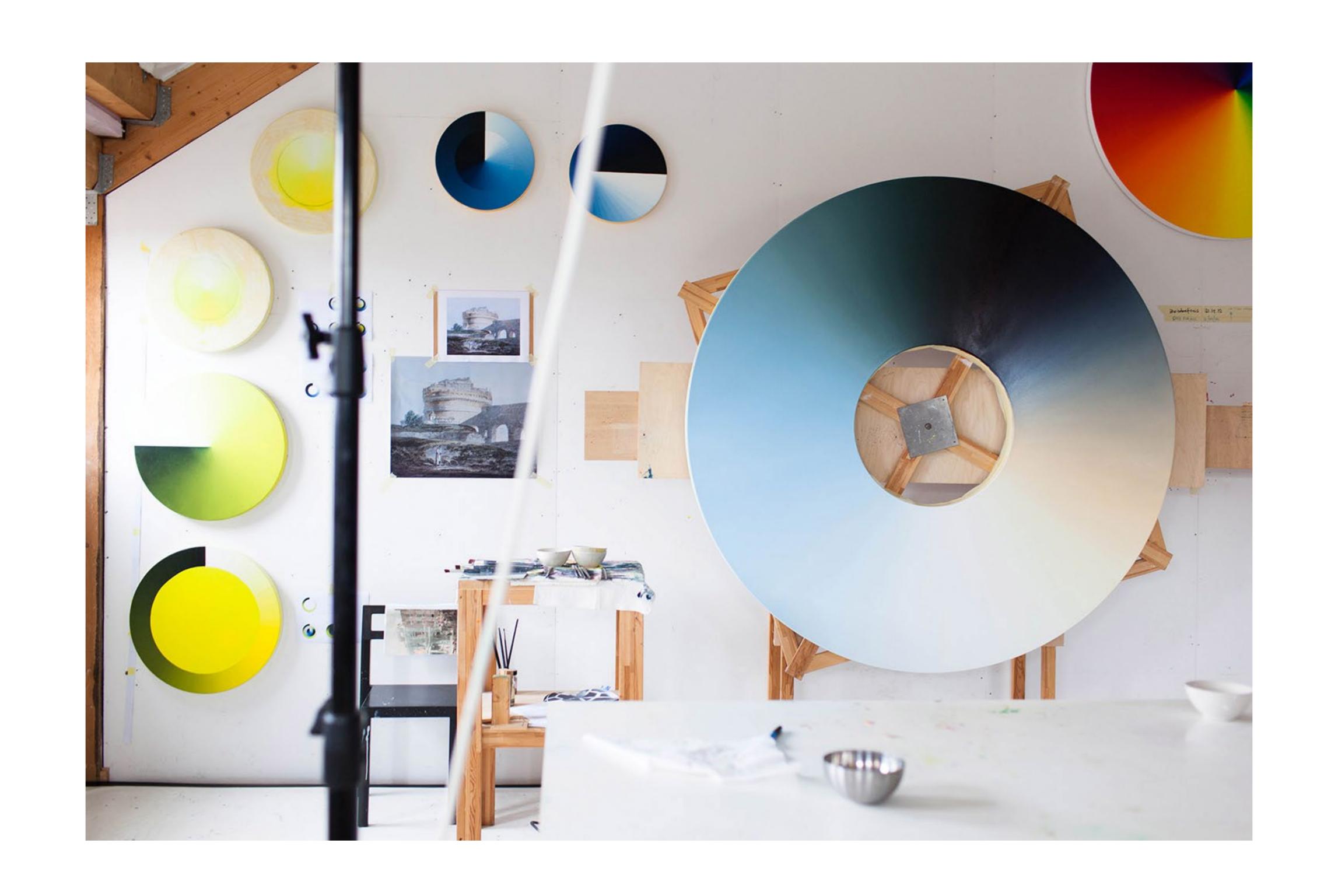


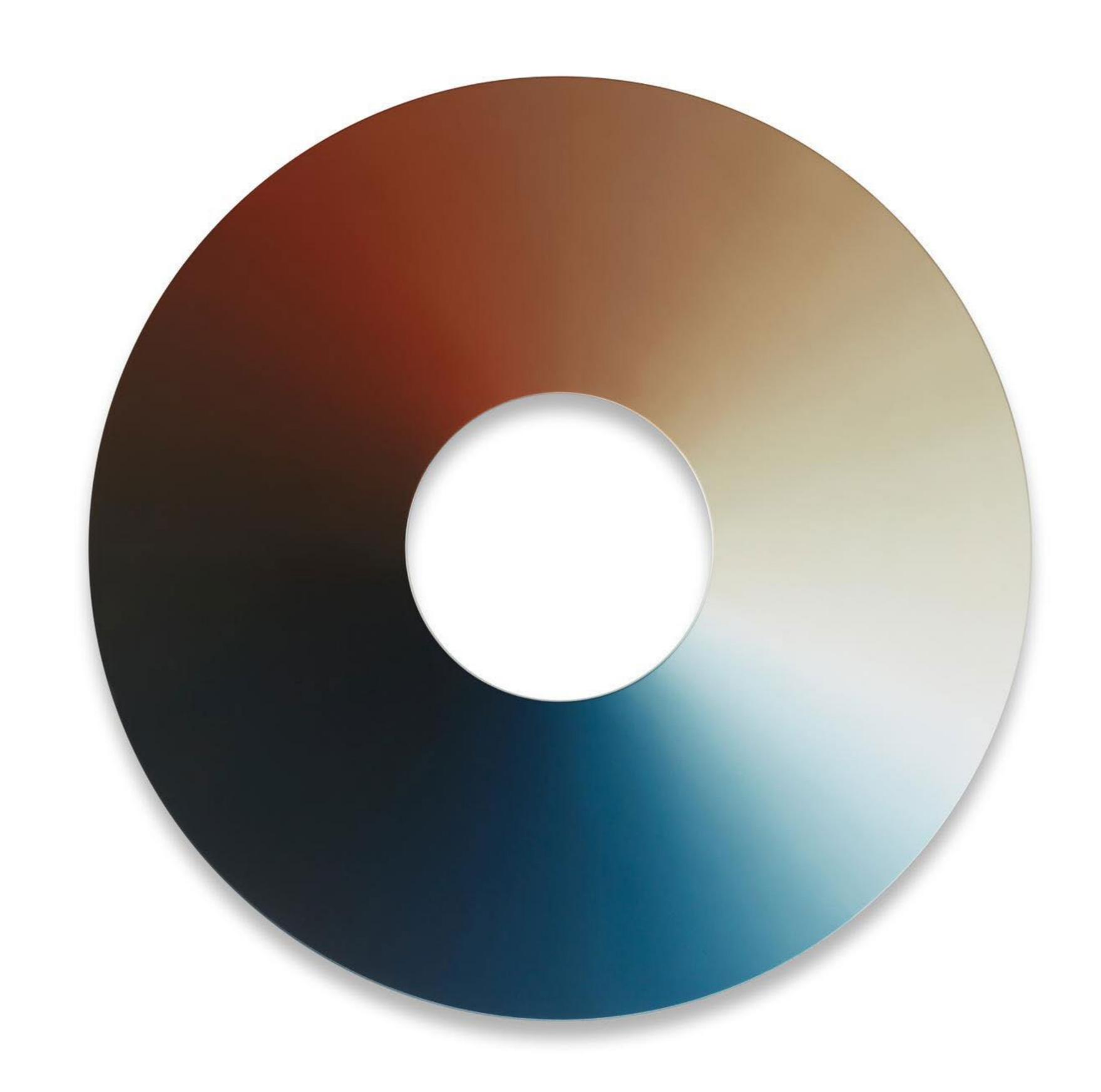






JMW Turner, Ancient Rome, 1839





Is there a link of the pure color schemes of a place, to it's perception of...

... beauty / ugliness
... lightness / darkness
... openness / enclosedness
... order / chaos

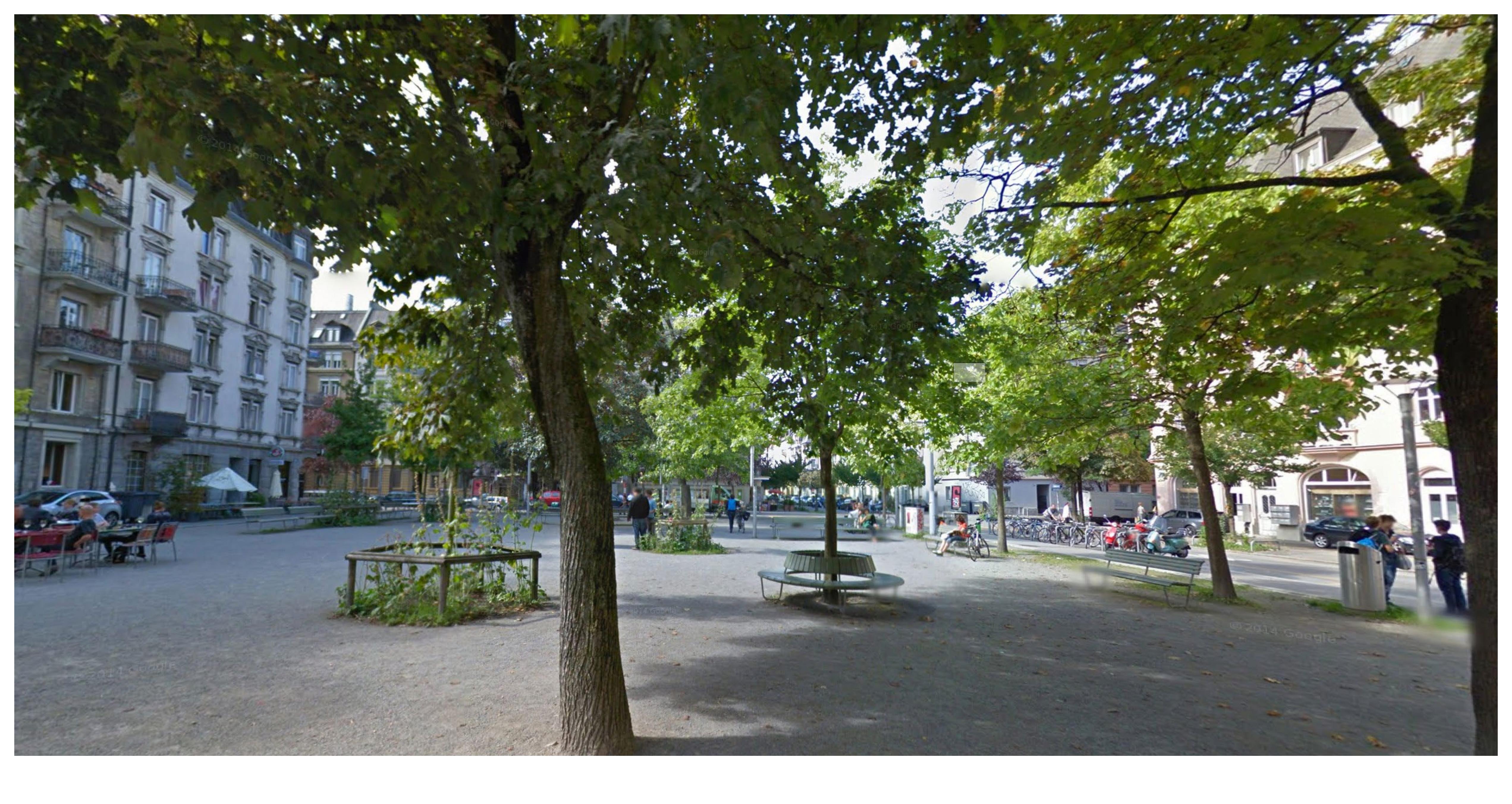


## Method

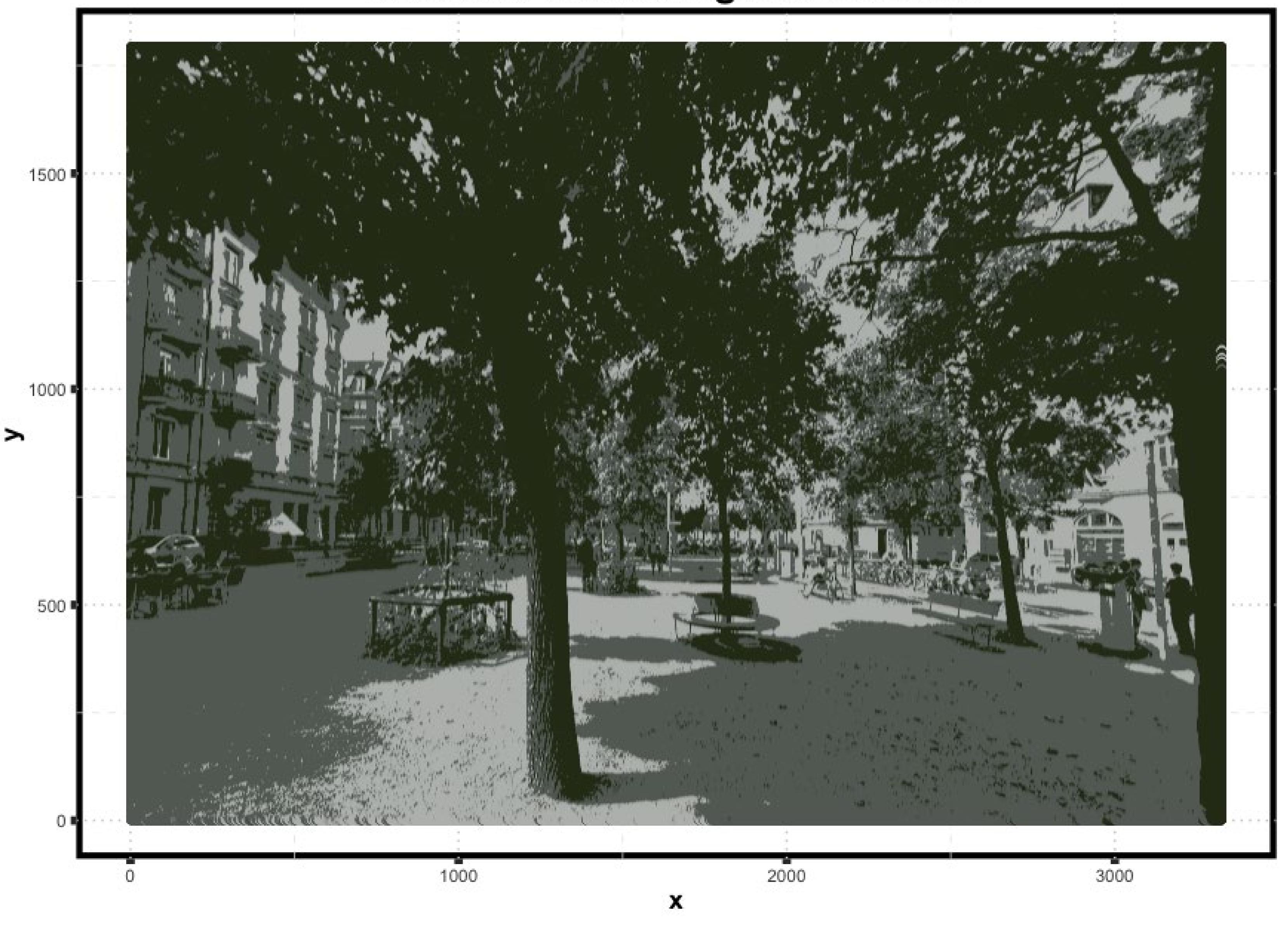




Make a Photosphere of the Place



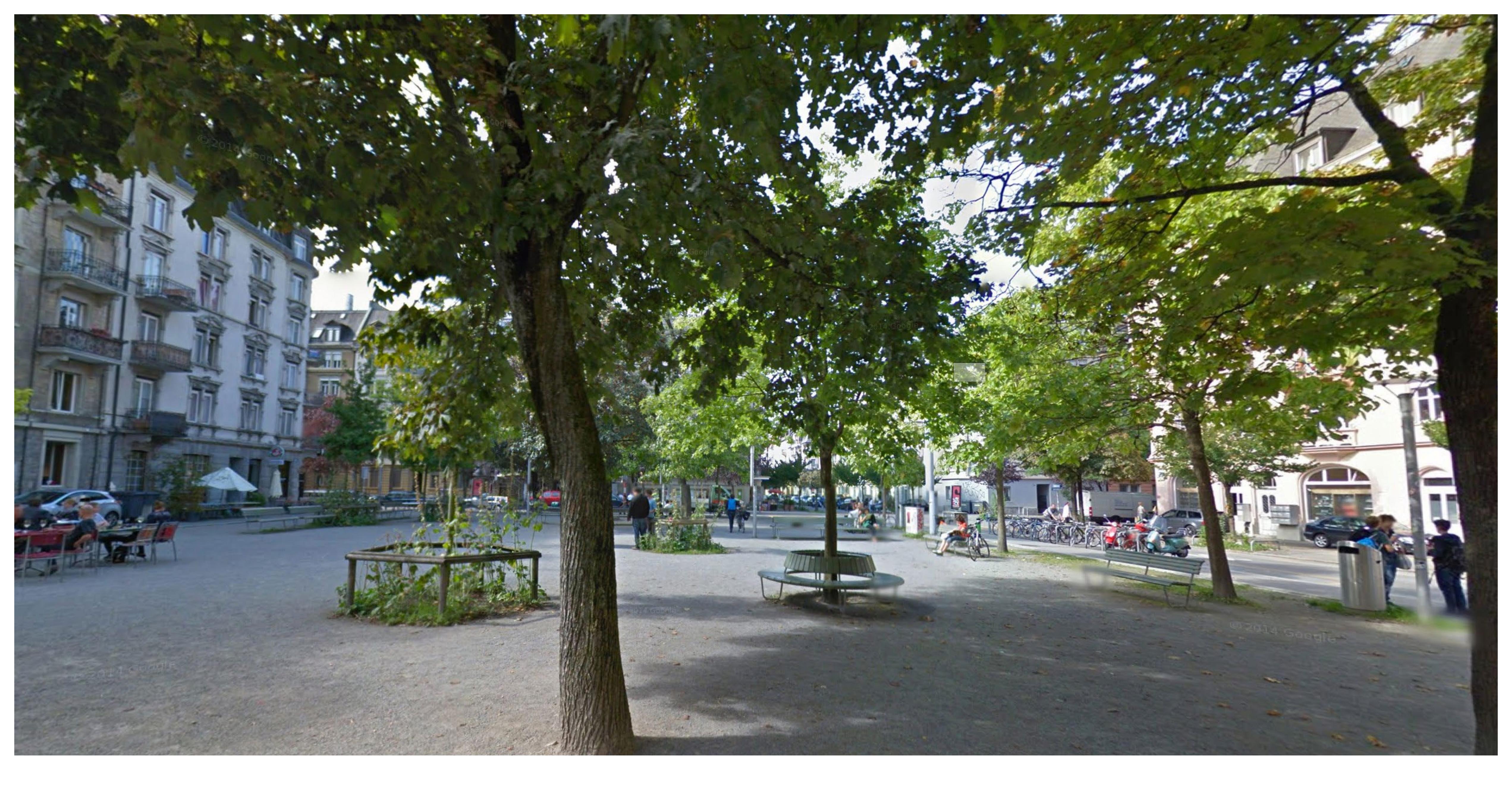
## k-Means Clustering of 3 Colours



Run a k-means algorithm in R. Then wait. Then get a coffee. (Caluclation time aprox. 30 Minutes)

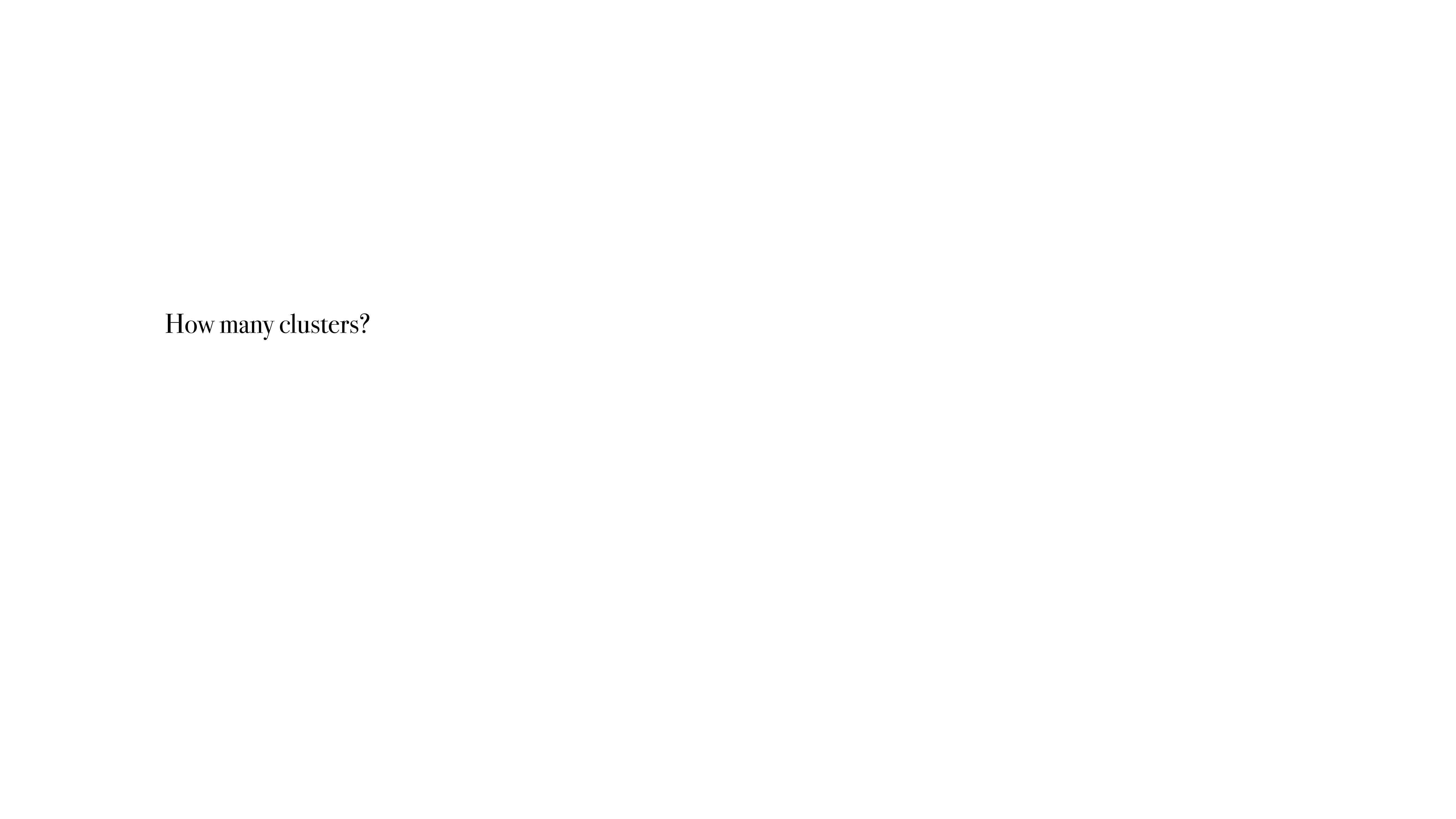


Or just save it in Photoshop as a color reduced PNG-File. Sorry, no coffee here. (Caluclation time aprox. 2 Seconds)

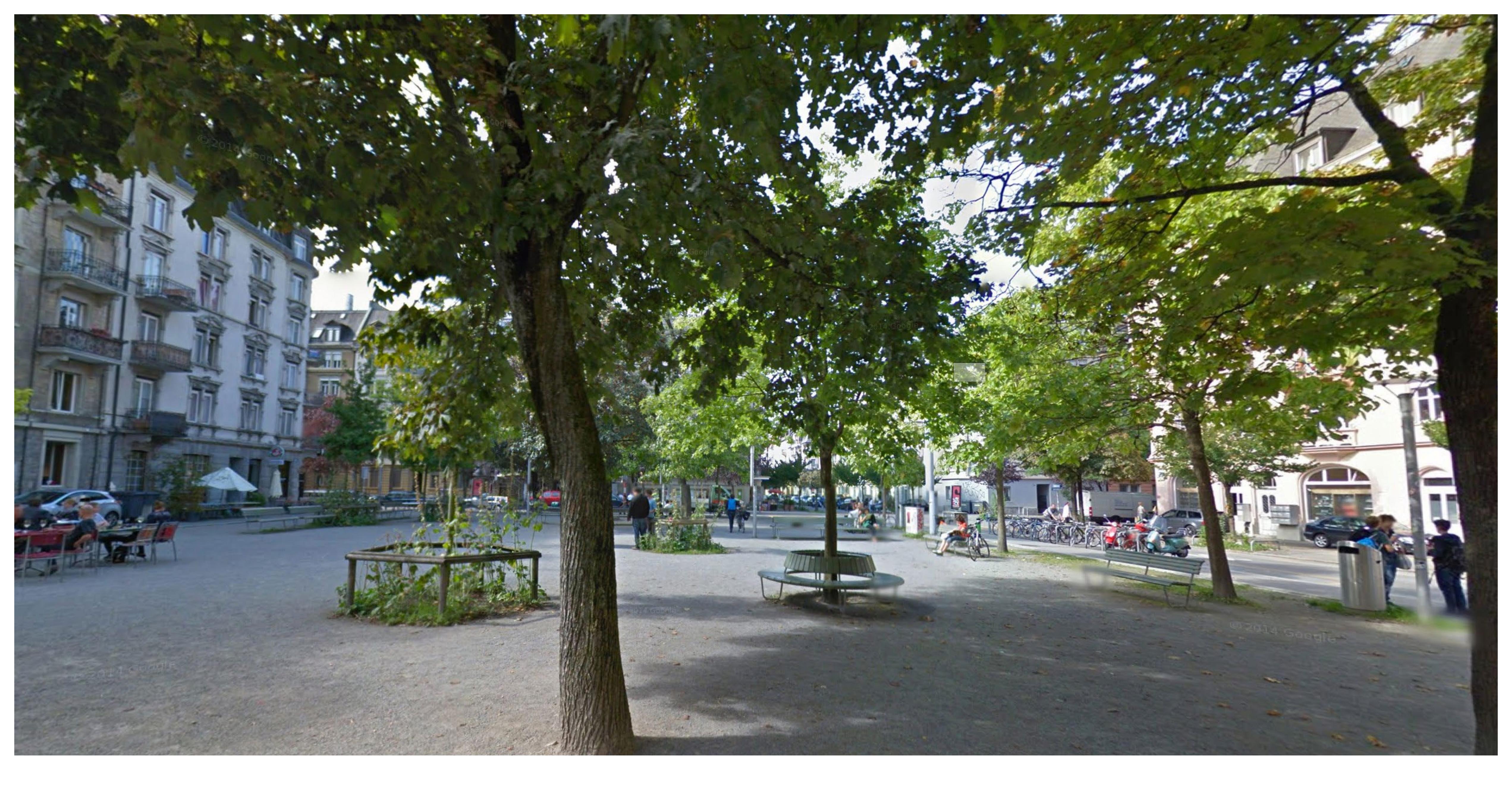




Again, just the three most dominant colors.







Original, all colors are there.



12 most dominant colors.

```
sketch_160425b | Processing 3.0.2
 00
   sketch_160425b
   import generativedesign.*;
   import processing.pdf.*;
   import java.util.Calendar;
   boolean savePDF = false;
43 PImage img;
   color[] colors;
   String sortMode = null;
   void setup(){
   //size(3814, 1907); // ESUM Foto-Spheres
    //size(1860, 1046); //16:9 30 Colors
   size(3326, 1794); // WHAT EVER
   //MacBook Pro Retina 15" 2880x1800
   //size(2880, 1800); // Retina
    //size(1920, 1080); // FlyLo
   //size(1108, 598); // half
   //size(3072, 1963); //Turner
   colorMode(HSB, 10, 10, 10, 360);
    noStroke();
    noCursor();
    img = loadImage("Test-Image_12c.png");
    //img = loadImage("j-m-w-turner-ulysses.jpg");
   void draw(){
   if (savePDF) {
      beginRecord(PDF, timestamp()+".pdf");
      colorMode(HSB, 10, 10, 10, 10);
     noStroke();
   int tileCount = width / max(mouseX, 1);
   float rectSize = width / float(tileCount);
   // get colors from image
   int i = 0;
   colors = new color[tileCount*tileCount];
    for (int gridY=0; gridY<tileCount; gridY++) {</pre>
      for (int gridX=0; gridX<tileCount; gridX++) {</pre>
       int px = (int) (gridX * rectSize);
       int py = (int) (gridY * rectSize);
       colors[i] = img.get(px, py);
       i++;
    if (sortMode != null) colors = GenerativeDesign.sortColors(this, colors, sortMode);
   // draw grid
    for (int gridY=0; gridY<tileCount; gridY++) {</pre>
      for (int gridX=0; gridX<tileCount; gridX++) {</pre>
       fill(colors[i]);
       rect(gridX*rectSize, gridY*rectSize, rectSize, rectSize);
    if (savePDF) {
      savePDF = false;
      endRecord();
```

Pixelsorting algorithm in Processing. Sorting by Hue-Value.



12 most dominant colors sorted by Hue.



Without clustering. All colors are present. Sorted by Hue.



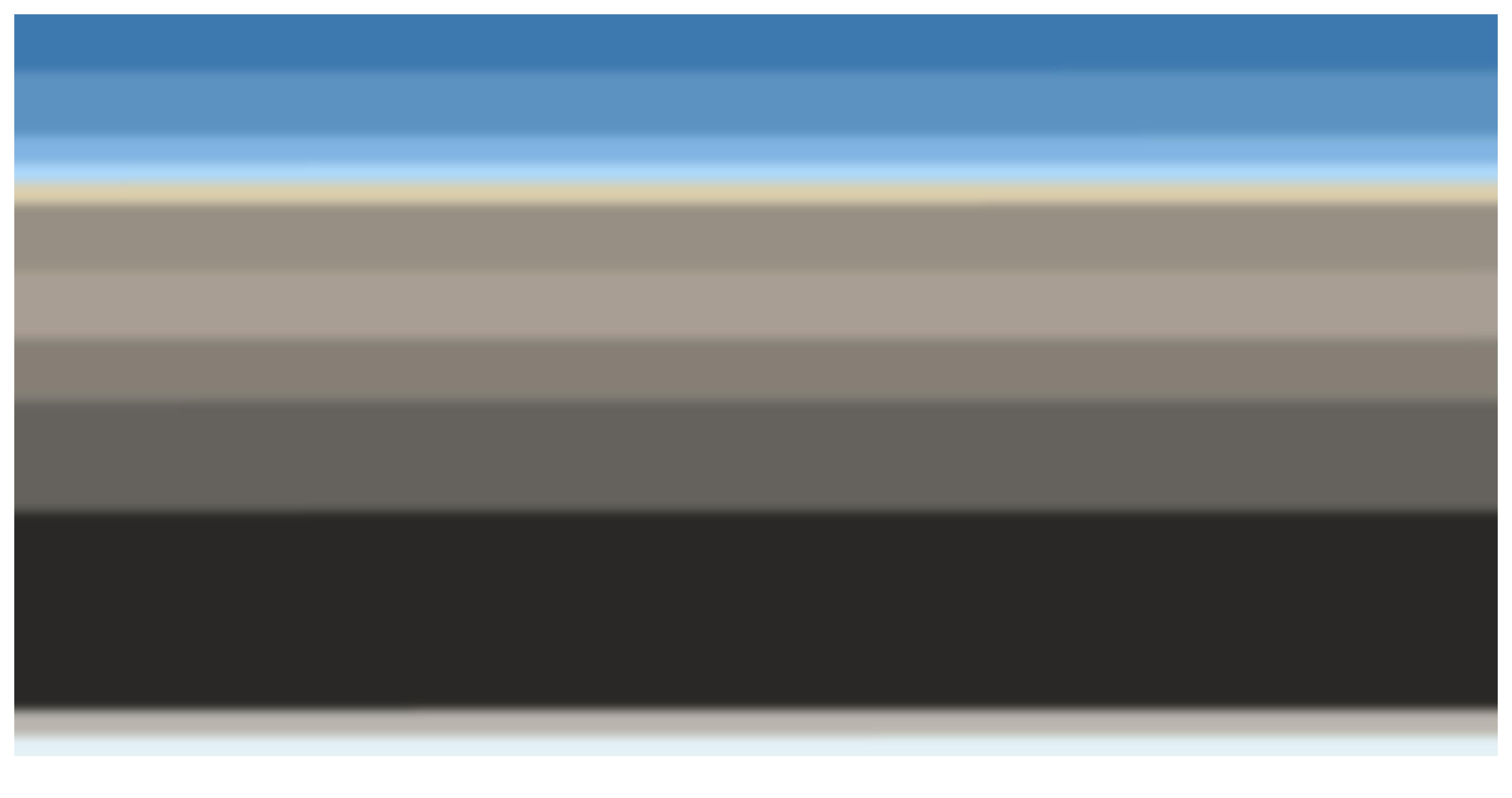
12 most dominant Colors sorted by Hue with Gaussian-Blur of 12 Pixels.

## Let's have a walk...

in a colorful way



Position 1 - 12 Colors



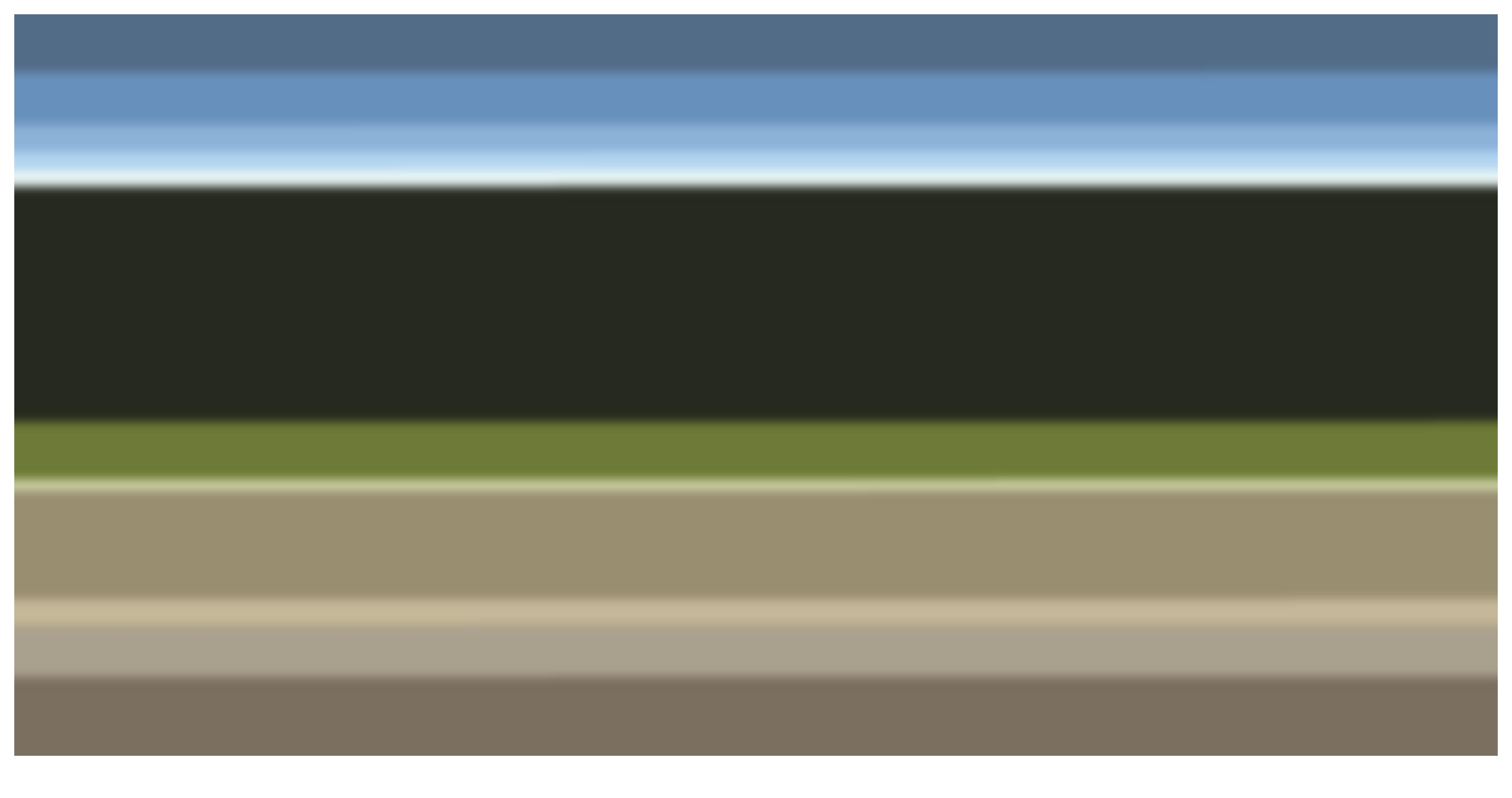
Position 2 - 12 Colors



Position 3 - 12 Colors



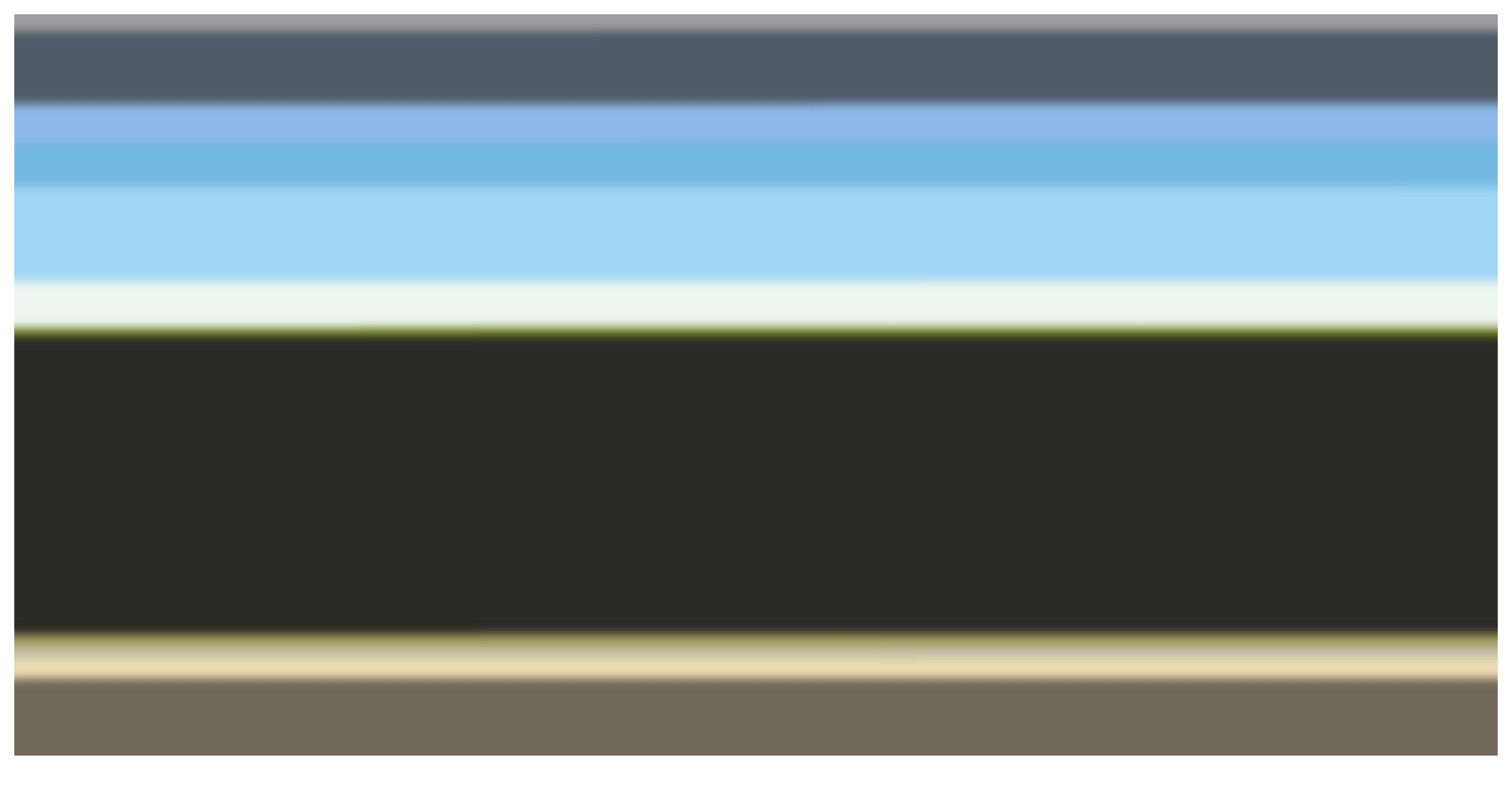
Position 4 - 12 Colors



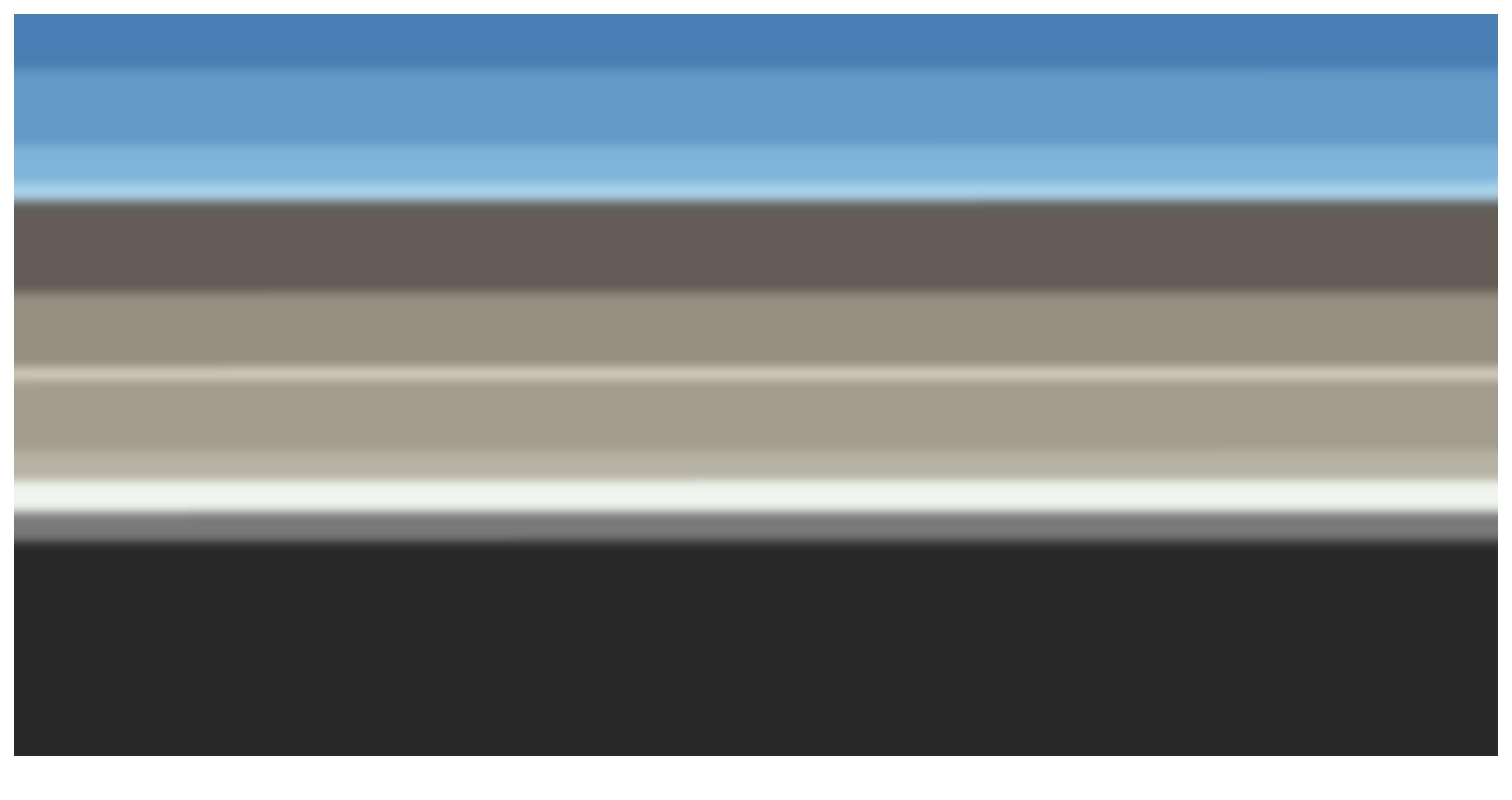
Position 5 - 12 Colors



Position 6 - 12 Colors



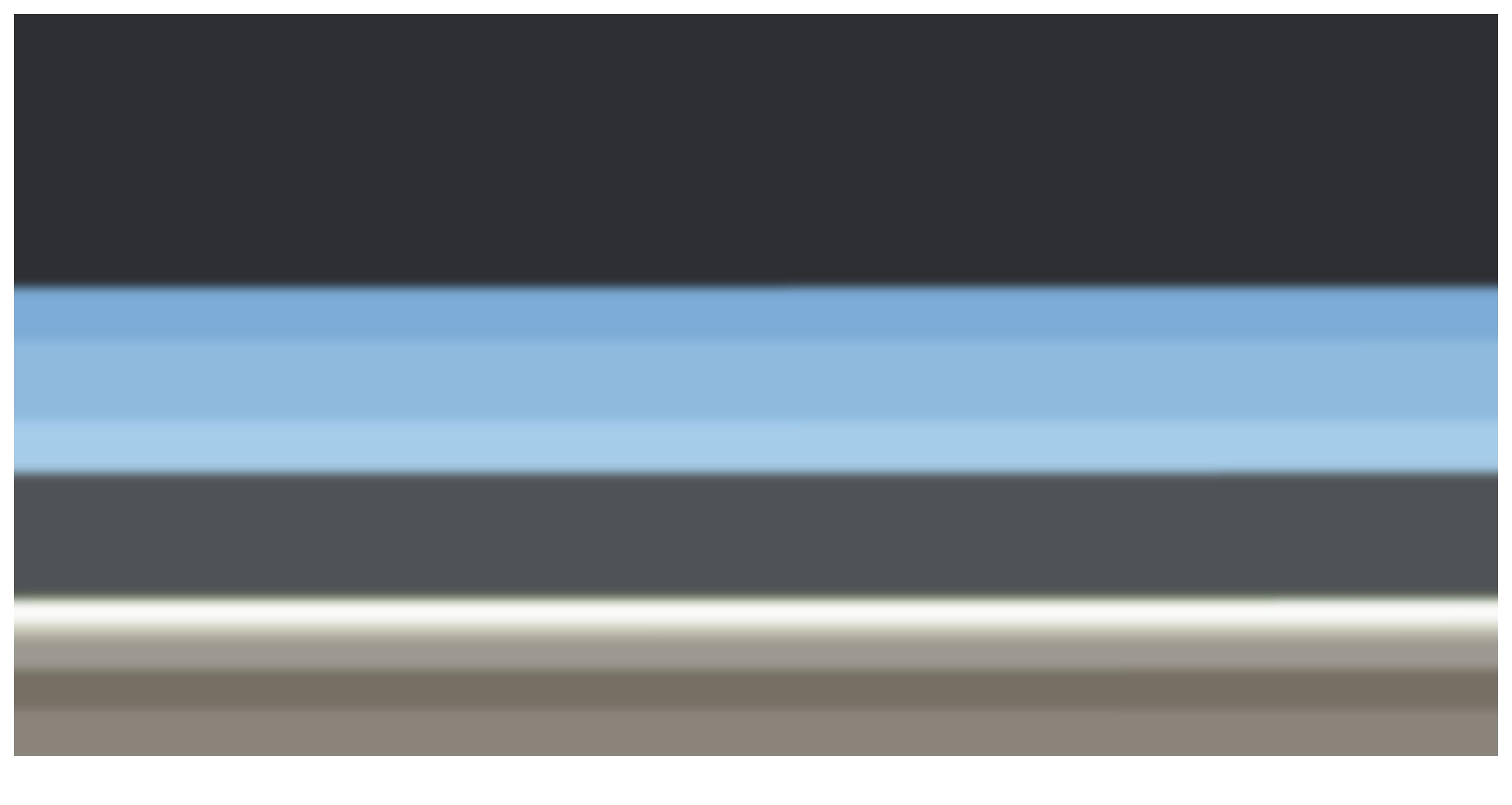
Position 7 - 12 Colors



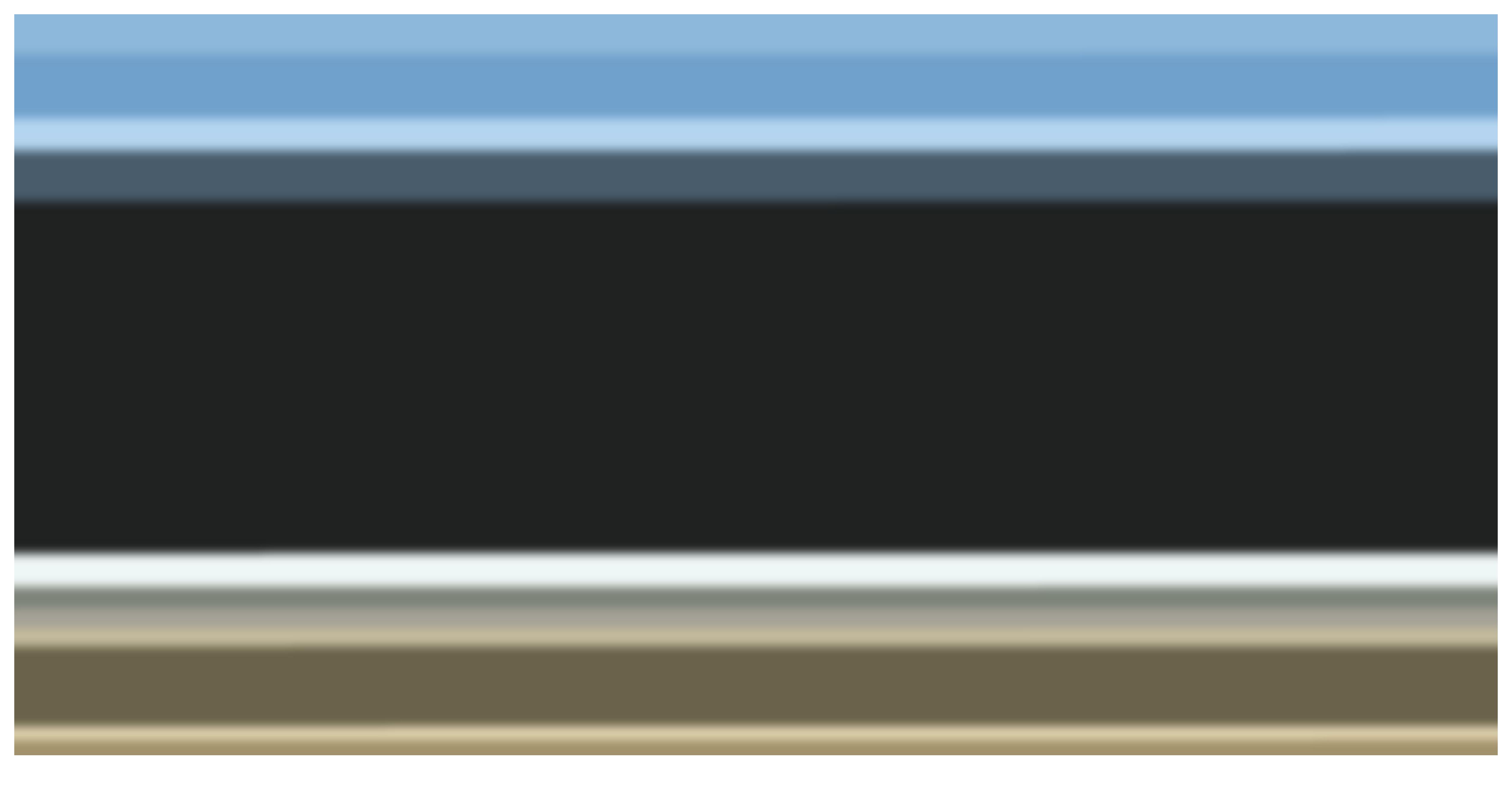
Position 8 - 12 Colors



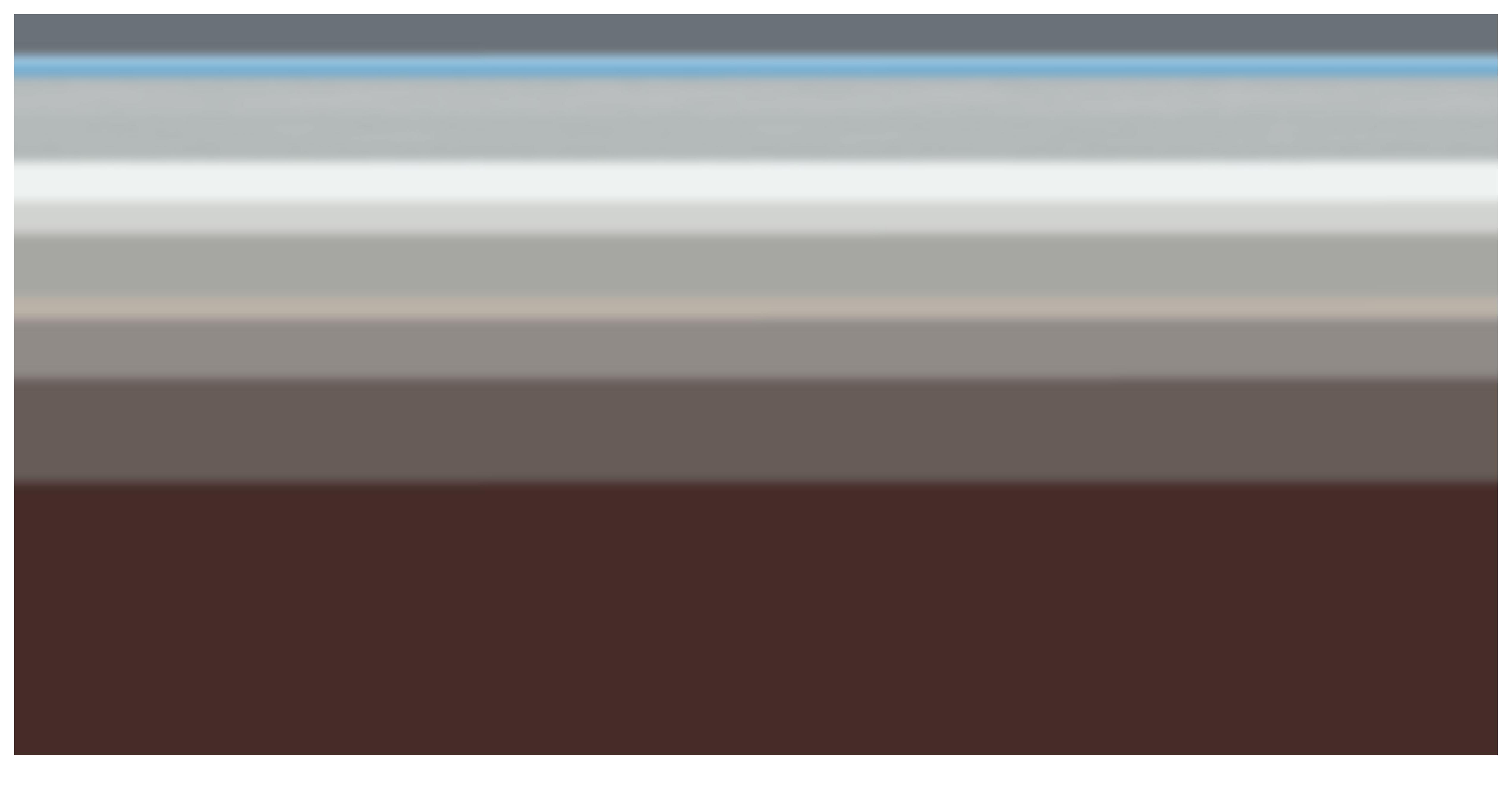
Position 9 - 12 Colors



Position 10 - 12 Colors



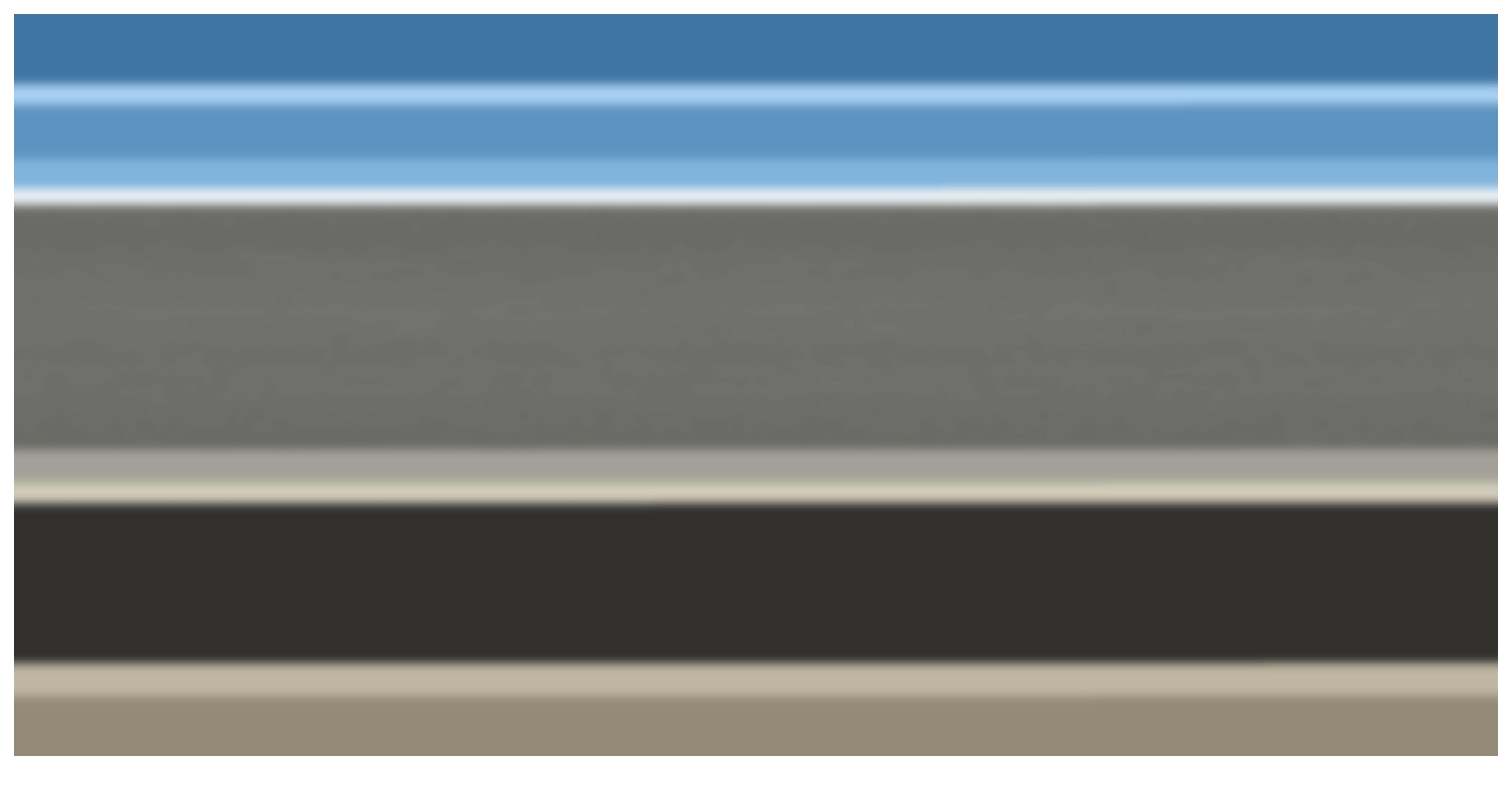
Position 11 - 12 Colors



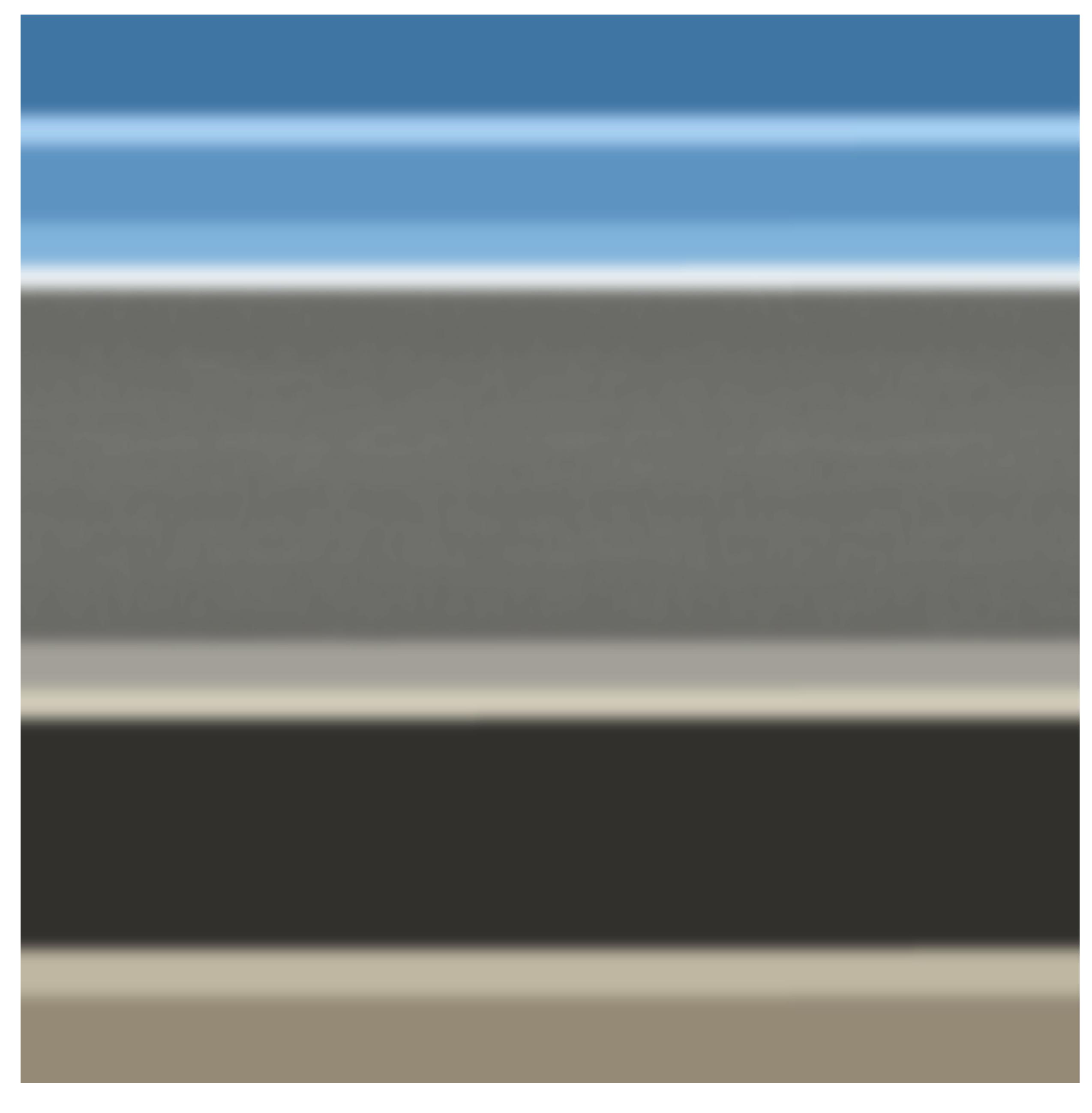
Position 12 - 12 Colors - Red Sunblind





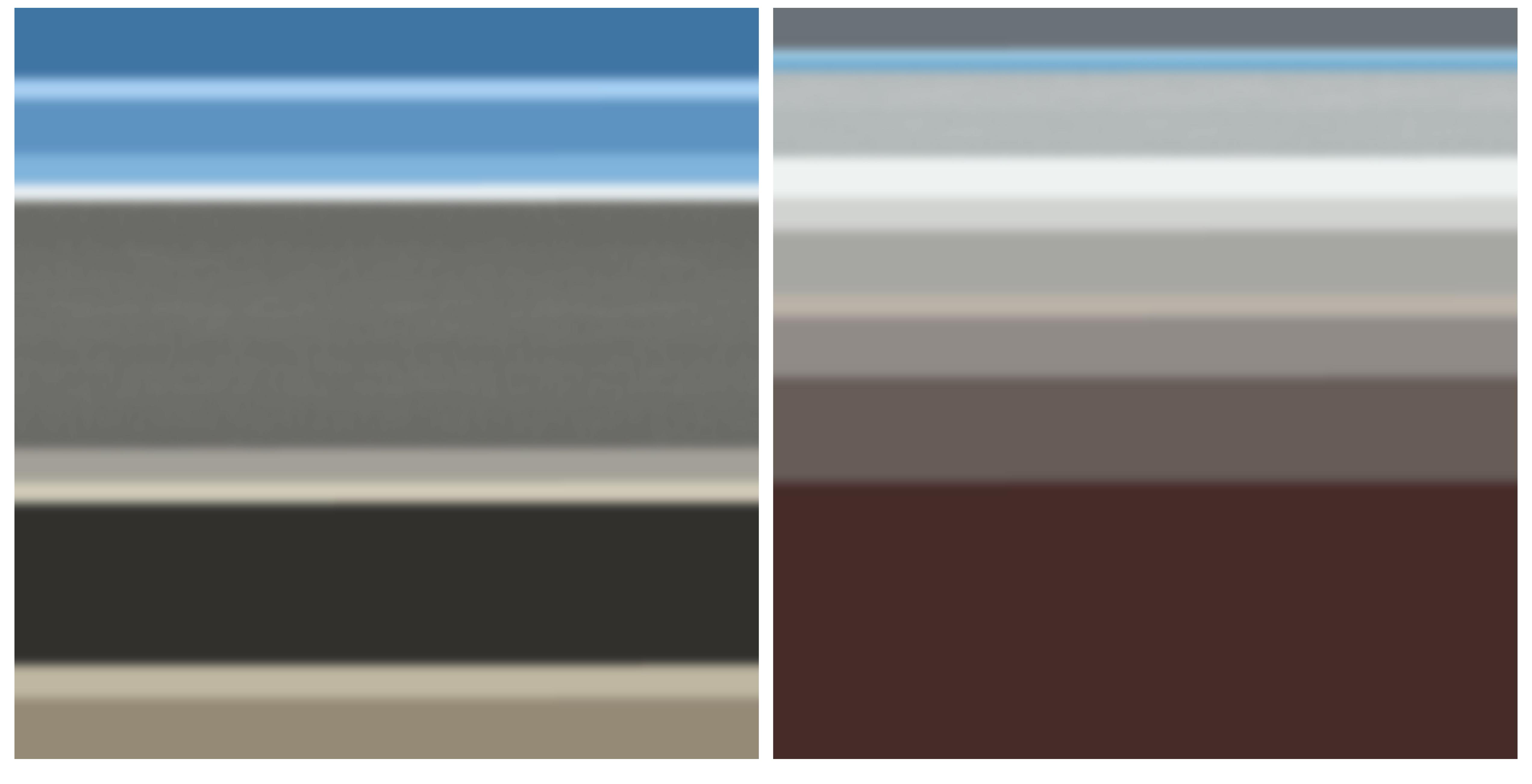


Position 12 - 12 Colors - Sky





12 Colors
Position 12, Sky



Pos 12, Sky

Pos 12, Red Sunblind



Position 13 - 12 Colors



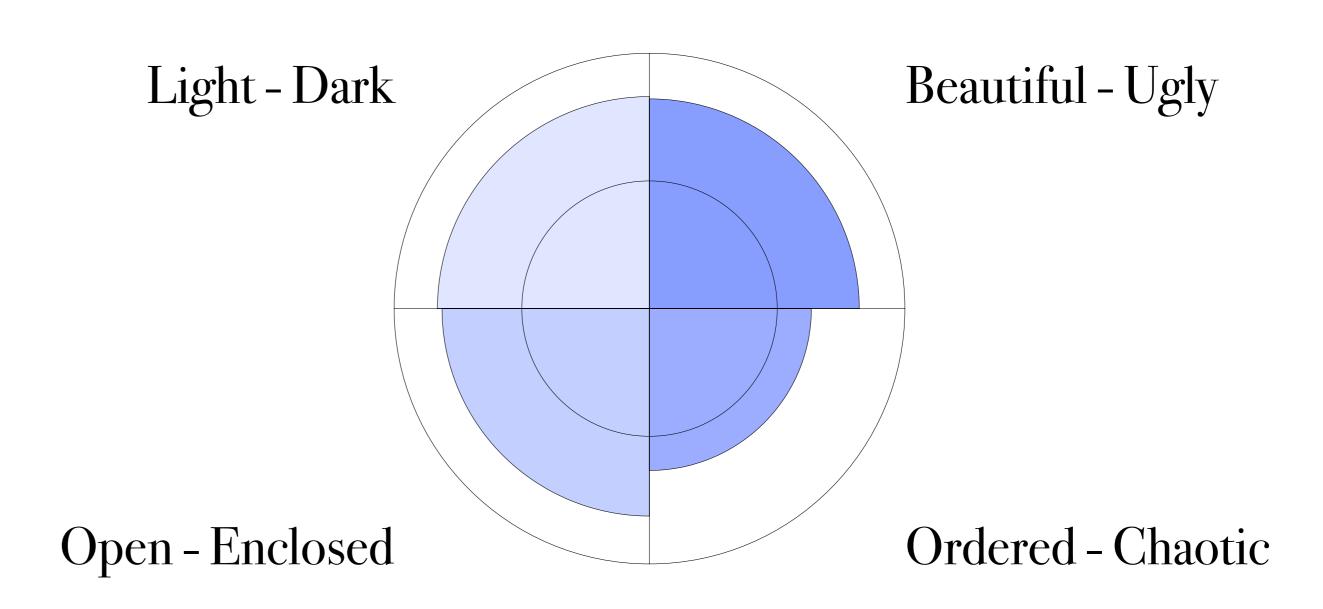
Position 14 - 12 Colors

## Outstanding Positions

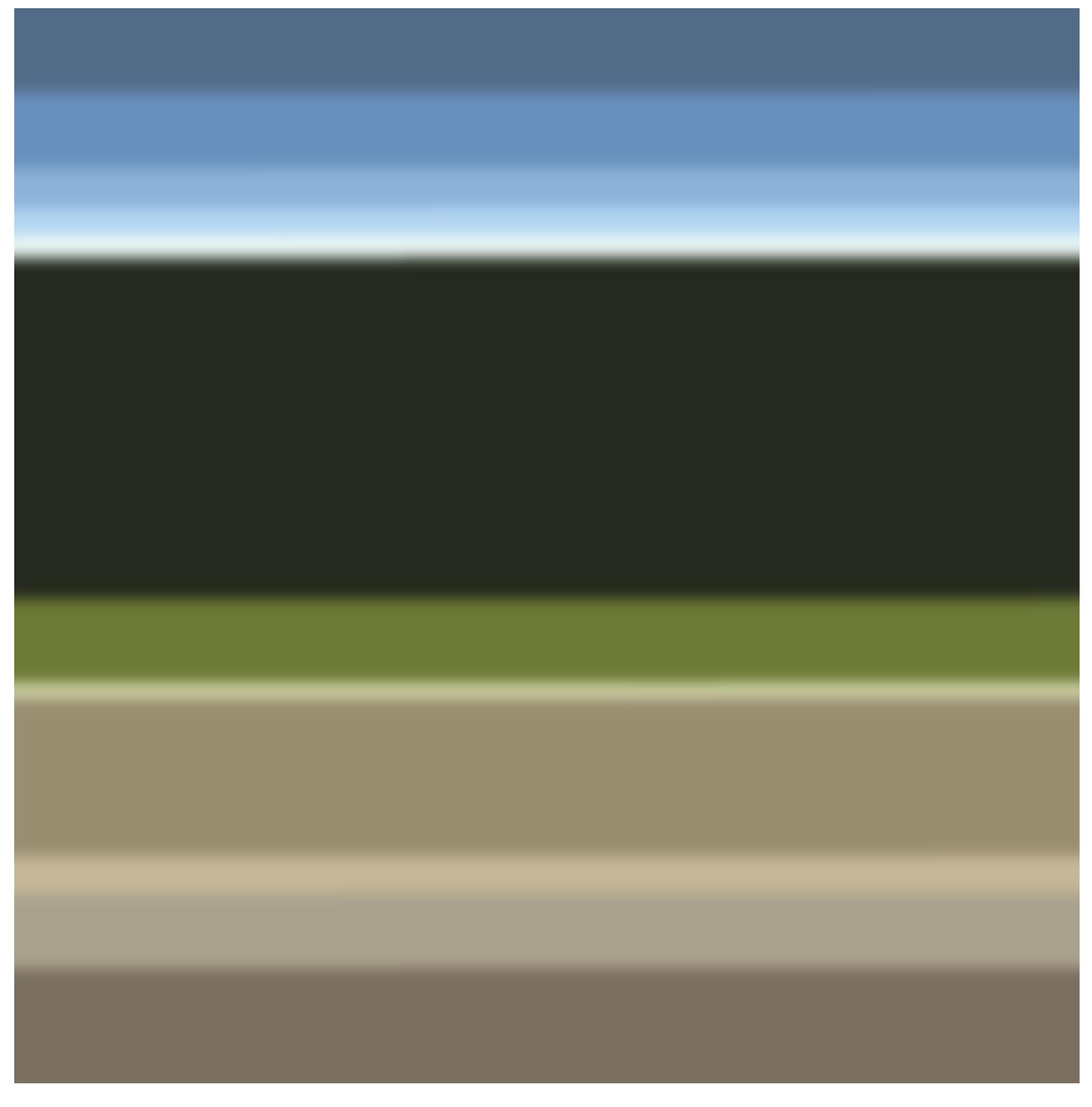
according to the survey



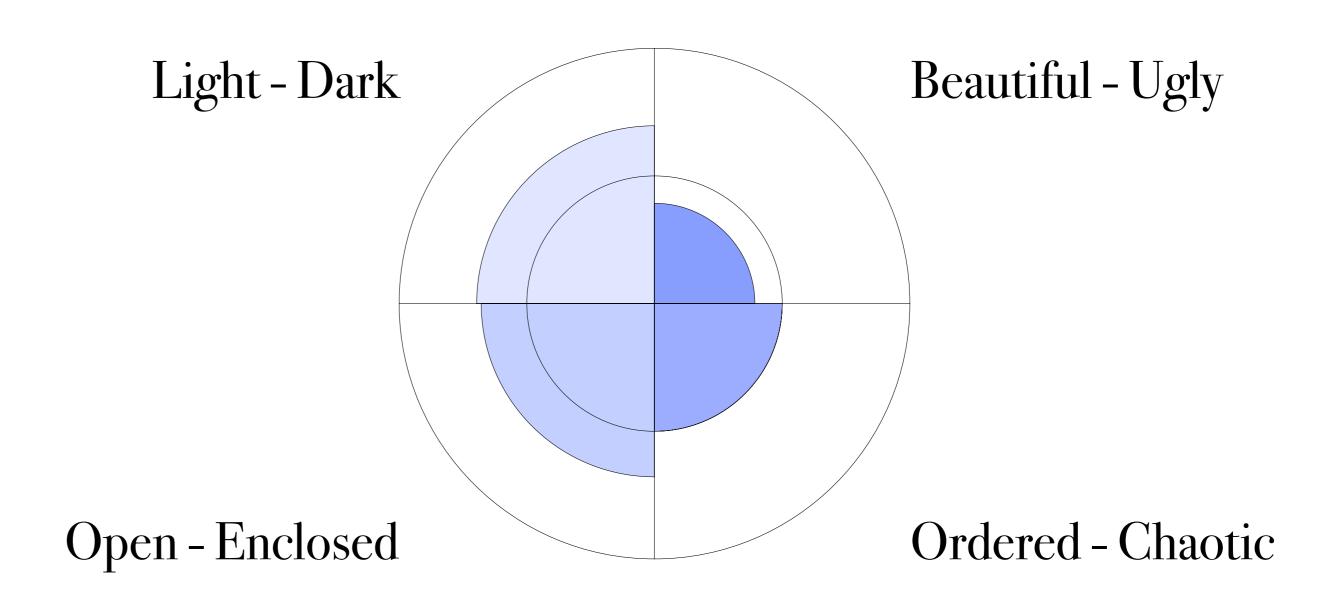




Position 3, rated as most beautiful





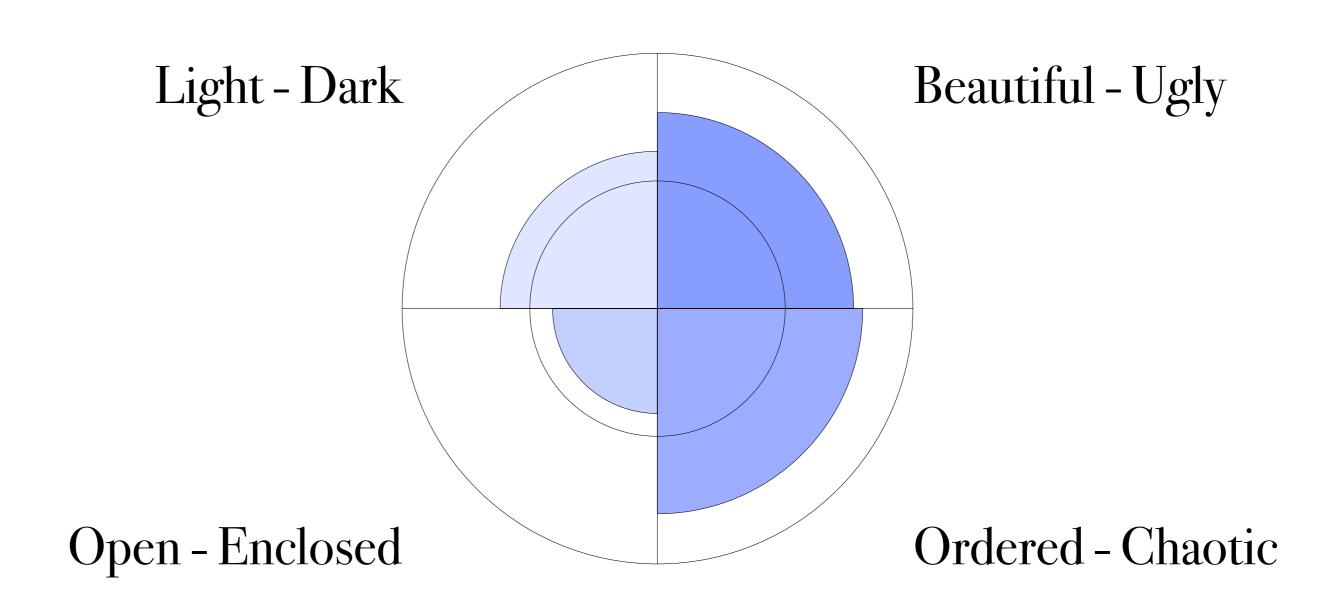


Position 5, rated as ugliest



12 Colors

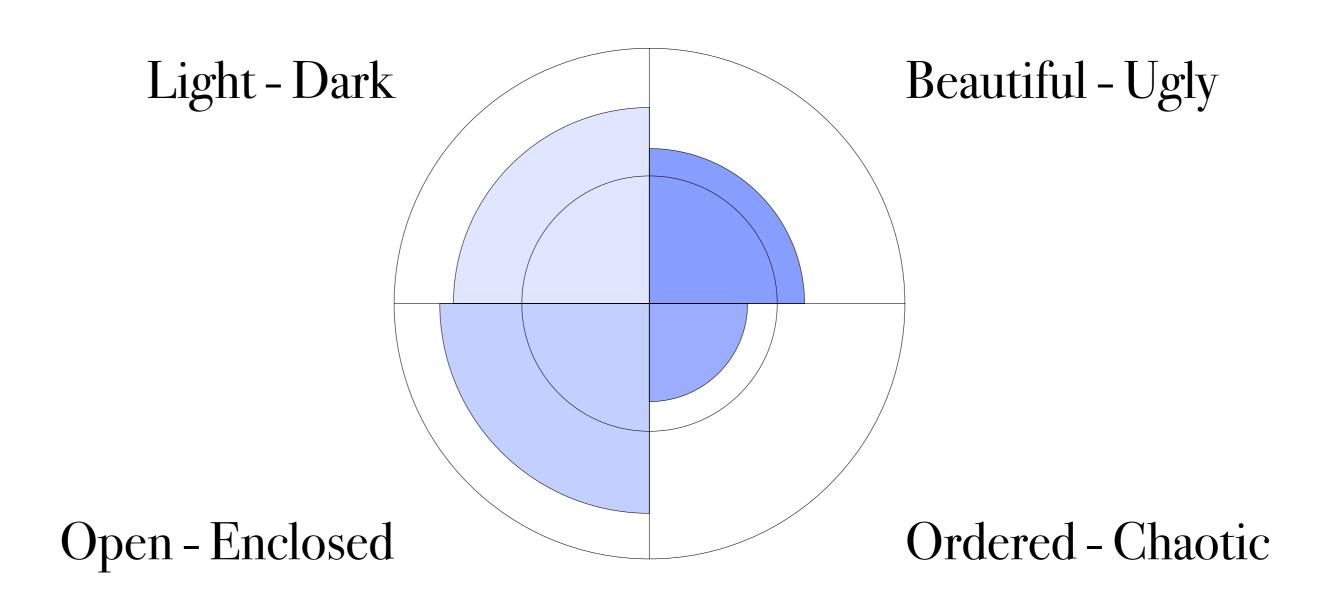




Position 7, rated as enclosest





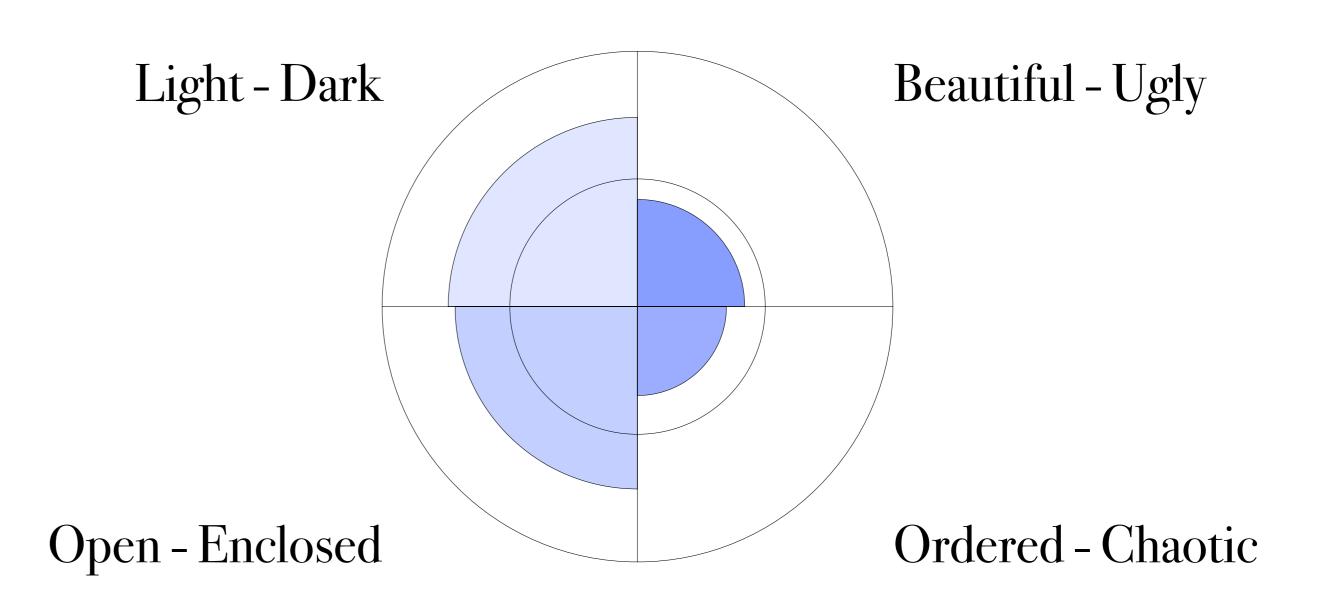


Position 11, rated as most open



12 Colors

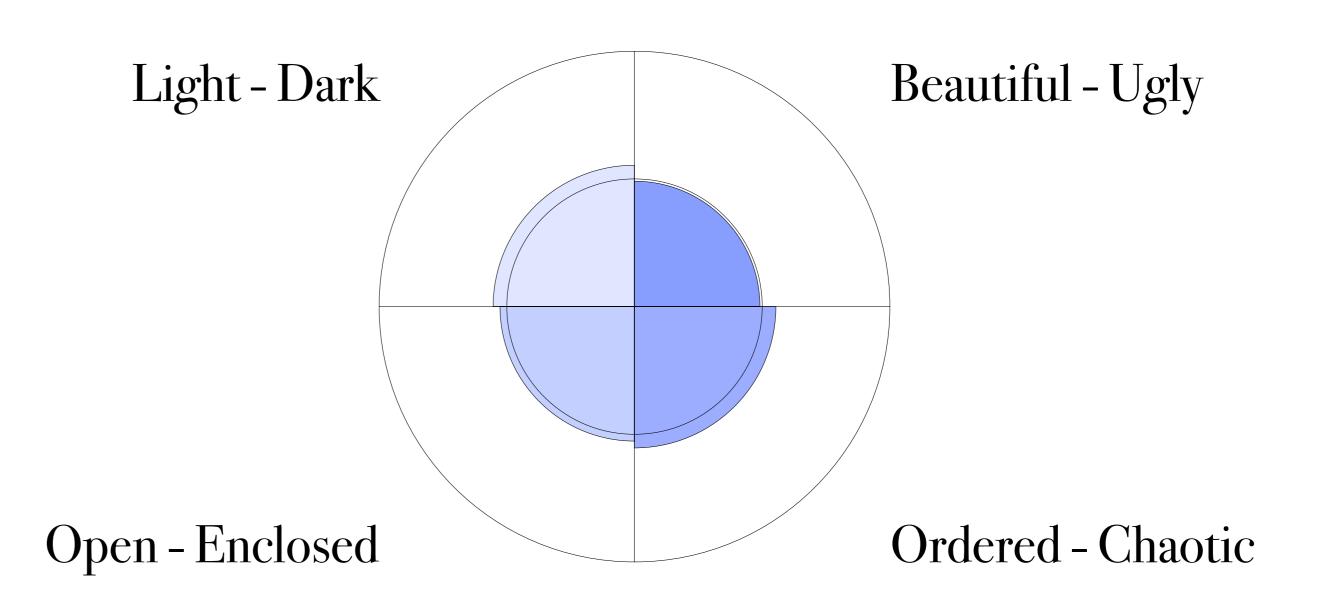




Position 13, rated as most chaotic



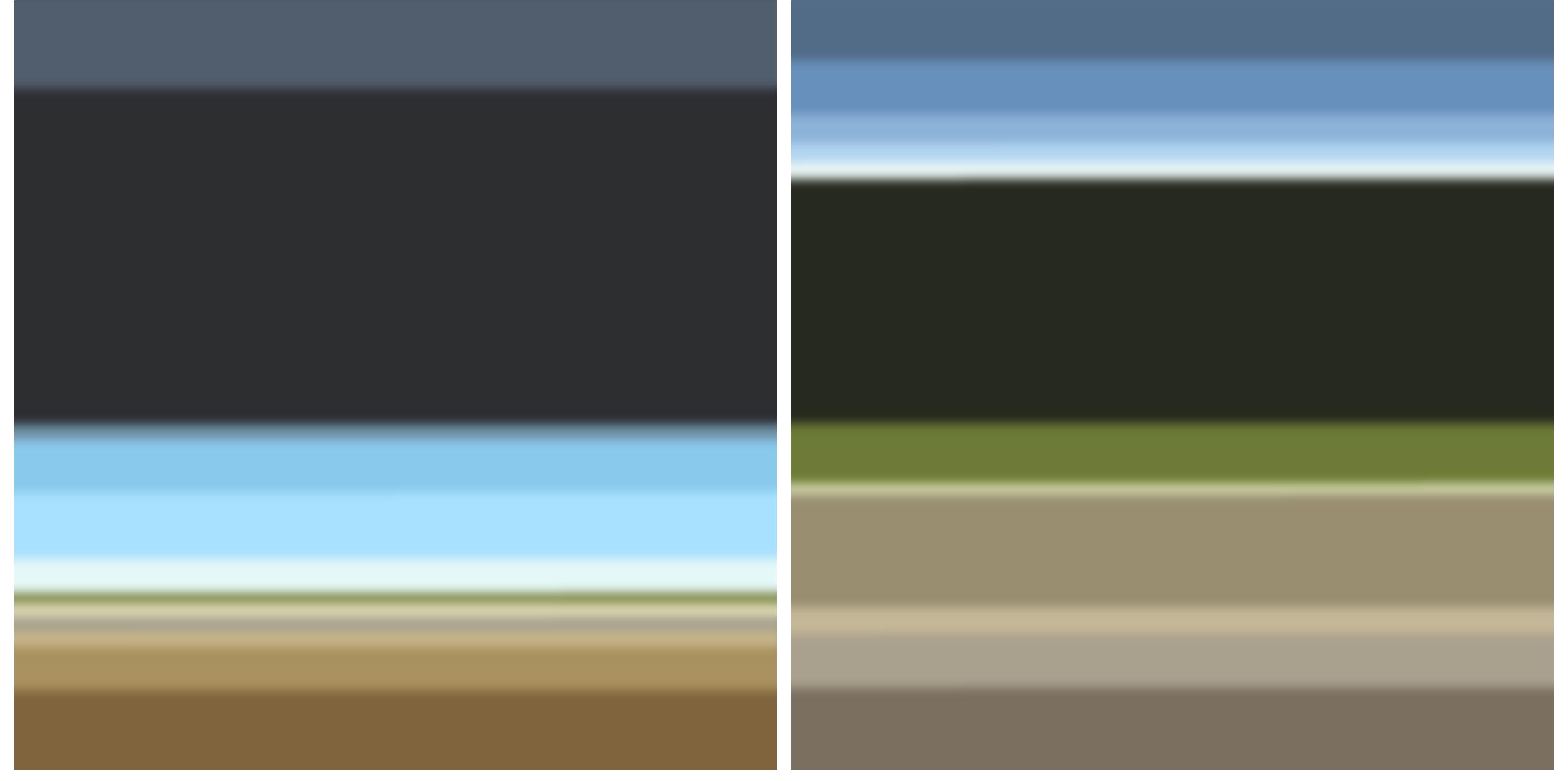




12 Colors Position 14, rated as darkest

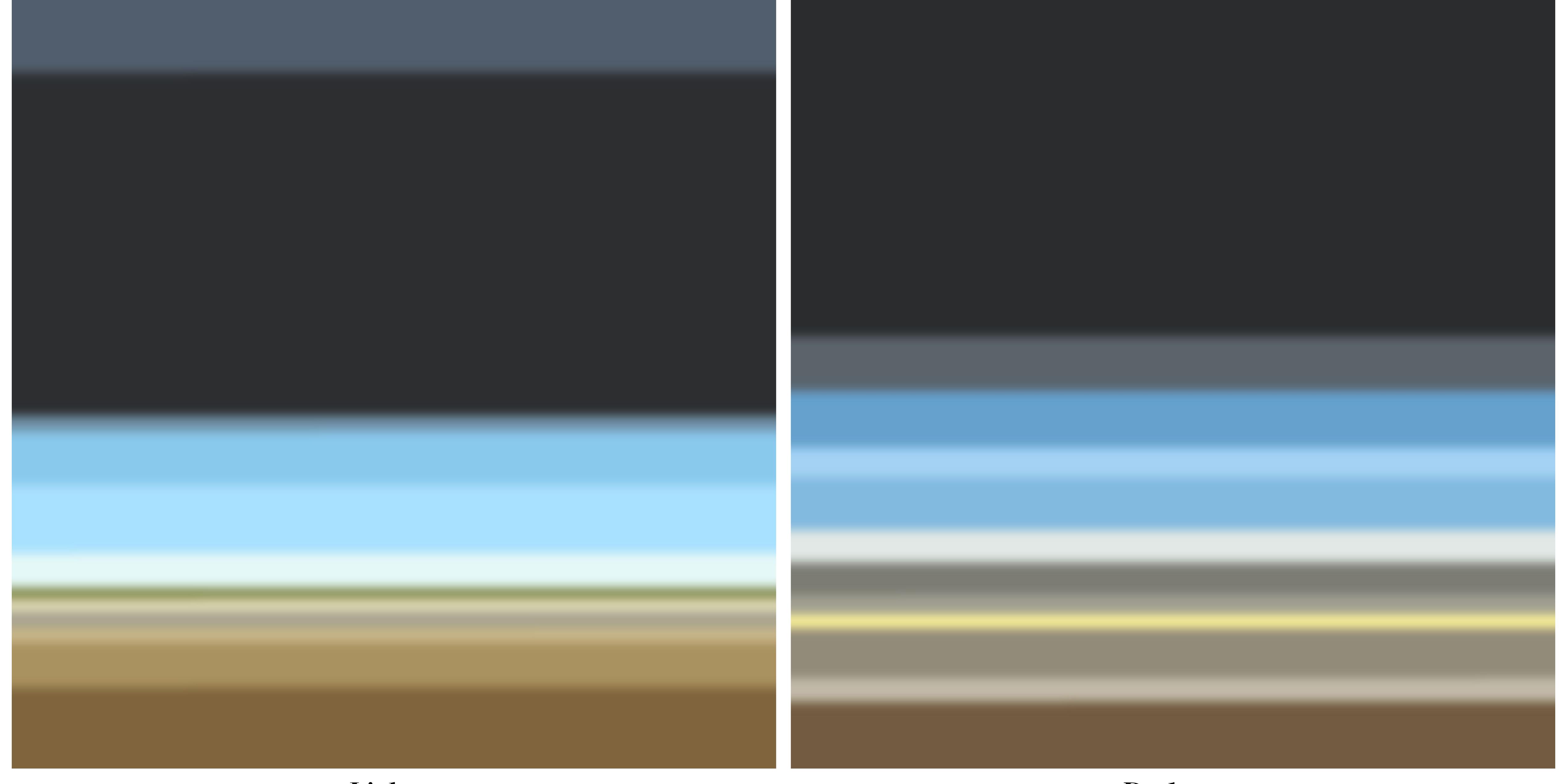
## Best & Worst

according to the survey



Most Beautiful
Position 3 + 64%

Ugliest
Position 5 - 22%



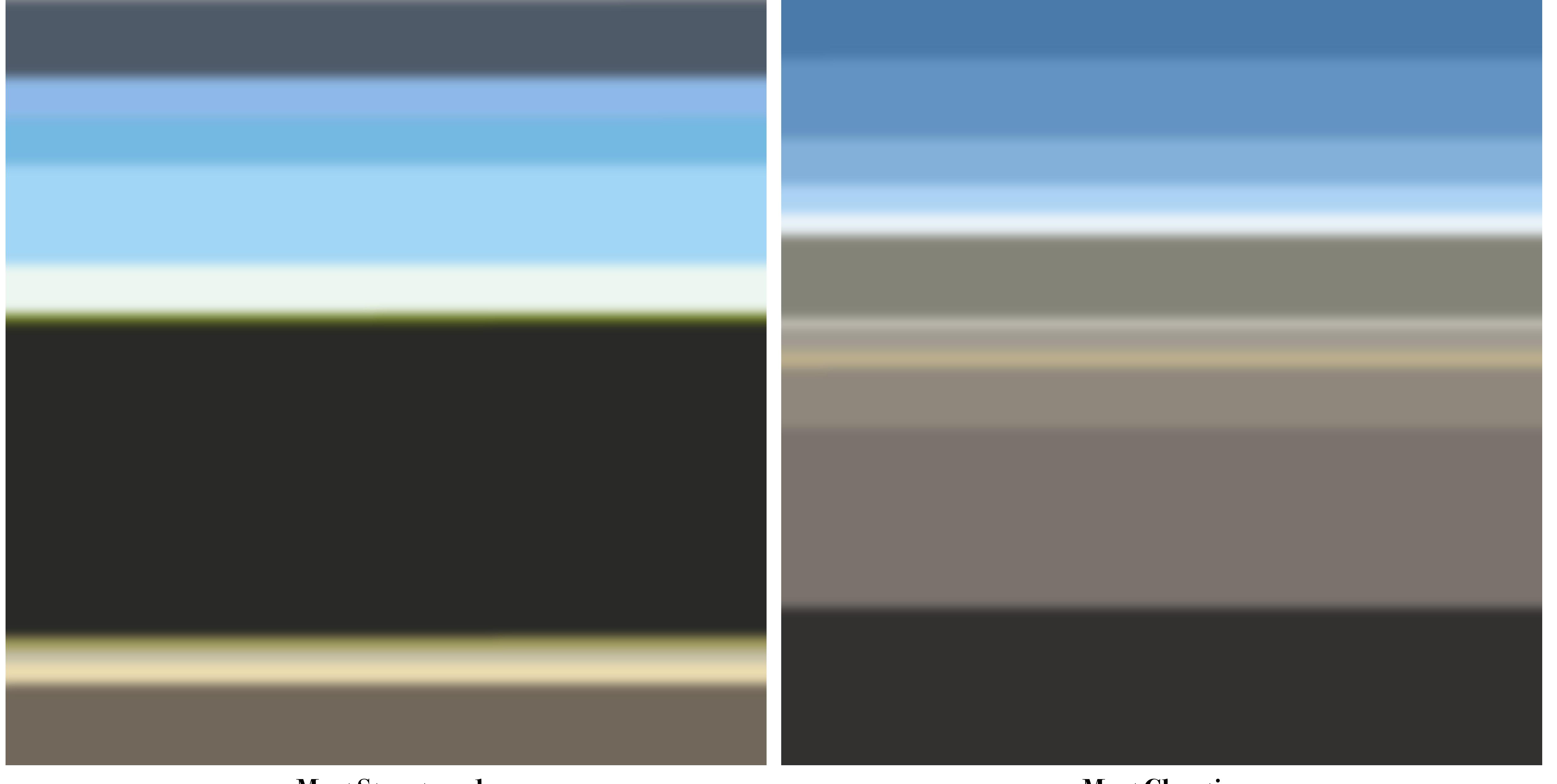
Lightest
Position 3 + 66%

Darkest
Position 14 + 11%



Most Open
Position 11 + 65%

Most Enclosed
Position 7 - 18%

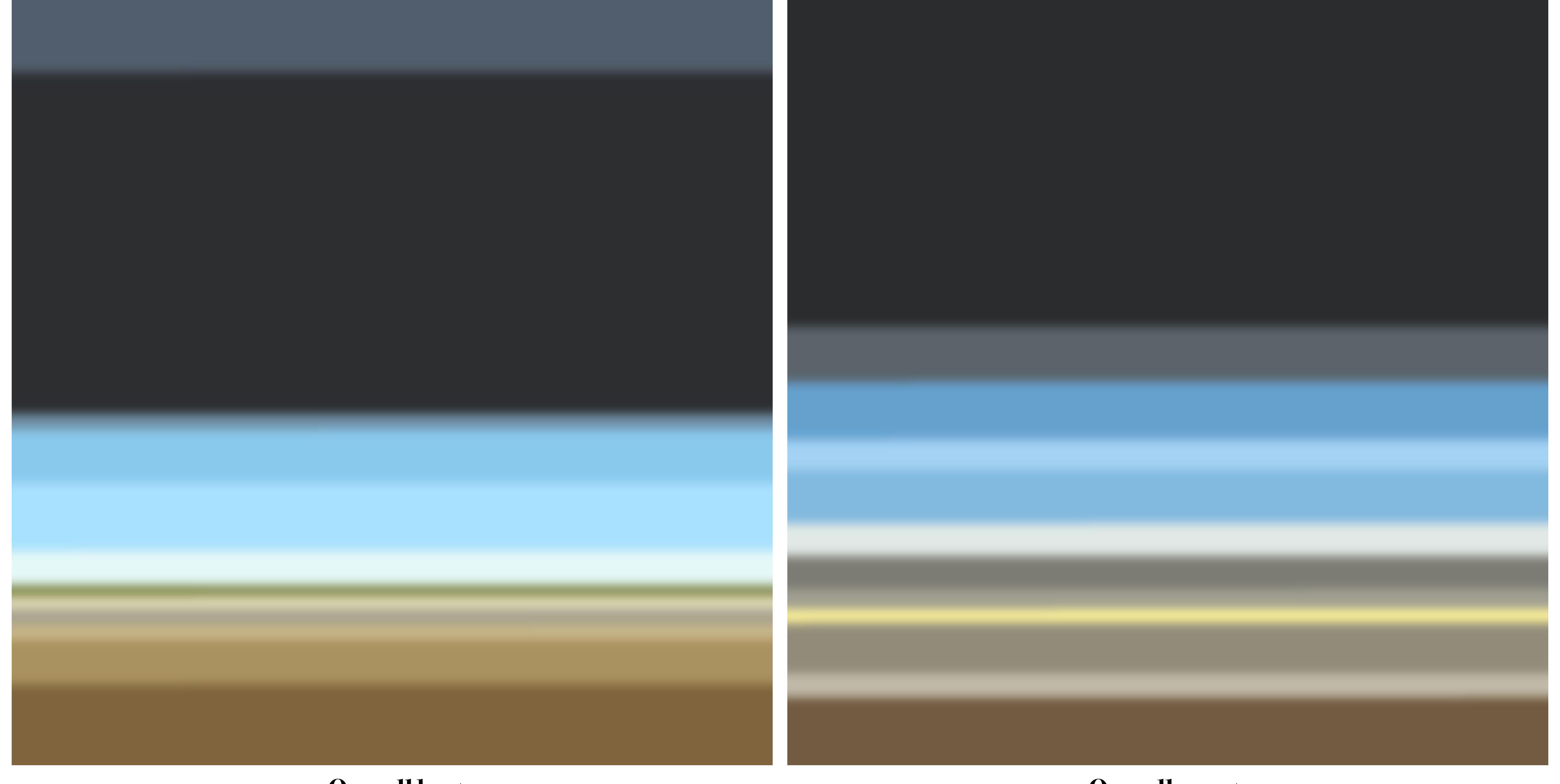


**Most Structured** 

Position 7 + 61%

Most Chaotic

Position 13 - 31%

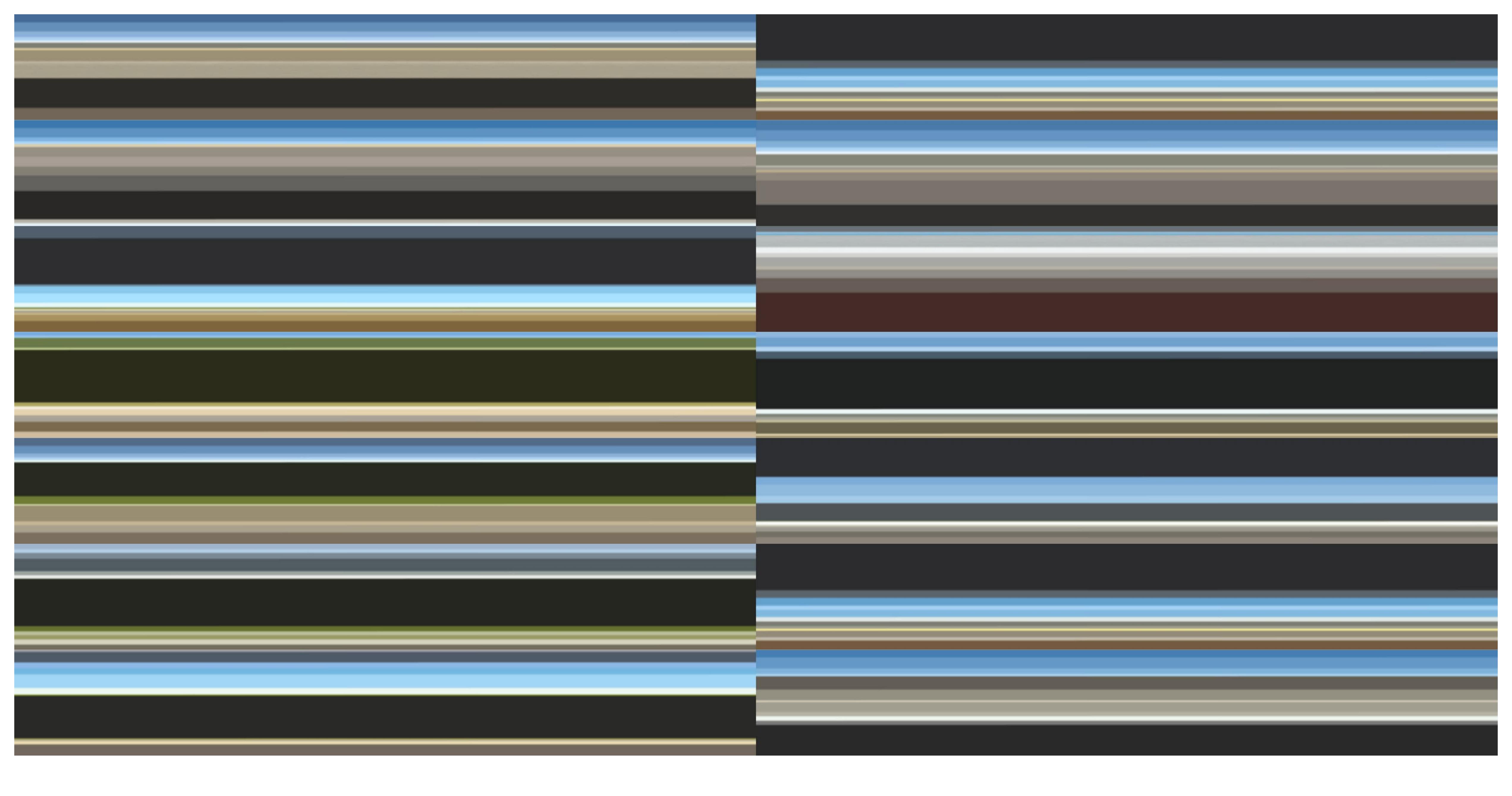


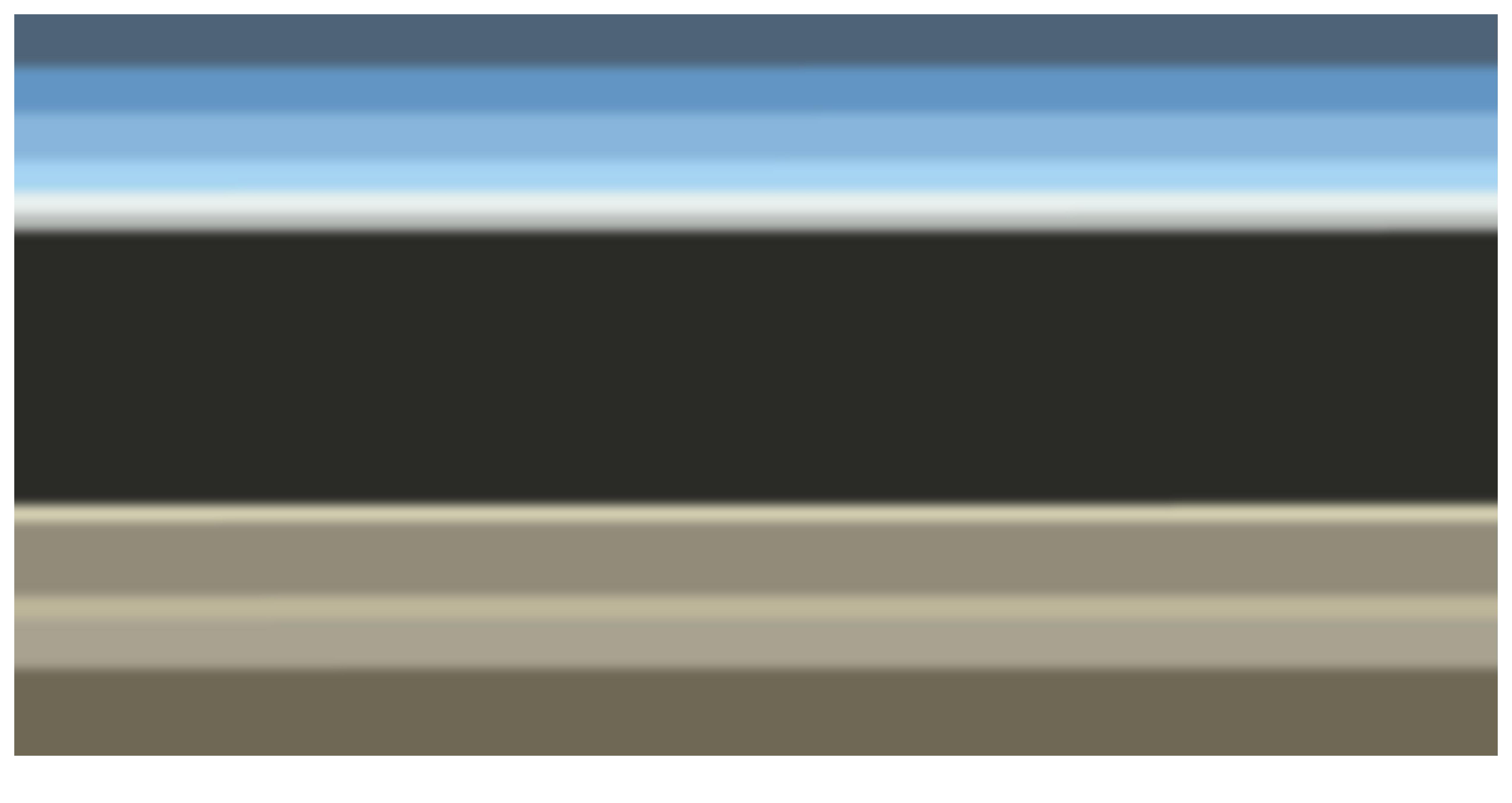
**Overall best** 

Position 3 + 55%

Overall worst

Position 9 + 4%





All Positions - 12 Most dominant Colors

Is there a link of the pure color schemes of a place, to it's perception of...

... beauty / ugliness
... lightness / darkness
... openness / enclosedness
... order / chaos



## Hard to say. Because...

... big difference in daytime, weather, etc.

...spherical distortion.

... no color metering, white and color balance, etc.

... camera is specific to one point.

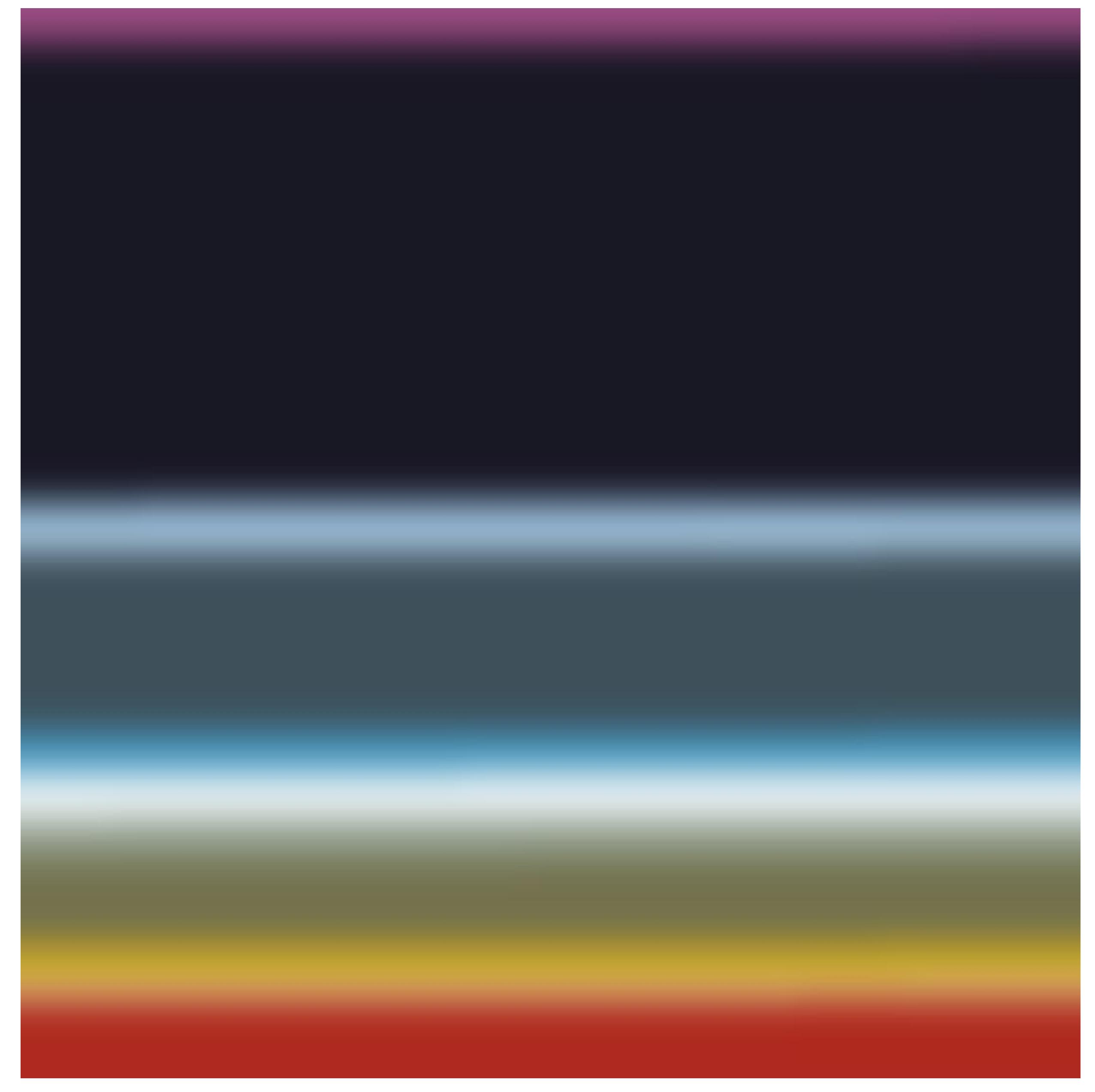
...people judge not by point but by place.

... neather green (plants) or blue (clear sky) influence.

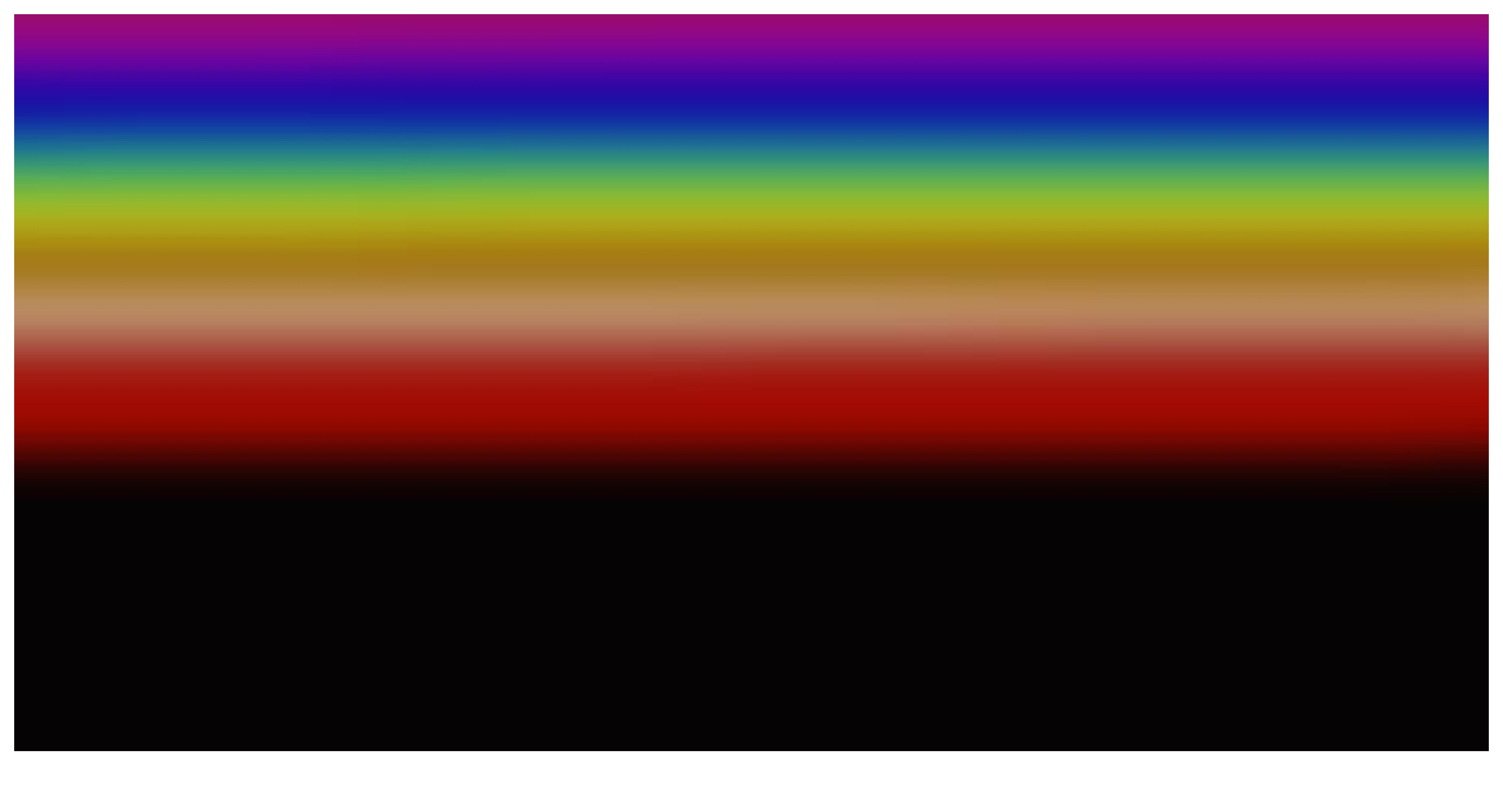
## Other Stuff

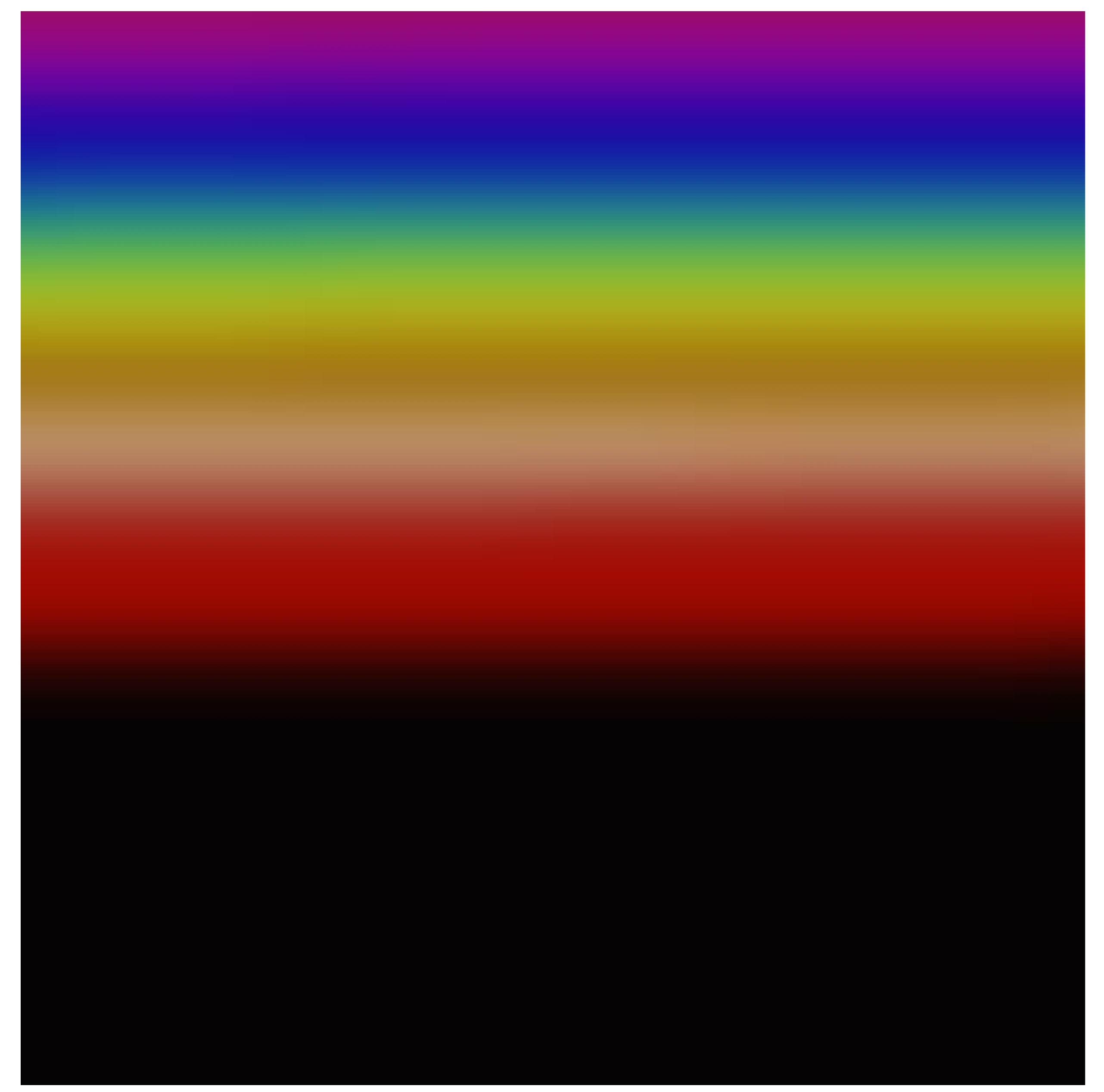
just for fun...







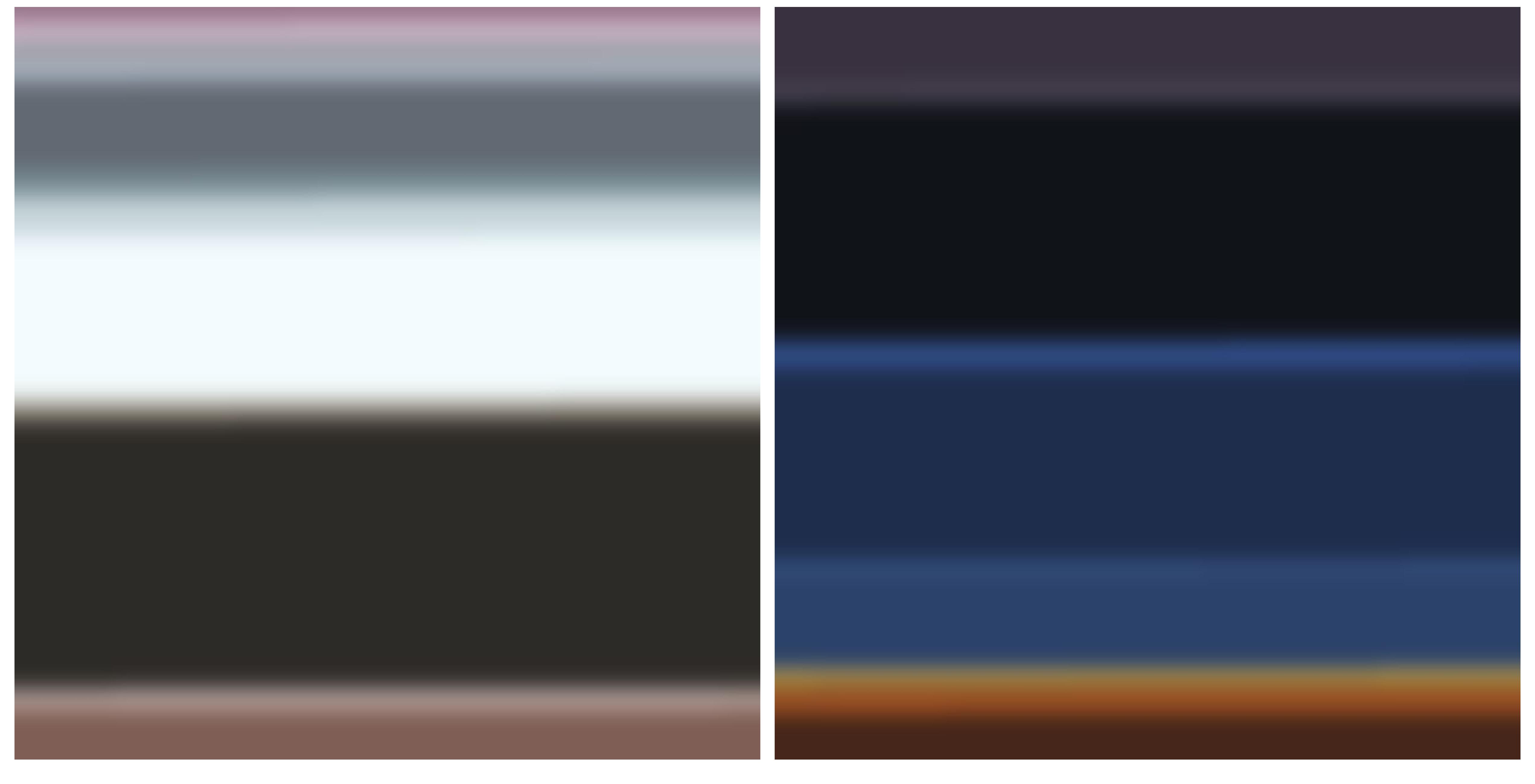












Favela by Day, 12 Colors

Favela by Night, 12 Colors

one more thing...







12 Colors
Unicorn Rainbow Kitten

