

Color Palette Design Doc

Objective and stuff

Application would be a single frontend component composed of probably html, css and javascript. No backend service should be used.

Application will serve artists, photographers, UI designers and similar people. They would be able to find suitable colors that they are looking for or maybe even more things via this application. Application should give feedback regarding similar, contrasting or interesting colors when a starting color is chosen.

Application should have a color selection tab. This tab should hold 3 different views; Hue+Saturation, Hue+Luminosity, Saturation+Luminosity based color selection.

There will be 3 main windows within the app. Color picker, Color analyze (**name will be changed) and color palette. Color picker part will always have the color zoom part. The color picker will have the option to choose 1 of the 3 color picking views; Hue+Saturation Wheel, Hue+Luminosity Map, Saturation+Luminosity Pyramid. By default Hue+Saturation Wheel will be shown and others will be hidden. Color analyze (**) will also have 3 options, color harmony wheel, contrasts, zones. By default color harmony wheels will be shown and others will be hidden.

There should be a Donation button (via some service) and some advertisements (just for server costs) within the application.

Some starting definitions;

Color; Hue*Saturation + Luminosity

Color; Red + Green + Blue

Red, Green and Blue are pure colors. Yellow is Red+Green. Cyan is Green+Blue. Magenta is Red+Blue.

Red, Green, Blue is also depicted as RGB with 0-255 range of value for each of these pure colors. 255,0,0 means Red, 0,255,0 means Green and 0,0,255 means Blue. But they don't need to be exactly at that value to be considered as their individual colors. 1-255,0,0 also means Red, 0,1-255,0 Green, 0,0,1-255 Blue.

Other colors can be generated by mixing these 3 pure colors. 255,255,0 means Yellow, 0,255,255 Cyan, 255,0,255 Magenta. 255,255,255 is White. Black is special and it will be defined in the Luminosity part of this paper. But you can think that 255,255,255 is White and any equal ratio of pure colors used also creates White (example; 10,10,10). Hmm, a question; what is 255,127,0? It is Red + Yellow in equal ratios. Definition of Hue better describes things like that.

Red, Green, Blue can be put into a chamber to be defined in another way. Red can be considered at 0 degree, Green at 120 degree and Blue at 240 degree. From this definition Yellow would be at 60 degree, Cyan at 180 degree, Magenta at 300 degree. This chamber defines the Hue value.

These degrees within the Hue chamber show much a color is mixed with another. For mixing please only consider the following colors; Red, Green, Blue, Cyan, Magenta and Yellow. 150 degrees can be thought as halfway between 120 degree and 180 degree. Because it is halfway, the mixing ratio of 120 degree (Green) and 180 degree (Cyan) is equal. If we chose 130 degrees, it would not be at an equal ratio. Closest colors are still at 120 degrees and 180 degrees. $|120-130|=10$ degrees and $|180-130|=50$ degrees, $50/10=5$. There is a 5 to 1 ratio for this case. Because 130 degrees closer to 120 degrees, Green would be 5 parts and Cyan would be 1 part.

We have been talking about ratios of color but not about how much they are used to create the color. Saturation is the definition of how much color is contributed. When we need to talk about Saturation, we also need to talk about White and Black. Saturation completely disregards White and Black. So given that color with RGB value 255,127,127, we need to remove the white from it. The 127,127,127 part is White and the 128,0,0 part is Red. Saturation is the max value of the RGB values. For this case, saturation is 128. If we see a color with the following RGB; 123,234,0. Saturation of it would be 234.

We have defined so many colors till now, but not the Black. Black is actually the absence of color. Let's continue with an example RGB value; 10,210,20. In order to find Black value, just take the maximum color from the RGB which is 210 for the example and just subtract it from 255. Black would be 45 for this example. With both White and Black defined we can also define the Luminosity. Luminosity is actually the ratio of White to Black. At 0% luminosity, there is no White. Similarly at 100% luminosity there is no Black. At 50% White and Black are at the same ratio.

We have defined Hue, Saturation and Luminosity so far. Let's define our first RGB color with it. Hue; 230 degree, Saturation 200, Luminosity %25. 230 degrees of Hue is between 180 and 240 degrees. $|230-240|=10$, $|230-180|=50$ which creates a 5 to 1 ratio. In order to find the exact values of Cyan (180 degree) and Blue (240 degree), we need to use the Saturation. $5+1=6$, $200/6=33,33$ which is not an integer. But we can still use it. Blue is closer to our color and Cyan is more far away. We can set the value of Cyan as 33 and $200-33=167$ as Blue for saturation. Our current colors are 0,33,33 (Cyan) and 0,0,167 (Blue) for a total of 0,33,200. In the case of the luminosity, the maximum value of our current RGB is 200. $255-200=55$ can be used for luminosity values. At %25 Luminosity, $25/(100-25)=1/3$ would be the ratio of White to Black and $25/100=1/4$ can be directly used for finding White's value. $55*(1/4)=13,75$ with rounding up 14 would be the White. Our RGB value of White is 14,14,14. Final color is $0,33,200+14,14,14=14,47,214$.

Why do we care about Hue, Saturation and Luminosity? Isn't RGB enough? Actually, RGB is good for computers, but for humans and especially artists who are interested in color Hue, Saturation and Luminosity is more useful.

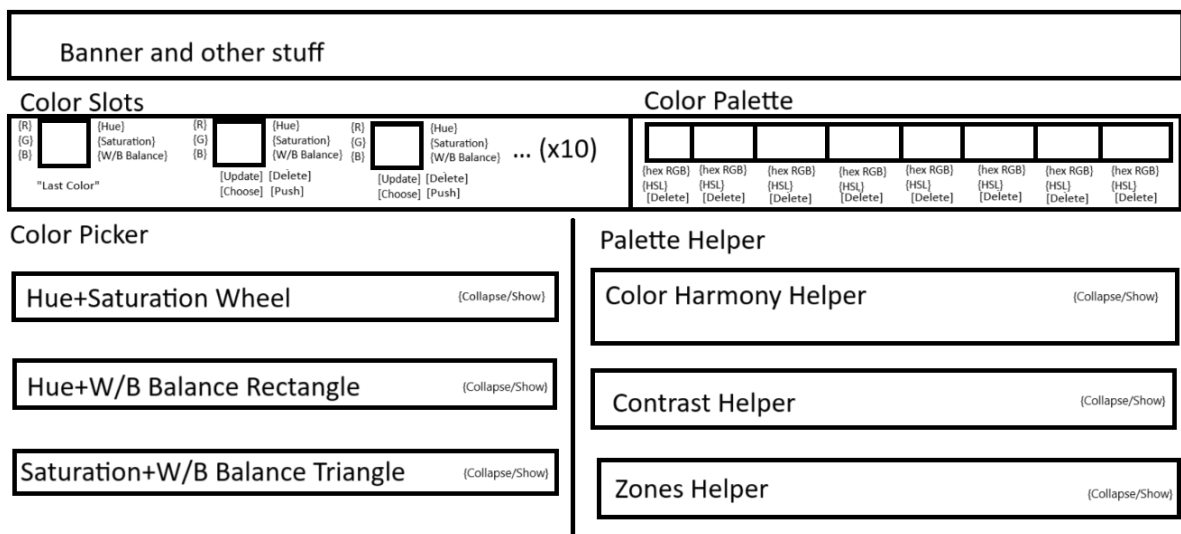
***UPDATE: W3 Defines luminosity in a different way. Consider our luminosity as White/Black!!! Change any occurrence of it as **W/B** when creating the application.

Palette

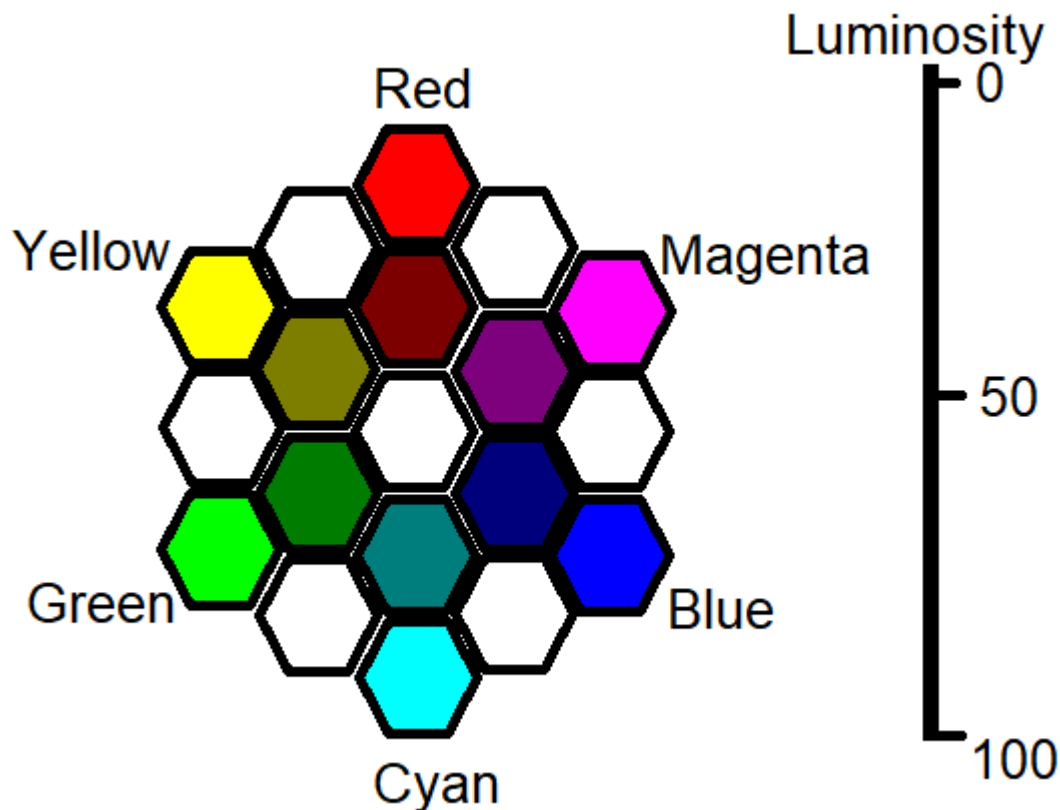
This is the main part of the application. This would have 2 different parts. First part is the chosen color part. When a user hovers around a color within the application and clicks on it, that color should be stored in the “Last Color” box. There should be at least 10 color slots within there (excluding the “Last Color”). Color slots should show the color’s RGB values, CMYK values and Hue-Saturation-Luminosity(W/B Balance) values. Users should be able to update these color slots with the value of the “Last Color” slot. There should be an option to delete the color (updating it with FFFFFFFF) within the chosen slot. Also there should be a button to push the color slot into the color palette’s first empty slot.

In the palette part, there should be 8 palette slots. Also, these slots should have an option to erase the color within the palette slot. Palette slots should show the color’s RGB values, CMYK values, Hue-Saturation-Luminosity(W/B Balance) values and HSL values below.

When a color gets chosen from color slots, any “Palette Helper”’s base color should be updated as that color.



Hue+Saturation Wheel



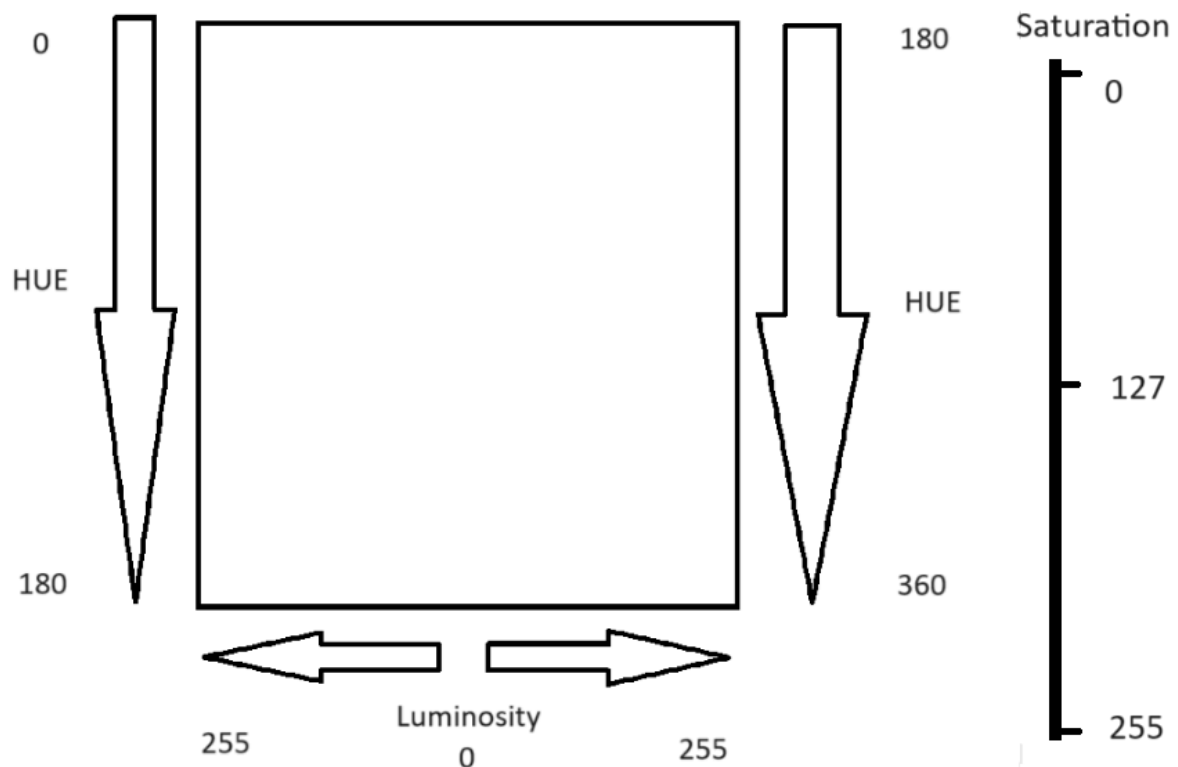
An example of a very simple wheel can be found above. In the top (at Hue 0 degree) Red can be found. Counter-clockwise degree of Hue increases. Saturation increases from center to corners of the wheel. Because Hue, Saturation and Luminosity are 3 different variables, it is almost impossible to show it in a 2D space efficiently. Luminosity can be set with a slider.

Wheel needs to be as big as each possible Saturation value. Center of the wheel is at Saturation 0 with Hue as undefined or maybe 0. Wheel increases 1 by 1 till Saturation value of 255.

Each cell needs to be colored based on the cell's position with degree of place as Hue and distance from center as Saturation. Luminosity(W/B Balance) of the cell needs to be set from the slider's value. Even though Luminosity(W/B Balance) is shown as 0-100, there should be at least 256 slider points. Definition of percentage can be completely dropped if necessary and continue on with a similar thing of the Saturation as for Luminosity(W/B Balance).

This wheel is for people who want to choose a color just by seeing the colors distribution by Hue and Saturation. Luminosity(W/B Balance) can be set manually through the slider and wheel will update all colors at the same time.

Hue+W/B Balance Rectangle



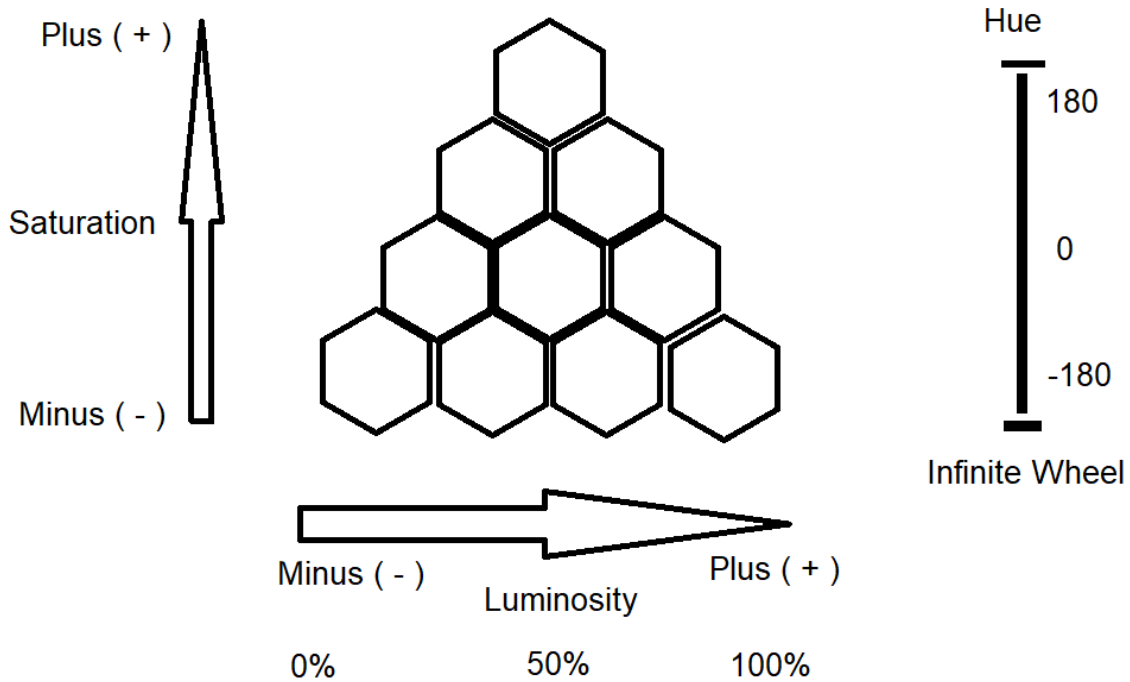
Hue can be from 0 to 360. But because of the odd shape it creates Hue is split into two parts; 0-180 and 180-360. Those parts are put together in a way to achieve maximum contrast by Hue.

Luminosity(W/B Balance)'s range gets affected by the value of Saturation. With low Saturation values, Luminosity(W/B Balance) can vary much more than in a high Saturation situation.

For this view, colors will be shown as rectangles rather than hexagons. Colors start from the center (up to down or vice versa line). Based on Saturation, most left and most right lines will be removed.

This view is for people who want to choose a color just by seeing the colors distribution by Hue and Luminosity(W/B Balance). Saturation can be set manually through the slider and wheel will update all colors at the same time.

Saturation+W/B Balance Triangle



Hue should be an infinite wheel that can roll over 360 and shown as modulo 360.

Pyramid should be 0-255 long from bottom to top.

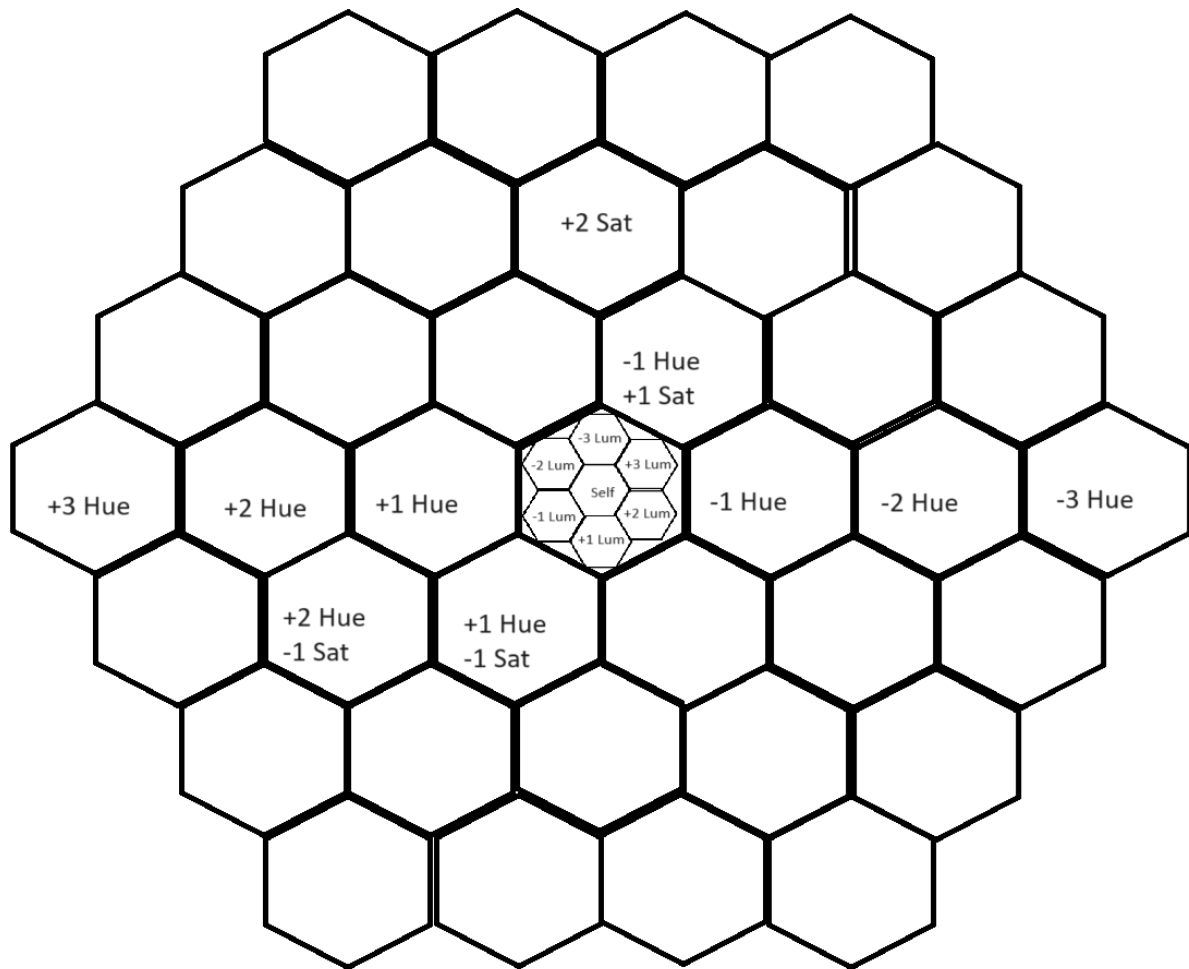
Other information can be easily understandable from the image.

This view is for people who want to choose a color just by seeing the colors distribution by Saturation and ~~Luminosity~~. Hue can be set manually through the slider and wheel will update all colors at the same time.

~~Color Zoom~~ (Abandoned)

~~Color Zoom view just shows neighboring colors of the chosen color. Neighboring colors range from -3 to +3 range of the chosen color by Hue, Saturation, Luminosity and combinations of them.~~

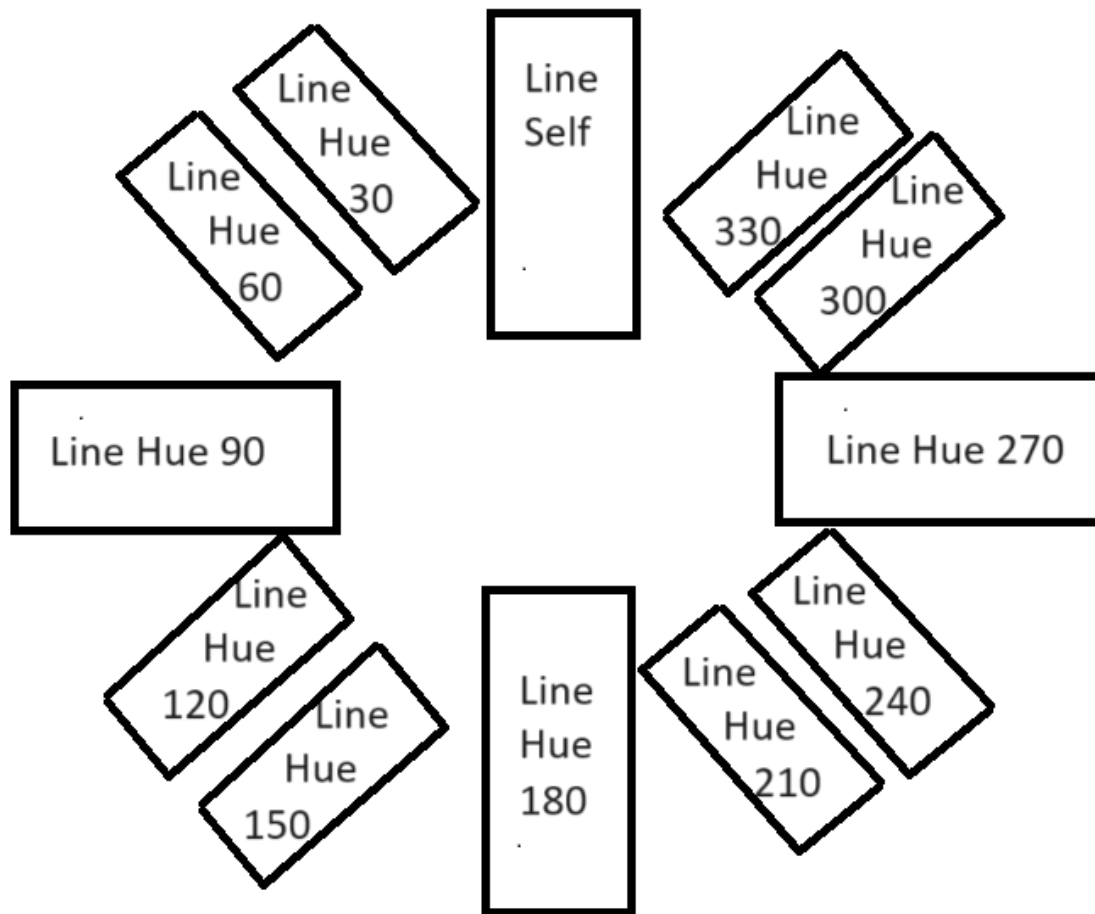
~~Other color choosing views may miss some minor details because of rounding errors. Because of that, the zoom feature is needed.~~



Color Harmony Helper

Color Harmony is an important thing for color palette creation. Choosing unrelated colors can create horrible results. The relations of colors are classified by their Hues and there are several different relationship classes. These relationship classes can be found below figure. Figure includes "Line"s, which are just groups of colors. Each line includes the Hue of the color and some other colors that may be used instead of it. The chosen color's Line is depicted as "Line Self".

When a harmony relationship class is chosen, "Line Self" and other lines that would be in the class should be highlighted. Relationship classes can be found below the figure.



Monochromatic (only self)

Diad positive (60 degrees of Hue and self)

Diad negative (300 degrees of Hue and self)

Complementary (180 degree of Hue and self)

Split complementary (150 degrees and 210 degrees at the same time and also self)

Triad (120 degrees and 240 degrees same time and also self)

Analogous (+30 and 330 at the same time and also self)

Double complementary positive (self, 30, 180, 210 degrees at the same time)

Double complementary negative (self, 330, 180, 150 degrees at the same time)

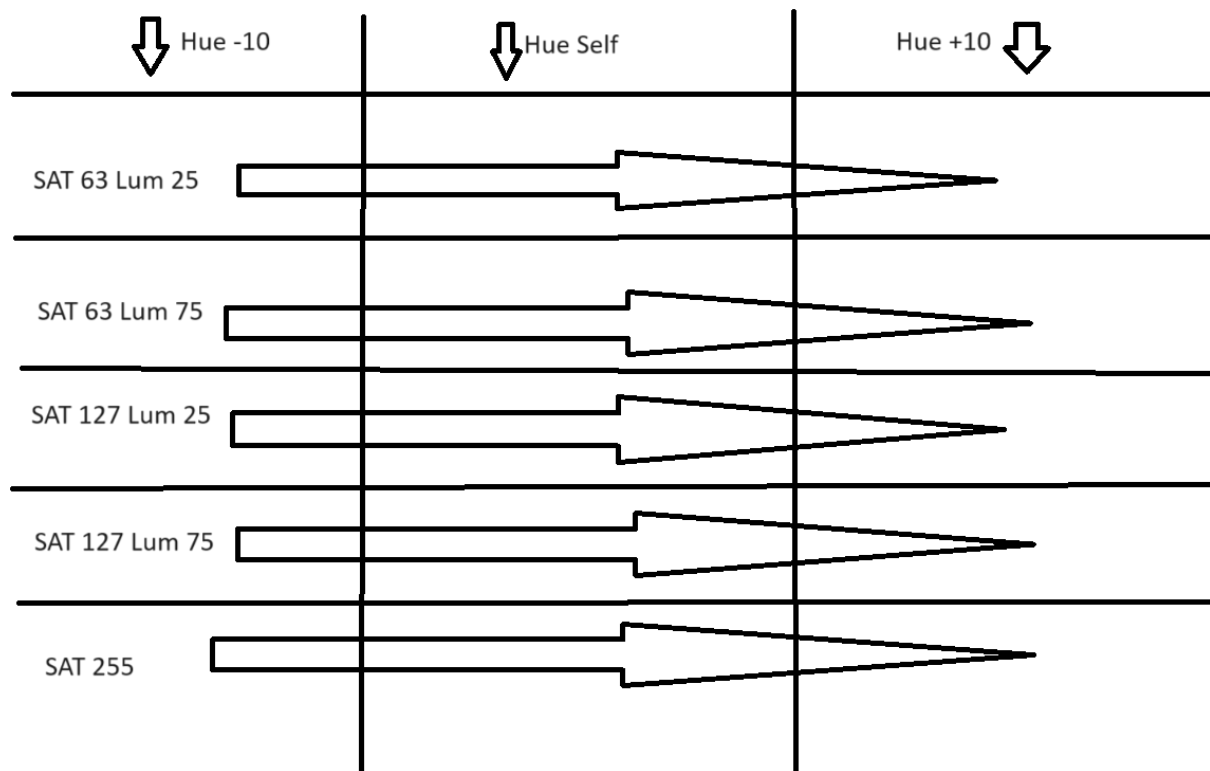
Rectangular tetrad positive (self, 60, 180, 240 degrees at the same time)

Rectangular tetrad negative (self, 300, 180, 120 degrees at the same time)

Square tetrad (self, 90, 180, 270 degrees at the same time)

Polychromatic (self, 60, 120, 180, 240, 300 degrees at the same time)

Lines' internal details are still open of discussion but current details can be found below figure.

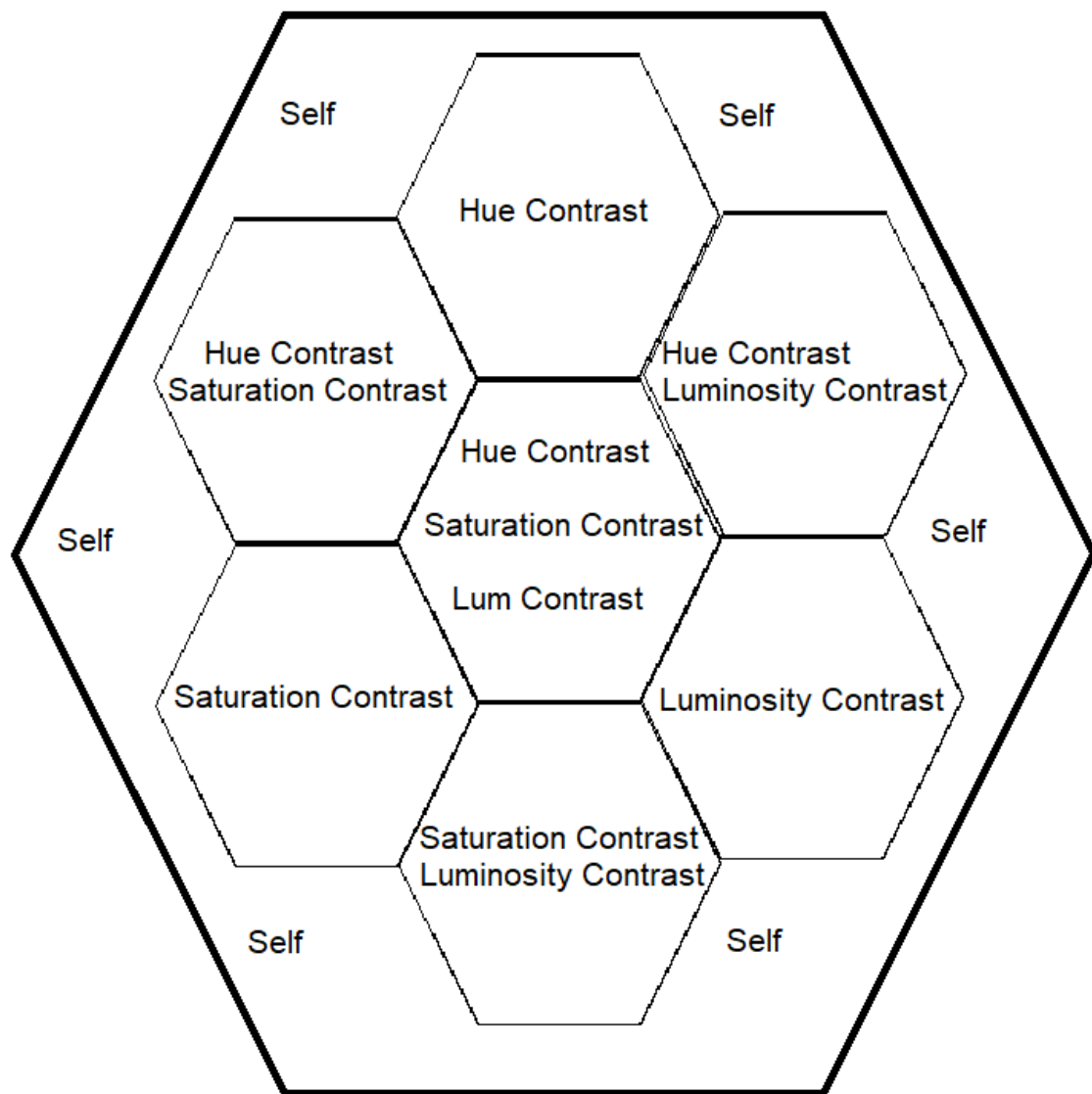


Figure; Single line. At the center north to south column Hue is the same as the chosen color. Left column should be at -10 Hue, the right column should be at +10 Hue. The top row should have the same Saturation and Luminosity-values. Other rows' Saturation and Luminosity-values can be found within the figure.

Contrast Helper

Each color has the values of Hue, Saturation and Luminosity. These values have complementary or "contrast" values. For an example of the Hue value of 210, contrast of it is $210+180=390$ which is equal to 30 degrees of Hue. For an example of the Saturation of 200, $200-127=73$ is the distance of the center value of 127 and contrasted value is $127-73=54$. If we had chosen 100 instead of 200, $100-127=-27$ and then $127-(-27)=154$ would be the contrasted value. Contrast for Luminosity is similar to Saturation. Distance to the center point can be used to find the contrasted value.

Below figure shows the view of contrasts. Bigger hexagon includes the chosen color and smaller hexagons within the bigger one are for showing individual or combined contrast values of the chosen color.



Zones Helper

Zones are actually a photography topic but it can still be used to create color palettes. (<https://progradedigital.com/understanding-and-using-the-zone-system-in-photography/> can be used to read more about zones)

For our application zones can be thought of as creating different Saturation and Luminosity distributions. Hue would be the same as the chosen color but Saturation and Luminosity would be changed based on chosen zone distributions.

Zones are defined as how much gray (from pure white to pure black) a color is. Based on how much white there is, the zone is defined. RGB value of 10,10,10 is at Zone 0, similarly RGB value of 250,250,250 is at Zone 10. But for our case we won't be just interested in the gray. Zone distributions can be found within the below table.

Zone table shows the starting value, middle value, end value and a specially chosen “Identity” value. Within the Zones view of the application, users would be able to choose which one of those value choices would be used for zones.

We will create our colors by Saturation, White and Black in Zone view. 3 numbers are required here. By default 7, 2, 1 would be chosen and the users will have the option to change it. When we add up the zones’ values together, most of the time it won’t add up to 255 (not RGB value but Saturation + White + Black value) and sometimes value can be more than 255. Just because of that we need to give options to users to change how it should be calculated.

Possible options are; equally, biggest and smallest. For the equal calculation; if the total number is not equal to 255, add/remove the difference from each of the Saturation, White and Black values equally. Similarly for the biggest and smallest ones, changes should be done the biggest or the smallest ones first. If it is not enough choose the second ones. Also if there are 2 (or maybe even 3) biggest or smallest values, do the changes equally to them. There is still a chance of values going bigger than 255 or remaining below 255. If more than 255, just remove the extra. For the below of 255, add the remaining number to the biggest one(s).

We have 3 values and we can show the permutation of these values within our application like S7-W2-B1, S7-W1-B2, S2-W7-B1, S2-W1-B7, S1-W7-B2, S1-W2-B7. There are 6 possibilities, we can use the hexagon style or any other style to show it. It is not a big deal, can be chosen however wanted.

With the calculated zone values, create the RGB value with the following equation; White + Hue*Saturation.

Zone	Start	Middle	End	Identity
0	0	11	22	0
1	23	34	45	25
2	46	57	69	50
3	70	81	92	76
4	93	104	115	101
5	116	127	139	127
6	140	151	162	154
7	163	174	185	179
8	186	198	209	205
9	210	221	232	230
10	233	244	255	255