

# Couleurs et valeurs

Par Vincent Lesbros

13 mai 2024 - Cyclonium

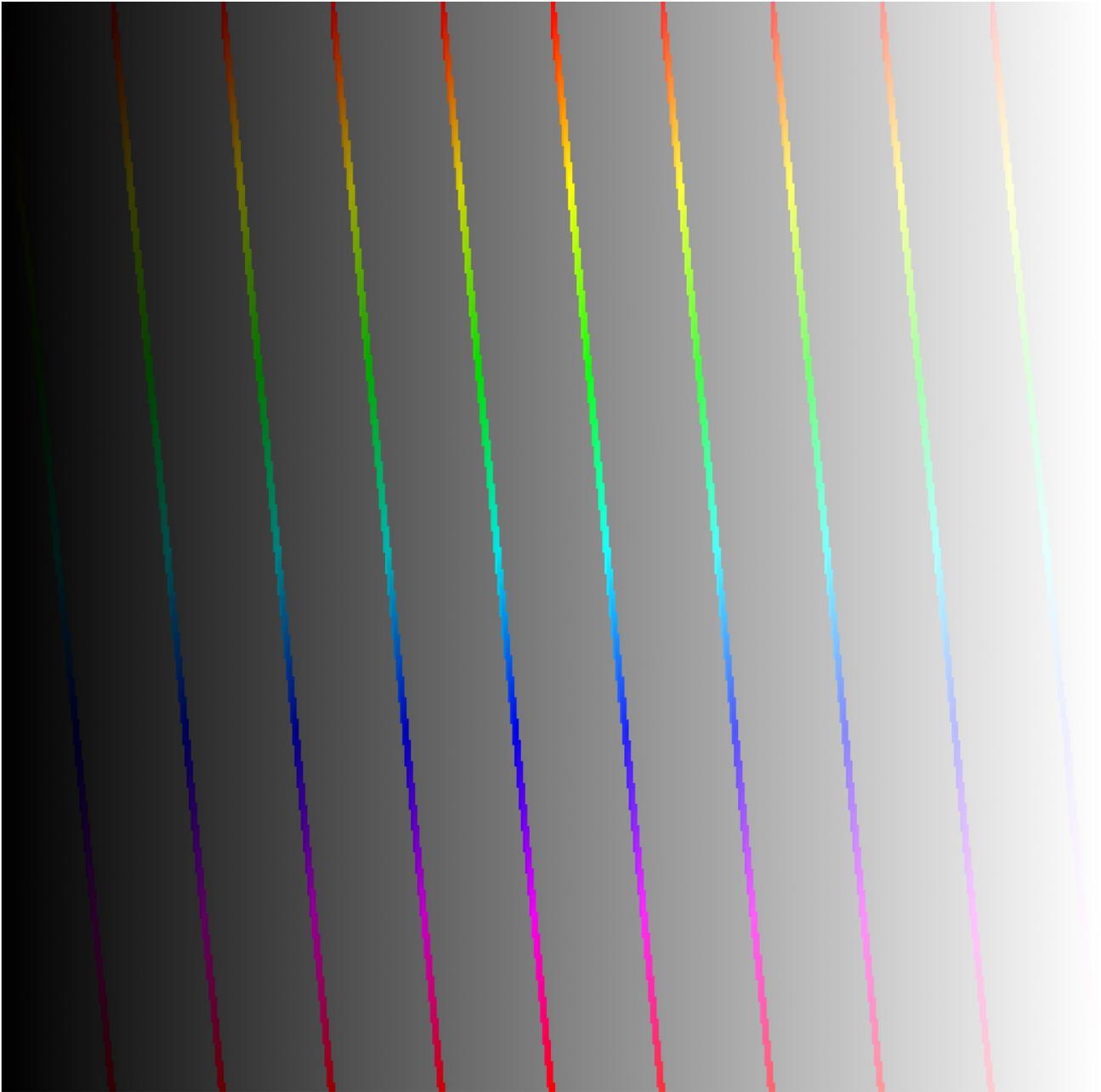


Figure 1

Mode HSL

Nombre de couleurs RVB distinctes : 8068 sur une population de 50000

Nombre de tours de teinte : 10.

Sur un fond dégradé du noir au blanc de gauche à droite, on dispose les couleurs d'une gamme en fonction de la teinte verticalement et de la valeur horizontalement. Les couleurs de la gamme sont choisies avec une saturation maximale.

On s'attendait, avec cette disposition, à obtenir un contraste minimal entre les teintes colorées et les valeurs de gris du fond. Cependant, à l'œil, nous ne sommes pas satisfaits : dans la zone centrale, les teintes colorées semblent plus claires que le fond.

La formule de calcul de la luminance, dans le système HSL à partir des composantes RVB, est la suivante :

```
r /= 255, g /= 255, b /= 255; // normaliser [0, 1]
let max = Math.max(r, g, b), min = Math.min(r, g, b);
let l = (max + min) / 2;
// la luminance est l * 100 (pour être exprimée en pourcentage).
```

C'est donc la moyenne entre la plus grande et la plus petite des composantes RVB de la couleur. Or il existe d'autres systèmes utilisés pour calculer la luminance. En particulier le système BT 709. Dans celui-ci, la luminance est calculée à partir de trois coefficients dont la somme est égale à 1 :

Coefficients :  $R = 0.2126$   $V = 0.7152$   $B = 0.0722$ .

La luminance est alors :  $r * R + v * V + b * B$ , sur une échelle de 0 à 255.

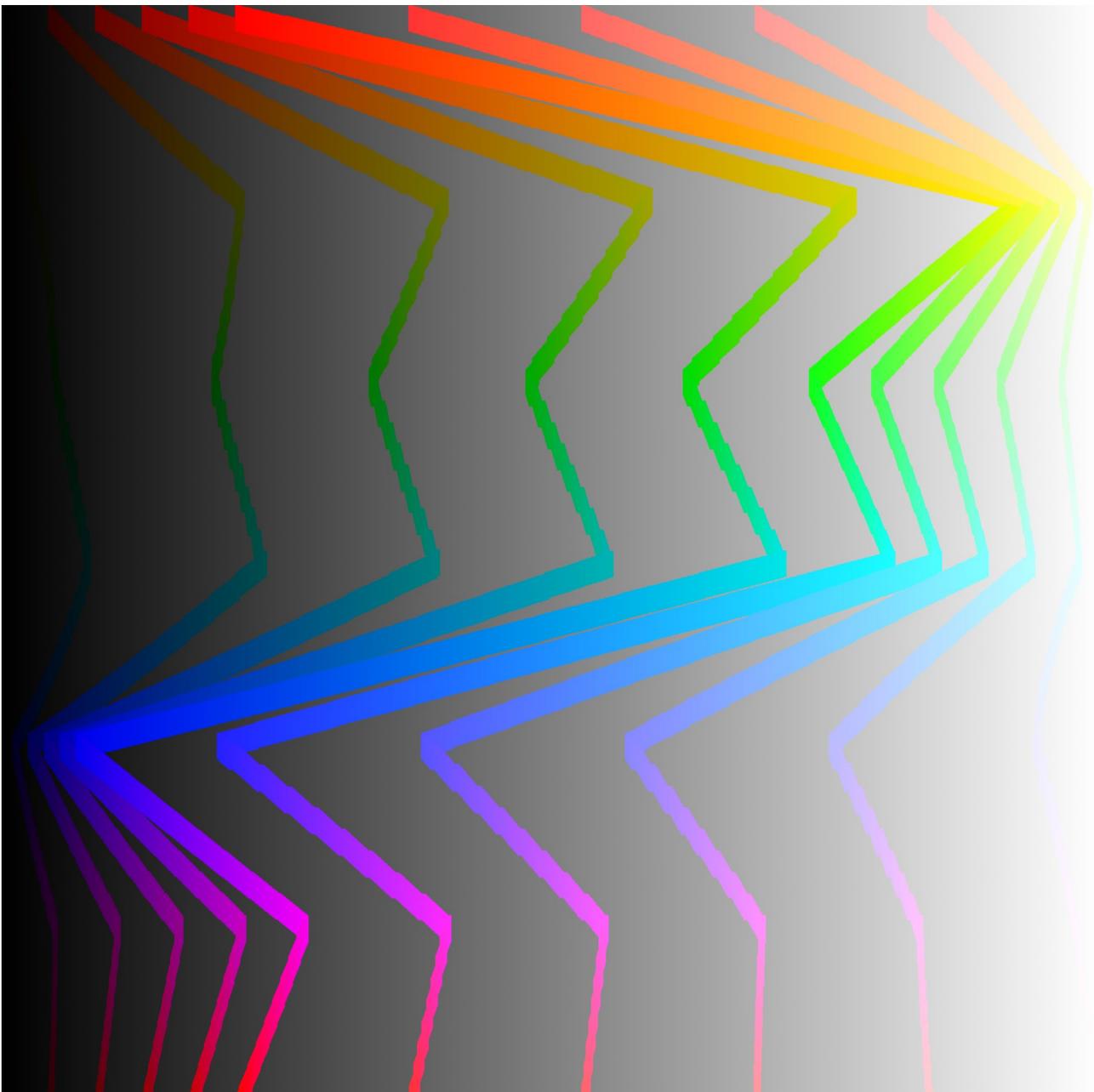


Figure 2

Coefficients :  $R = 0.2126$   $V = 0.7152$   $B = 0.0722$  (BT 709)

Nombre de couleurs RVB distinctes : 8068 sur une population de 50000

Nombre de tours de teinte : 10.

Mais la norme a été établie pour les écrans cathodiques et ne semble plus correspondre à la luminosité des couleurs sur les écrans à cristaux liquides.

On le voit, le coefficient attribué à la composante bleue est trop faible, ce qui a pour effet de placer les bleus trop à gauche, dans la partie sombre.

Je cherche donc de nouveaux coefficients pour essayer de correspondre à la vision des couleurs que j'ai sur mon écran.

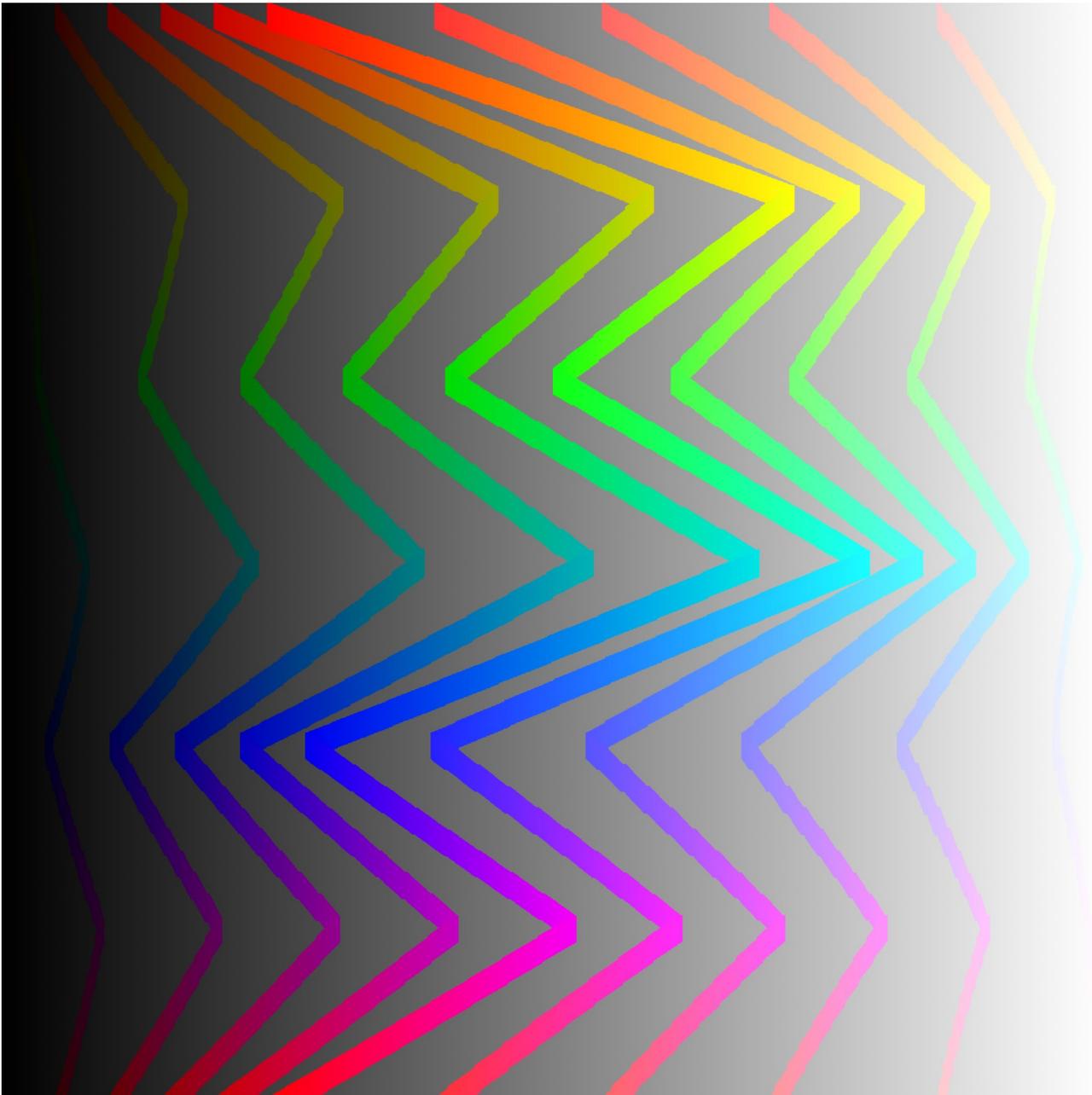


Figure 3

Coefficients :  $R = 0.24073804467105067$   $V = 0.4643142042886451$   $B = 0.2949477510403043$

Nombre de couleurs RVB distinctes : 8068 sur une population de 50000

Nombre de tours de teinte : 10.

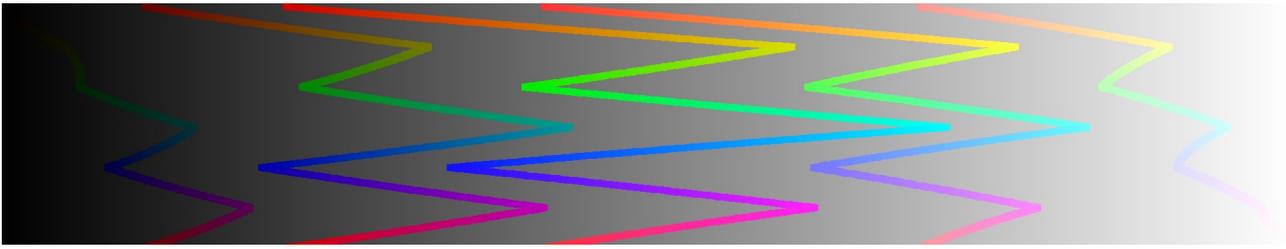


Figure 4 Légende : image de 1920x360 pixels  
Coefficients : R = 0.2714055671667086 V = 0.4309605598569525 B = 0.297633872976339  
Nombre de couleurs RVB distinctes : 3864 sur une population de 6000  
Nombre de tours de teinte : 5.

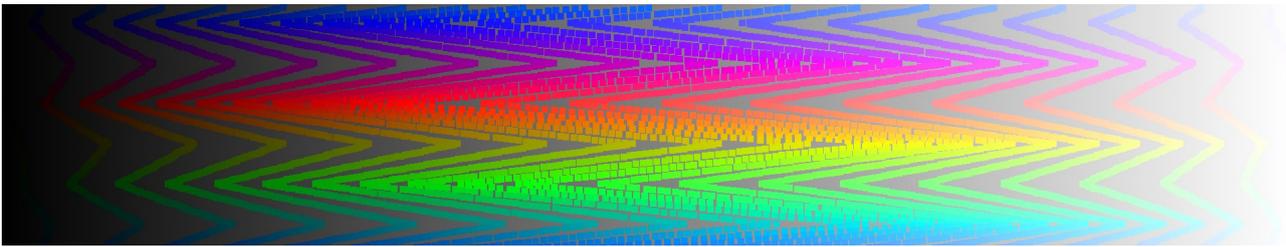


Figure 5 Légende : image de 1920x360 pixels  
Coefficients : R = 0.298879202988792 V = 0.3777477261228295 B = 0.3233730708883786  
Nombre de couleurs RVB distinctes : 5462 sur une population de 6000  
Nombre de tours de teinte : 20.  
Phase : 143.46201743462

Comme les rouges se trouvent aux extrémités du cercle chromatique vers  $0^\circ$  et vers  $360^\circ$ , j'ai ajouté un paramètre de phase de teinte pour pouvoir décaler l'image verticalement.

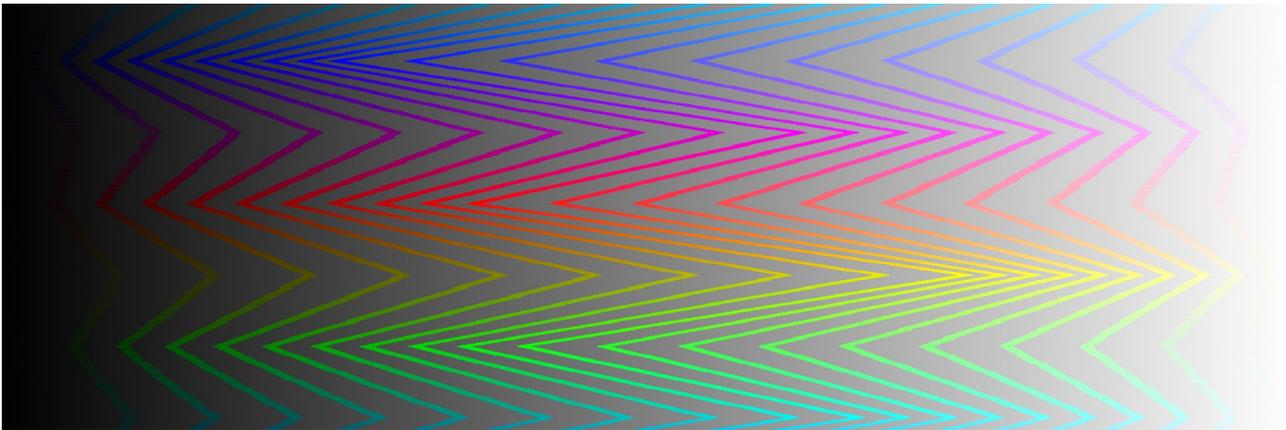


Figure 6 Légende : image de 2400x800 pixels  
Coefficients : R = 0.359817351598174 V = 0.37782036388953294 B = 0.26236228451229315  
Nombre de couleurs RVB distinctes : 15709 sur une population de 100000  
Nombre de tours de teinte : 20  
Saturation : 100  
Phase de la teinte : 168.

# Expérimentations

Page d'expérimentation : <https://www.cyclonium.com/atelier/couleurs/etalonnage.html>

## Première approche

En haut de la page, trois curseurs, Rouge, Vert, Bleu permettent de sélectionner une couleur. Un quatrième curseur sélectionne une valeur de gris.

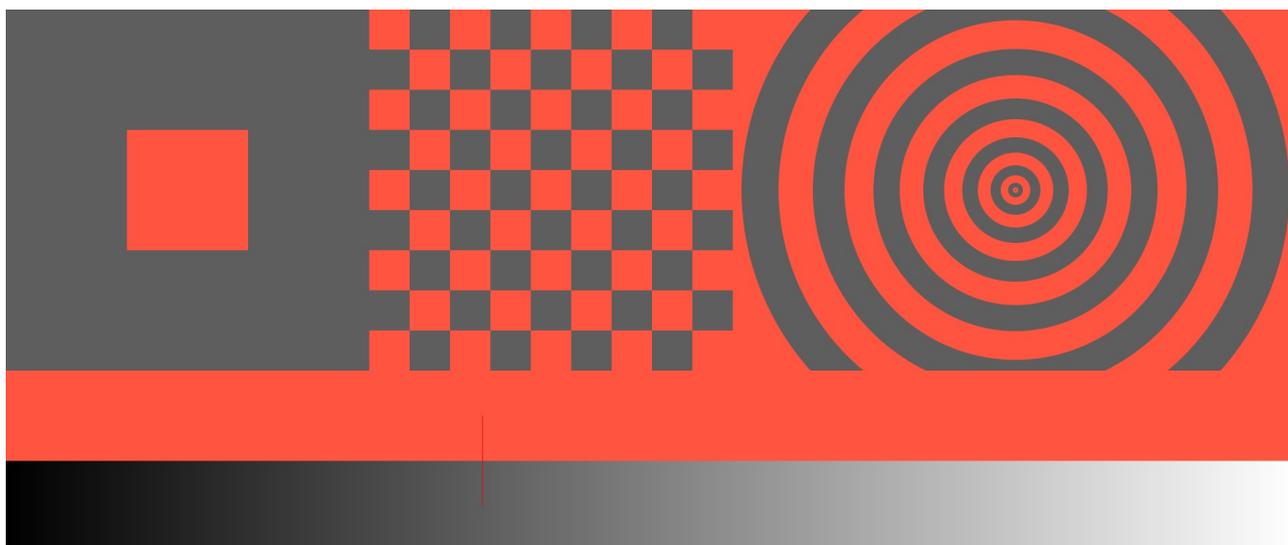


Figure 7 : En manipulant les curseurs, on obtient dans le premier canevas des images de ce type. La partie haute montre des mosaïques construites avec la couleur et la valeur de gris. La partie basse affiche un dégradé du noir au blanc juxtaposé avec la bande de couleur. Un petit trait vertical rouge indique la position de la valeur de gris sélectionnée par rapport au dégradé. On peut faire glisser ce curseur pour modifier la valeur de gris.

Le jeu est de rechercher, par tâtonnement, la valeur de gris correspondant le mieux à la valeur ressentie de la couleur.

Dans la partie basse, le contraste simultanément entre l'aplatissement de couleur et le dégradé de gris nous fait percevoir la partie droite de la bande de couleur plus sombre que la partie gauche, c'est une illusion d'optique : la couleur est la même partout.

Vers la gauche, le gris semble plus sombre que la couleur. Vers la droite, c'est l'inverse. Il s'agit donc de placer le curseur à l'endroit de la bascule, là où la différence semble la plus faible.

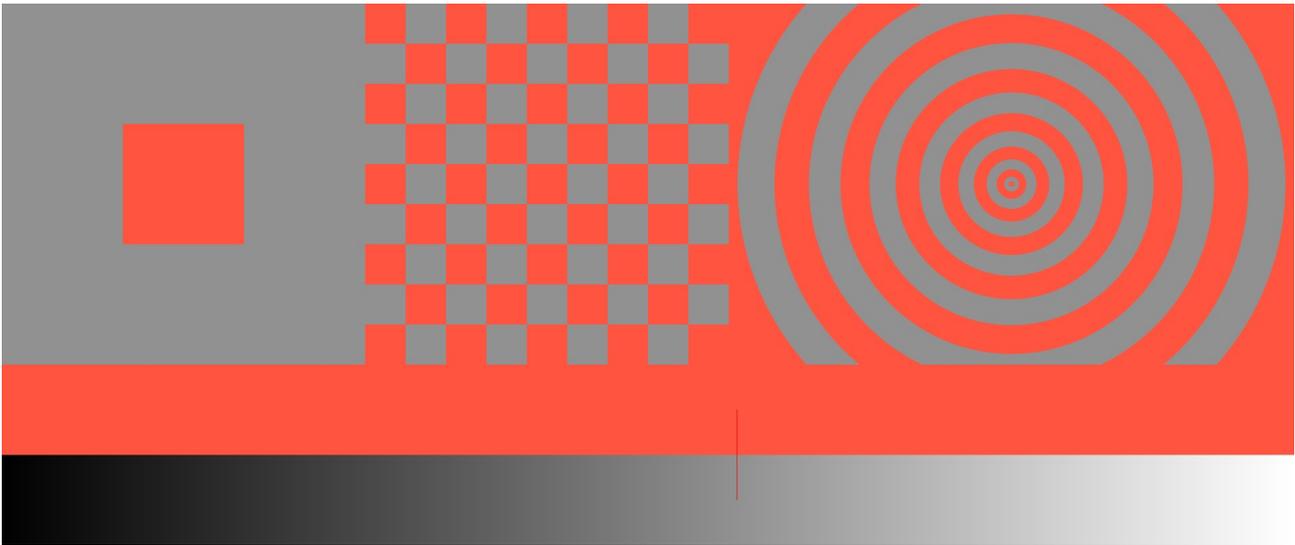


Figure 8 : Pour mon écran, dans mes conditions d'éclairage, la position idéale semble être celle-ci dessus.

Le bouton « Valider l'échantillon » ajoute ma mesure dans la table située en dessous de l'image.

<b>code</b>	<b>rouge</b>	<b>vert</b>	<b>bleu</b>	<b>gris</b>
#ff5540	255	85	64	145

J'effectue une série de mesure :

<b>code</b>	<b>rouge</b>	<b>vert</b>	<b>bleu</b>	<b>gris</b>
#ff5540	255	85	64	145
#808080	128	128	128	128
#80b54b	128	181	75	155
#000000	0	0	0	0
#ff0000	255	0	0	118
#ffff00	255	255	0	211
#ff00ff	255	0	255	175
#00ff00	0	255	0	190
#0000ff	0	0	255	116
#a93b74	169	59	116	105
#64c22f	100	194	47	154

On peut échanger les mesures faites avec un tableur comme LibreOffice Calc.

Pour cela, dans la page on utilise le bouton « Sélectionner la table » et on peut copier les données vers le tableur.

	A	B	C	D	E	F
1						
2						
3	code	rouge	vert	bleu	gris	
4	#ff5540	255	85	64	145	
5	#808080	128	128	128	128	
6	#80b54b	128	181	75	155	
7	#000000	0	0	0	0	
8	#ff0000	255	0	0	118	
9	#ffff00	255	255	0	211	
10	#ff00ff	255	0	255	175	
11	#00ff00	0	255	0	190	
12	#0000ff	0	0	255	116	
13	#a93b74	169	59	116	105	
14	#64c22f	100	194	47	154	
15						
16						

Figure 9 : Dans l'autre sens, pour reprendre les données du tableur et les réintroduire dans la page, il faut sélectionner uniquement les cellules de valeur, pas les intitulés ni la colonne de code, copier, aller sur la page, utiliser le bouton « Sélectionner la table » et coller.

Dans la page, un clic sur un code de la table rappelle la ligne correspondante et permet de rééditer un échantillon.

## Détermination des coefficients

À partir de cette table de mesure, on peut essayer de calculer les coefficients R, V, B par la méthode des moindres carrés.

Bouton : « Calculer les coefficients ».

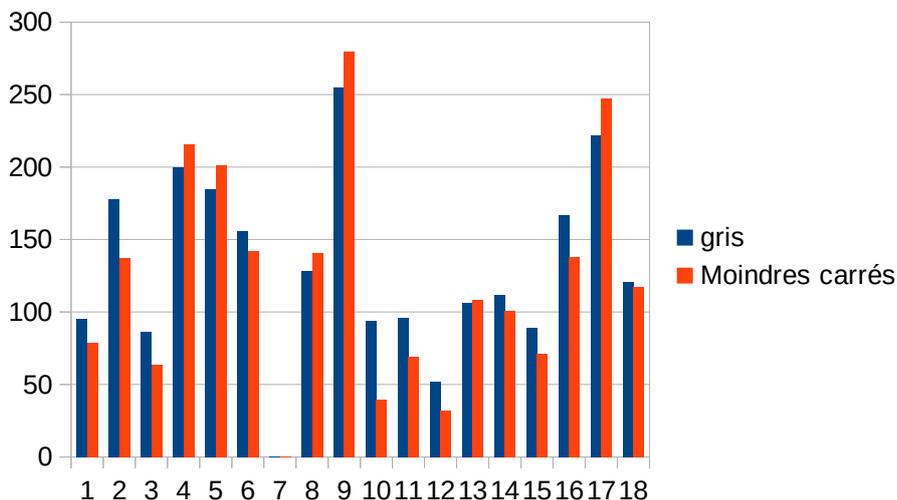


Figure 10 : Exemple de comparaison entre les niveaux de gris mesurés et les niveaux de gris recalculés par les coefficients trouvés d'après les échantillons par la méthode des moindres carrés.

Note : il faut suffisamment d'échantillons pour que cela soit calculable. La somme des coefficients doit être égale à 1, donc j'applique une règle de trois sur le résultat pour ajouter cette contrainte.

## Évaluer une couleur

En fonction des coefficients entrés ou calculés, le bouton « Évaluer la couleur » calcule le niveau de gris de la couleur courante.

## Seconde approche

Dans la section suivante de la page, on a à disposition l'outil utilisé pour générer les images du début de ce papier où les couleurs d'une gamme sont placées dans un dégradé de gris en fonction de la teinte, verticalement et de la valeur calculée depuis les coefficients actuels, horizontalement.

On contrôle :

- La taille des rectangles de couleur à tracer pour chaque échantillon de la gamme.
- La taille de l'image à générer.

Et pour la définition de la gamme :

- Le nombre de couleurs initial, la population, avant le filtrage imposant que chaque couleur de la gamme exprimée en RVB soit unique.
- Le nombre de « tours de teinte » : combien de fois parcourir le cercle chromatique au cours de la construction.

Voici le code de construction :

```
// construction de gammes
function pseudoCouleursCirculaires(taille, toursDeTeinte) {
  let couleurs = [];
  let pasDeLuminance = 100 / taille;
  let pasDeTeinte = 360 * toursDeTeinte / taille;
  let saturation = Number(document.getElementById("gammeSaturation").value);

  for (let c = 0; c < taille; ++c) {
    let luminance = c * pasDeLuminance;
    let teinte = (c * pasDeTeinte) % 360;
    couleurs.push(hslToRgb(teinte, saturation, luminance));
  }
  return couleurs;
}
```

Initialement, la saturation était fixée à 100 %, mais pour expérimenter, j'ai ajouté un curseur permettant de régler la saturation afin d'en voir les effets.

## Conclusion temporaire



Figure 6

En observant à nouveau la figure 6, où les coefficients sont  $R = 0.359817351598174$   $V = 0.37782036388953294$  et  $B = 0.26236228451229315$ , il semble que les couleurs dont une des composante est très forte par rapport aux deux autres possèdent une valeur largement sous-évaluée par la formule linéaire.

Cela indique peut-être qu'une formule de calcul de la valeur plus conforme à notre vision devrait non seulement utiliser les composantes mais également les min et les max des composantes.

### Le but

Mon but dans cette recherche est de générer des gammes de couleurs permettant de visualiser des plans de valeurs en utilisant des pseudo couleurs de manière à rendre visible des échelles de valeurs les plus grandes possibles.

En effet, si je possède un plan de valeurs avec des variations de l'ordre de 1 à 10000 par exemple, et que je construis une image en niveaux de gris pour la représenter, les valeurs doivent être rabattues à une échelle de gris de 0 à 255. Ce qui a pour conséquence de rendre invisibles les petites variations.

Les pseudo couleurs, mappant les valeurs sur une échelle colorée plus large, permettent de mieux rendre visible les petites variations. Ce procédé est souvent utilisé en cartographie ou en imagerie médicale.

Je souhaite en plus, que la fonction passant des valeurs à la couleur soit une bijection, et que l'impression visuelle des luminances reste cohérente.