Colors in images

Color spaces, perception, mixing, printing, manipulating . . .

Tomáš Svoboda

Czech Technical University, Faculty of Electrical Engineering
Center for Machine Perception, Prague, Czech Republic

svoboda@cmp.felk.cvut.cz

http://cmp.felk.cvut.cz/~svoboda

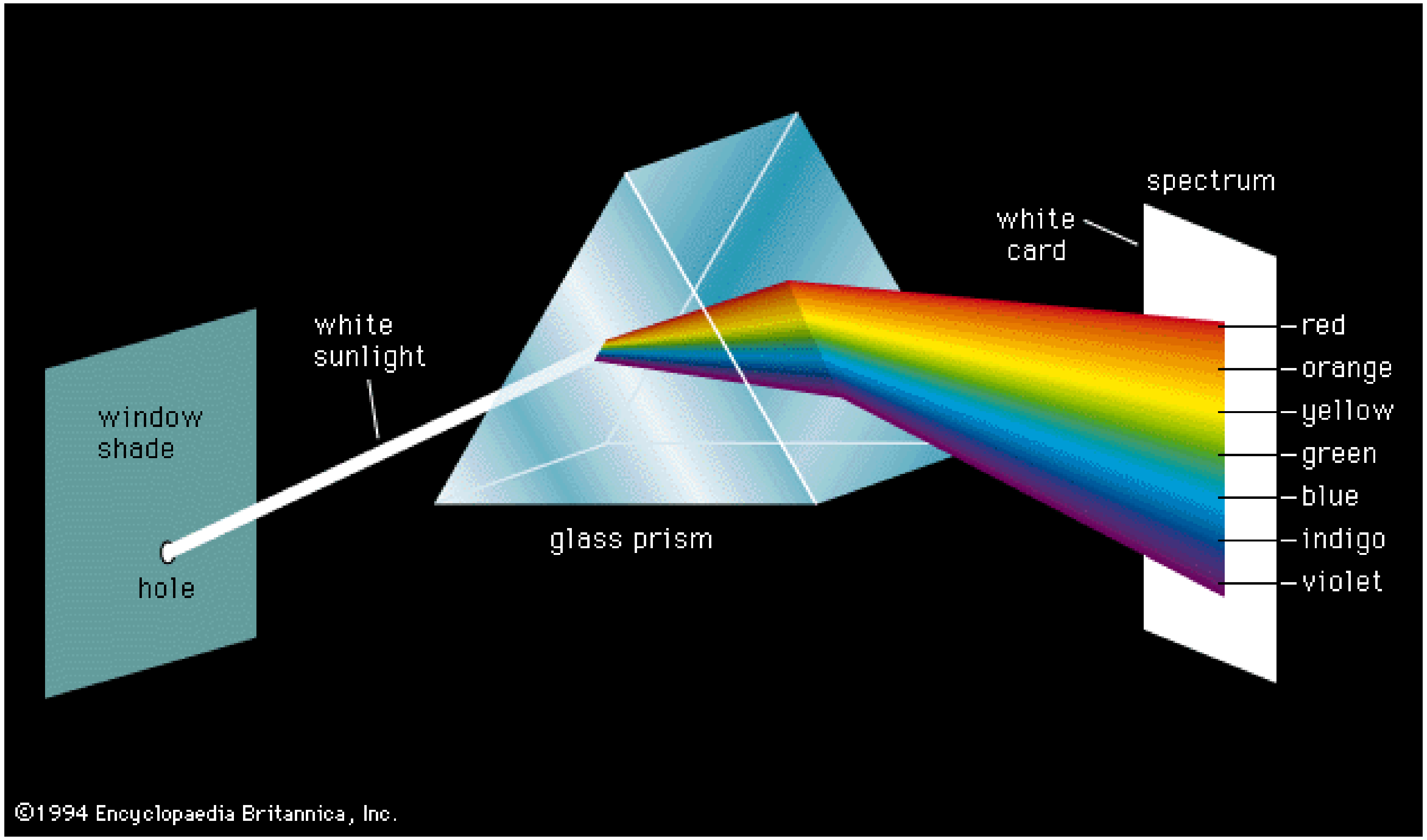
Warning

- rather an overview lecture
- pictorial, math kept on minimum
- knowing keywords you may dig deeper

Thanks Wikipedia for many images.

Color Spectrum

Color is a human interpretation of a mixture of light with different wavelength λ (projected into a retina or camera photoreceptors).



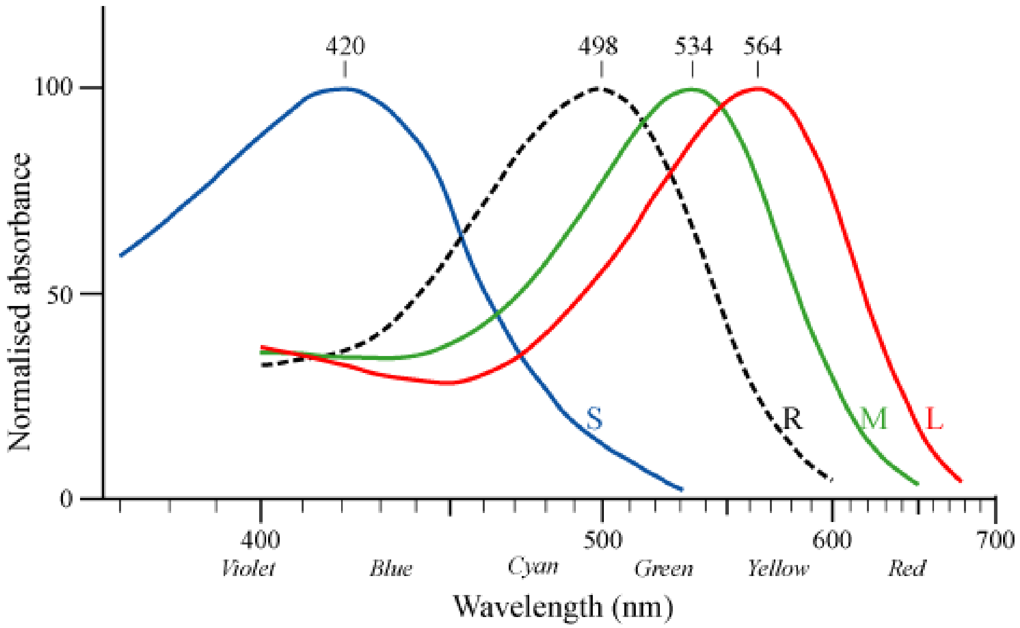
Isaac Newton's experiment (1666).

Perception of Light

- Human eye contains three types color receptor cells, or cones.
- Their sensitivity is a function of wavelength.
- Three peaks may be approximately identified in BLUE, GREEN, RED.
- Combination of the responses give us our color perception. tristimulus model of color vision.

Marking according to wavelengths

- S — short
- M — medium
- L — long

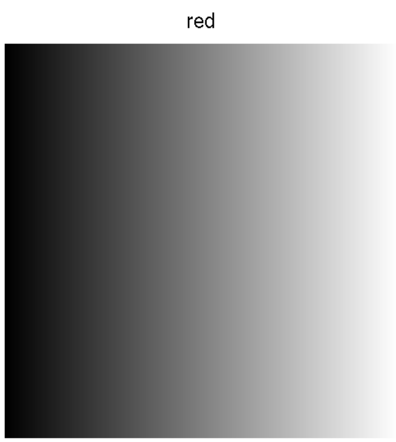


RGB color model

- A color point is represented by three numbers $[R, G, B]$
- $[R, G, B]$ have typically range $0 \ldots 255$ for most common 8-bit images

RGB color model

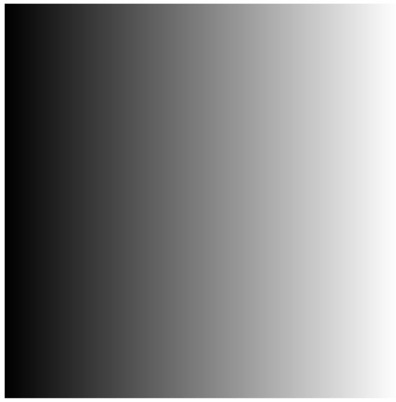
- A color point is represented by three numbers $[R, G, B]$
- $[R, G, B]$ have typically range $0...255$ for most common 8-bit images



RGB color model

- A color point is represented by three numbers $[R, G, B]$
- $[R, G, B]$ have typically range $0 \ldots 255$ for most common 8-bit images

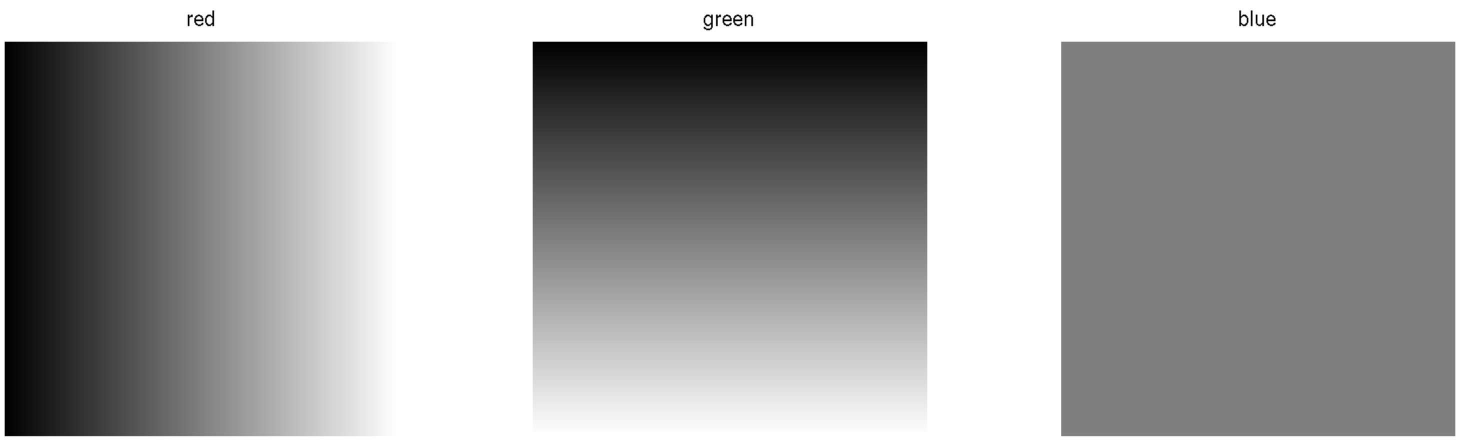
red



green

RGB color model

- A color point is represented by three numbers $[R, G, B]$
- $[R, G, B]$ have typically range $0 \ldots 255$ for most common 8-bit images

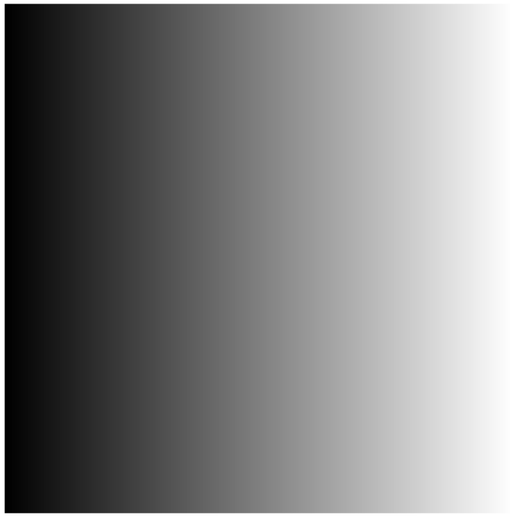


RGB color model

- A color point is represented by three numbers $[R, G, B]$
- $[R, G, B]$ have typically range $0 \dots 255$ for most common 8-bit images

RG only, B zero

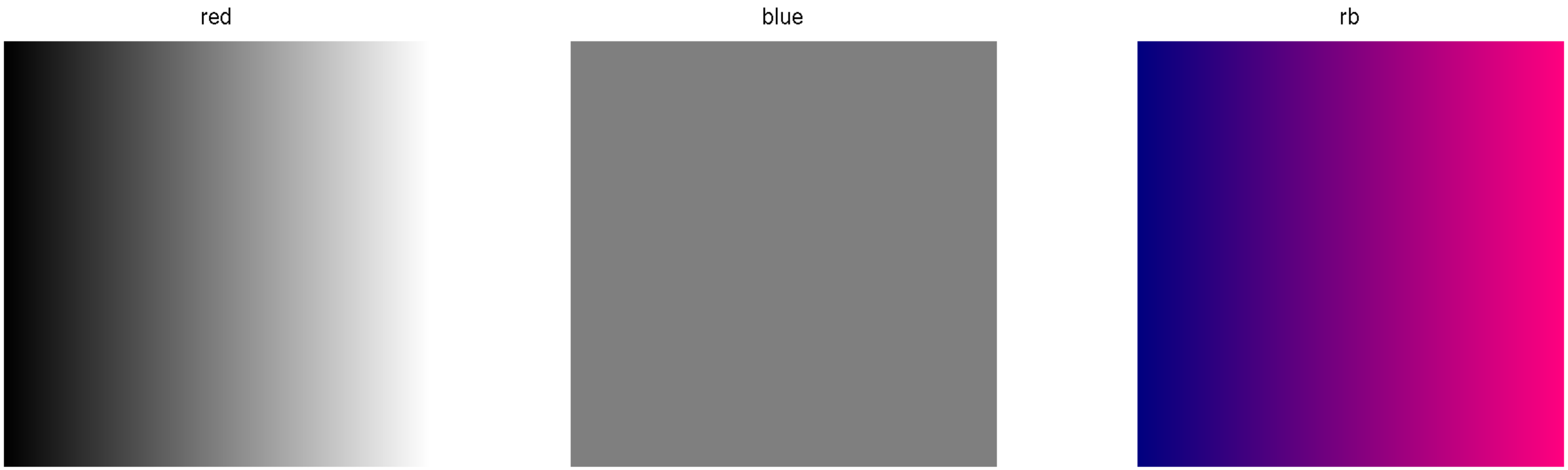
red



green

rg

RB only, G zero



GB only, R zero

green

blue

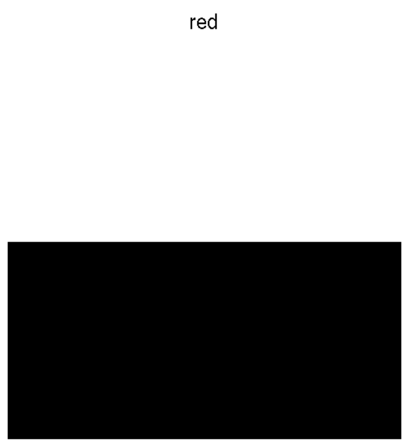
gb

Additive mixing

- computer screens, TV, projectors
- Primary colors: ones used to define other colors, [R,G,B]
- Secondary colors: pairwise combination of primaries, [C,M,Y] (Cyan, Magenta, Yellow)

Additive mixing

- computer screens, TV, projectors
- Primary colors: ones used to define other colors, [R,G,B]
- Secondary colors: pairwise combination of primaries, [C,M,Y] (Cyan, Magenta, Yellow)

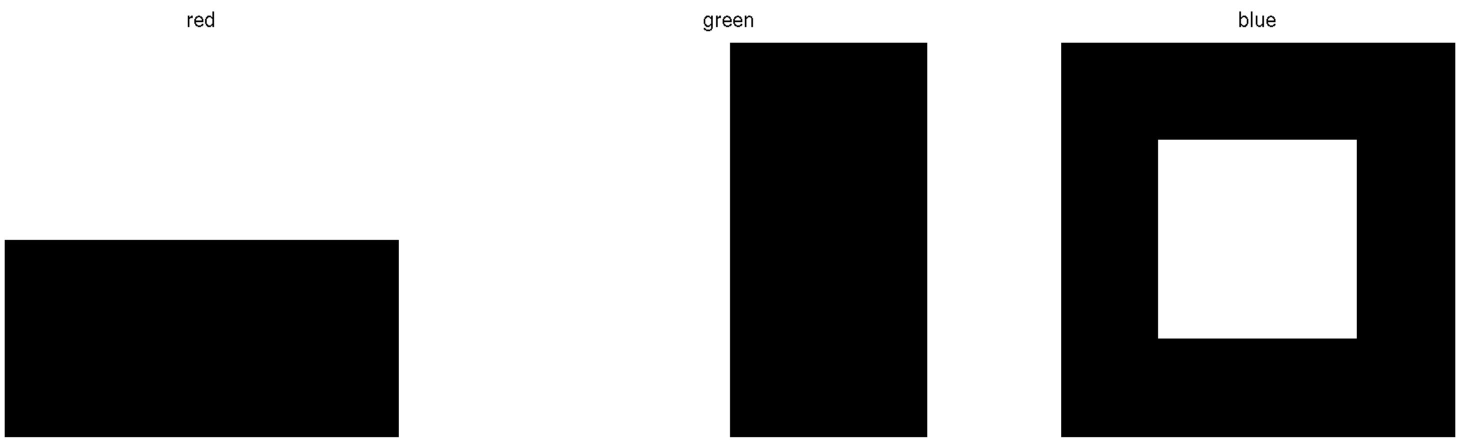


Additive mixing

- computer screens, TV, projectors
- Primary colors: ones used to define other colors, [R,G,B]
- Secondary colors: pairwise combination of primaries, [C,M,Y] (Cyan, Magenta, Yellow)

Additive mixing

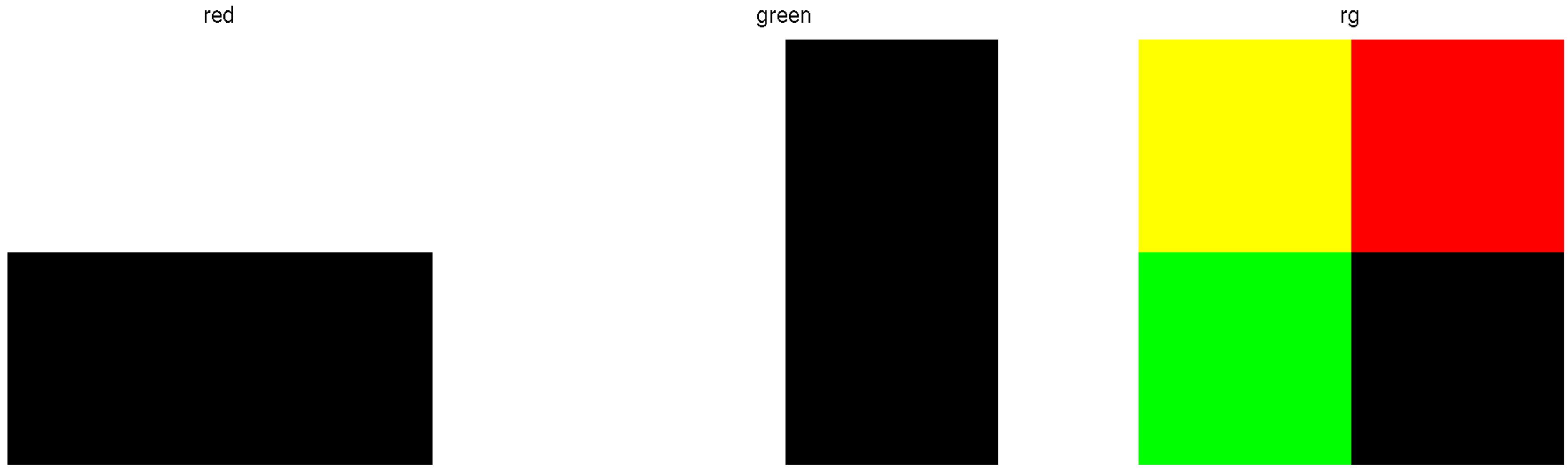
- computer screens, TV, projectors
- Primary colors: ones used to define other colors, [R,G,B]
- Secondary colors: pairwise combination of primaries, [C,M,Y] (Cyan, Magenta, Yellow)



Additive mixing

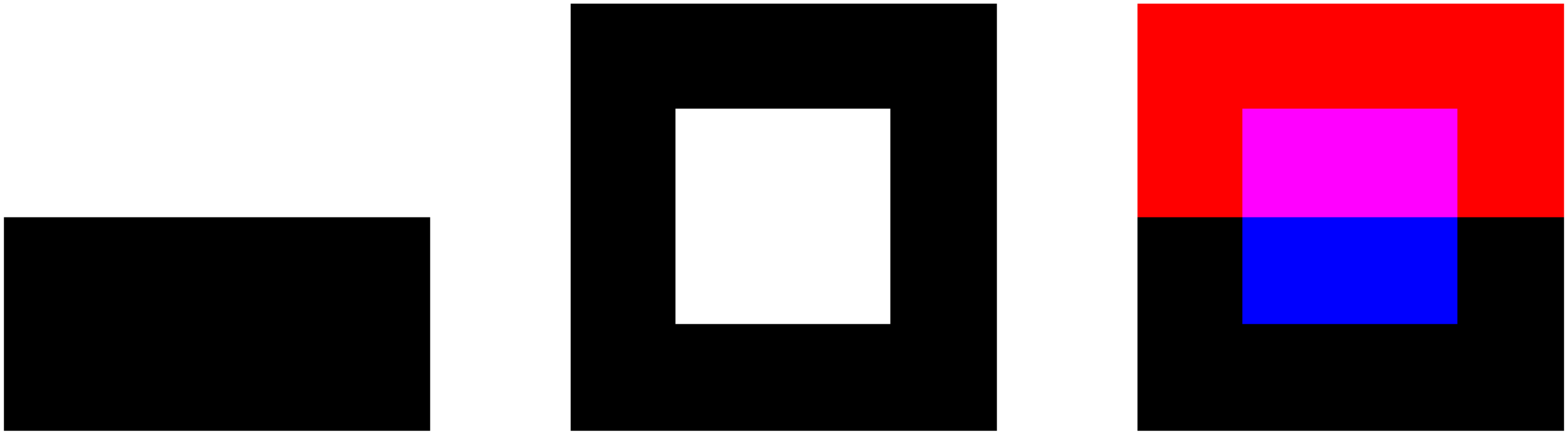
- computer screens, TV, projectors
- Primary colors: ones used to define other colors, [R,G,B]
- Secondary colors: pairwise combination of primaries, [C,M,Y] (Cyan, Magenta, Yellow)

RG only, B zero

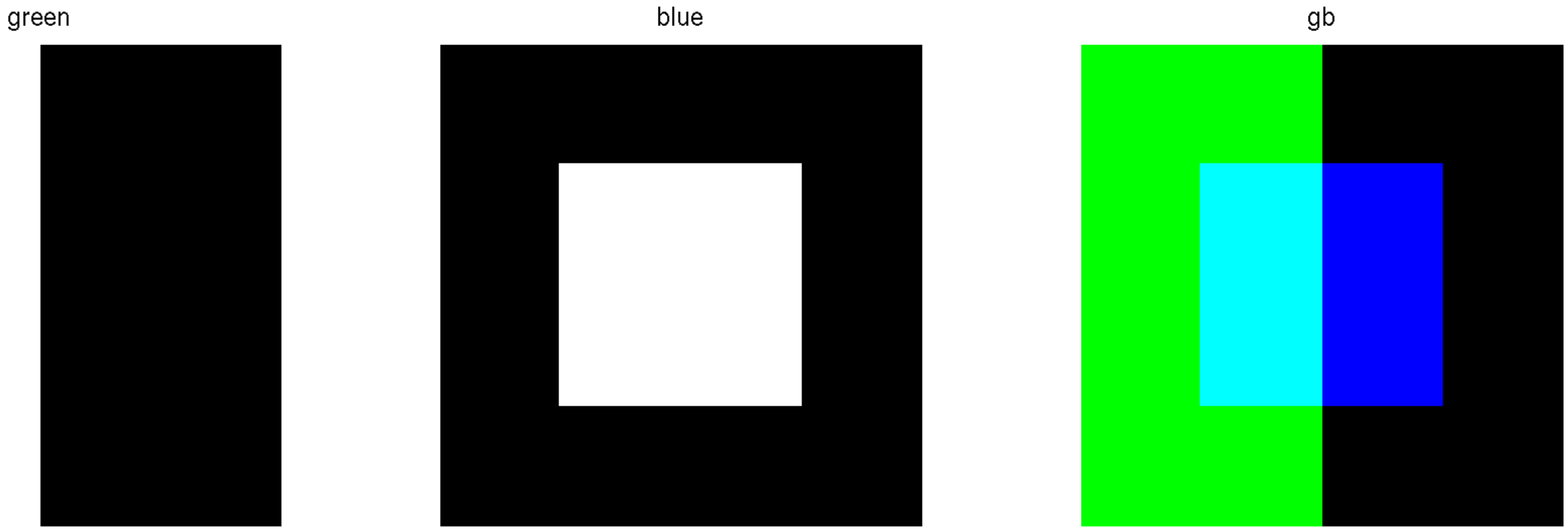


RB only, G zero

red blue rb

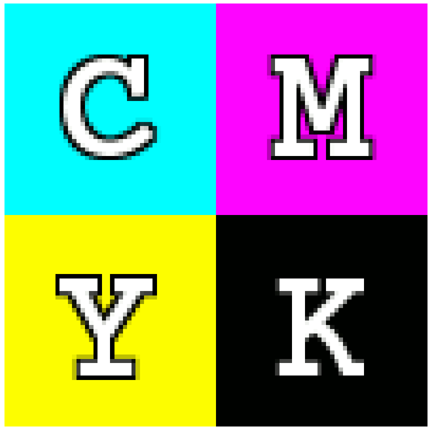


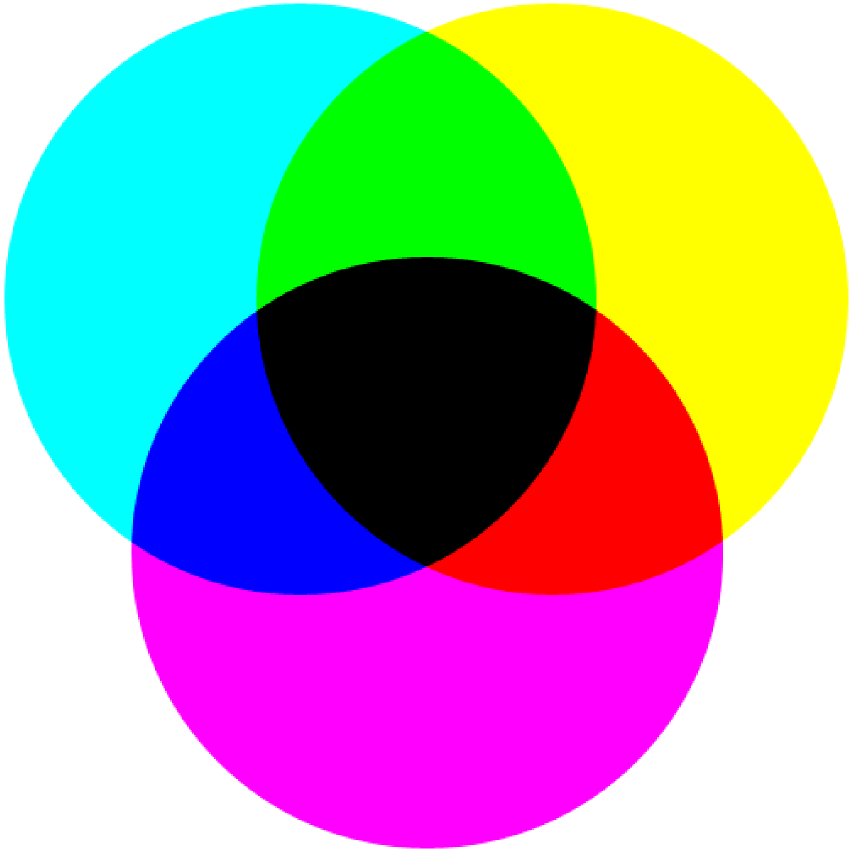
GB only, R zero



Subtractive mixing

- it works through light absorption
- the colors that are seen are from the part of light that is not absorbed
- paintings, printing, . . .
- Primary colors: ones used to define other colors [C,M,Y]
- Secondary colors: pairwise combination of primaries [R,G,B]





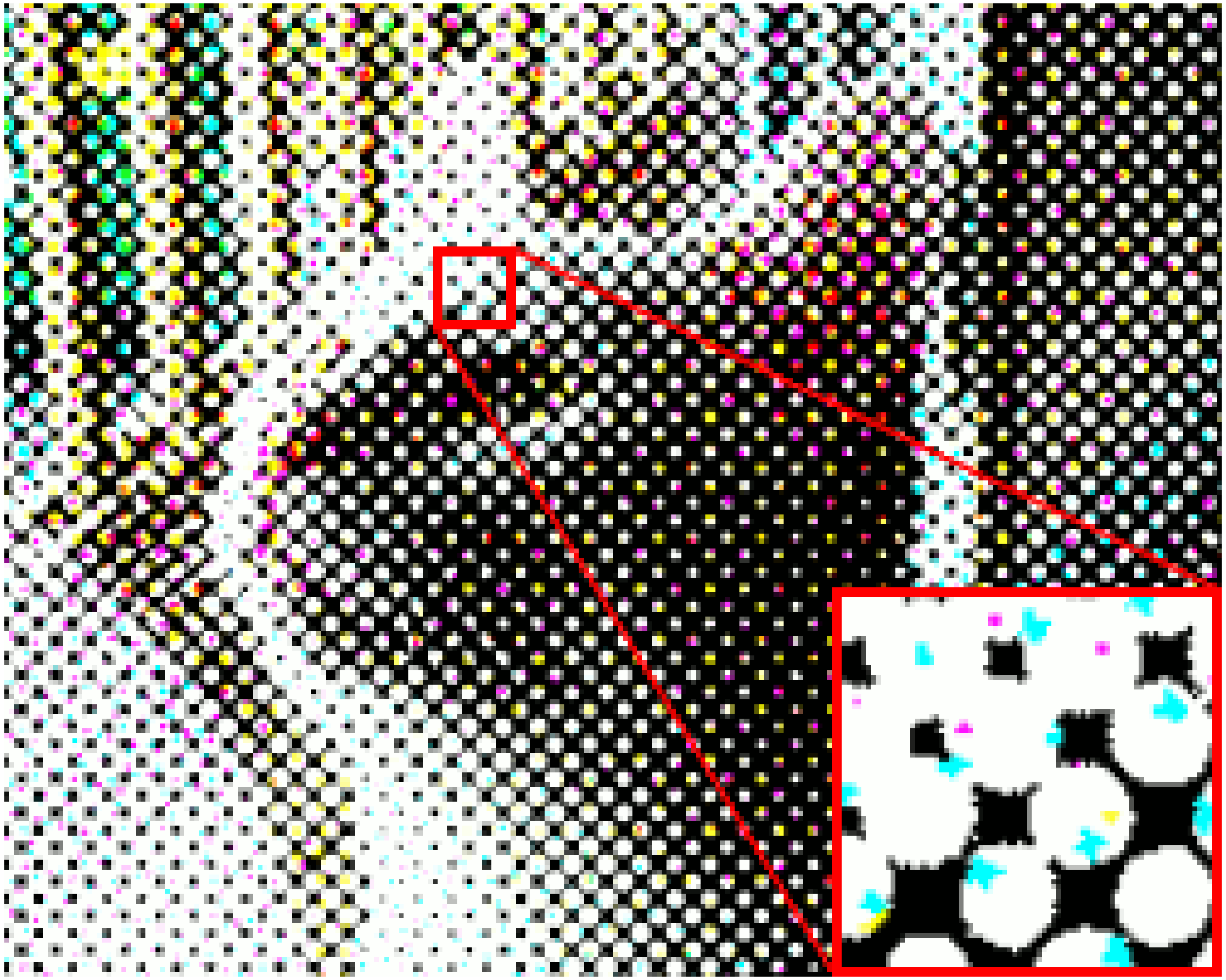
CMYK model

- color primaries [C.M,Y] should result black when all mixed together
- in practice, such black is not dense enough
- K = key (black) is added to the model

CMYK model

- color primaries [C.M,Y] should result black when all mixed together
- in practice, such black is not dense enough
- K = key (black) is added to the model

CMYK printing



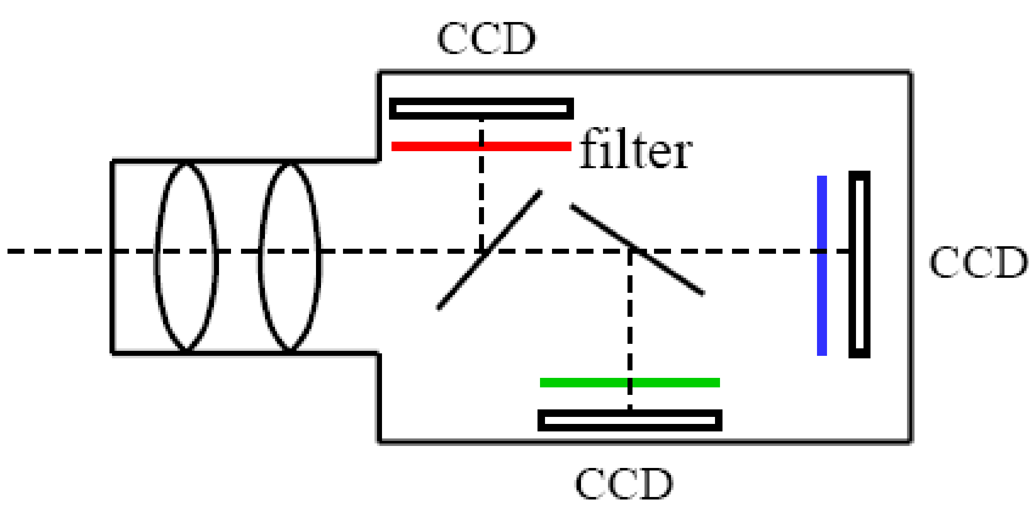
Capturing RGB values

- We know how to display, print color . . .
- How to capture?
- CCD generates output proportionally to amount of energy

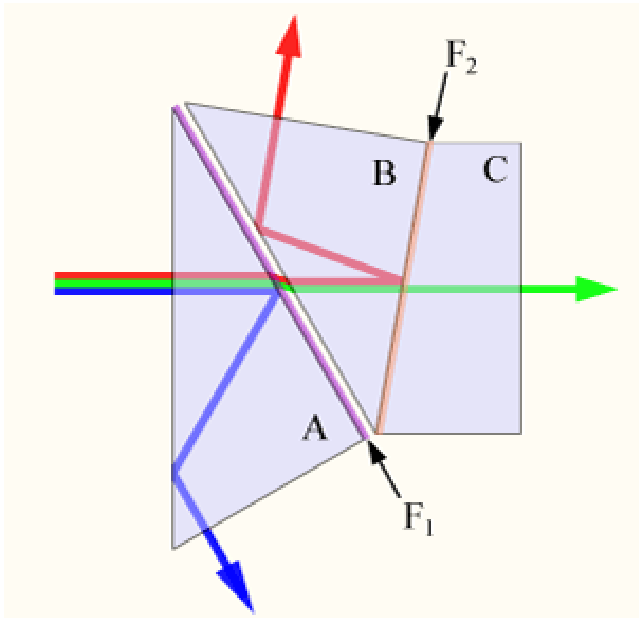
Capturing RGB values

- We know how to display, print color . . .
- How to capture?
- CCD generates output proportionally to amount of energy

3CCD camera with separating dichroic beam splitter



3CCD chip camera

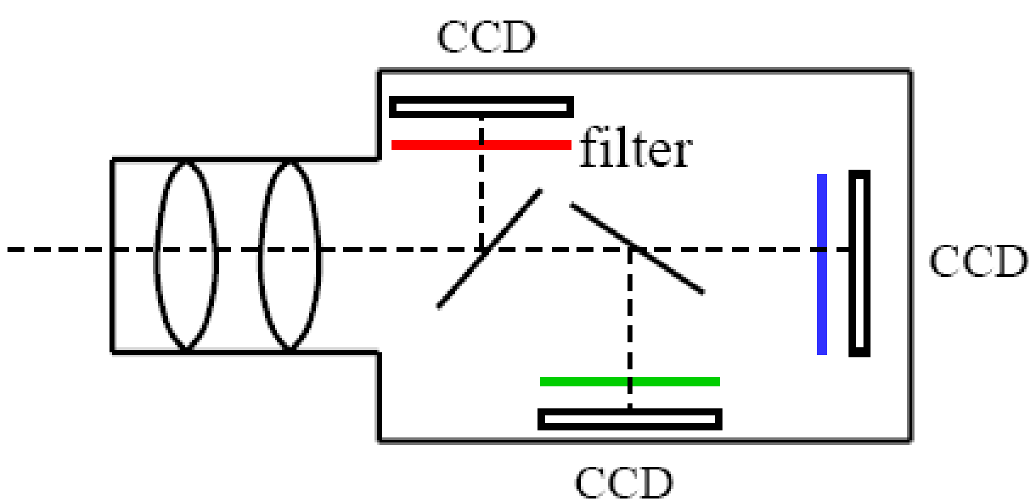


dichroic prism

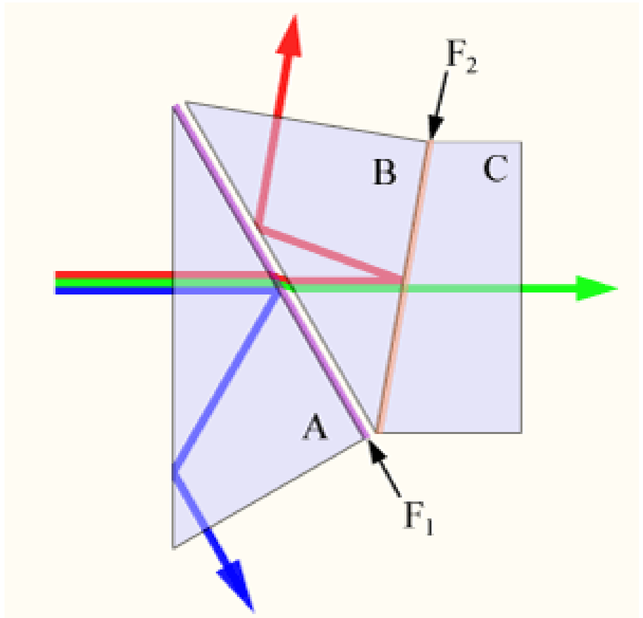
Capturing RGB values

- We know how to display, print color . . .
- How to capture?
- CCD generates output proportionally to amount of energy

3CCD camera with separating dichroic beam splitter



3CCD chip camera

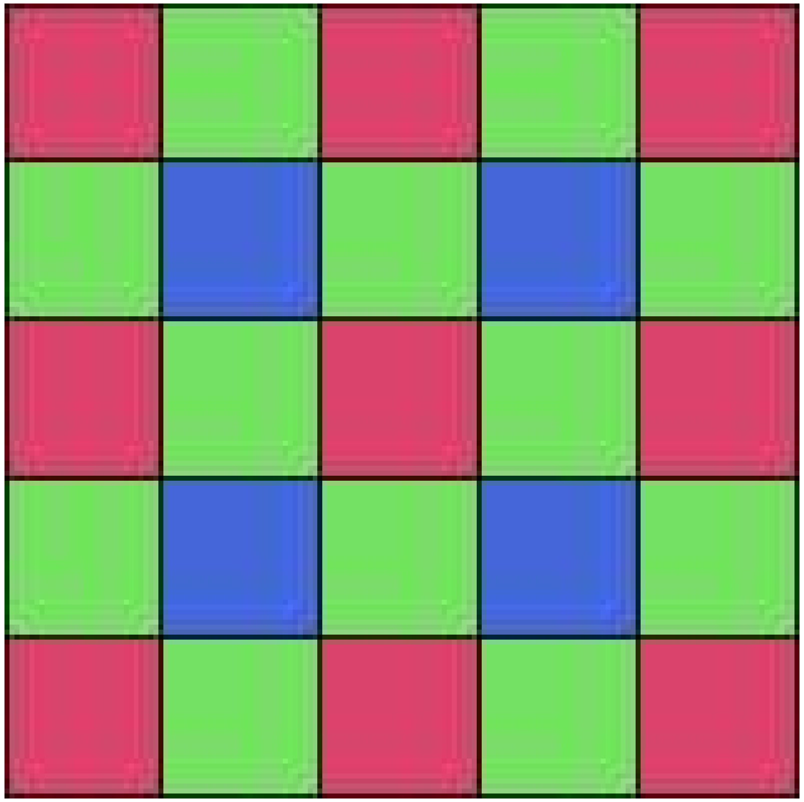


dichroic prism

Good: Color quality, Problem: price . . .

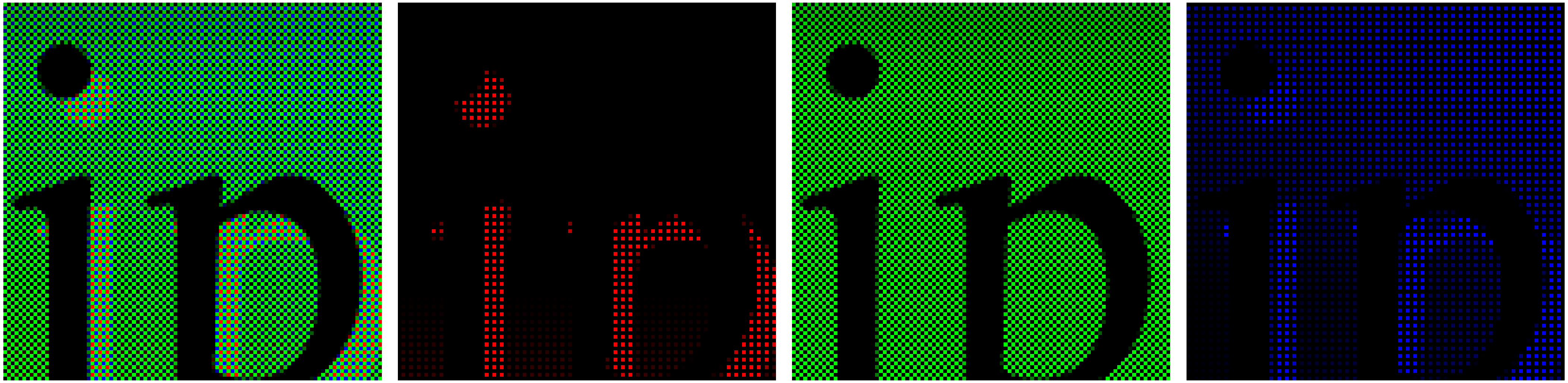
1CCD camera with Bayer filter

- use one chip
- place a selective filter in front of it
- 2:1:1, 2 to green, human eye is most sensitive to it
- combine values to make RGB image Demosaicking
- cheap but the image quality suffers
- this is, among other things, what makes difference between digital photo cameras



Demosaicking in images

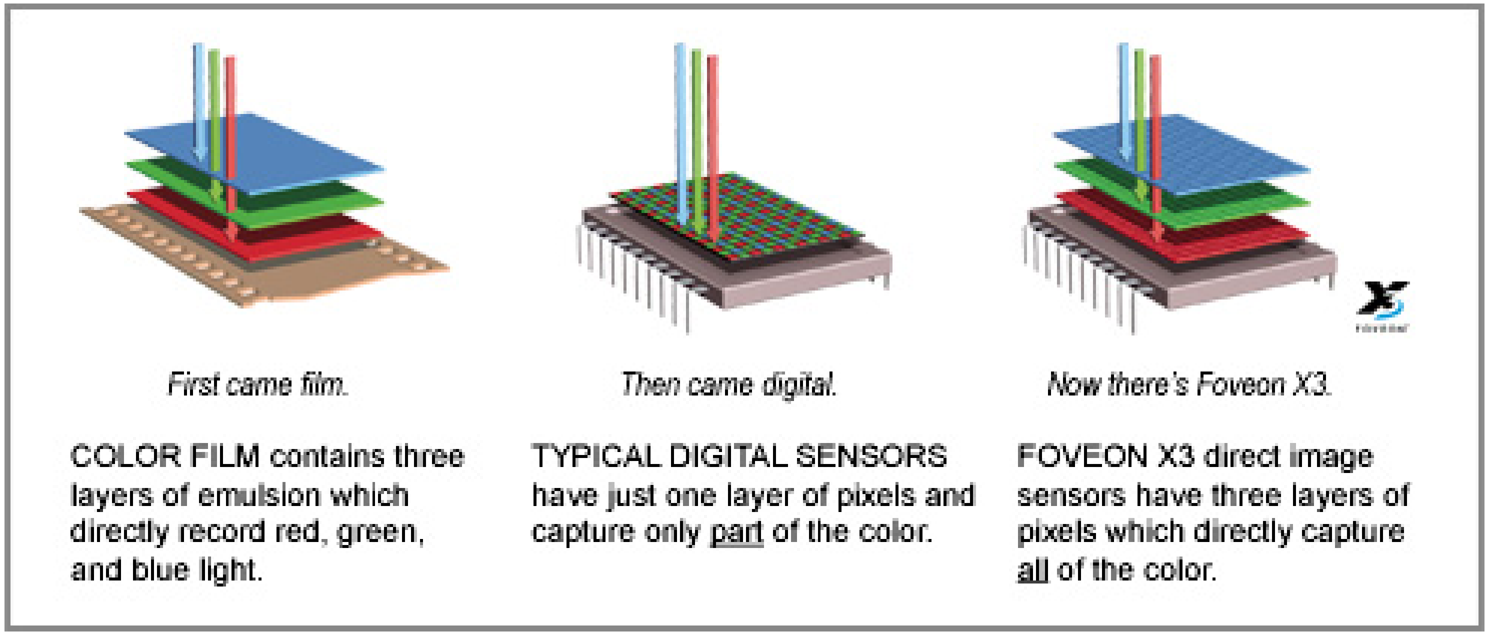
Demosaicking in images

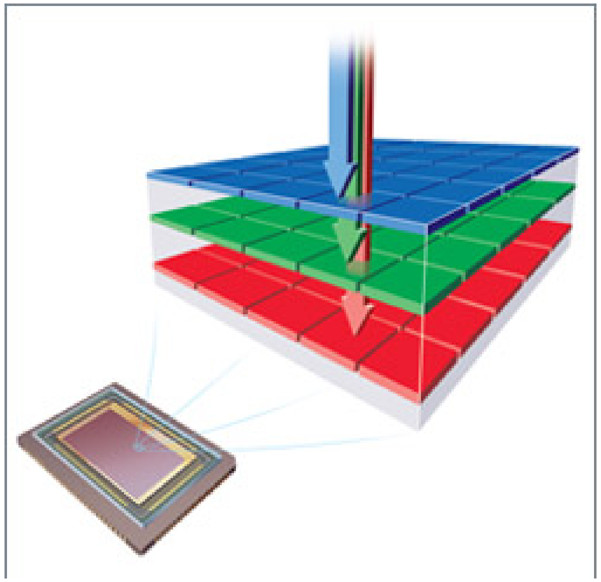


Demosaicking result

Many demosaicking method exist. 1CCD with a filter is still prevailing solution. Few expensive DV cameras in consumer level. A company Foveon found yet another way . . .

Color from “depths”





Capturing color — revisited

- Many demosaicking method exist.
- 1CCD with a filter is still prevailing solution.
- Few expensive DV cameras in consumer level.
- A company called Foveon found yet another way . . .

HSV color space

- Problem in RGB space: How would you create a color according to your design?
- RGB values do not correspond to human thinking about colors
- We are saying: pure red, deep purple, sky blue . . .

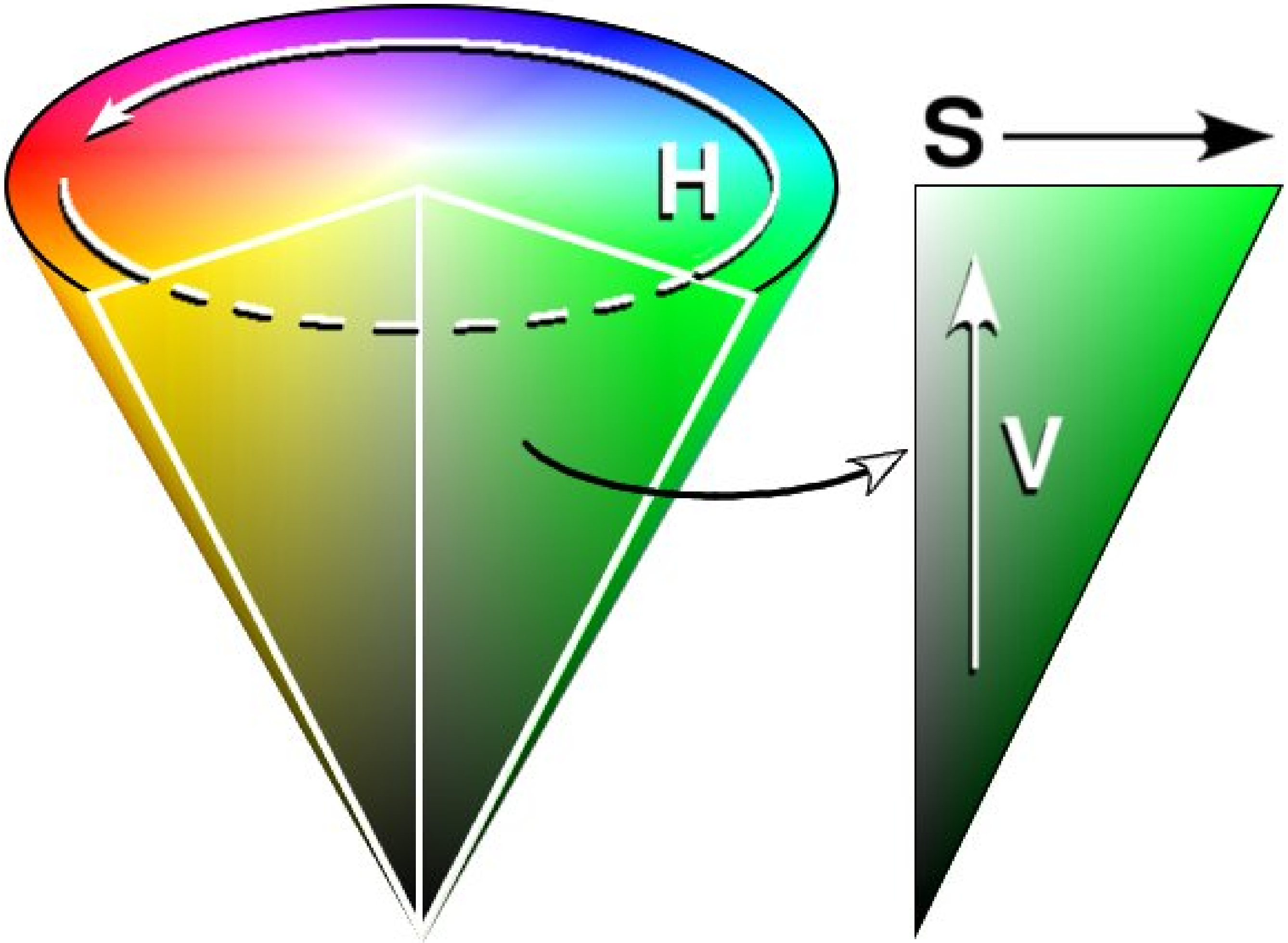
HSV color space

- Problem in RGB space: How would you create a color according to your design?
- RGB values do not correspond to human thinking about colors
- We are saying: pure red, deep purple, sky blue . . .

HSV Hue, Saturation, Value color space

- Hue is the color type (red, yellow, . . .)
- Saturation refers to color purity or vibrancy
- Value is the brightness of the color

HSV cone



Playing with saturation

original image

what a nice autumn!

Playing with saturation

original image

what a sad gray autumn!

Additive mixing — revisited

- Can we, assuming properly chosen [R,G,B], mix any color?

Additive mixing — revisited

- Can we, assuming properly chosen [R,G,B], mix any color?
- Well, almost any.

What is wrong?

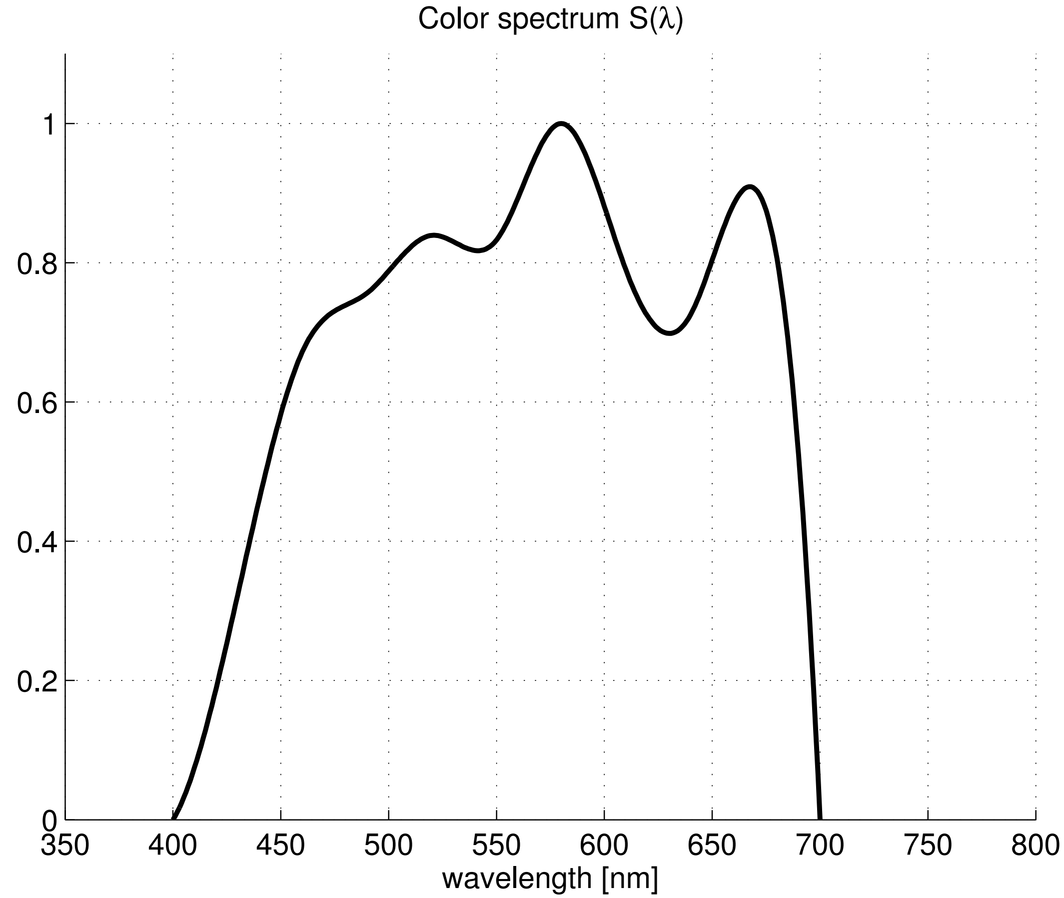
Additive mixing — revisited

- Can we, assuming properly chosen [R,G,B], mix any color?
- Well, almost any.

What is wrong?

- Blue and Green makes Cyan.
- But how to make monochromatic Cyan?
- blue + green - (little red) = monochromatic cyan
- but how to make negative values on screens?

What do you need to match any color?

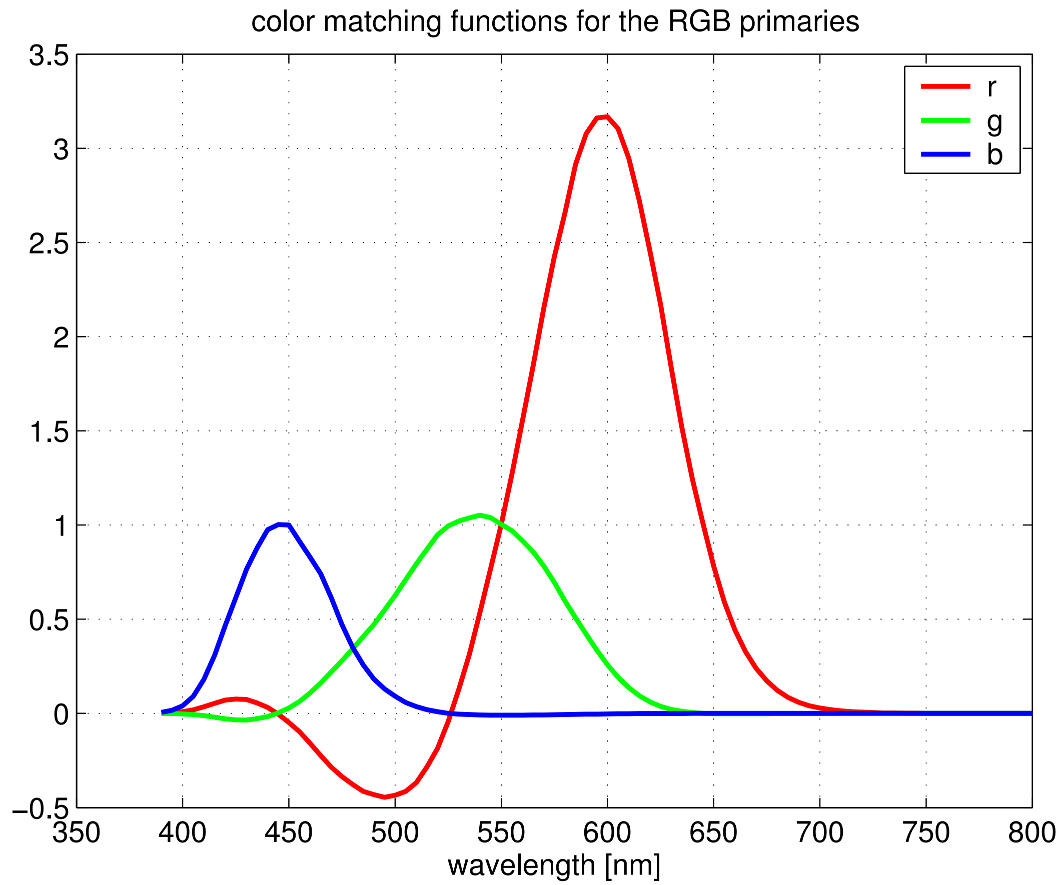


[1]

- Table of $S(\lambda)$ in predefined λ
- $S(\lambda) = P_1 \int f_1(\lambda) S(\lambda) d\lambda + P_2 \int f_2(\lambda) S(\lambda) d\lambda + P_3 \int f_3(\lambda) S(\lambda) d\lambda$ which gives us $[P_1, P_2, P_3]$ representation.

[1] Data tables can downloaded from http://www.cvrl.org

What RGB do you need to match any color?



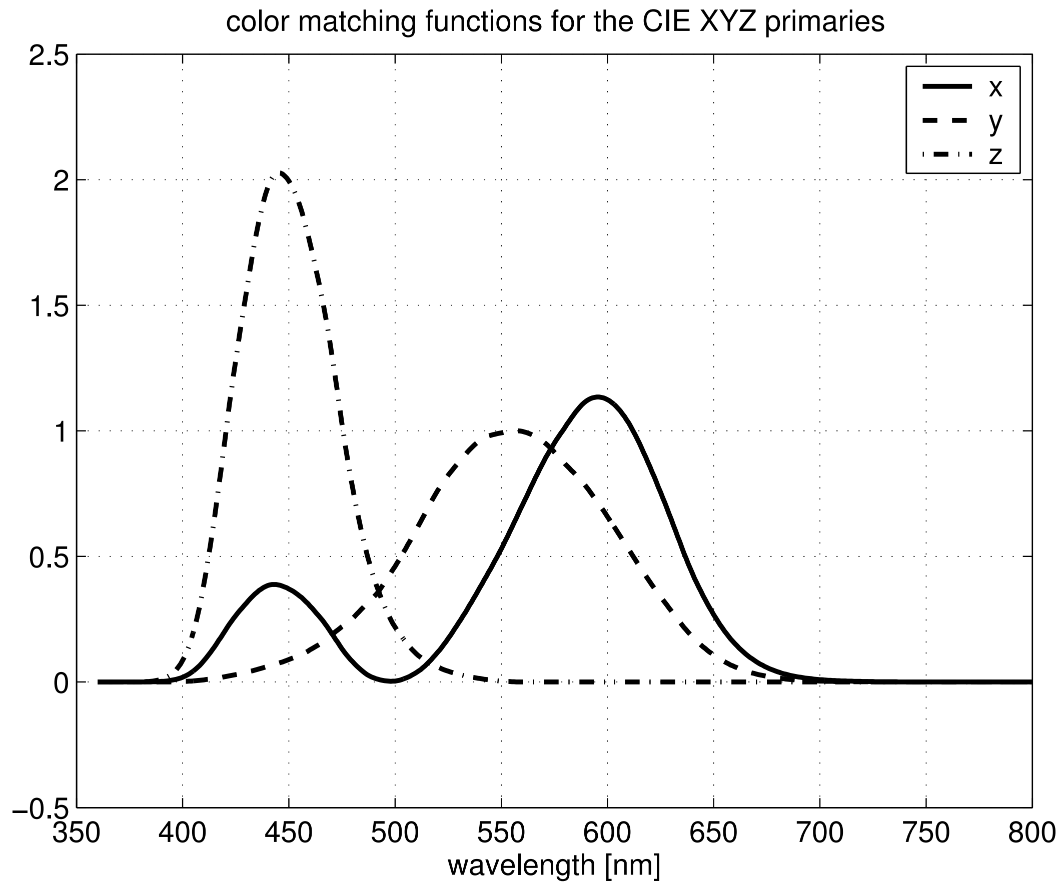
[2]

$$S(\lambda) = R\int r(\lambda)S(\lambda)d\lambda + G\int g(\lambda)S(\lambda)d\lambda + B\int b(\lambda)S(\lambda)d\lambda$$

Problem: How to realize devices with negative matching functions?

[2] Data tables can downloaded from http://www.cvrl.org

A way out — new primary "colors" CIE XYZ

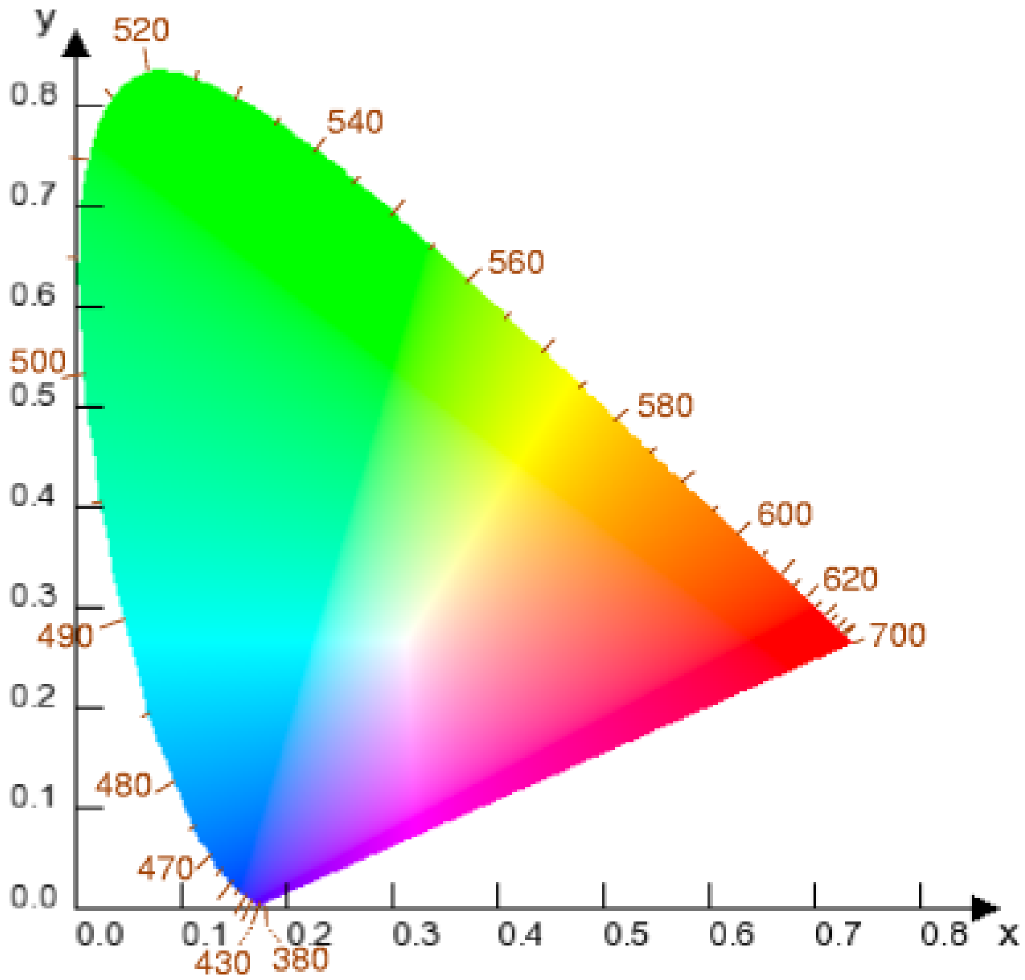


[3]

$$S(\lambda) = X\int x(\lambda)S(\lambda)d\lambda + Y\int y(\lambda)S(\lambda)d\lambda + Z\int z(\lambda)S(\lambda)d\lambda$$

[3] Data tables can downloaded from http://www.cvrl.org

CIE chromaticity diagram

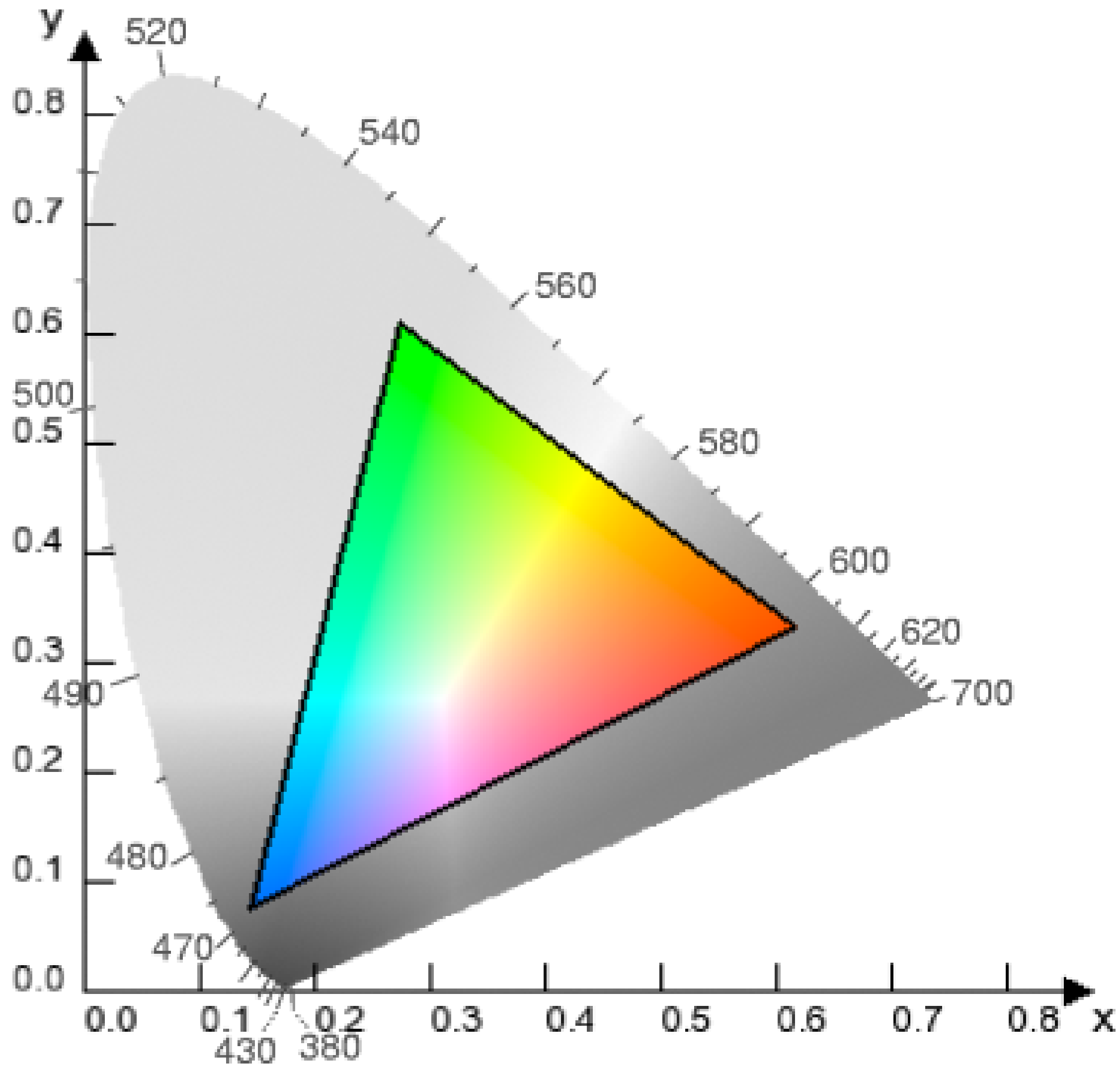


$$[x, y] = \left[\frac{X}{X+Y+Z}, \frac{Y}{X+Y+Z} \right]$$

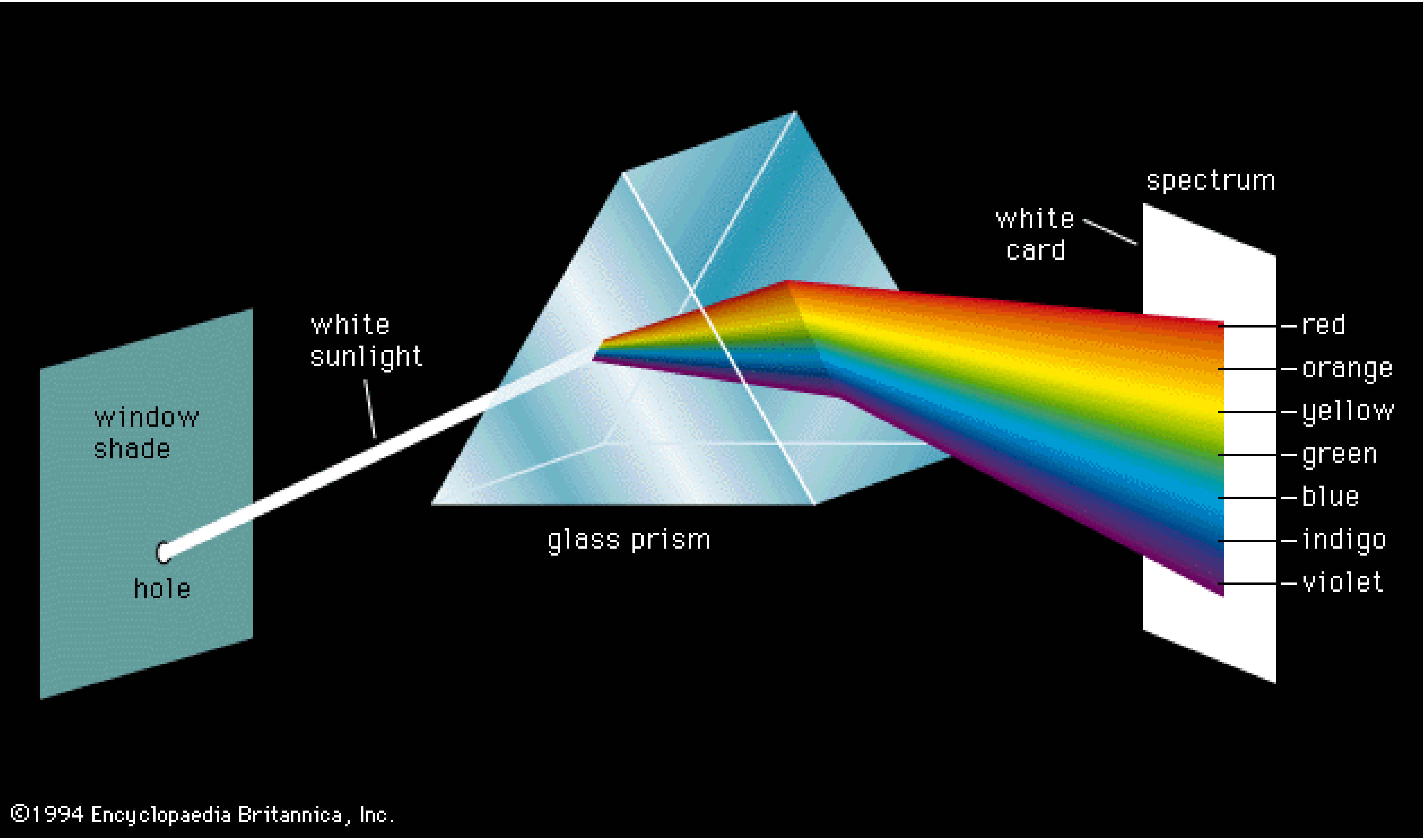
Do we see all colors on the screen?

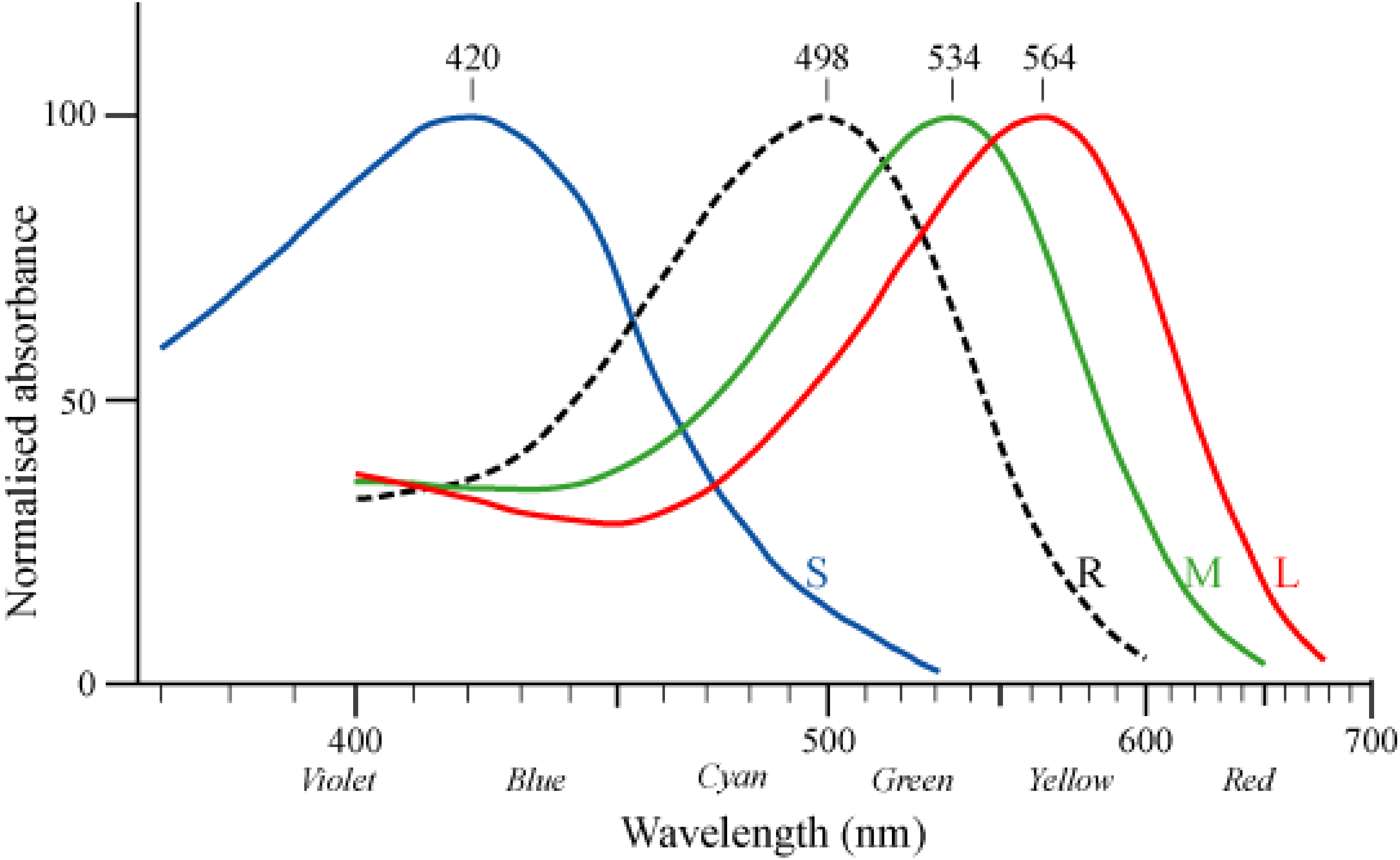
Do we see all colors on the screen?

No!



Typical gamut of a CRT monitor

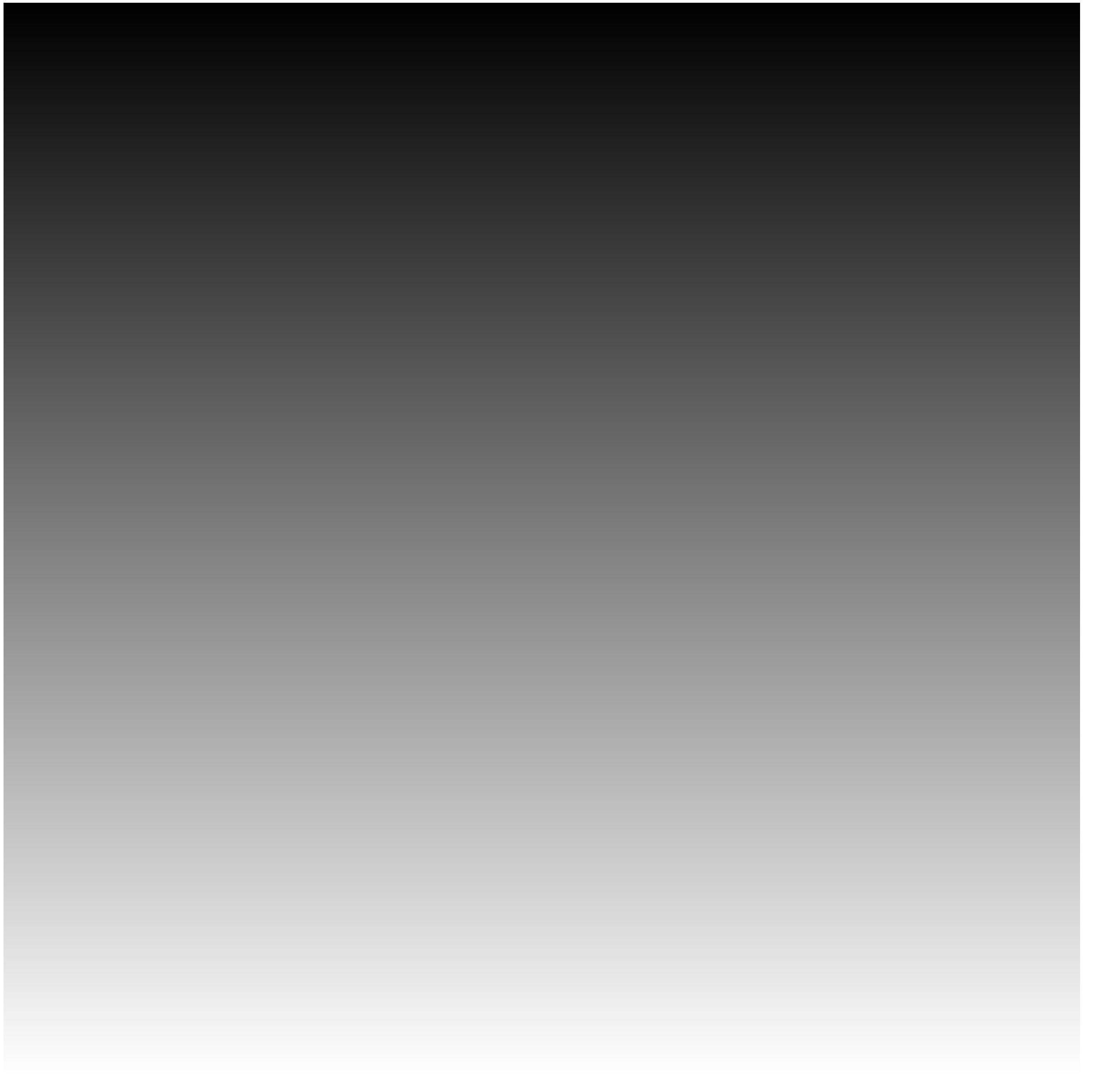

spectrum
white card
red
orange
yellow
green
blue
indigo
violet
white sunlight
window shade
hole
glass prism
©1994 Encyclopaedia Britannica, Inc.



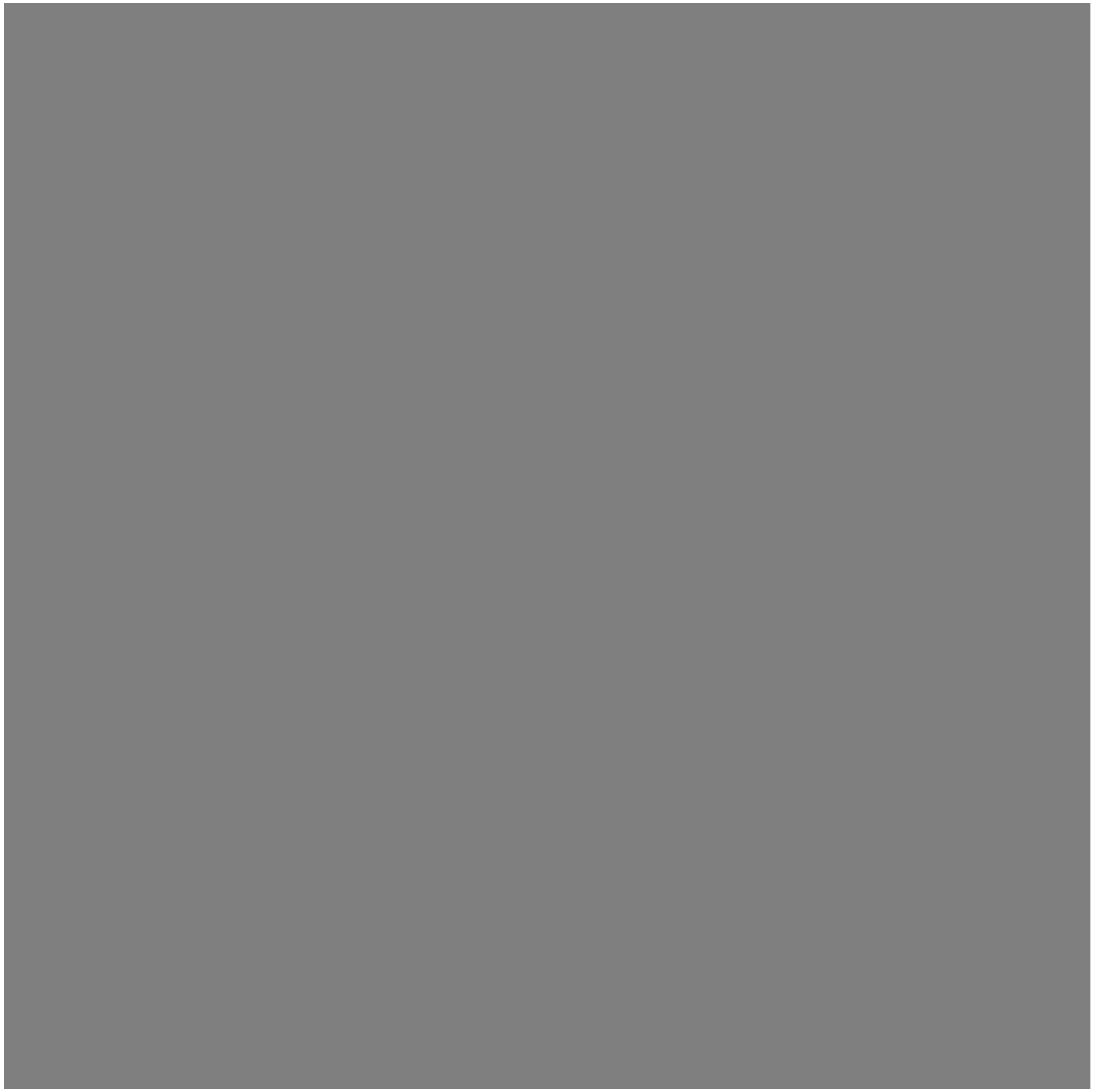
420
498
534
564
100
50
0
Normalised absorbance
S
R
M
L
400
500
600
700
Violet
Blue
Cyan
Green
Yellow
Red
Wavelength (nm)

red

green



blue

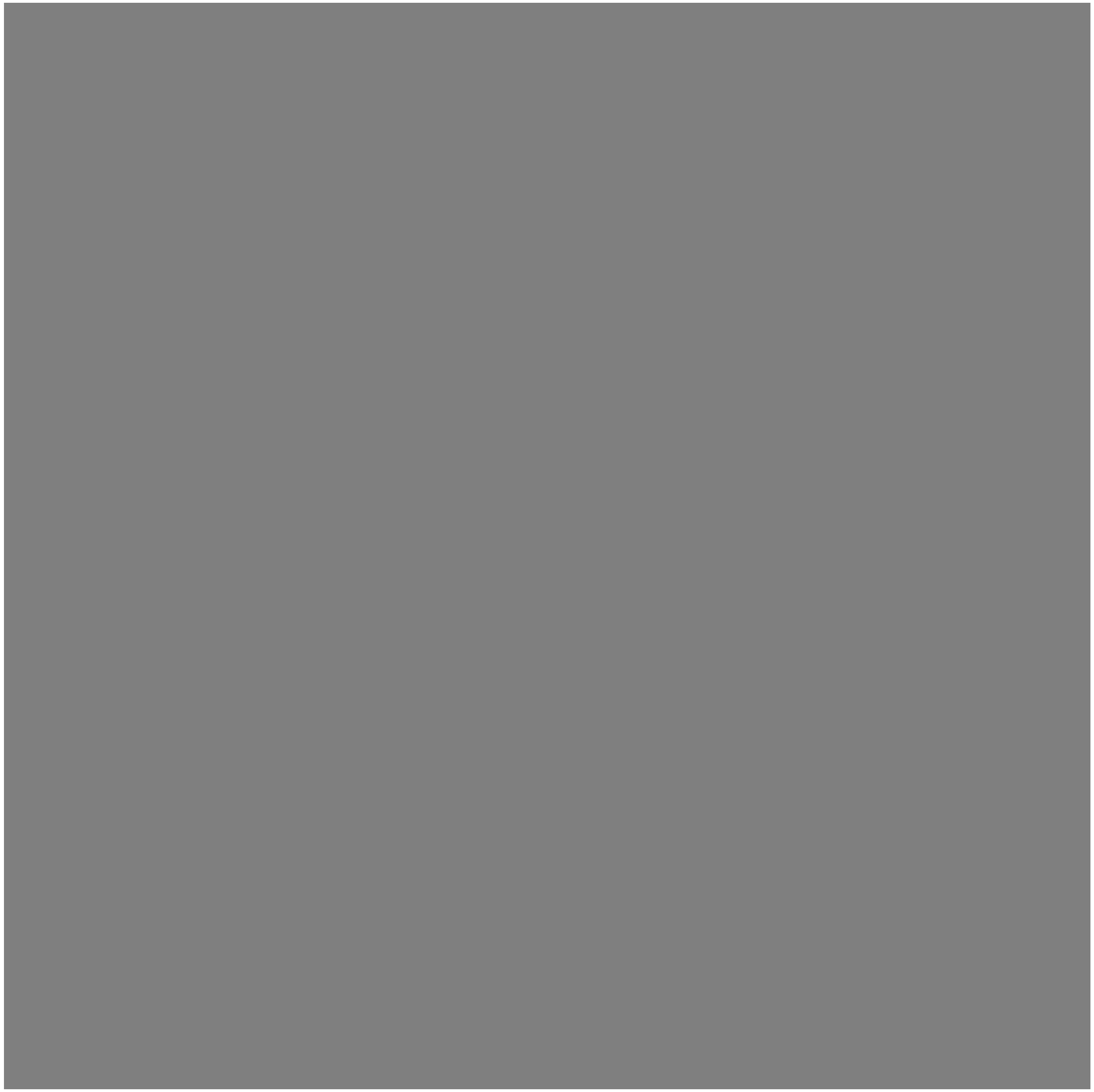


rgb

red

green

blue

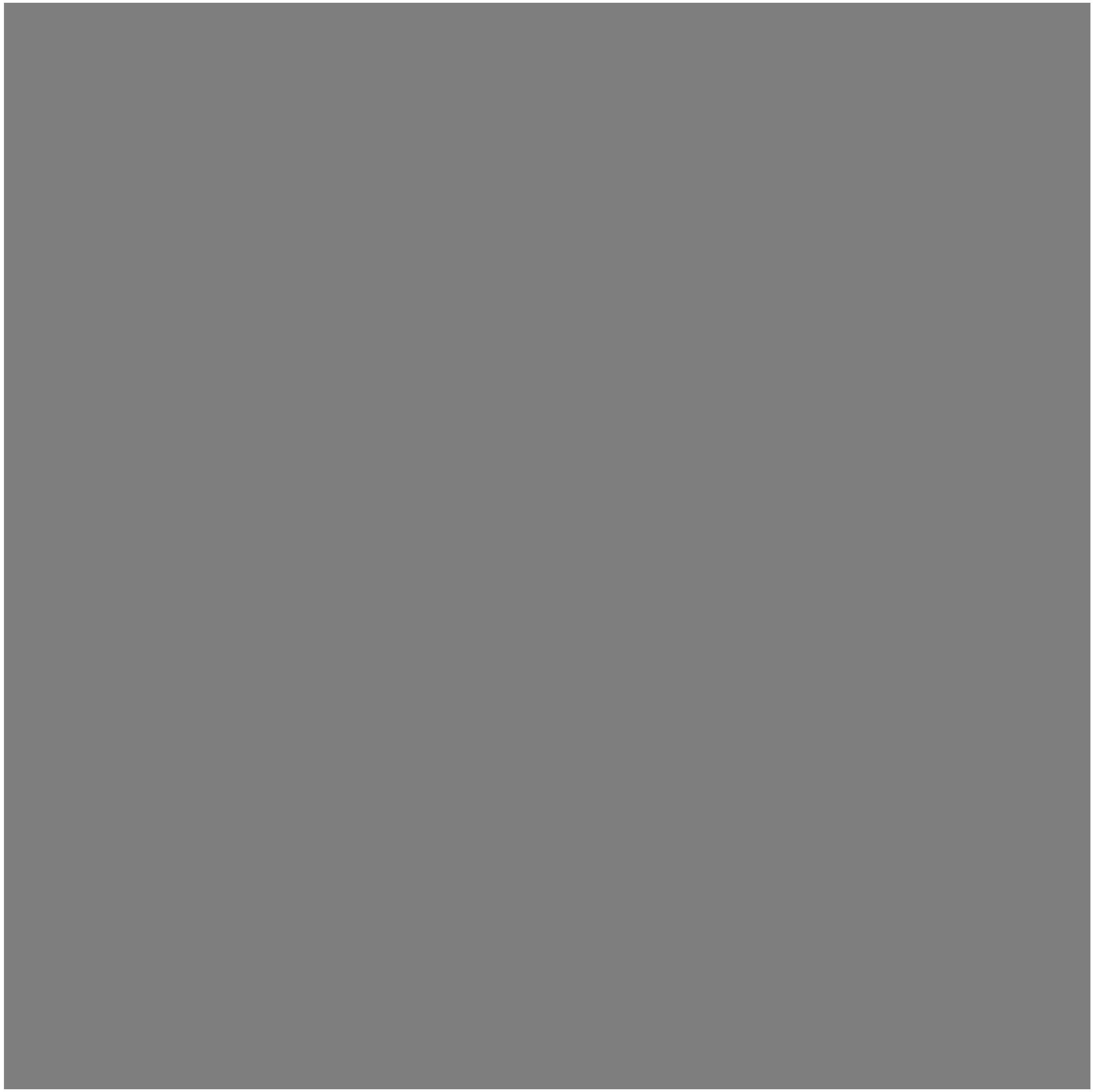


rgb

red

green

blue

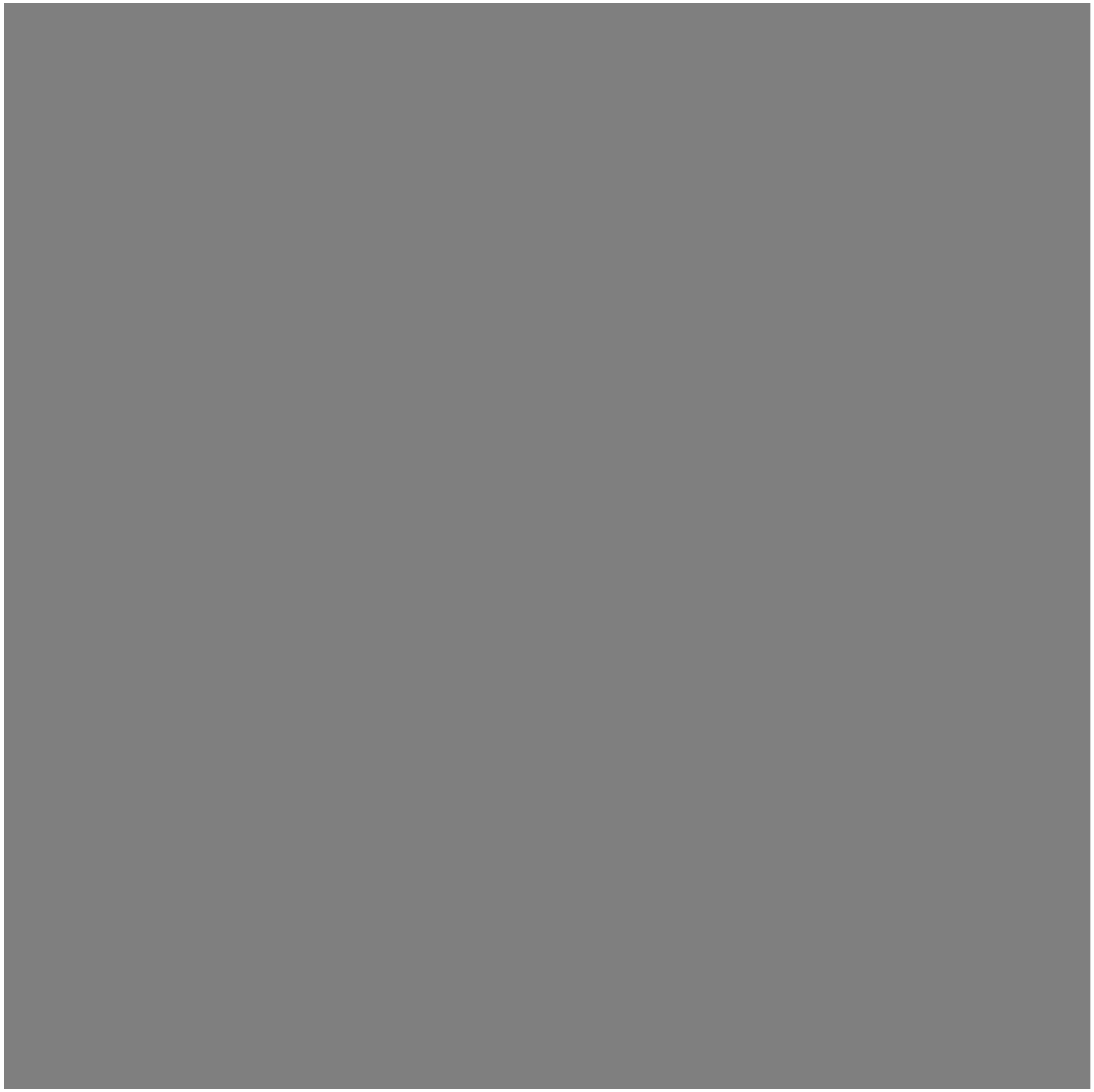


rgb

red

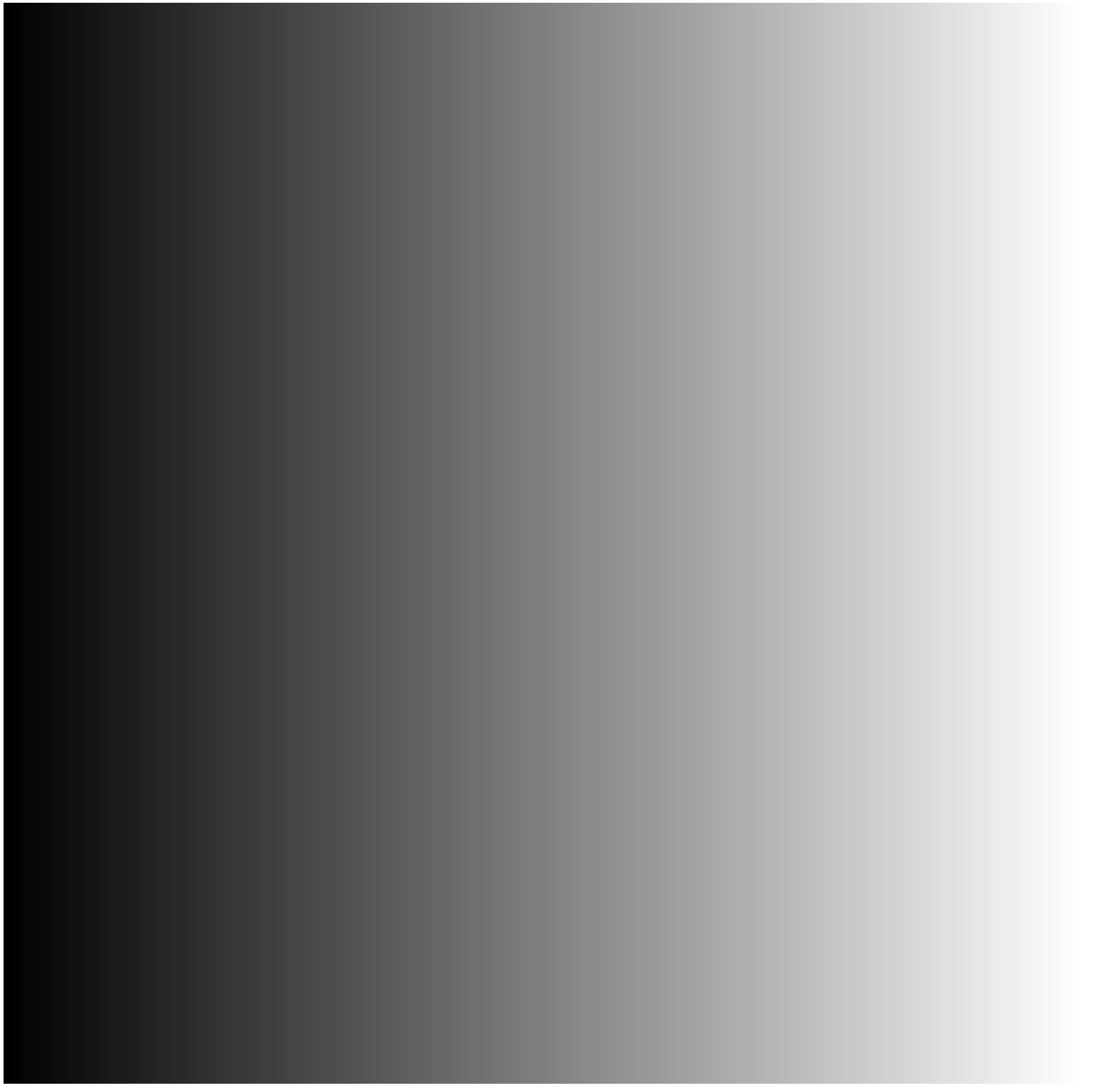
green

blue



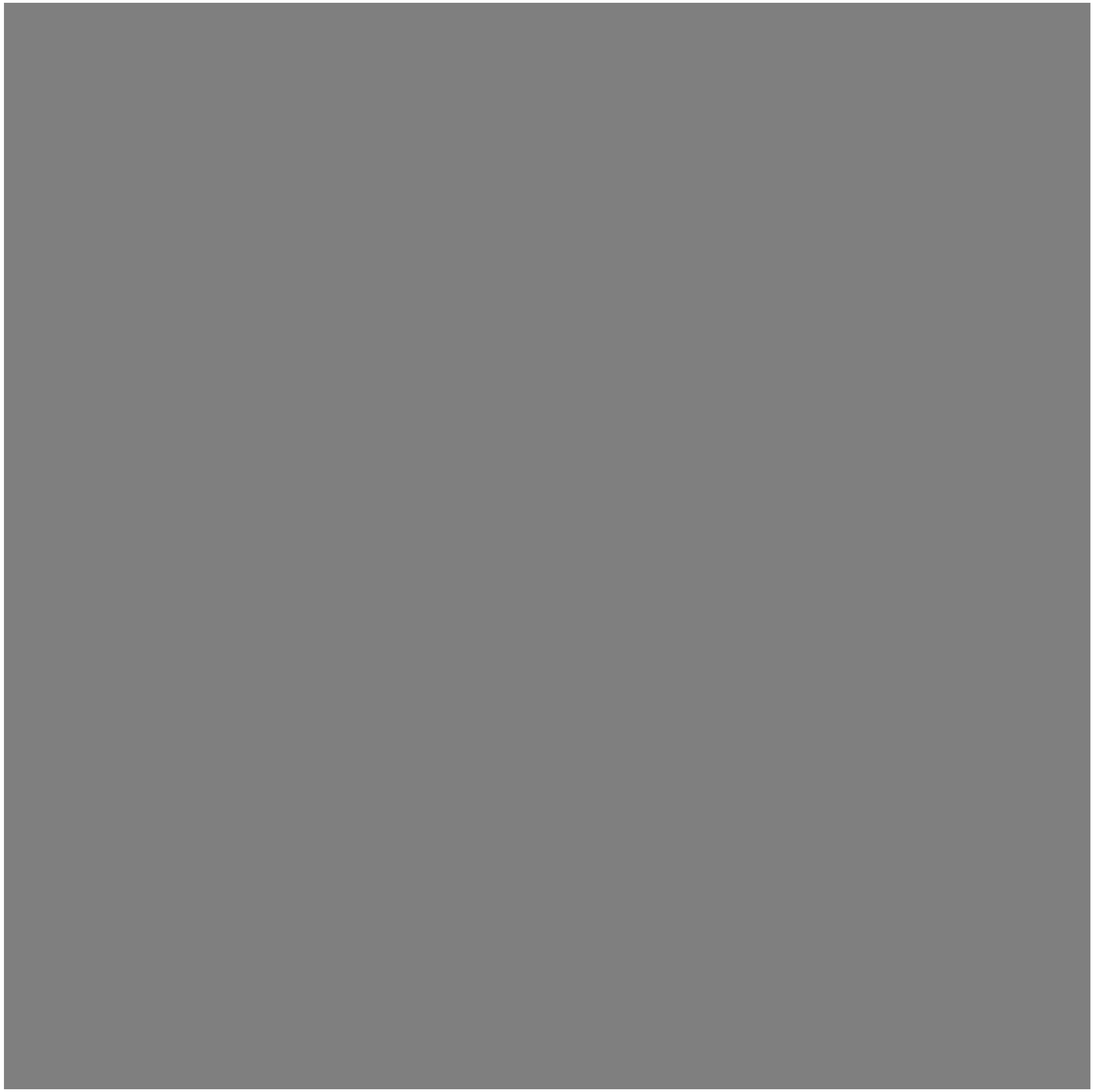
rgb

red



green

blue

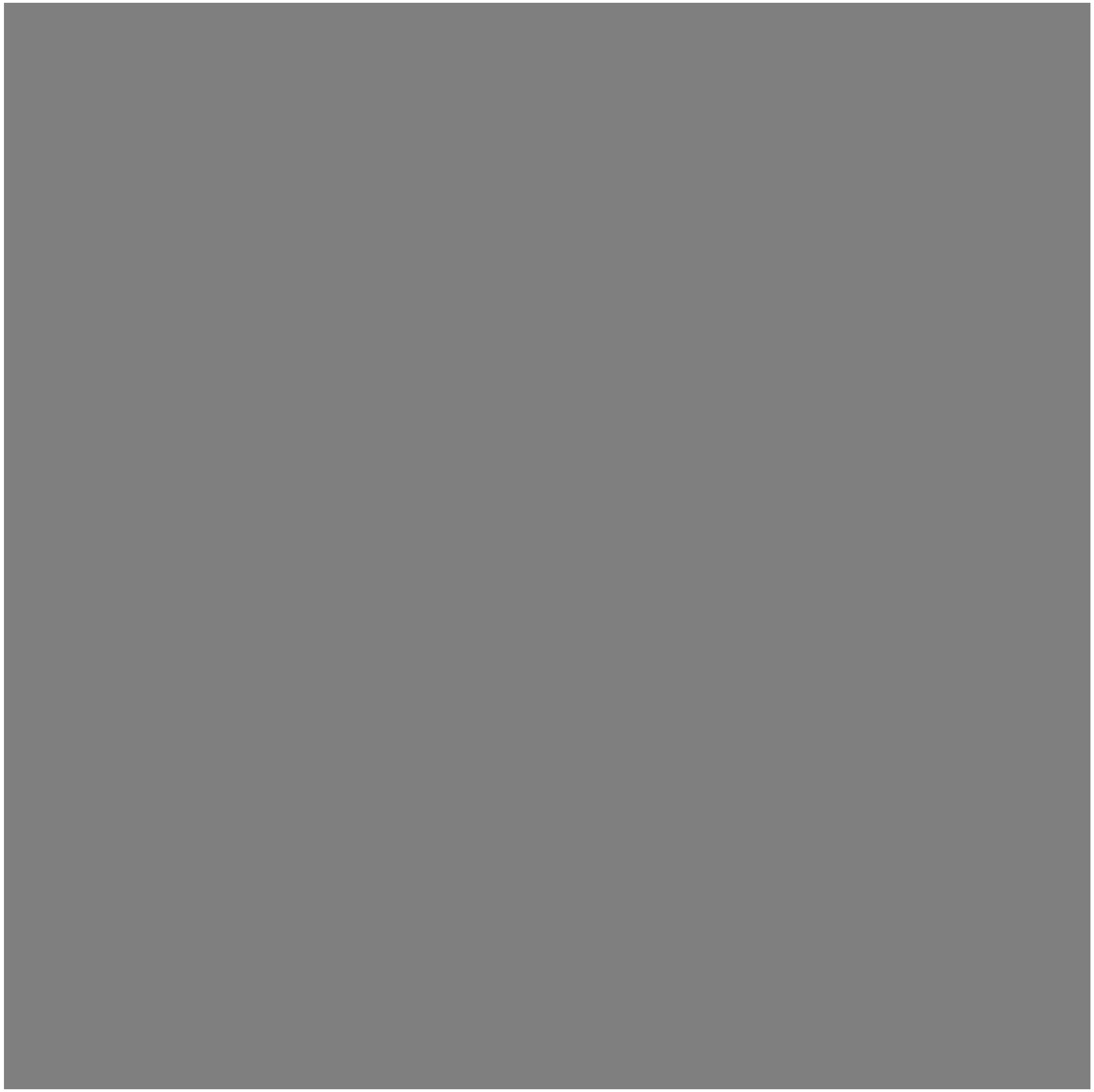


rgb

red

green

blue

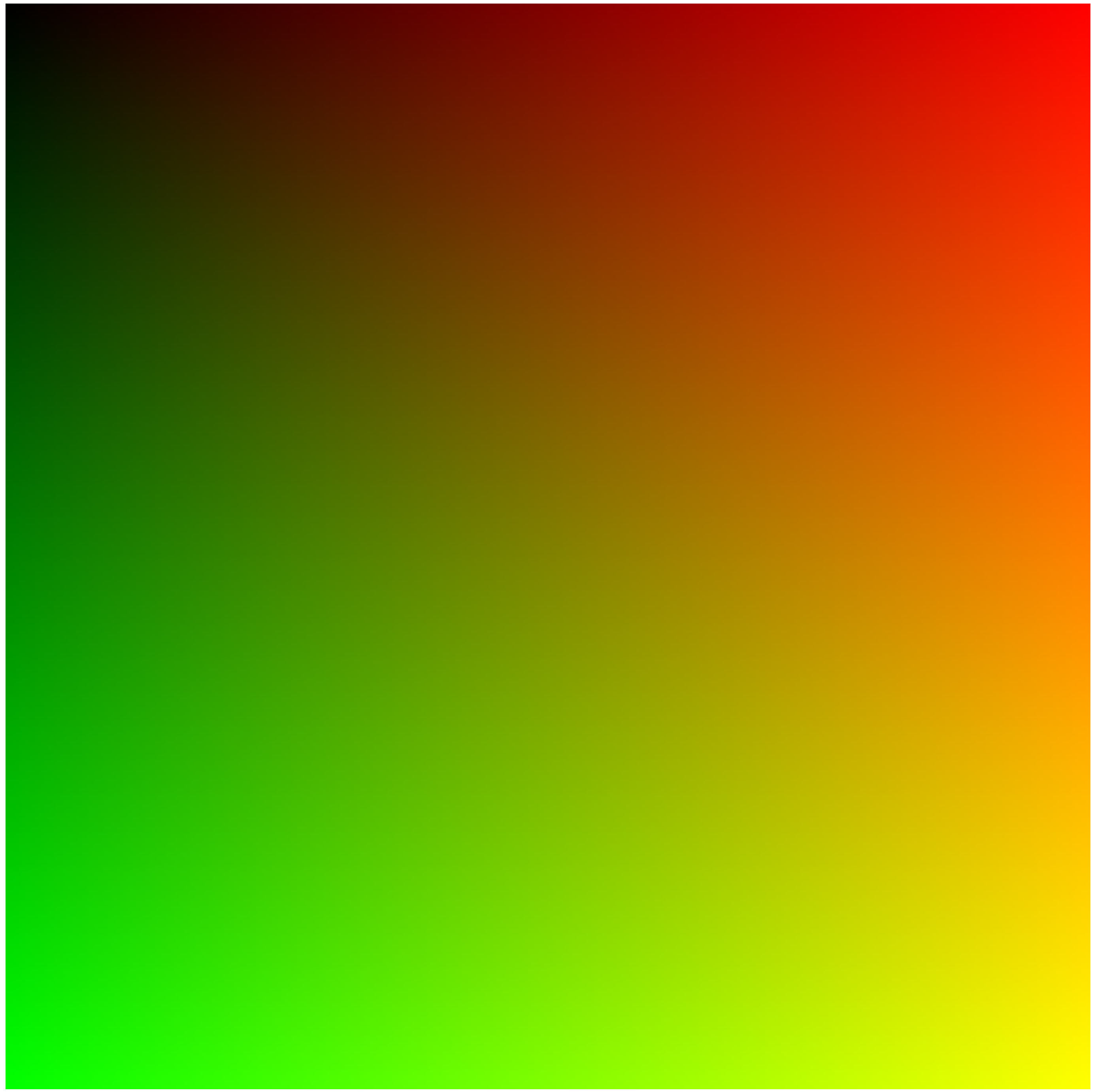


rgb

red

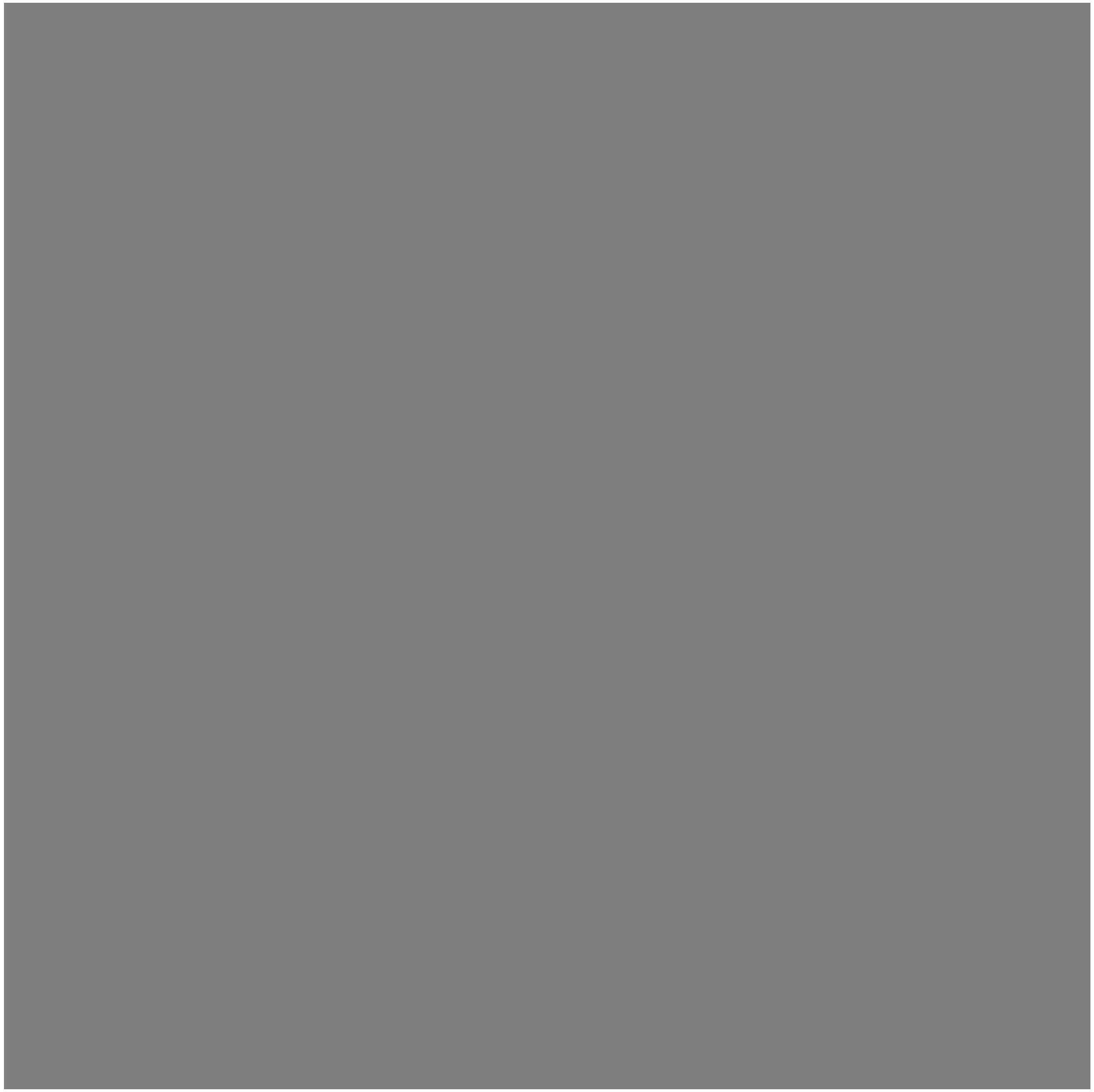
green

rg



red

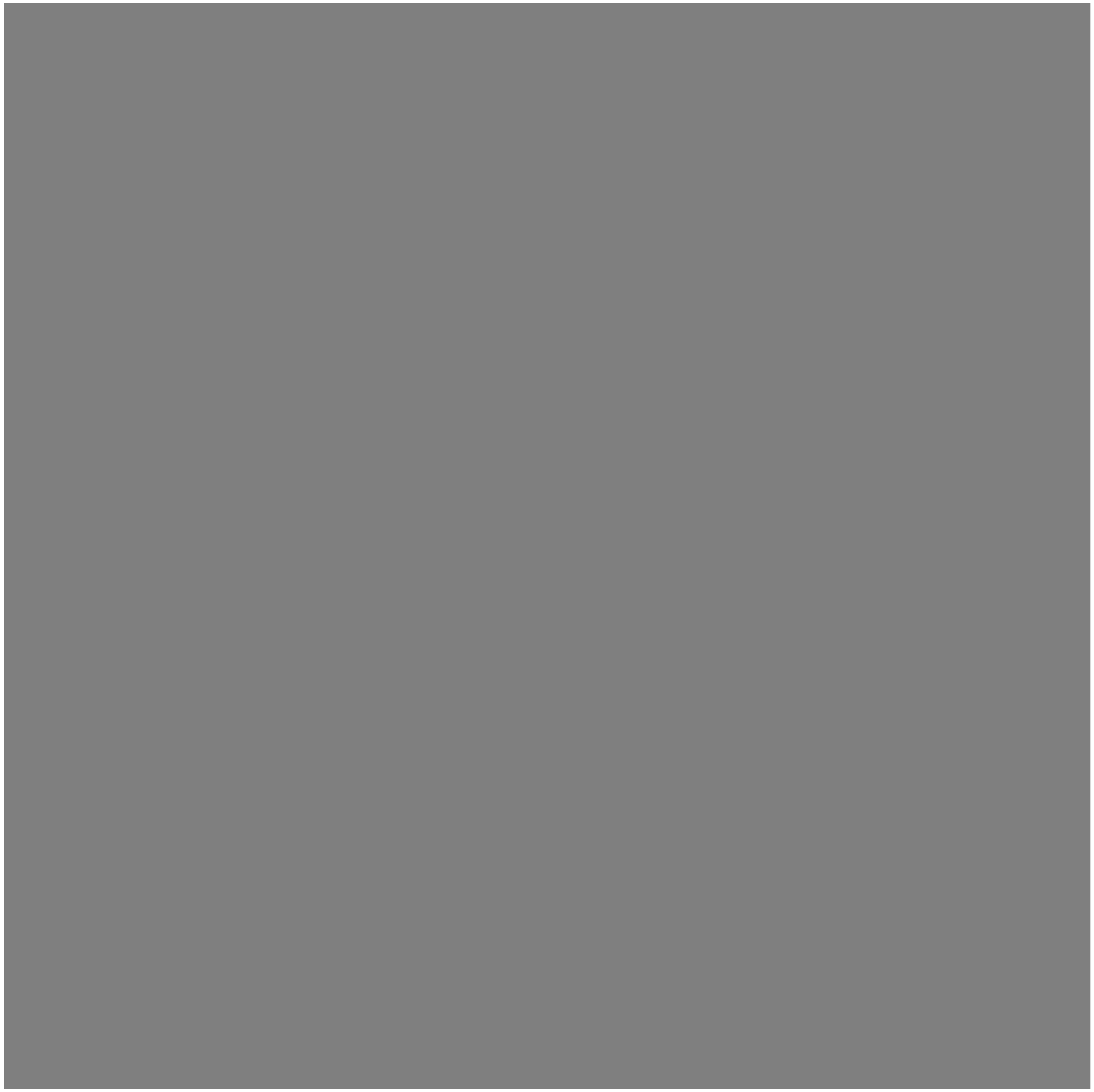
blue



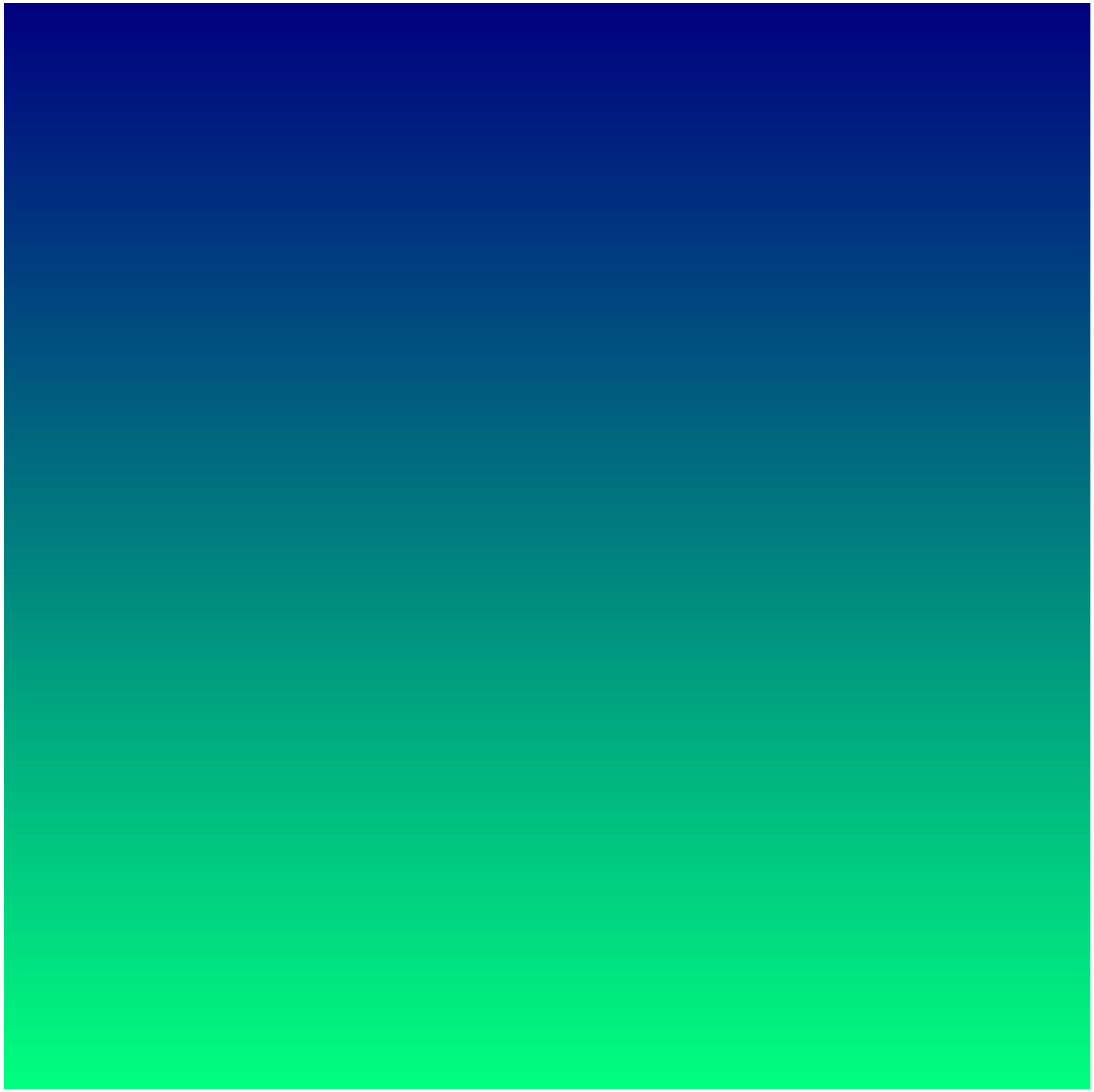
rb

green

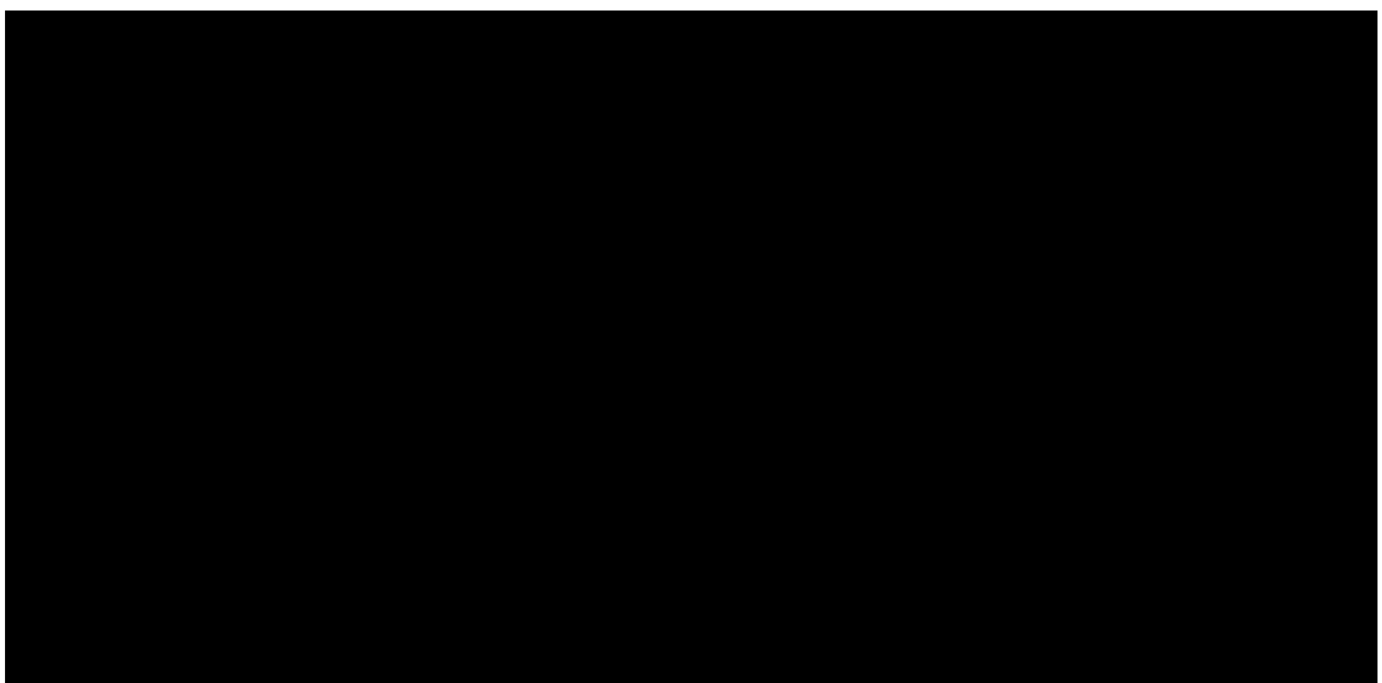
blue



gb

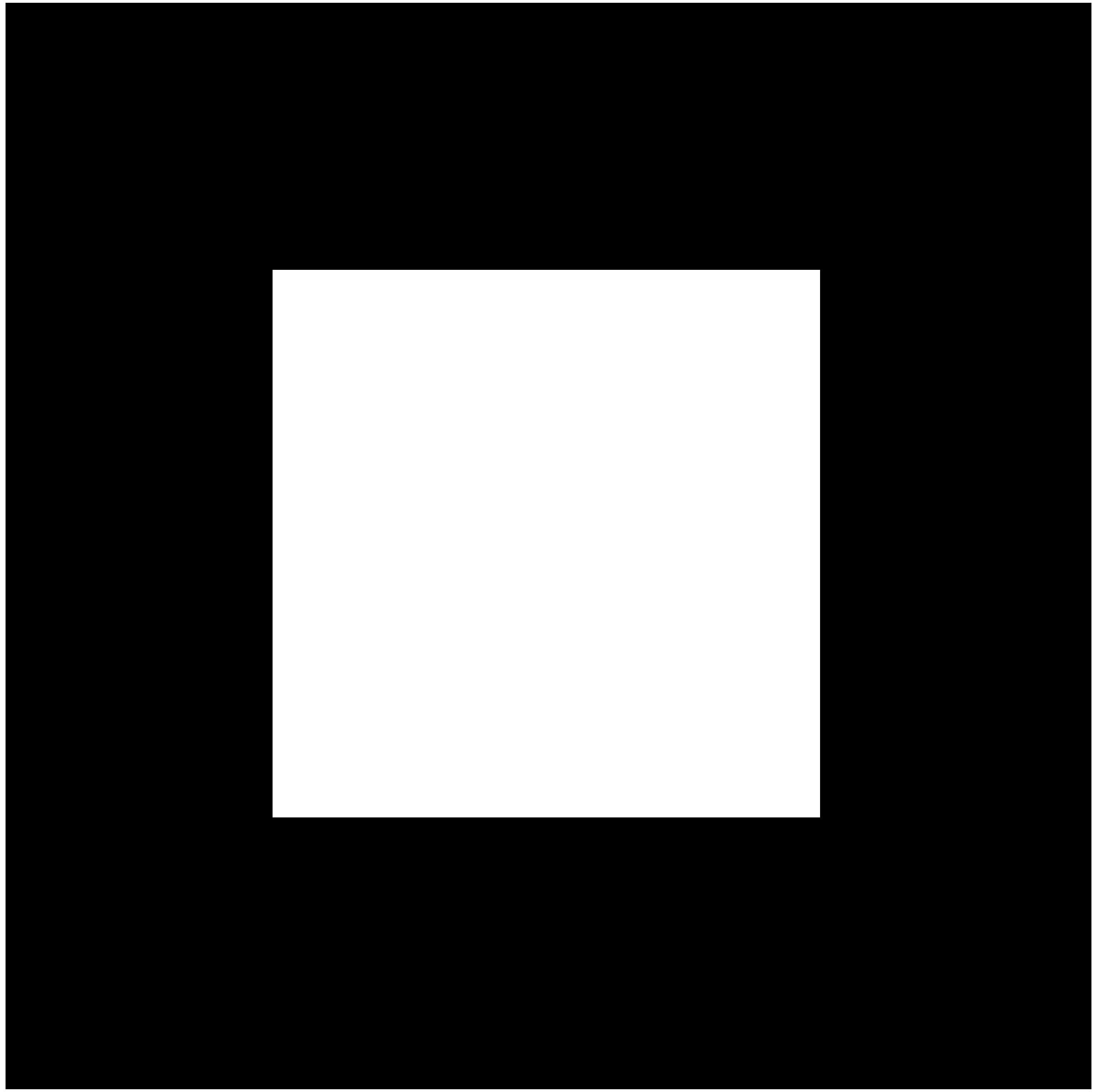


red

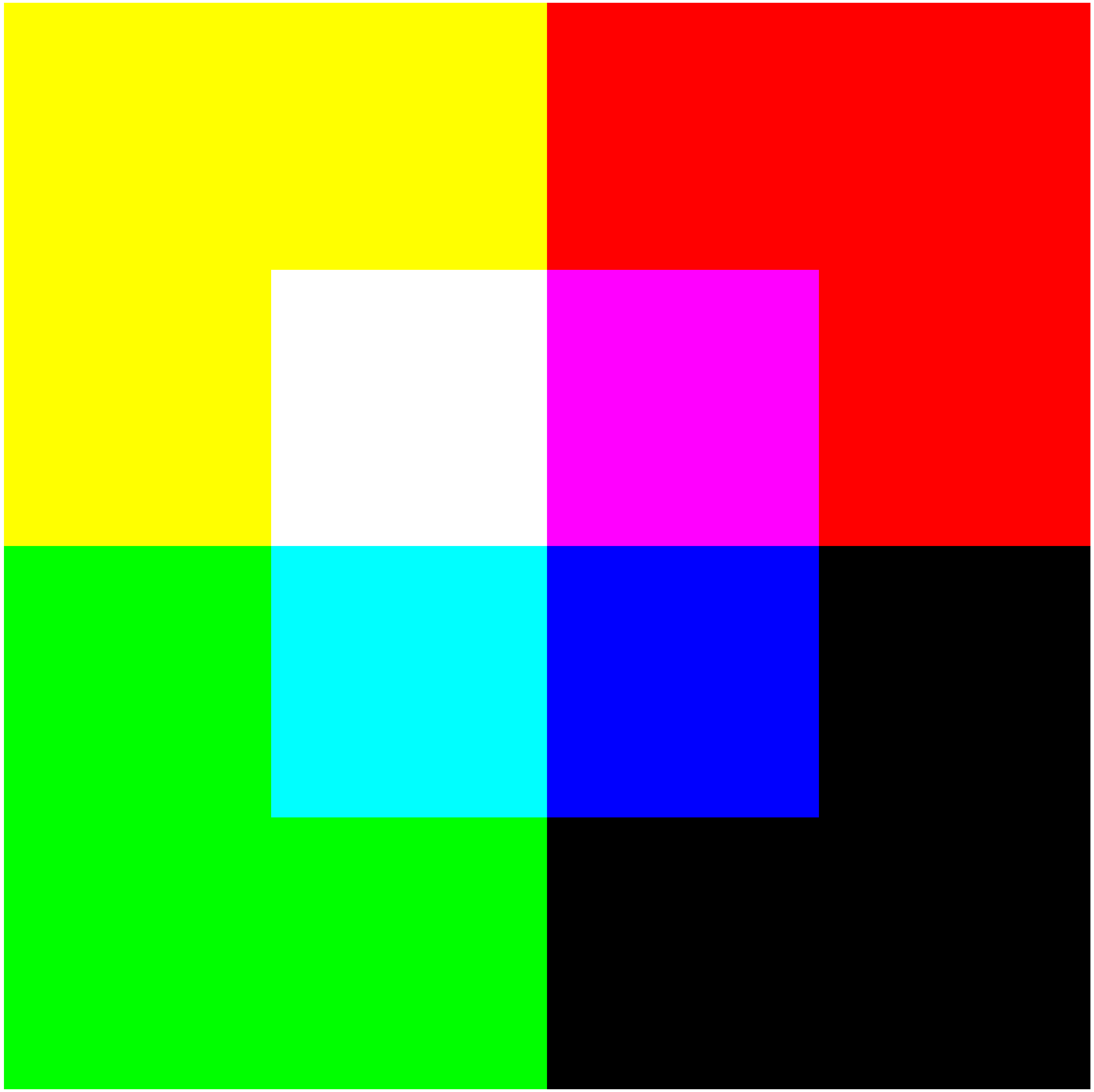


green

blue

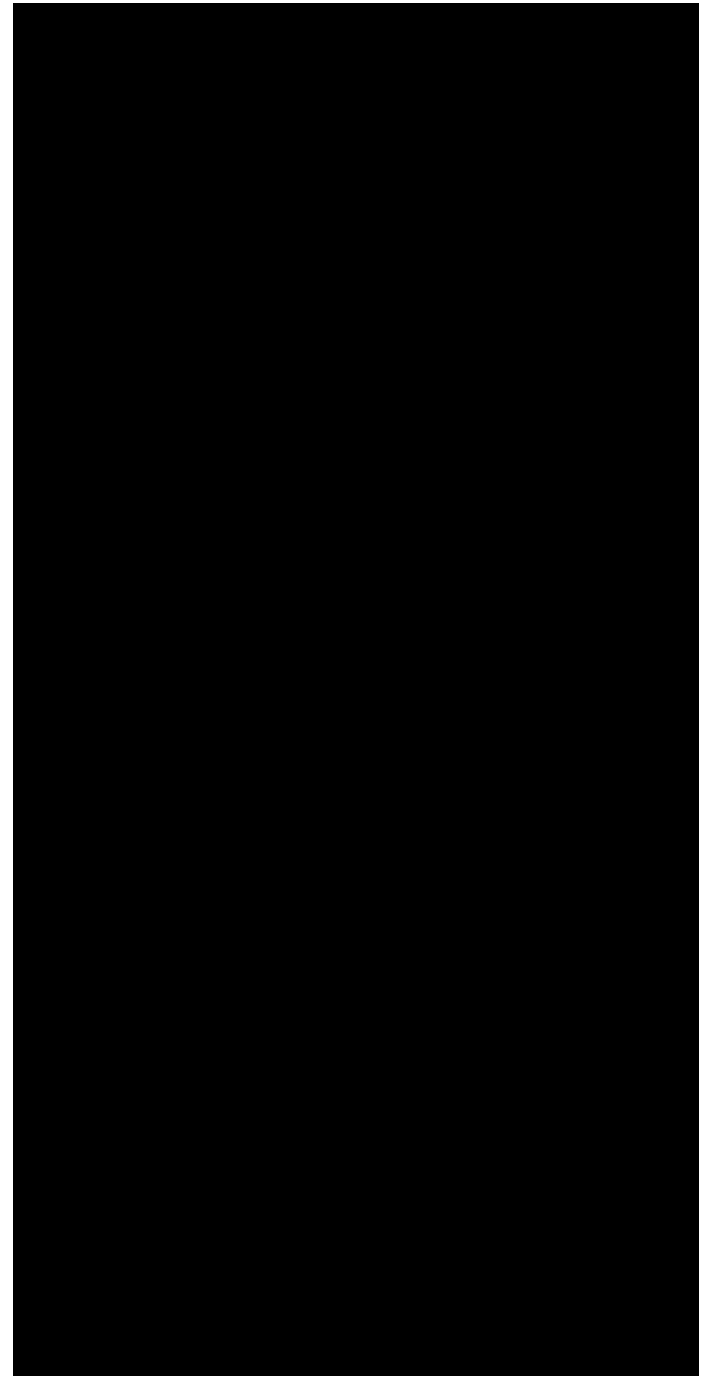


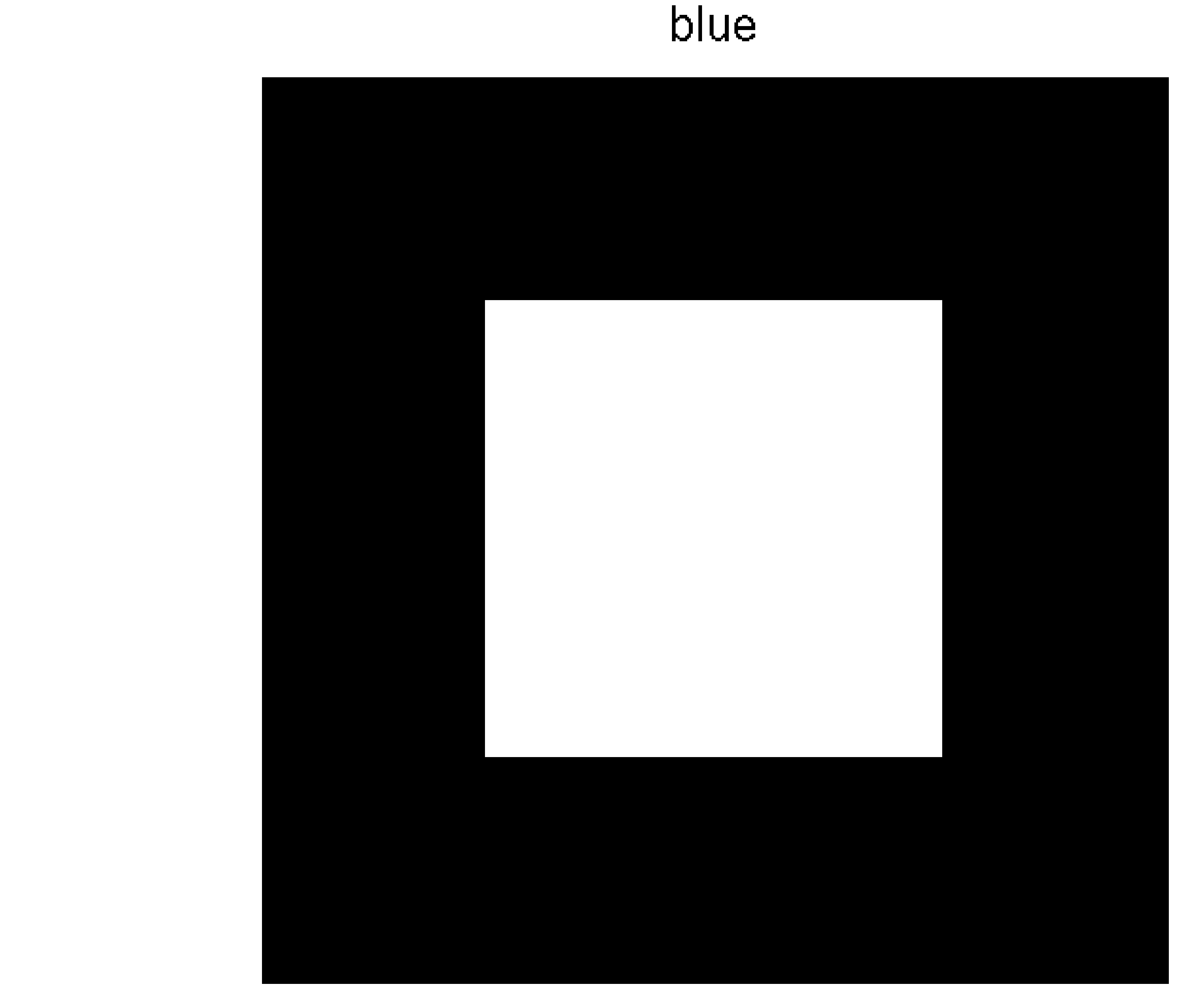
rgb



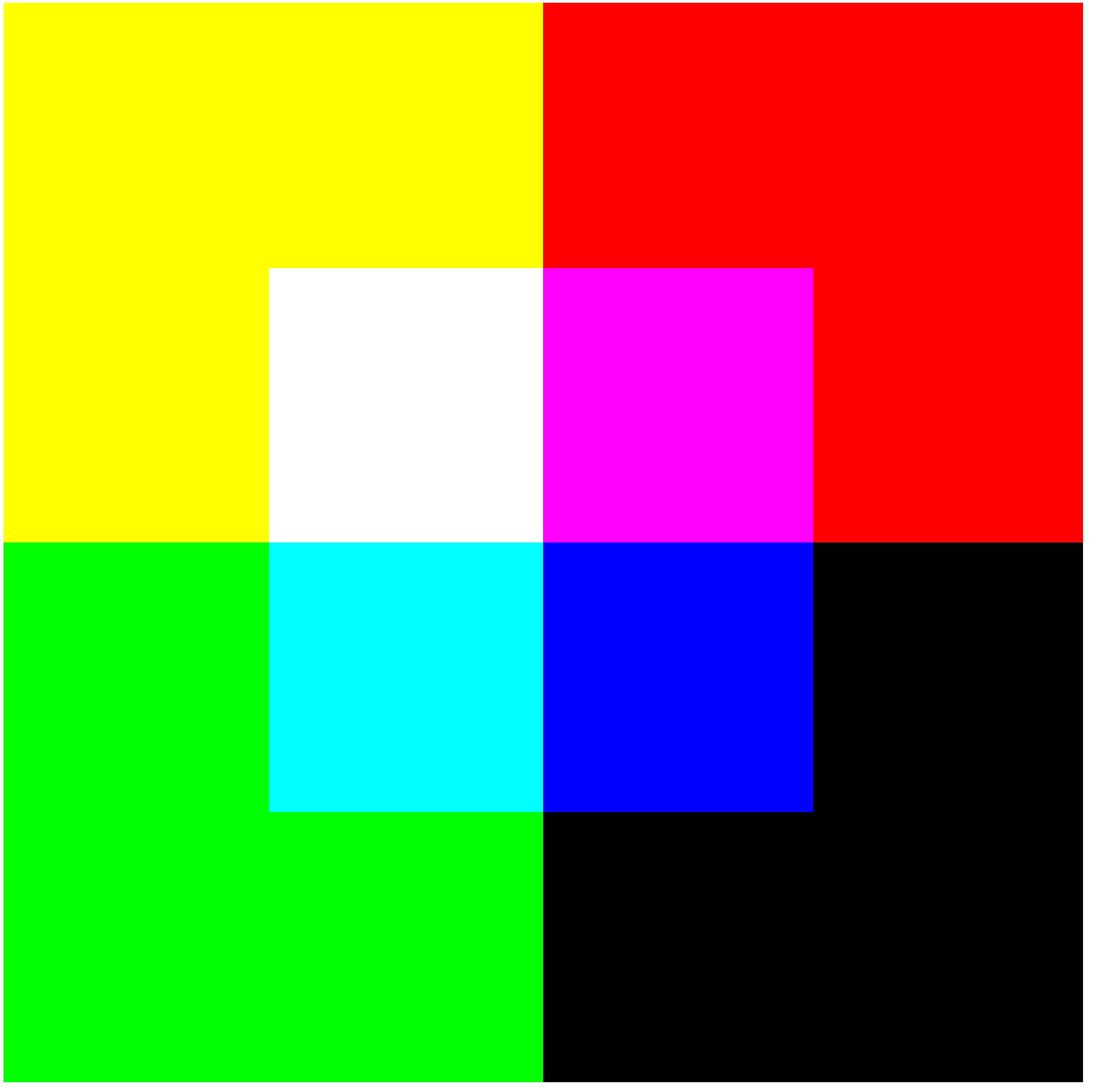
red

green



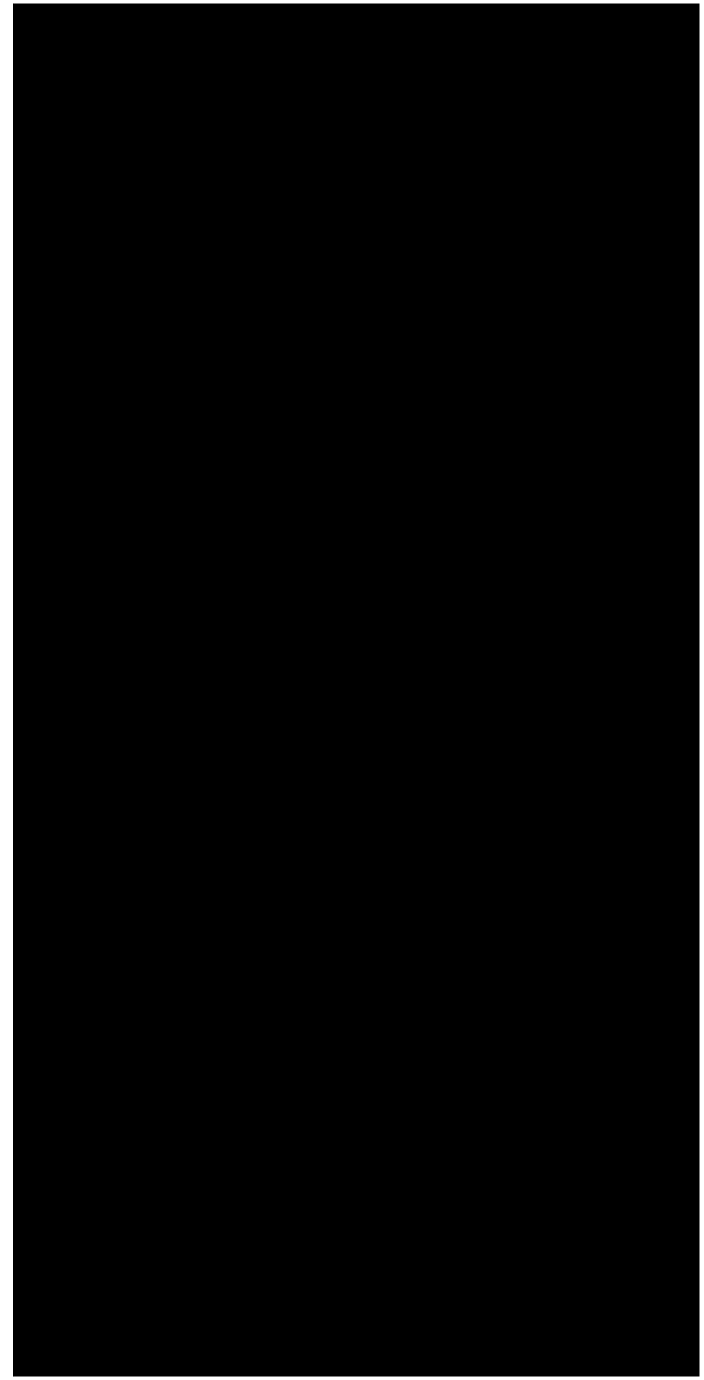

blue

rgb

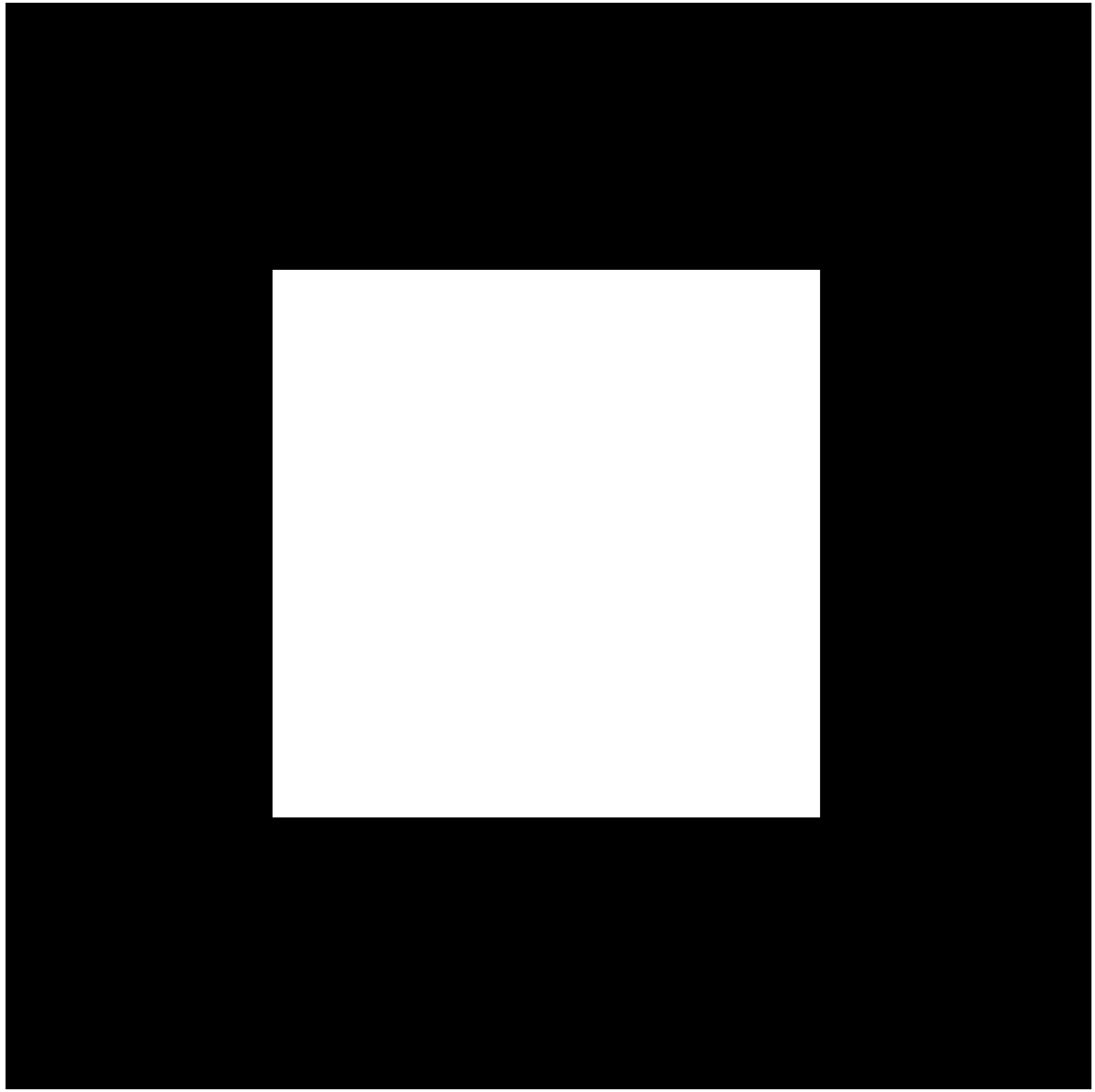


red

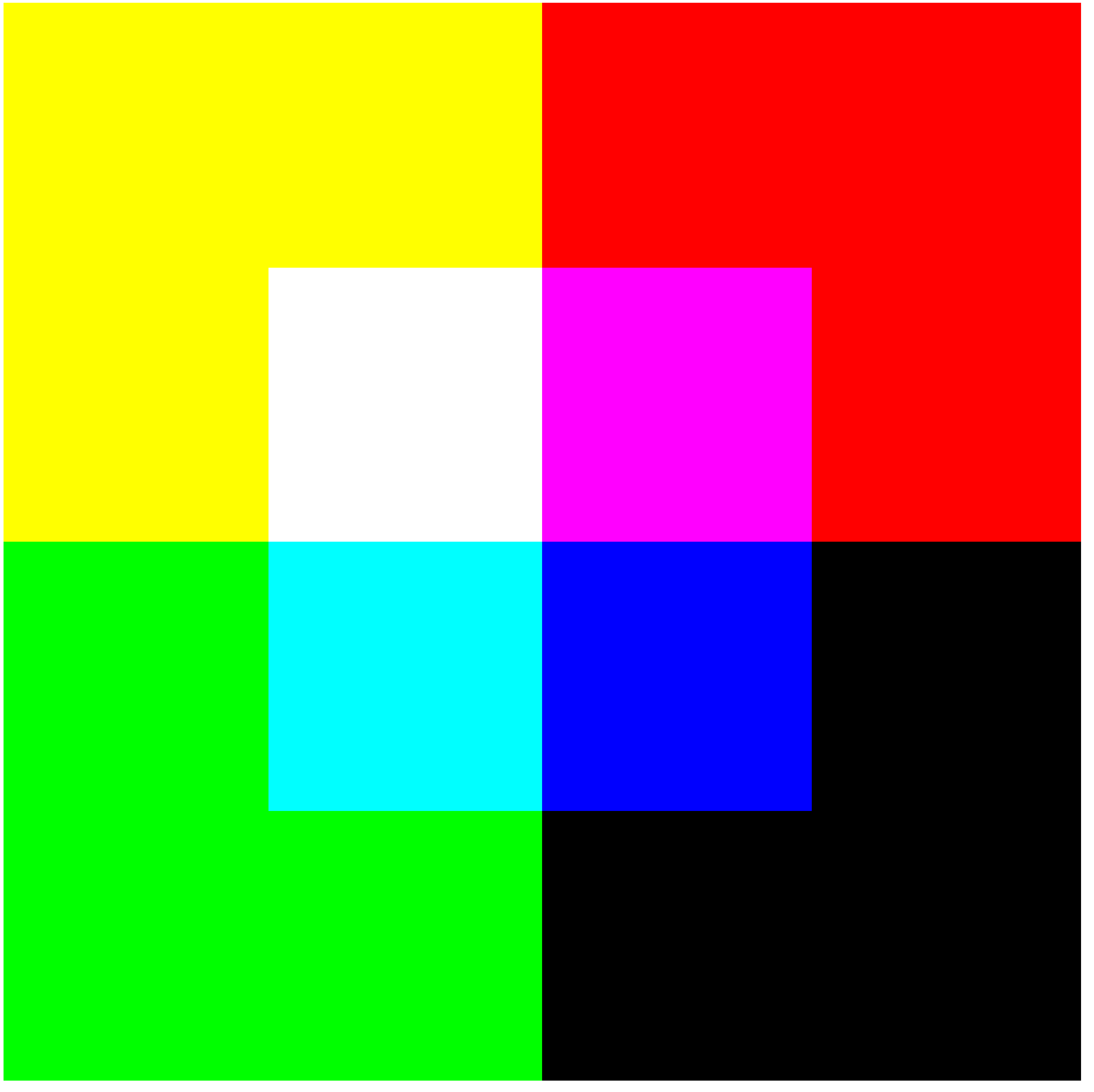
green



blue

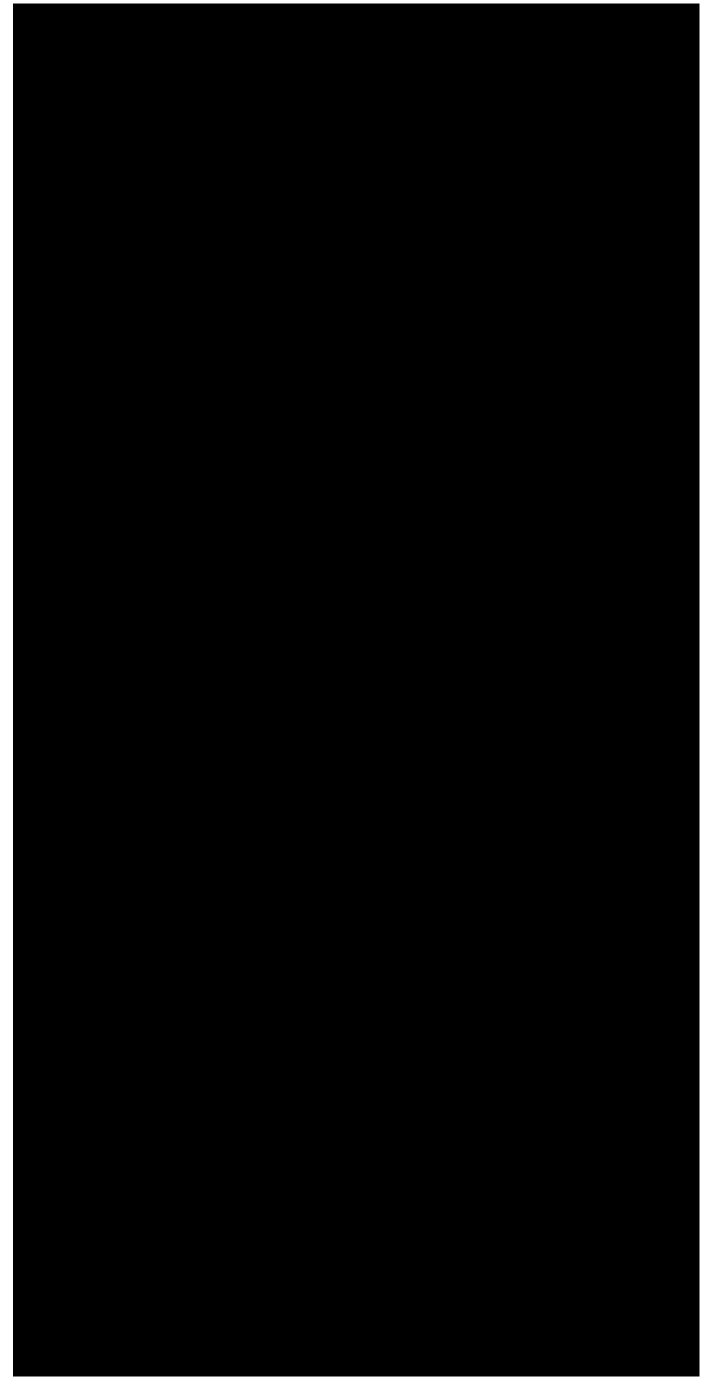


rgb

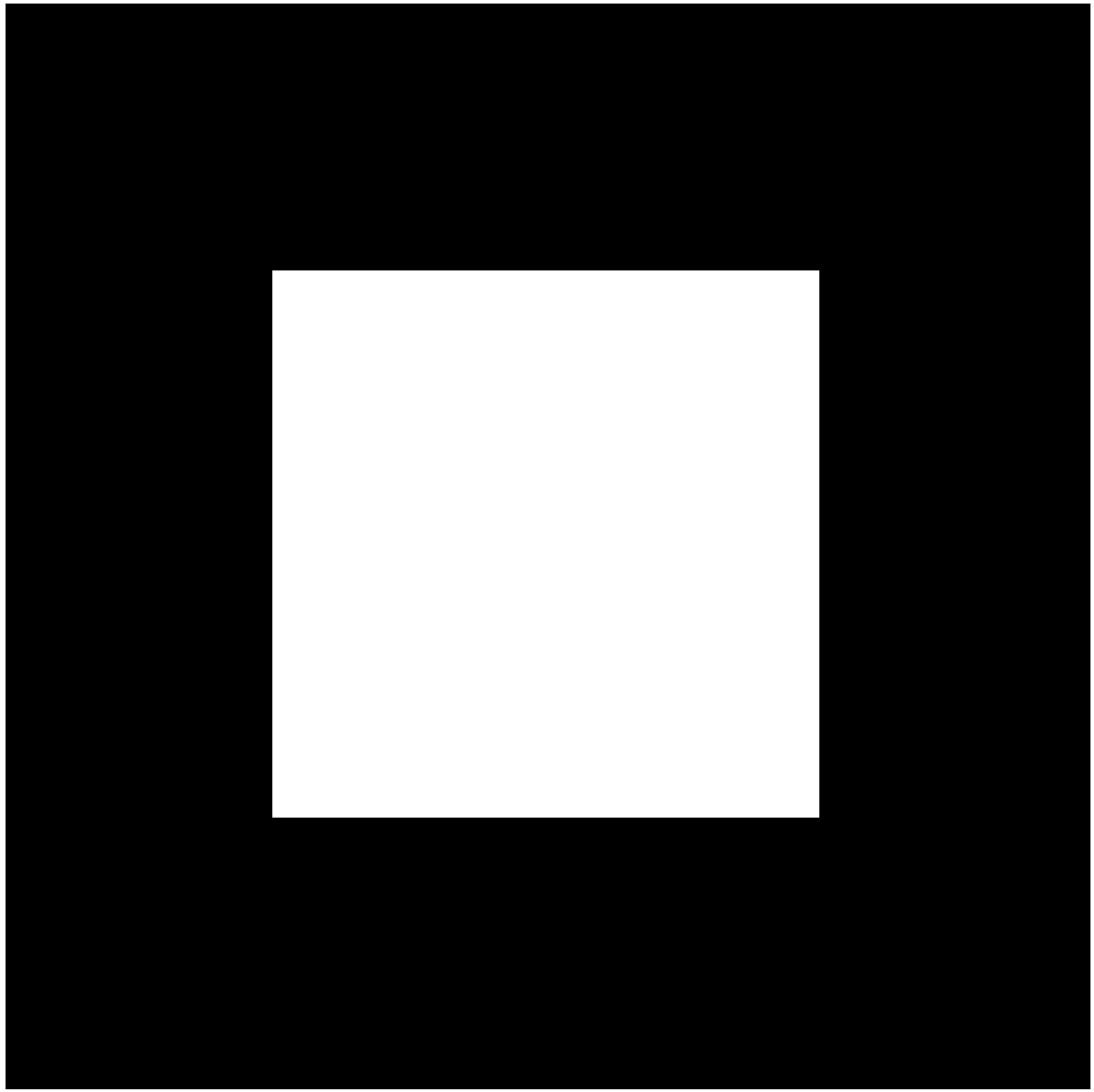


red

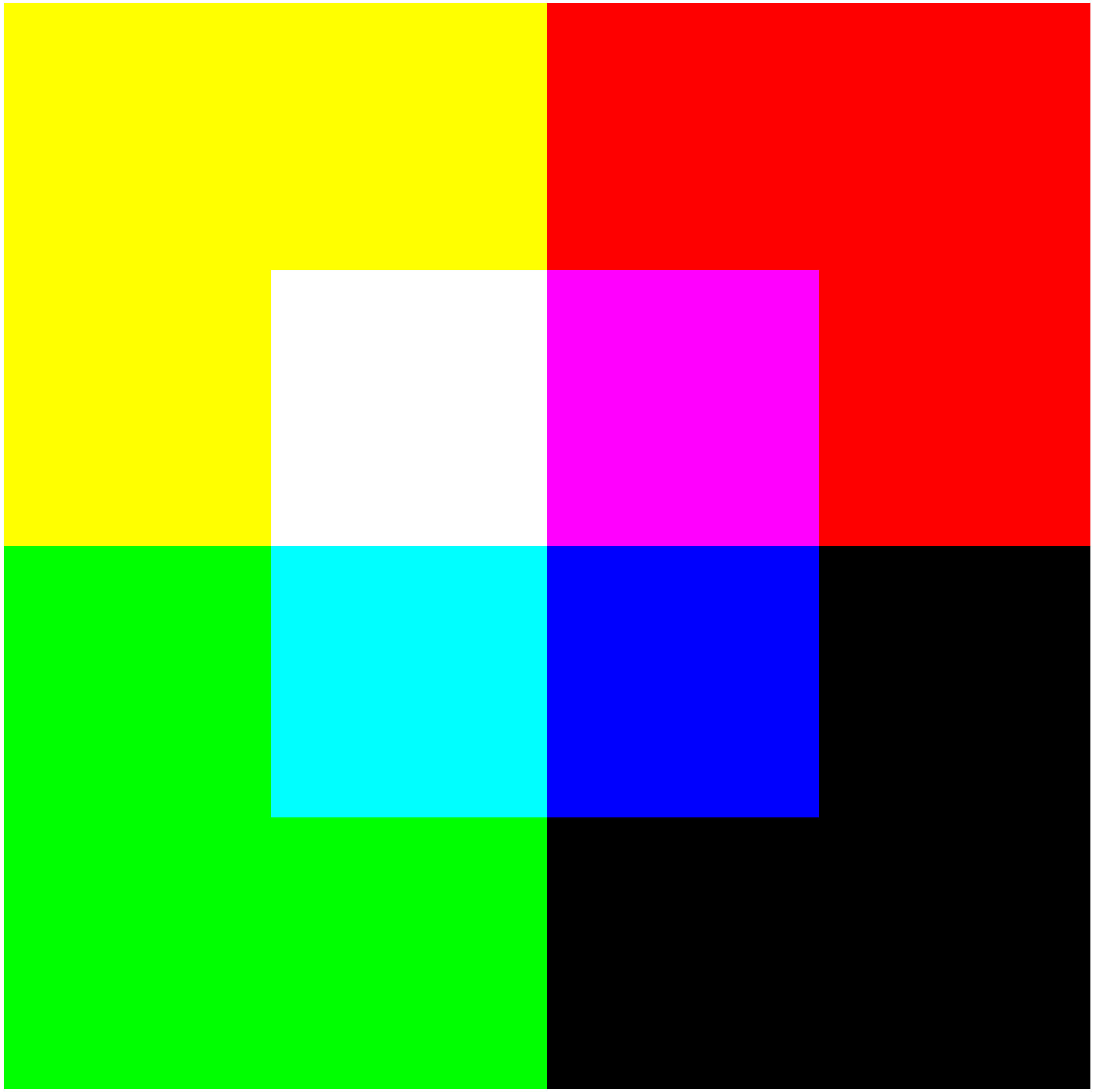
green



blue

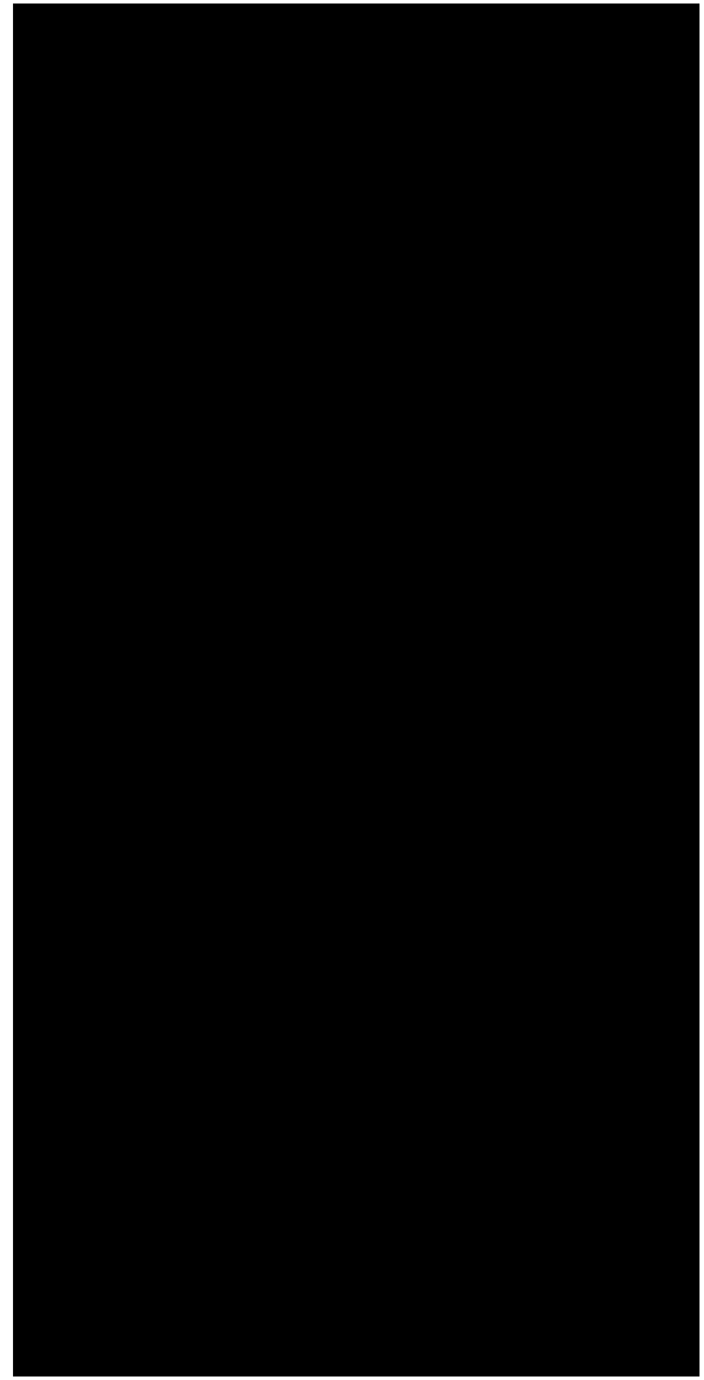


rgb

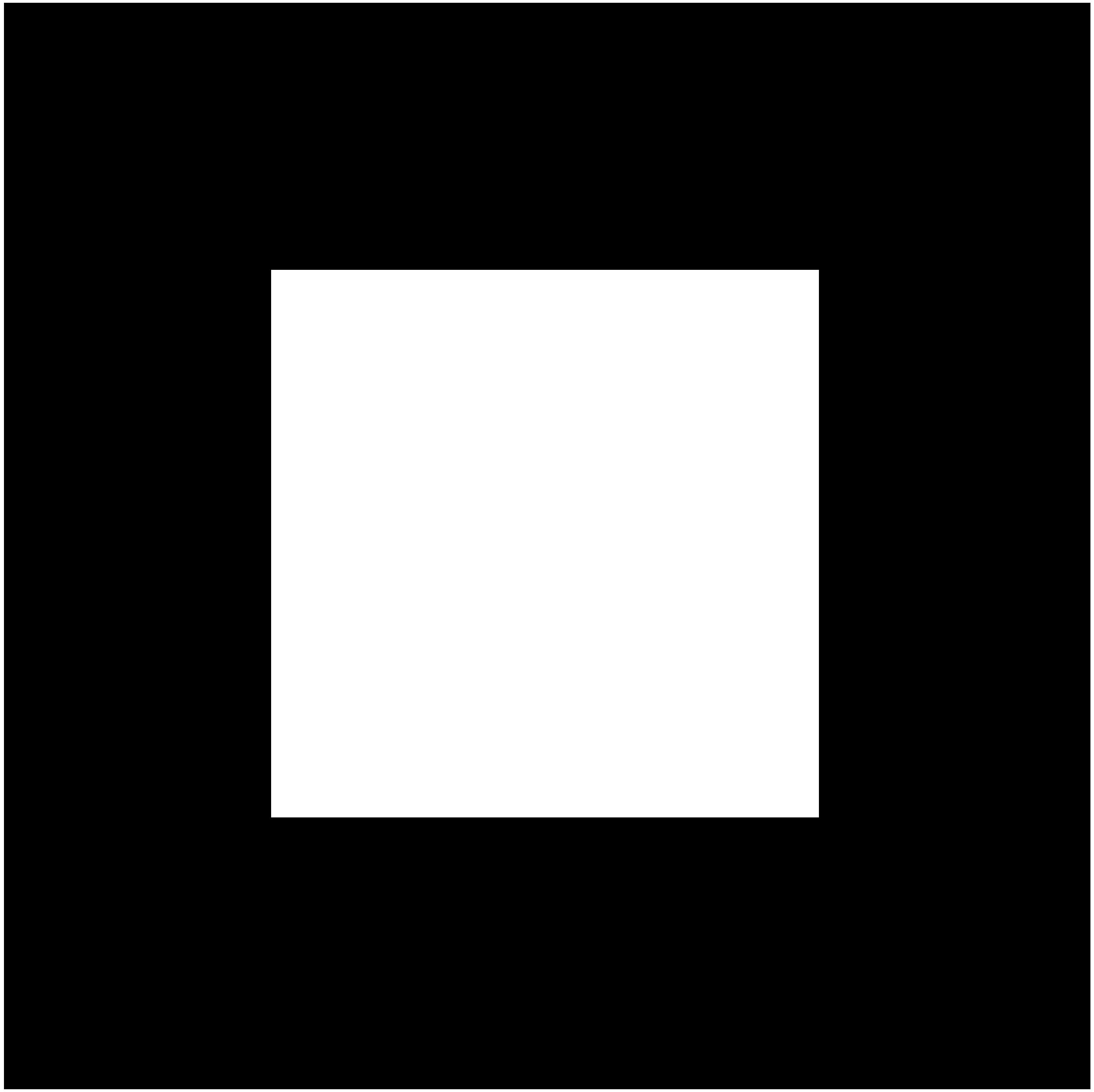


red

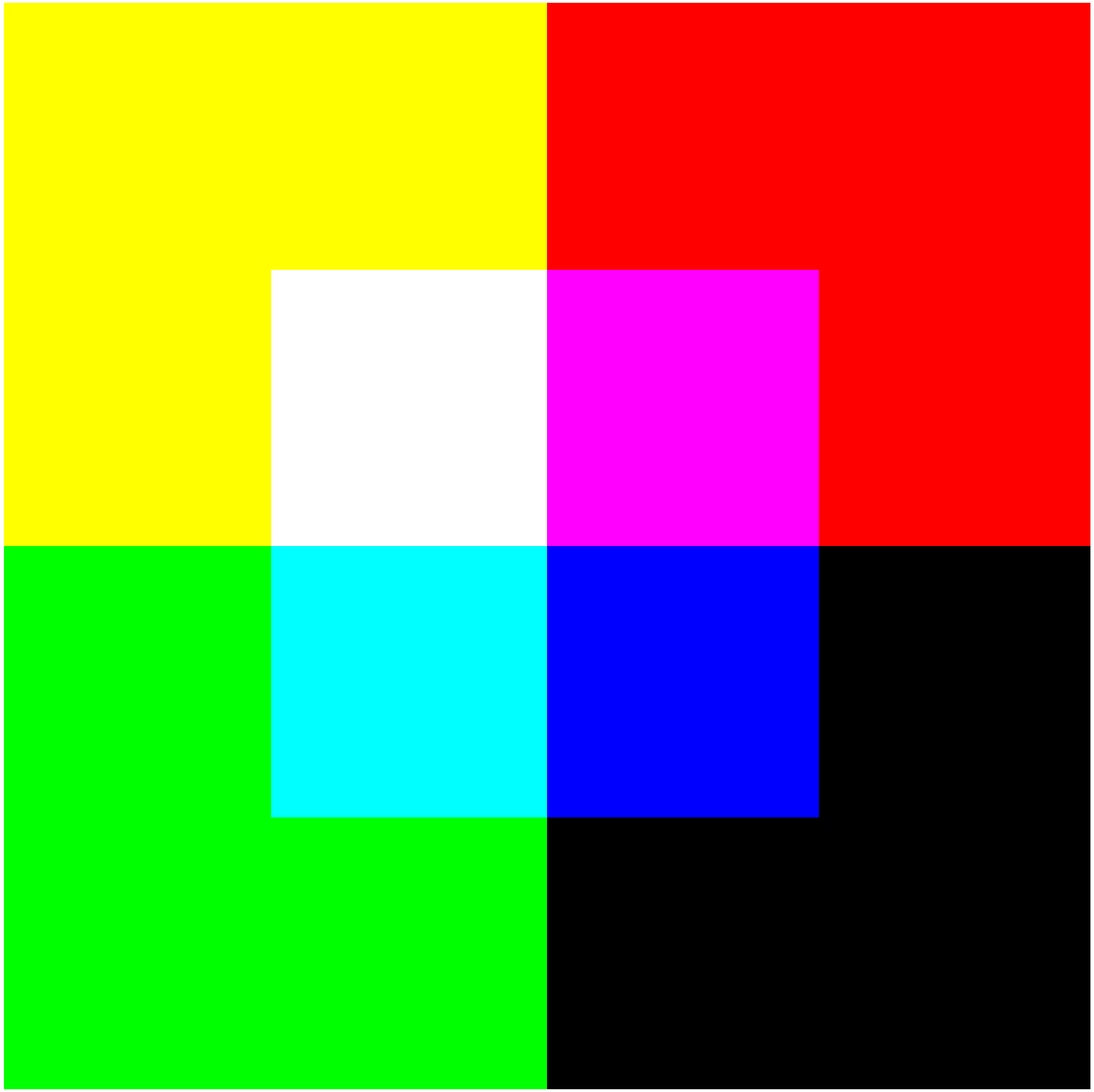
green



blue

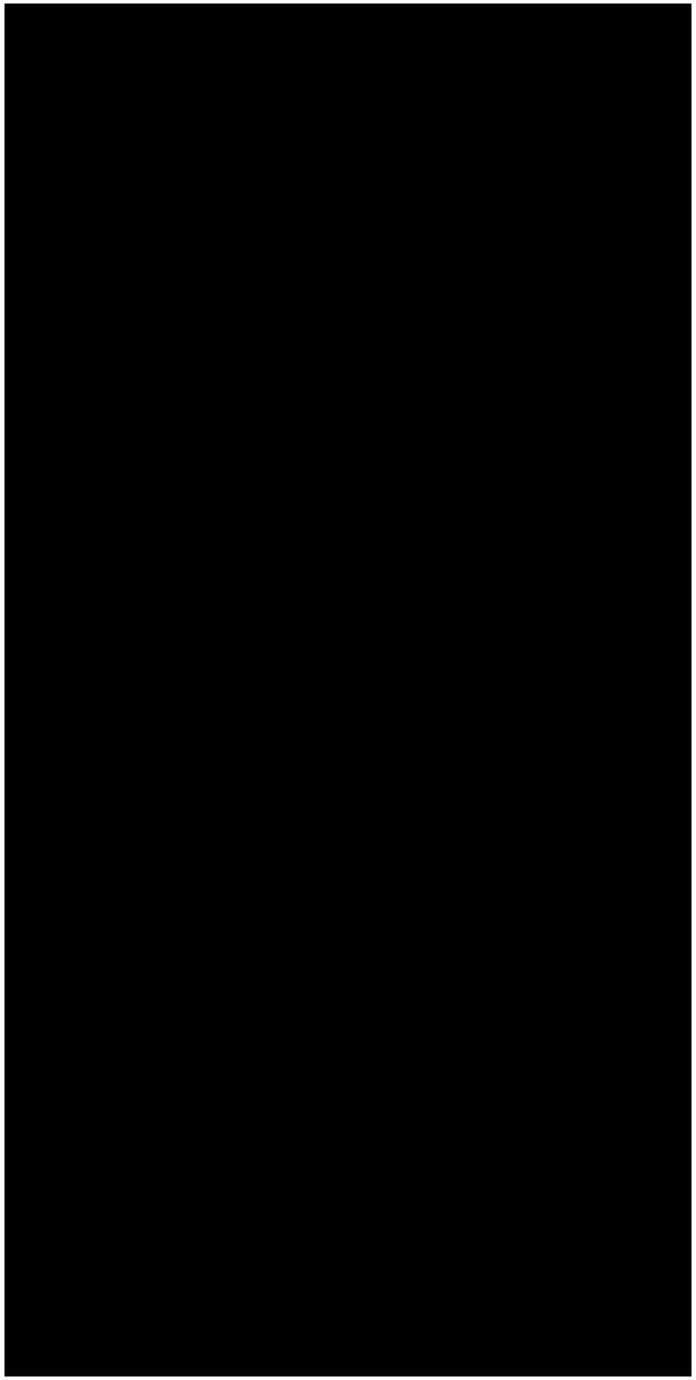


rgb

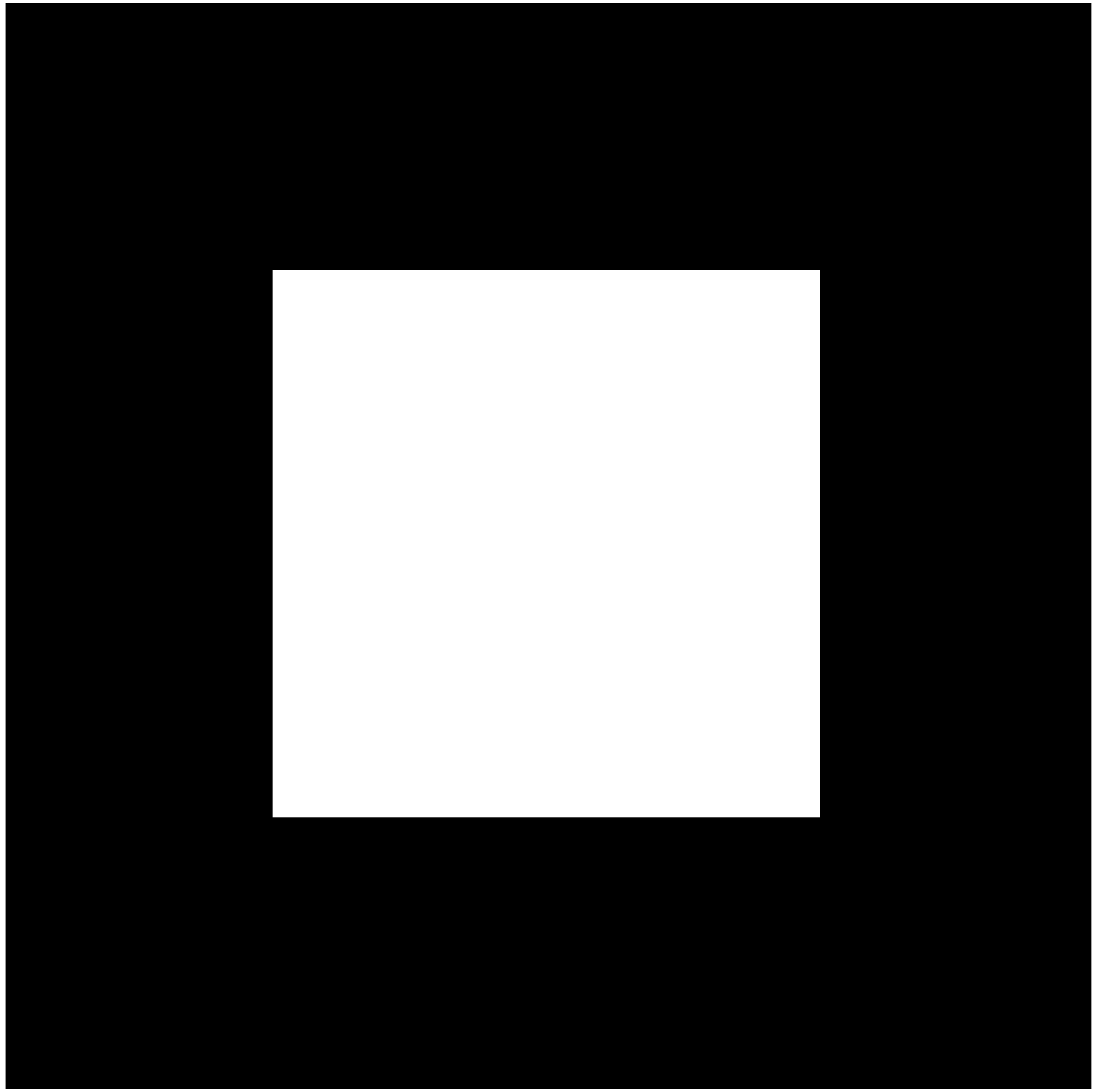


red

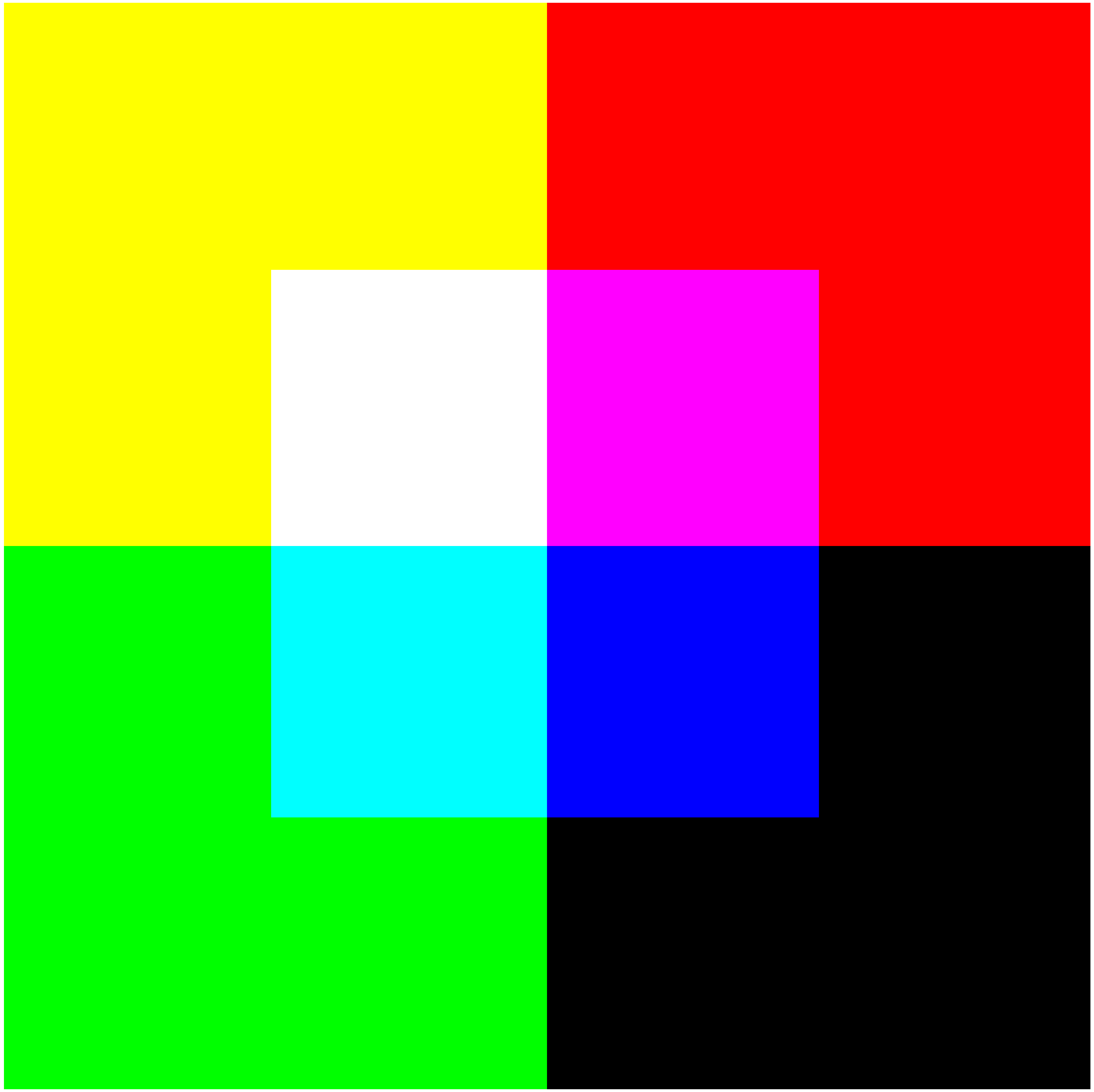
green



blue

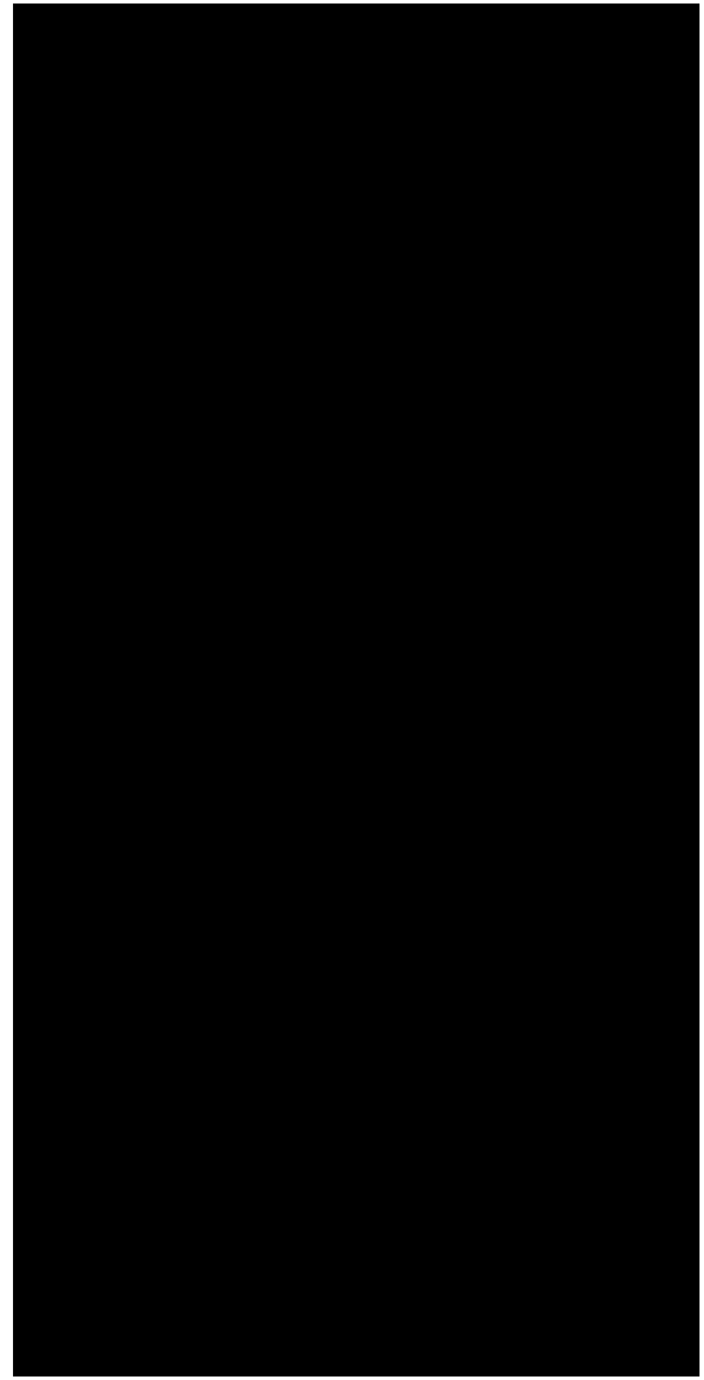


rgb

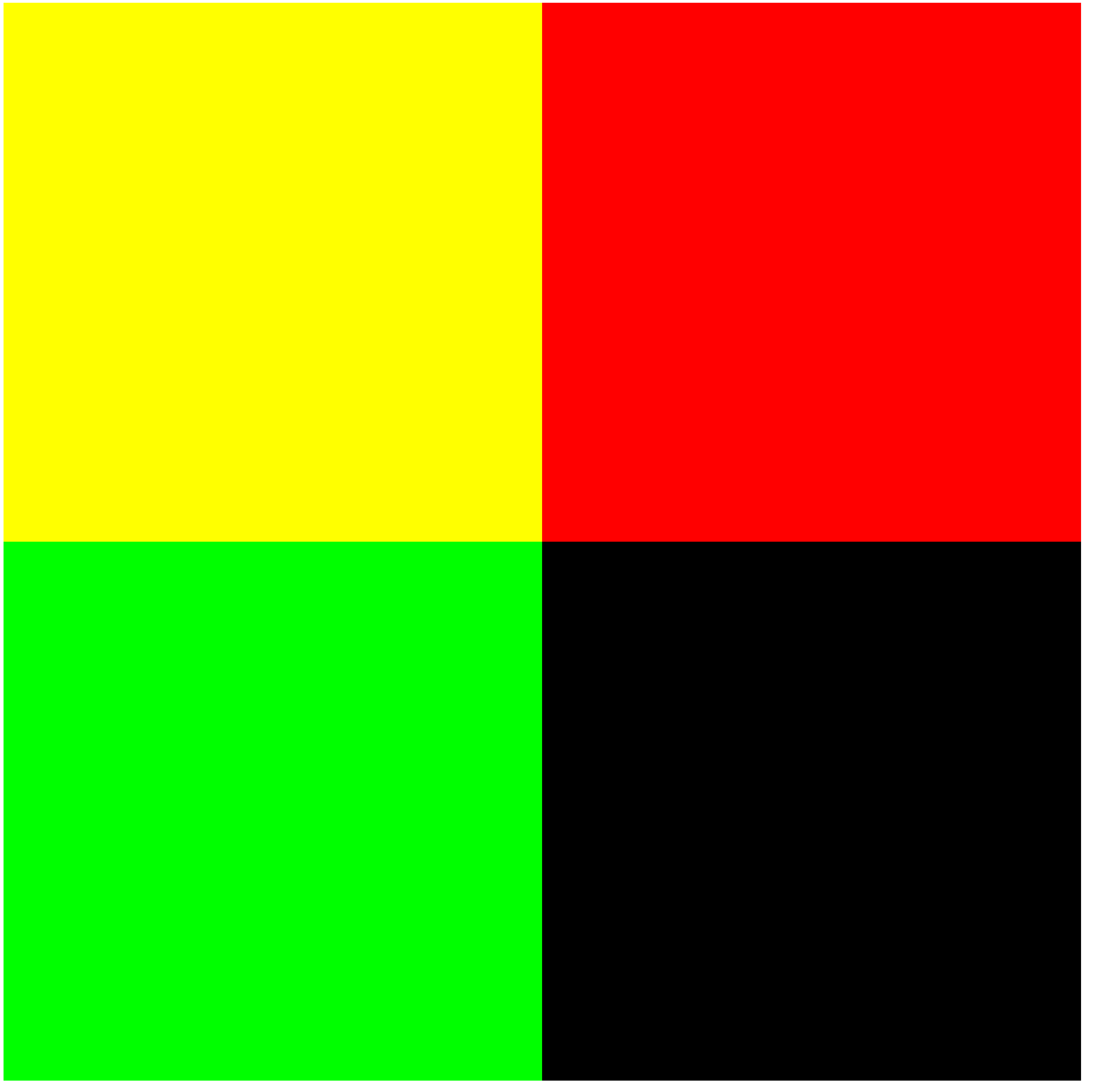


red

green

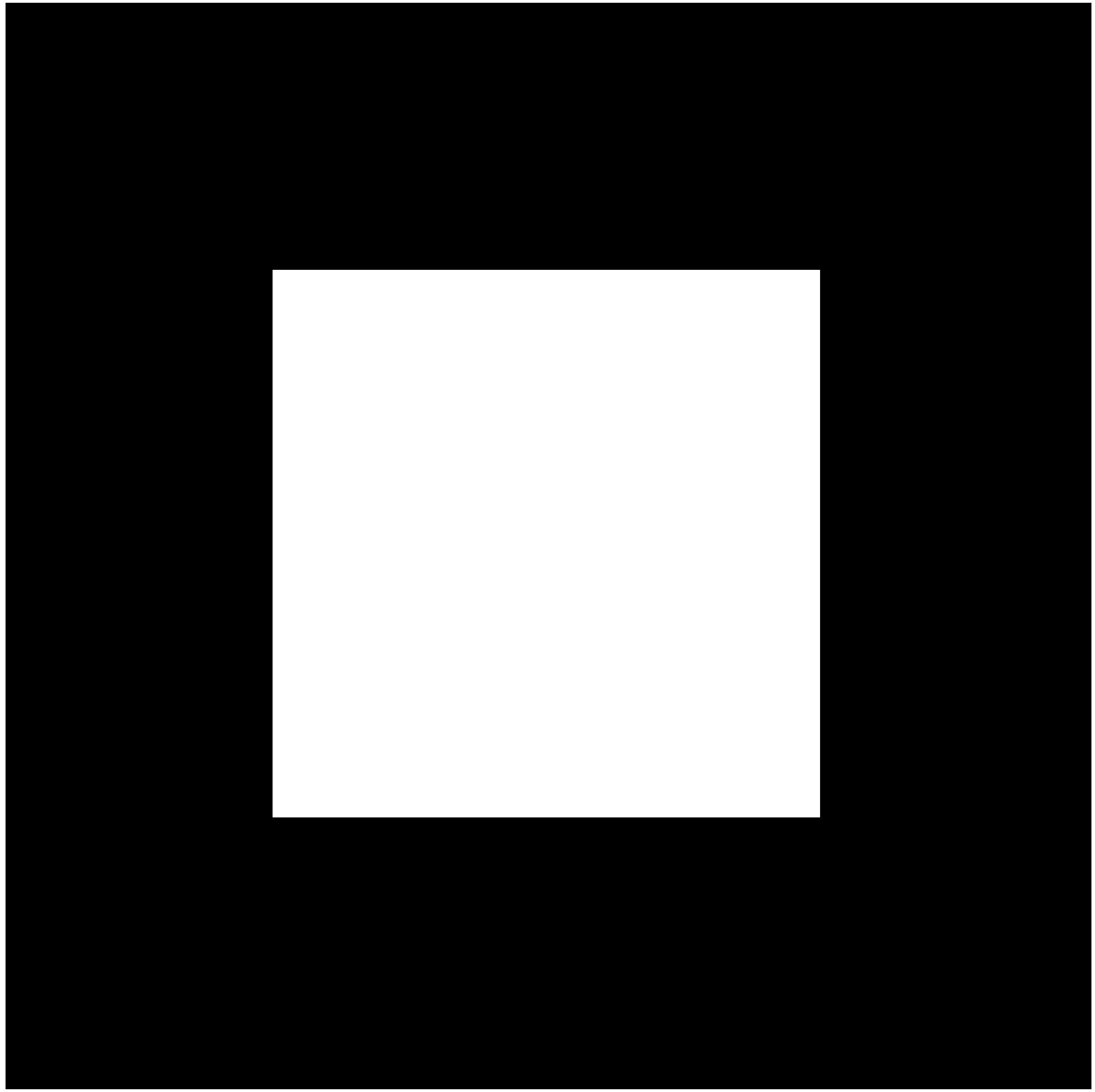


rg

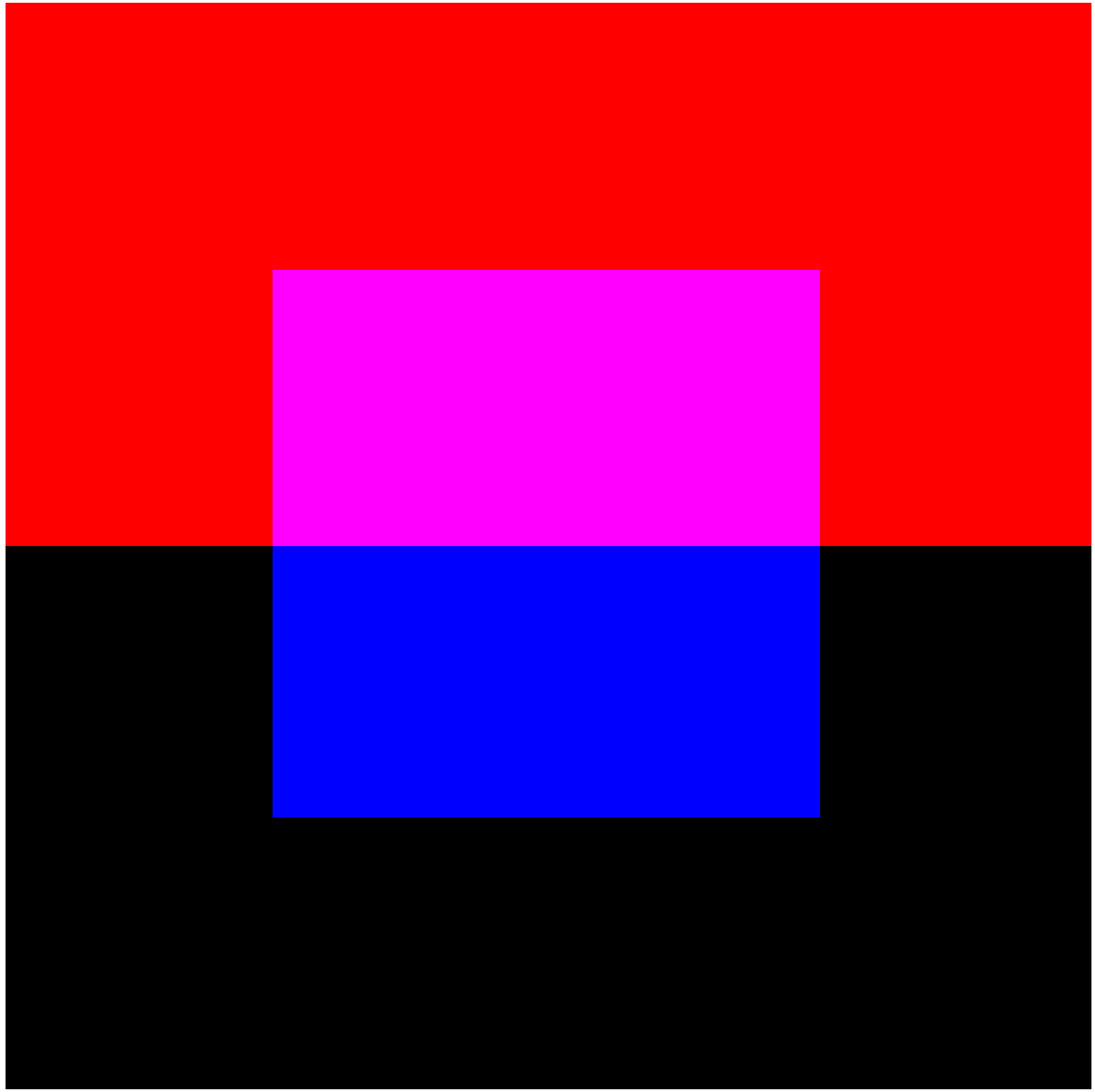


red

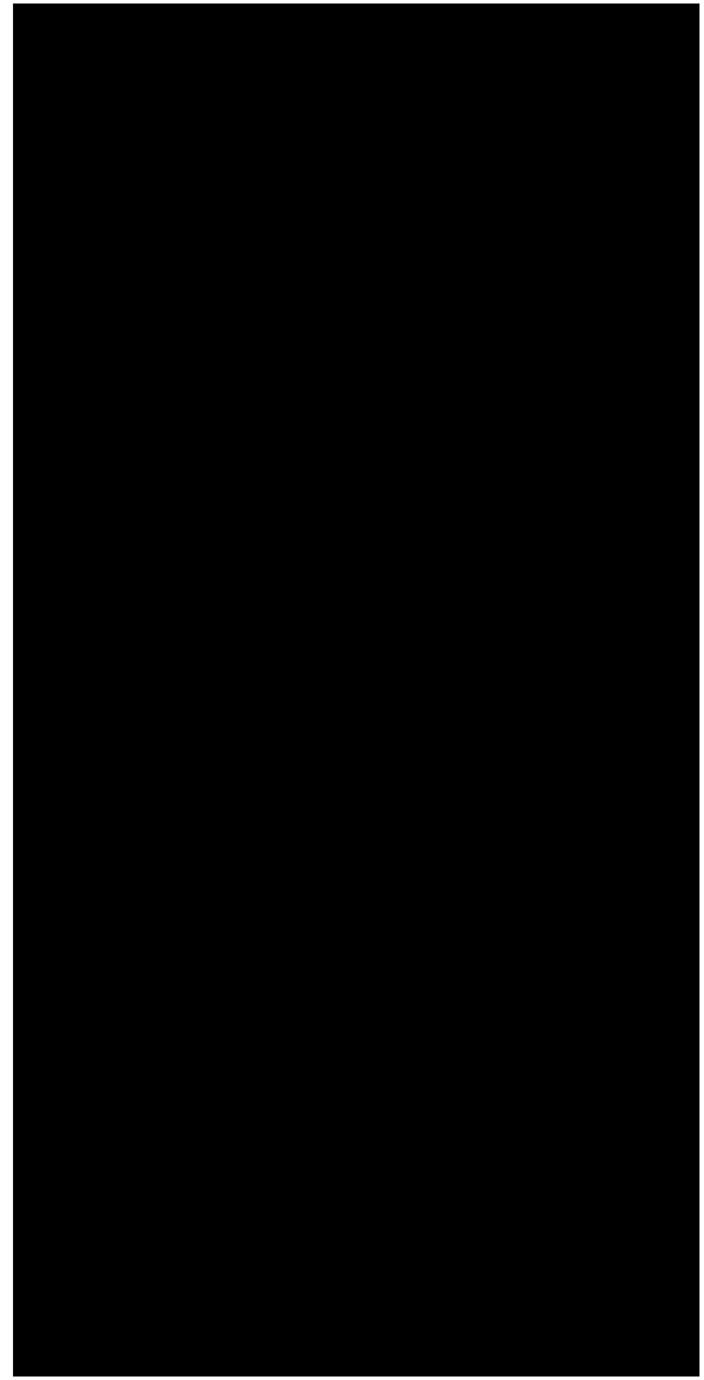
blue



rb

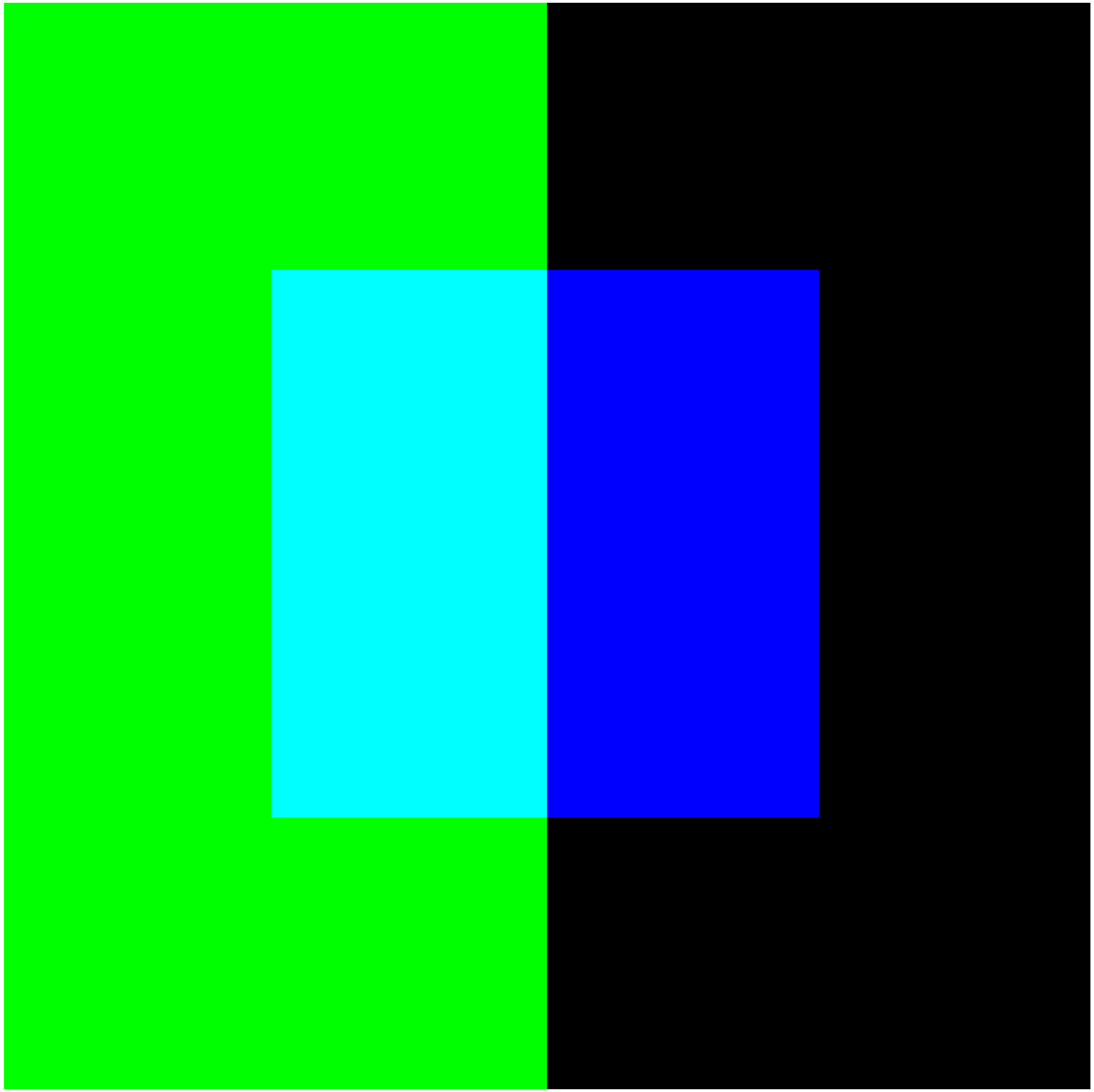


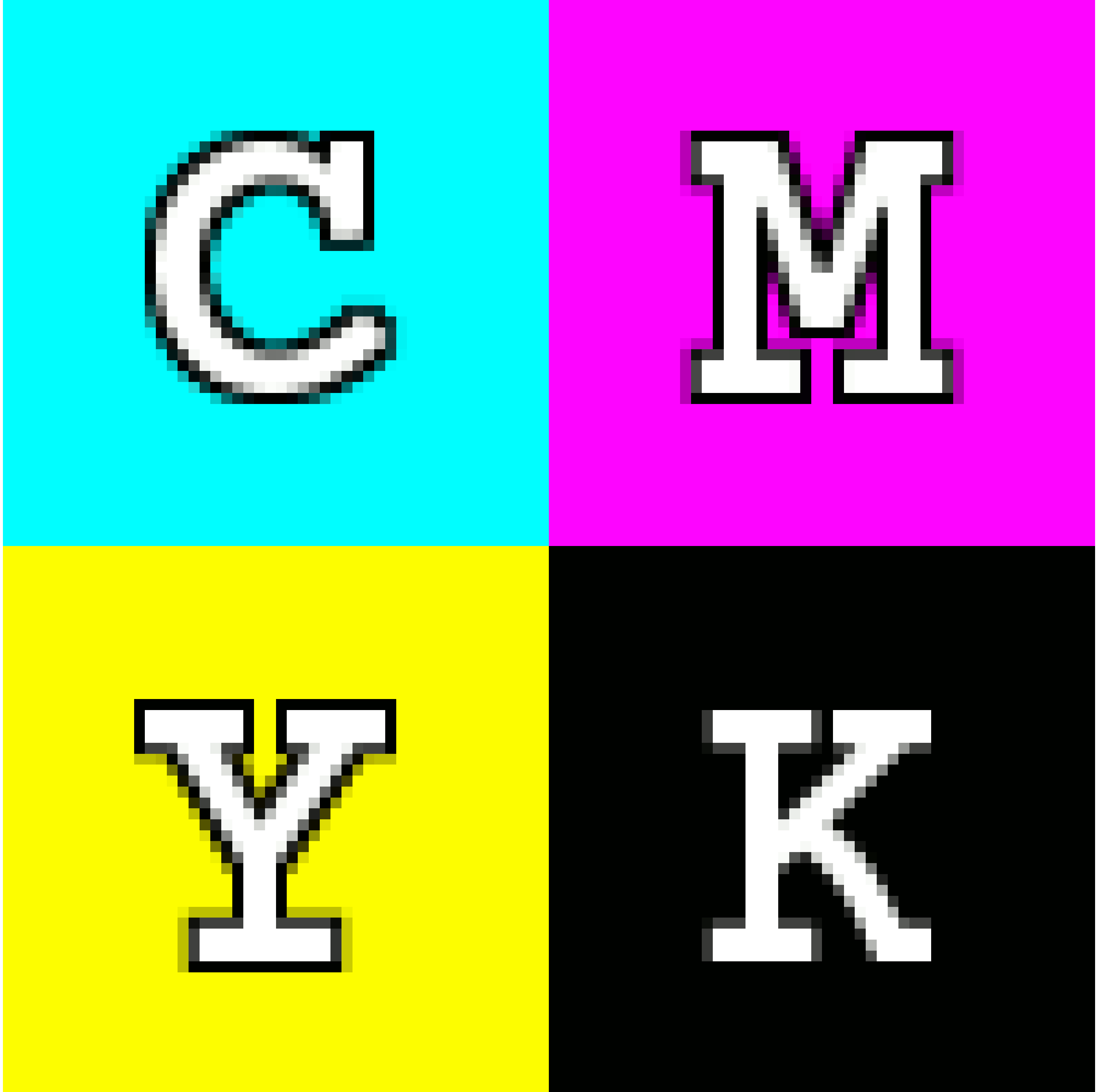
green

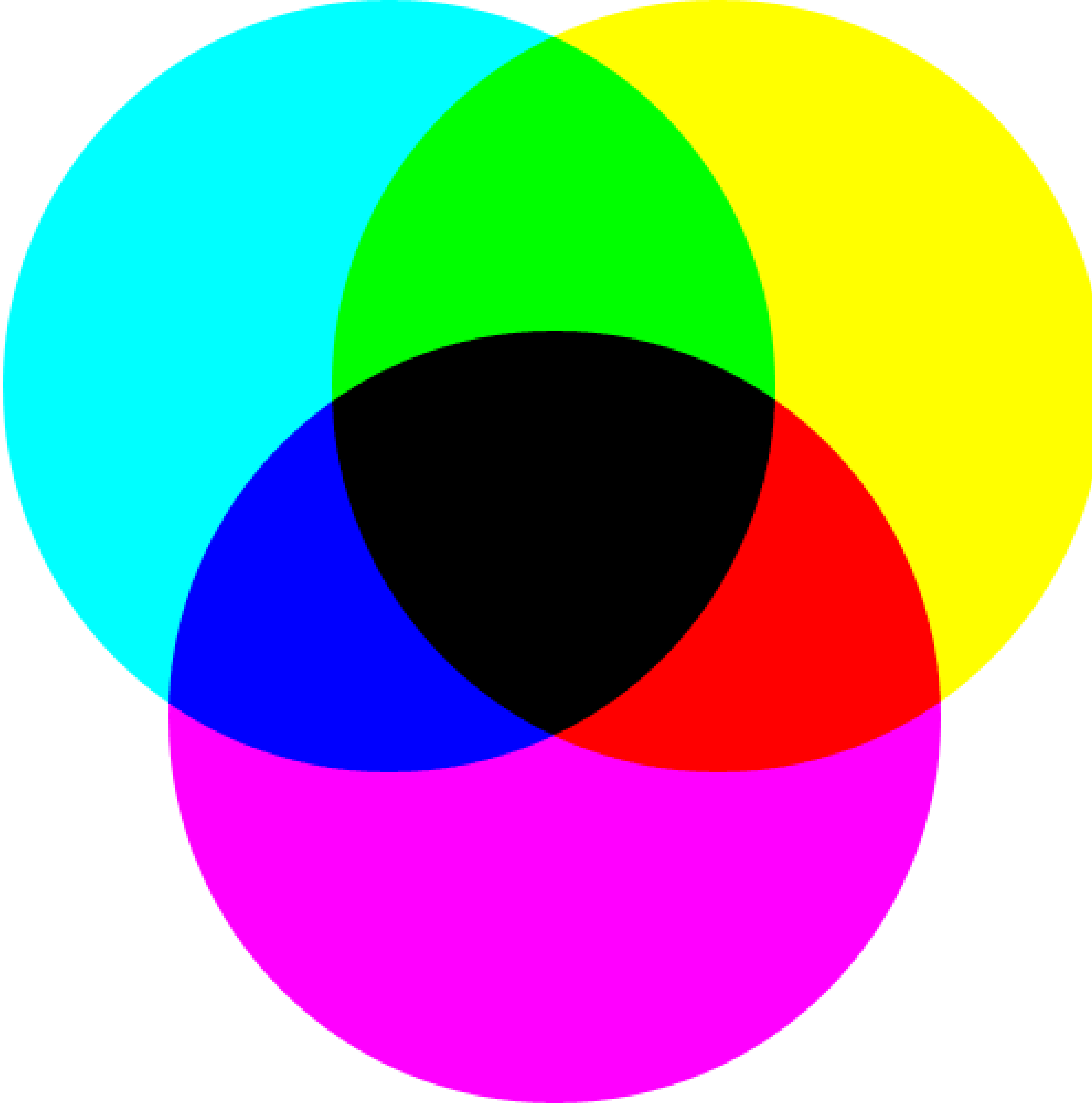


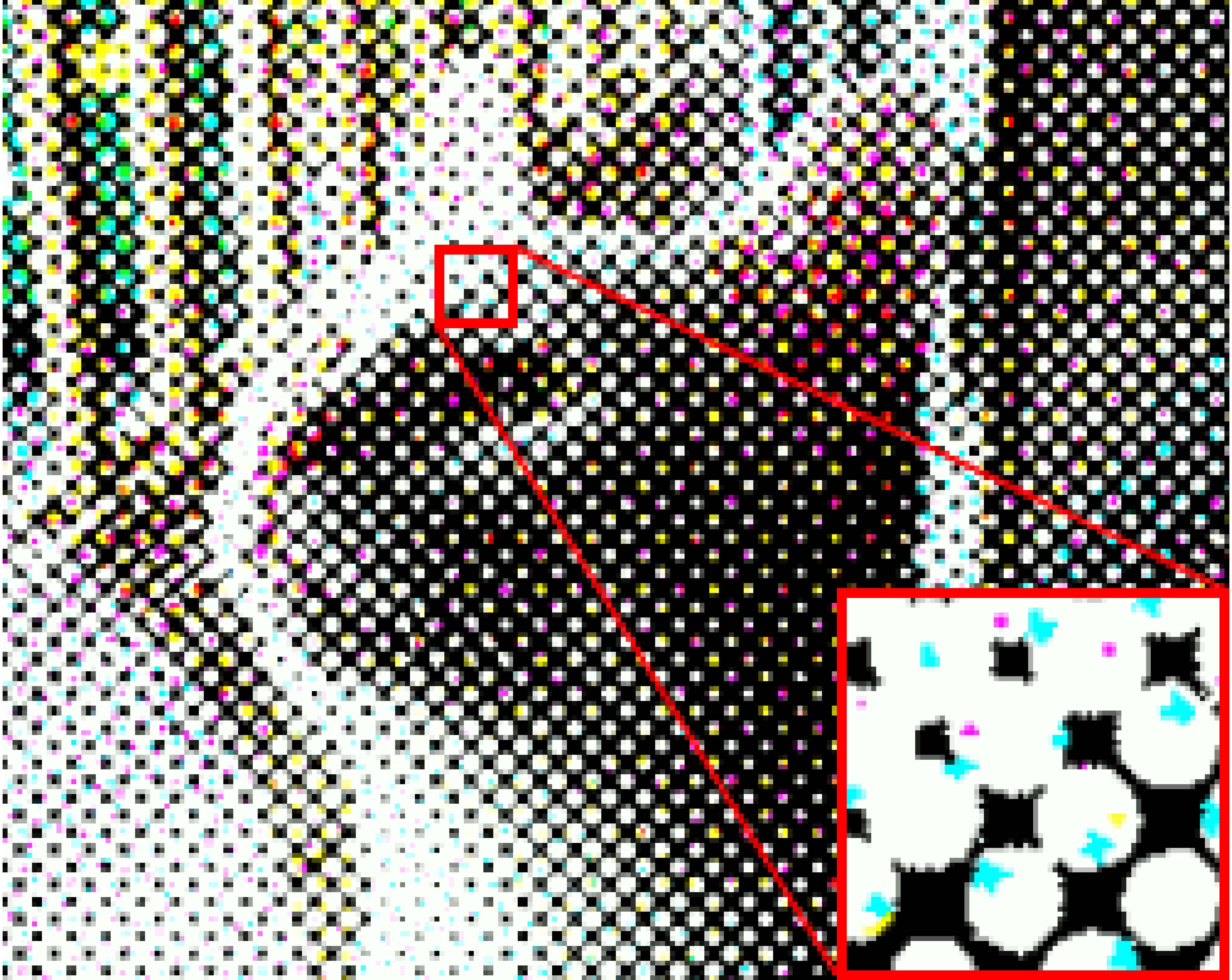

blue

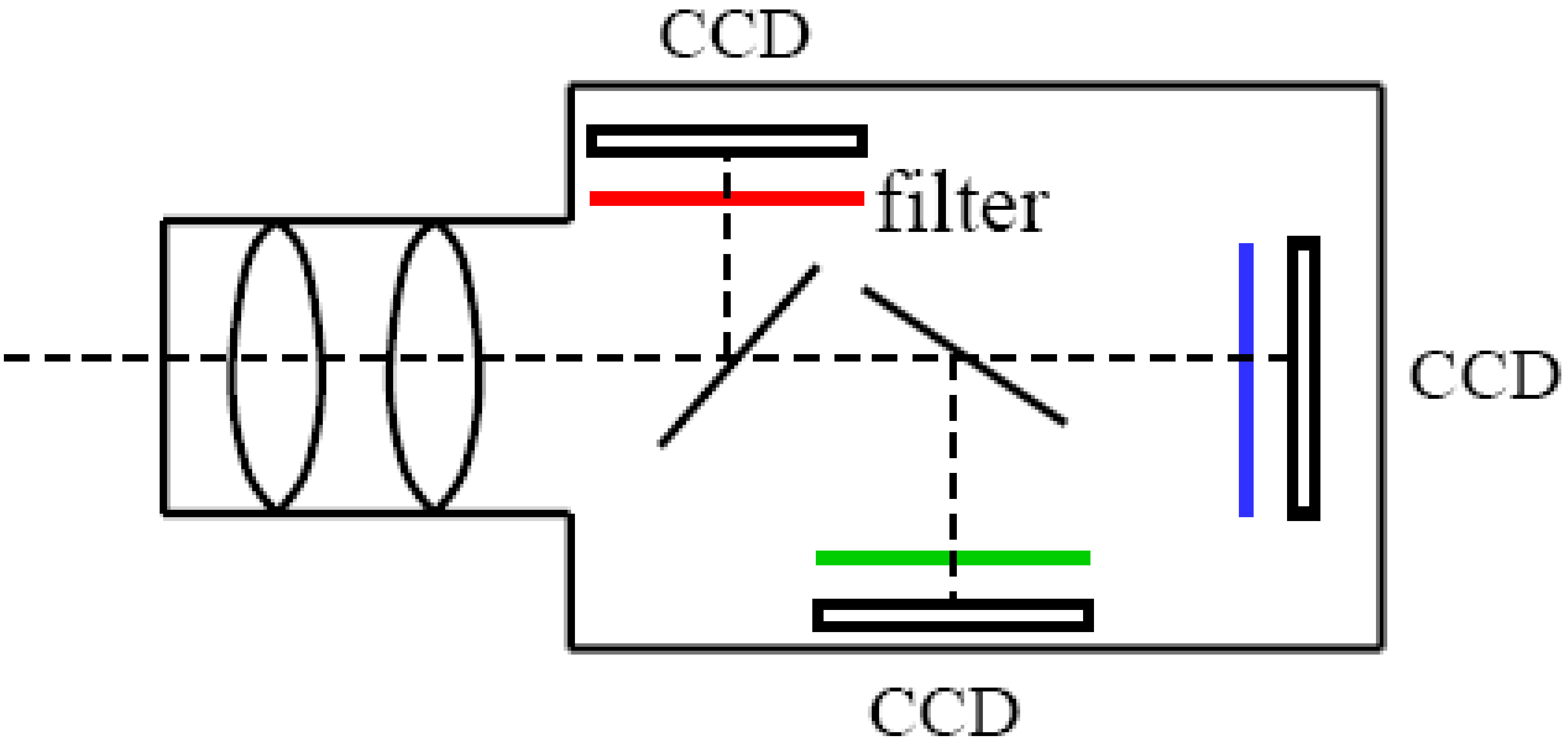
gb

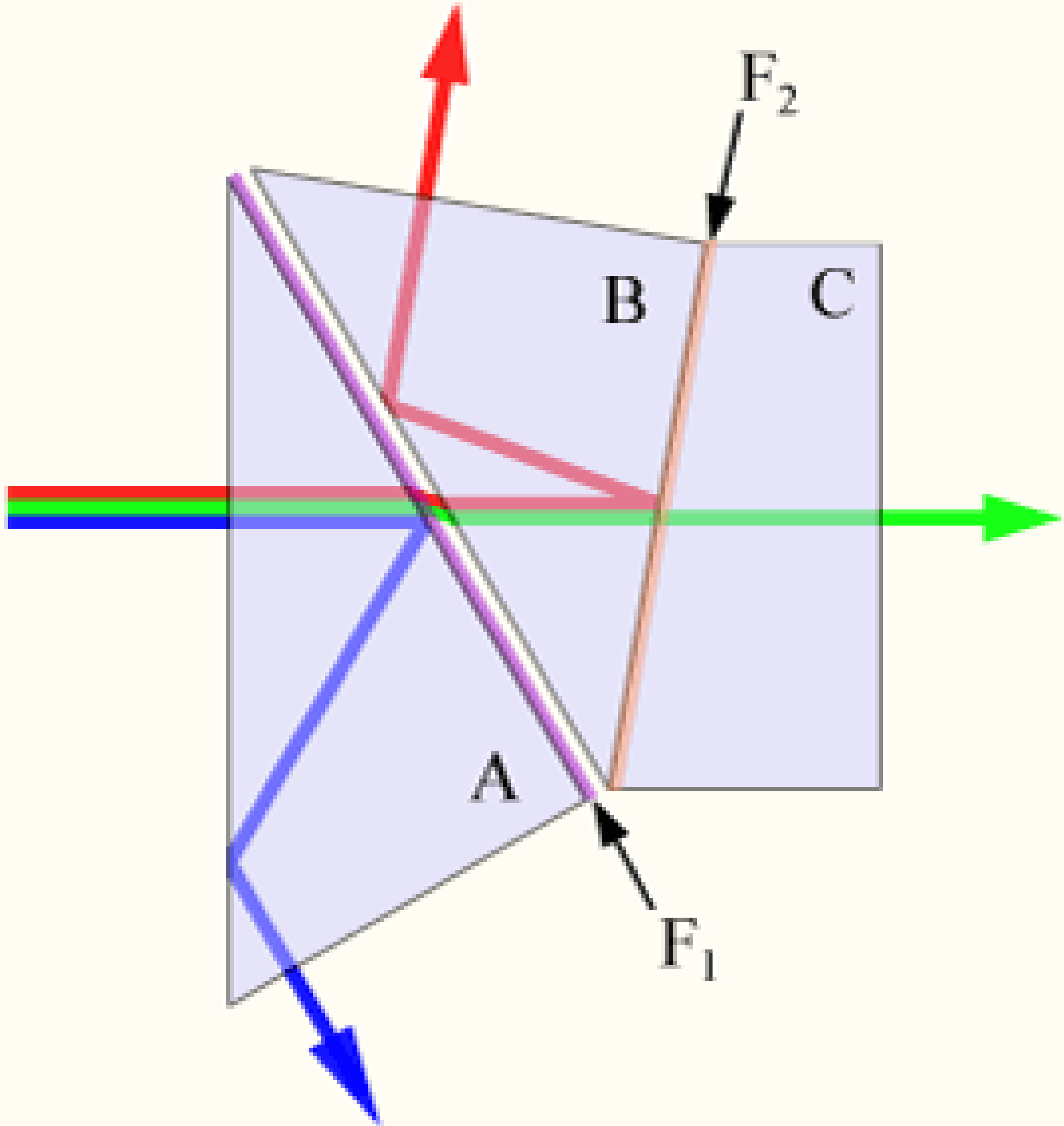



C
M
Y
K

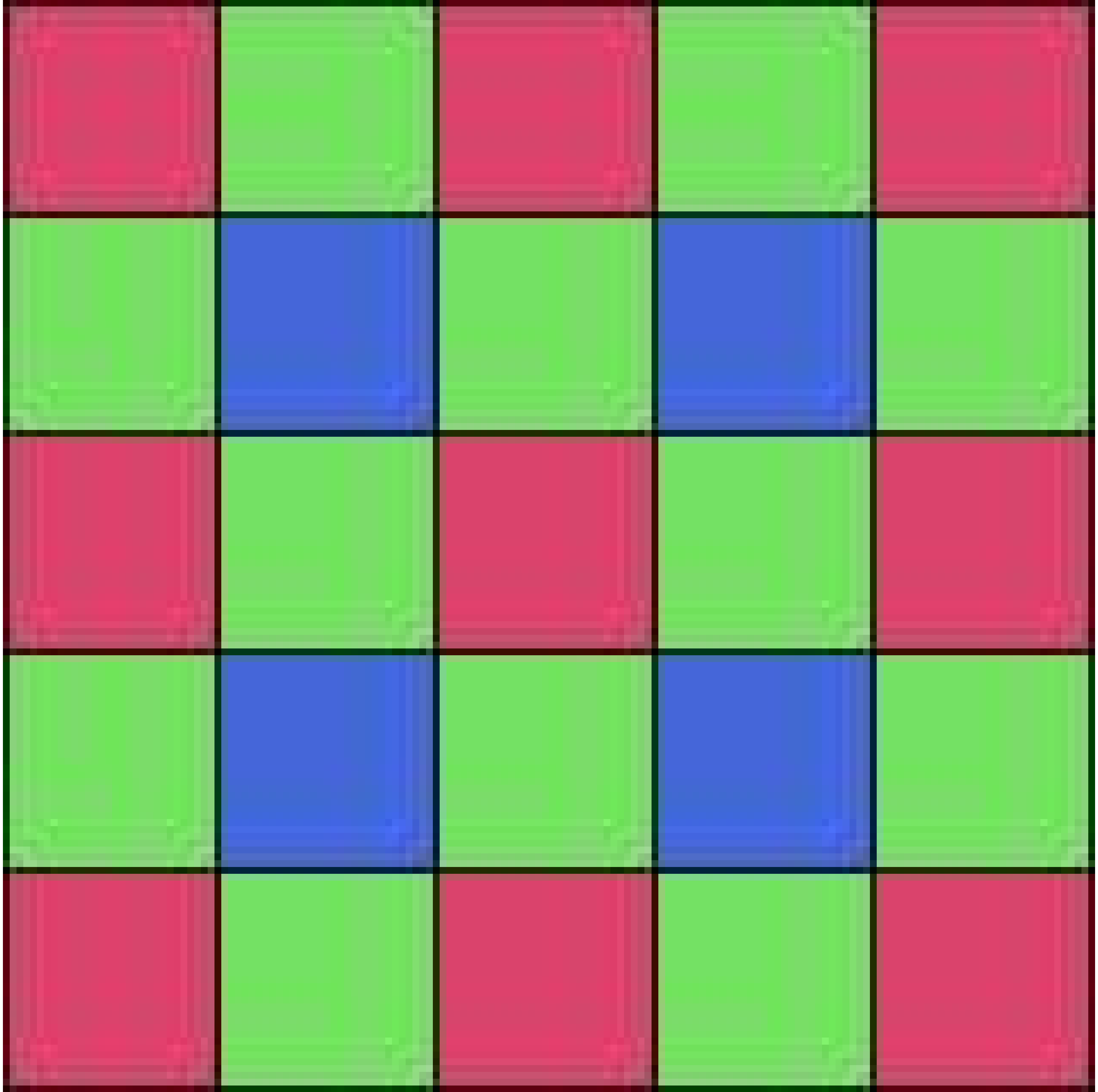





CCD
filter
CCD
CCD



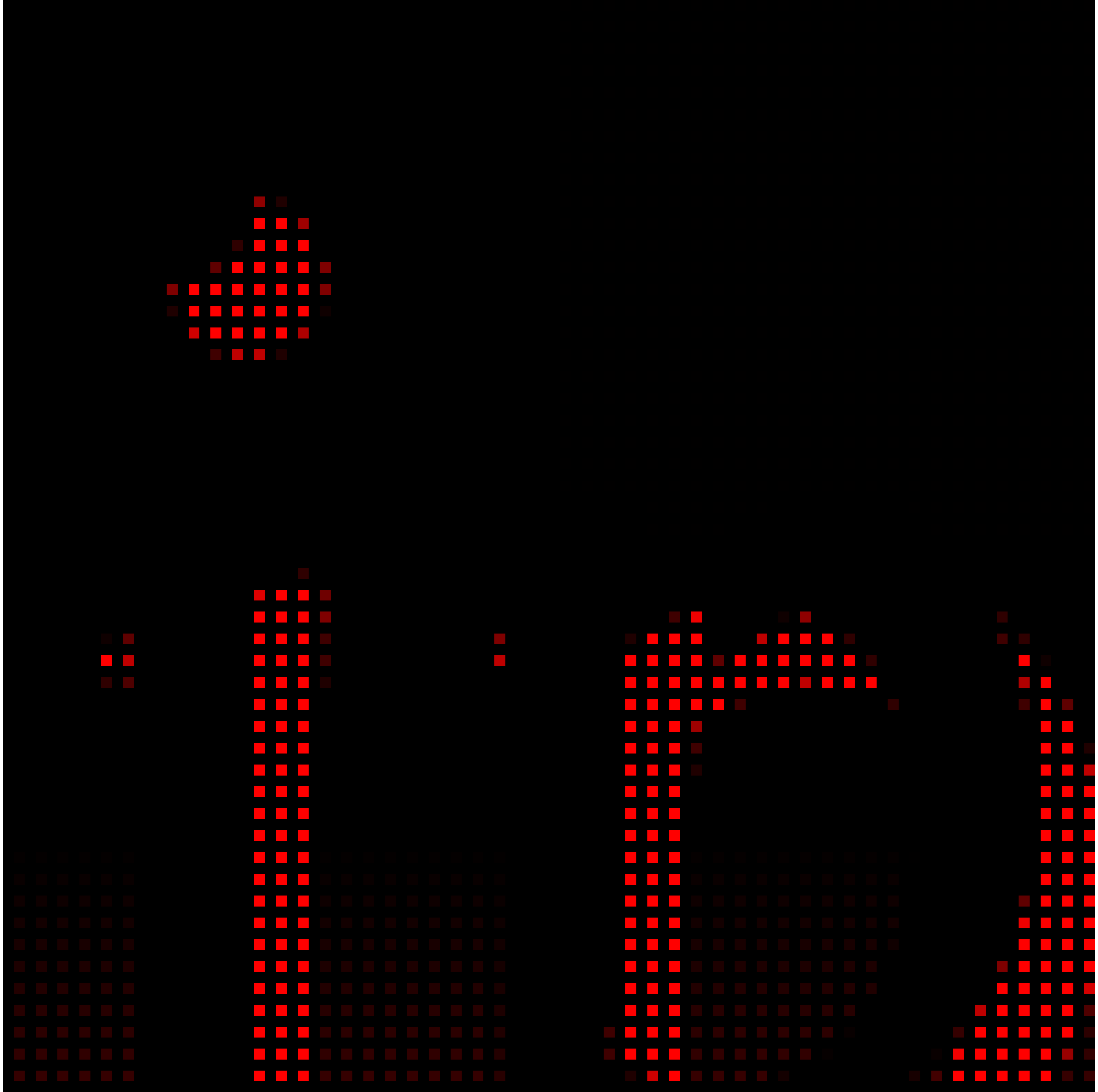
F2
B
C
A
F1



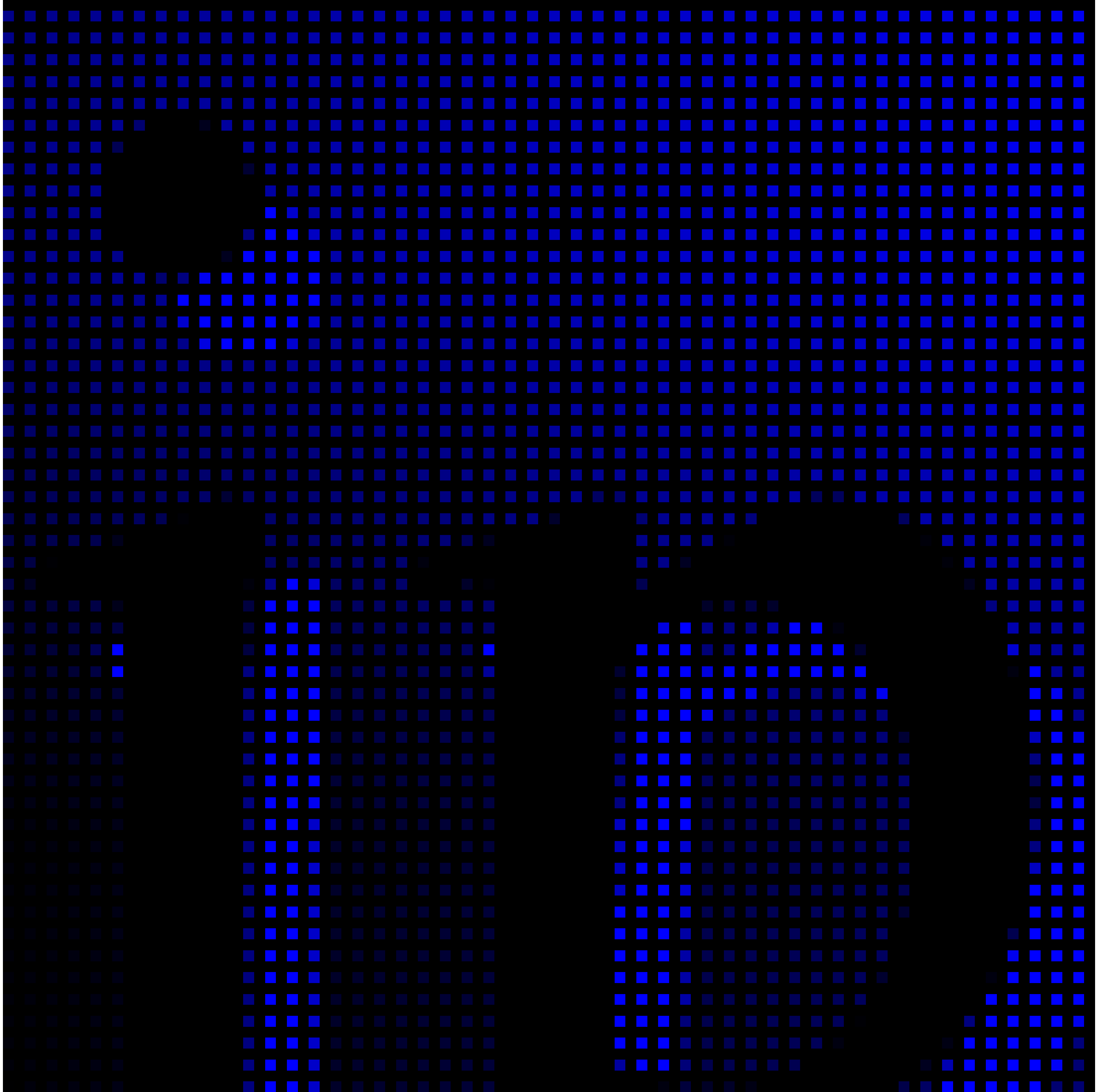
Wikipedia

Wikipedia

in

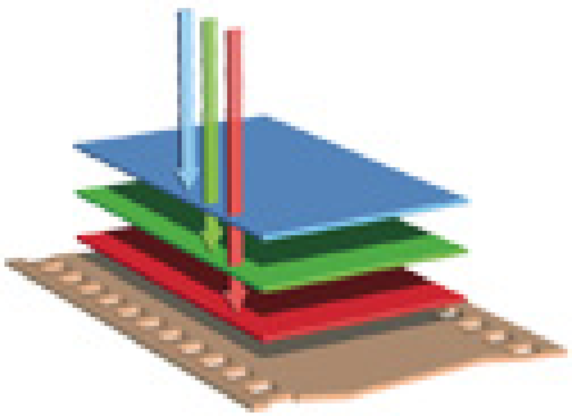


in



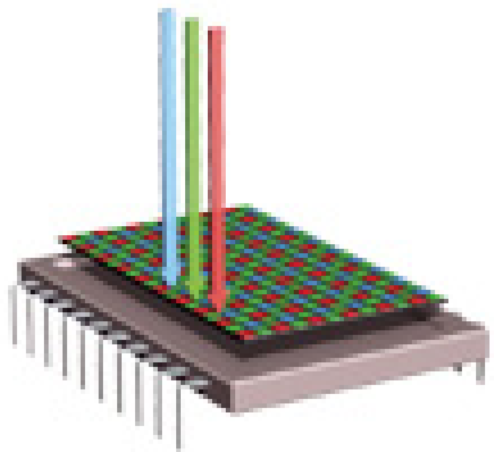
in

ip



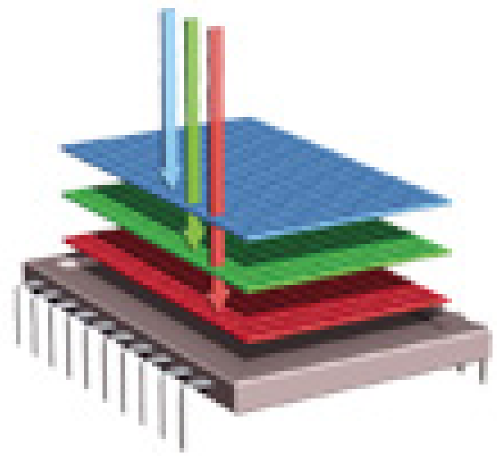
First came film.

COLOR FILM contains three layers of emulsion which directly record red, green, and blue light.



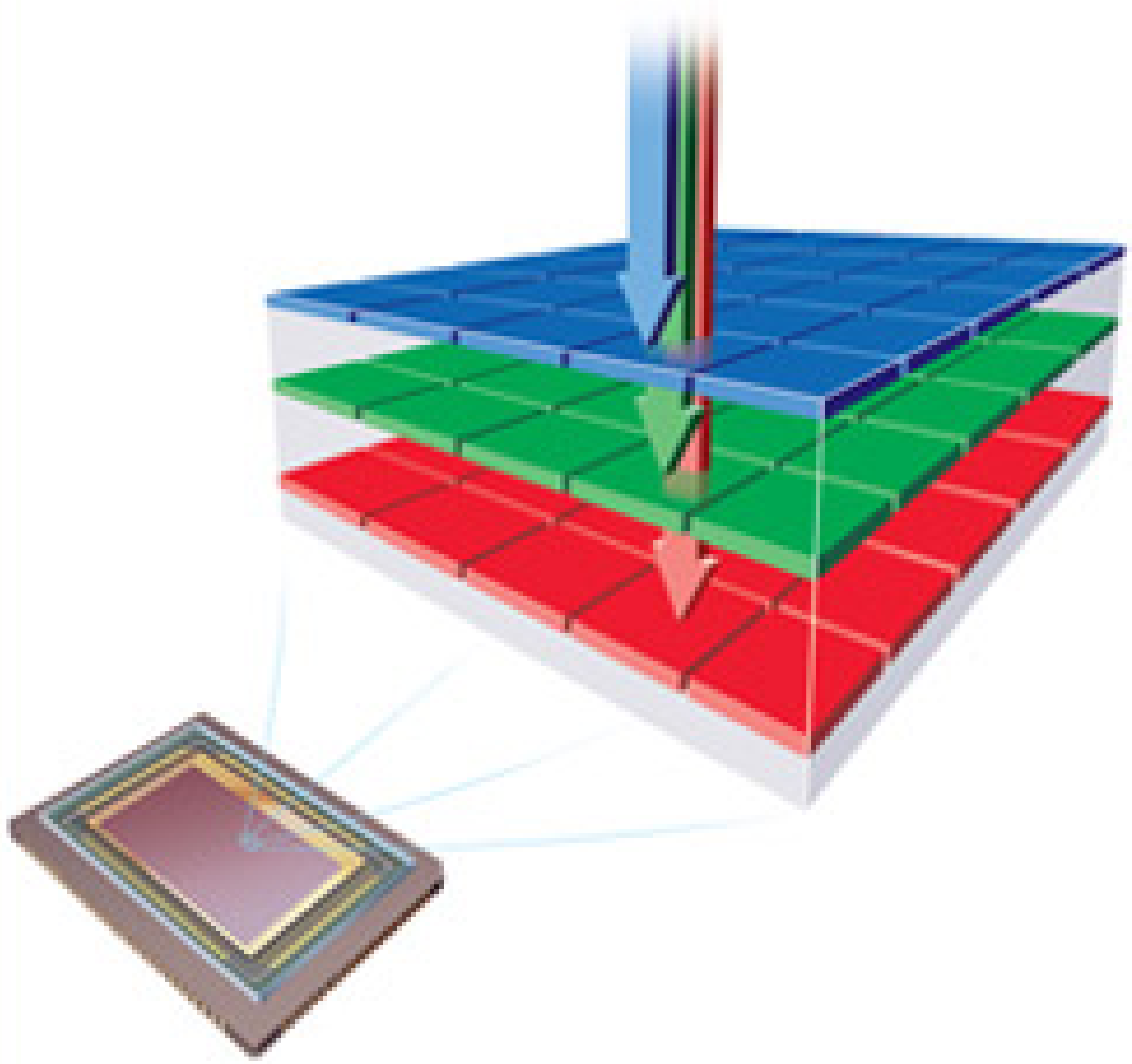
Then came digital.

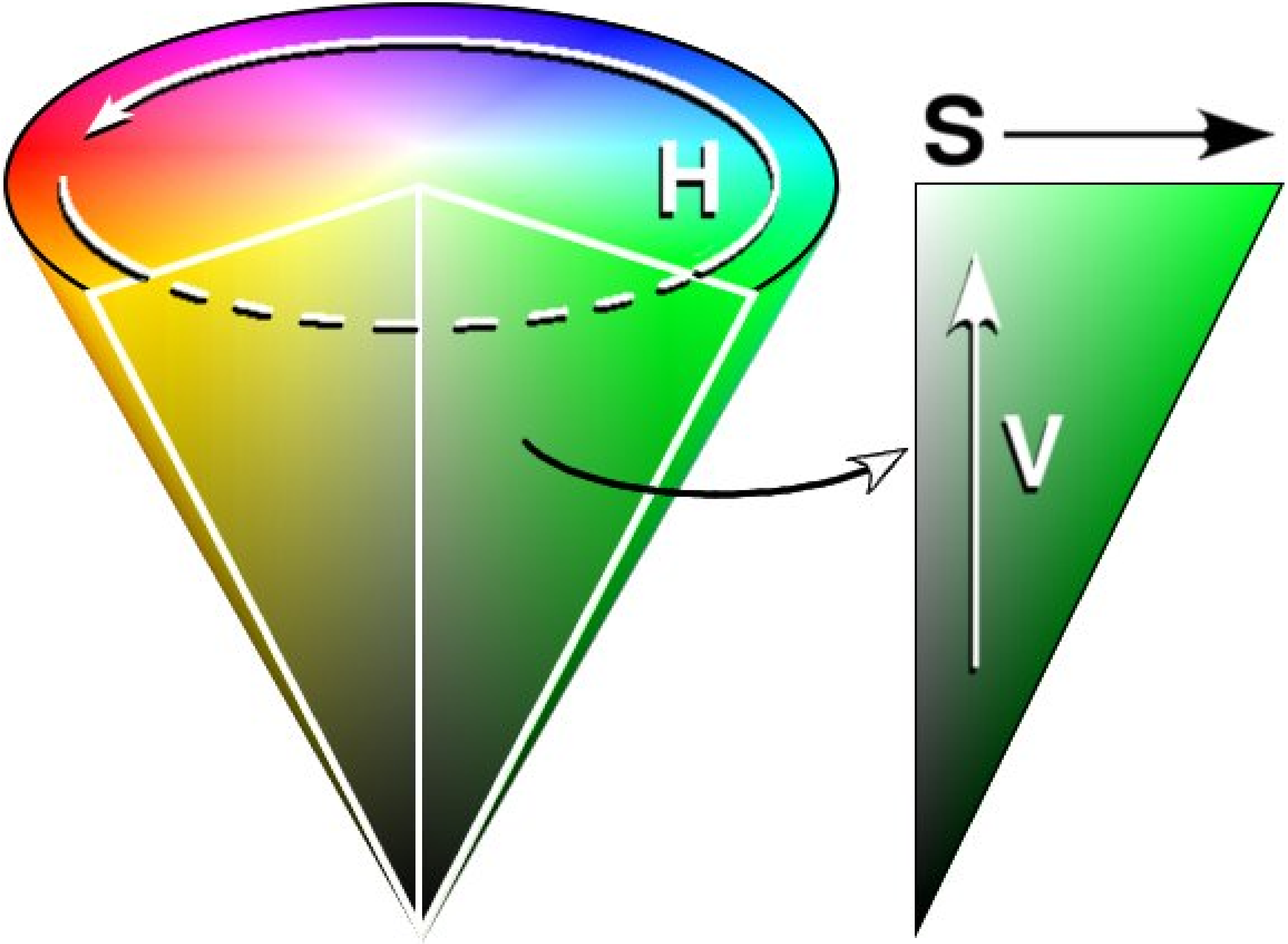
TYPICAL DIGITAL SENSORS have just one layer of pixels and capture only <u>part</u> of the color.



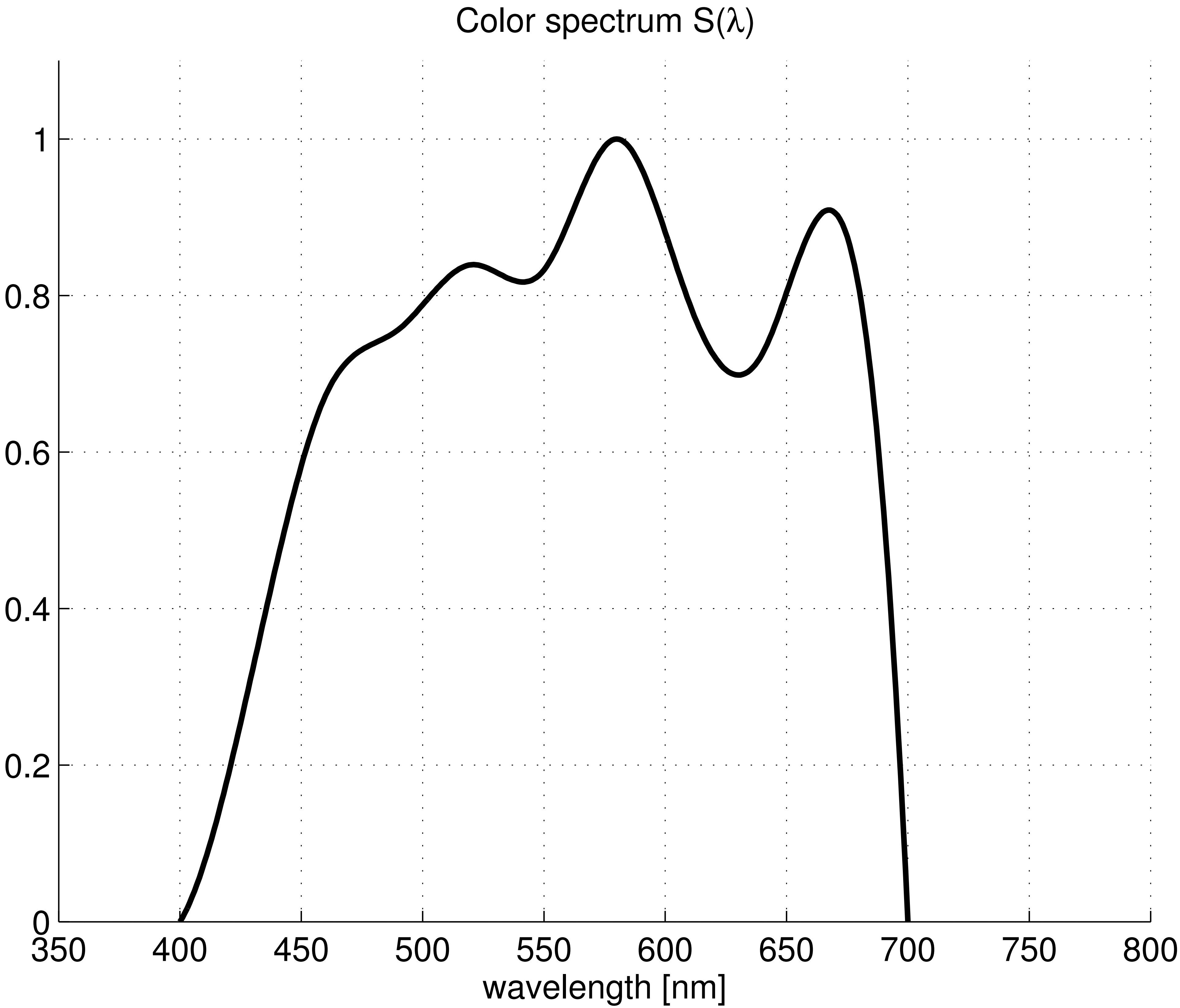
Now there's Foveon X3.

FOVEON X3 direct image sensors have three layers of pixels which directly capture <u>all</u> of the color.

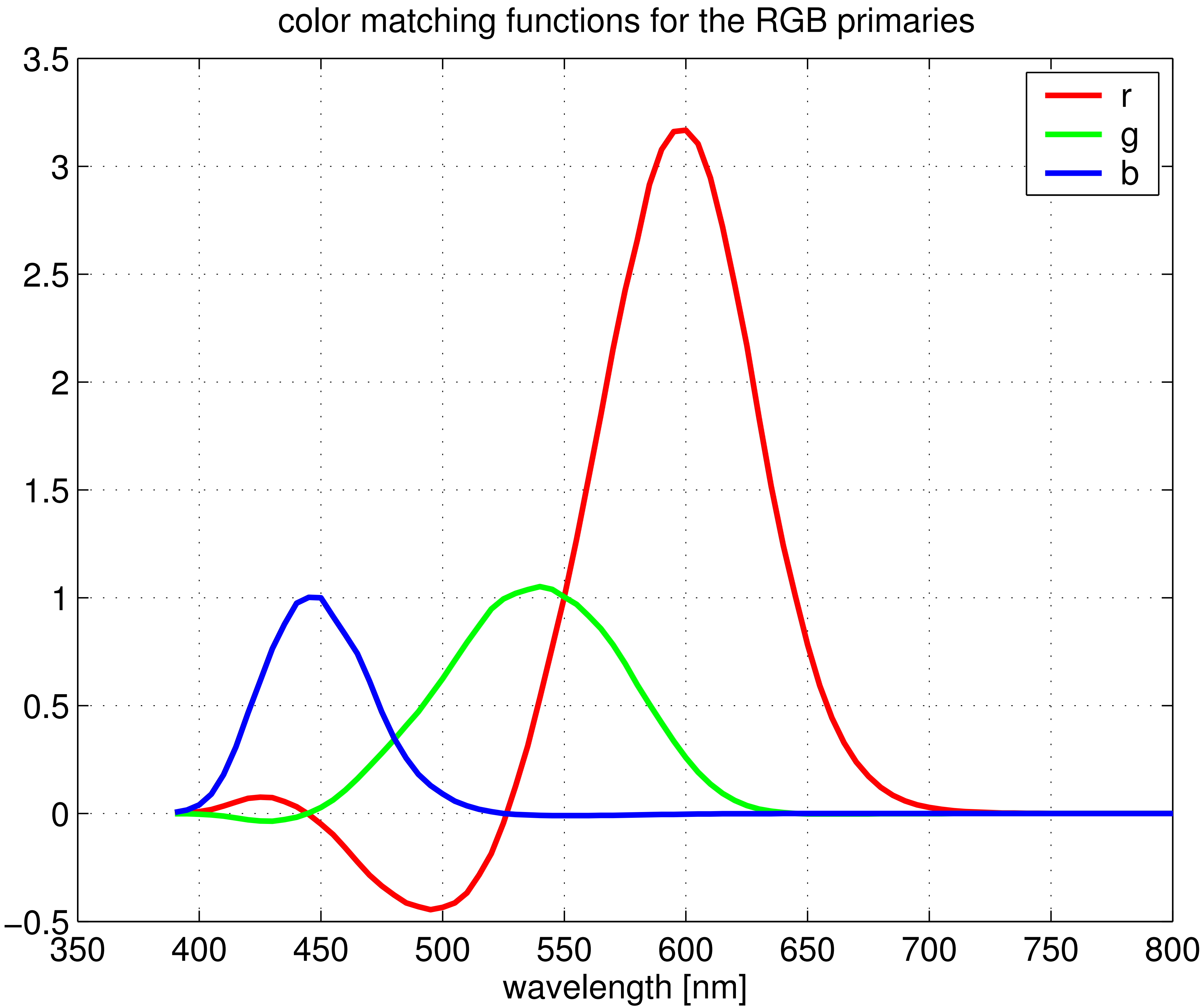


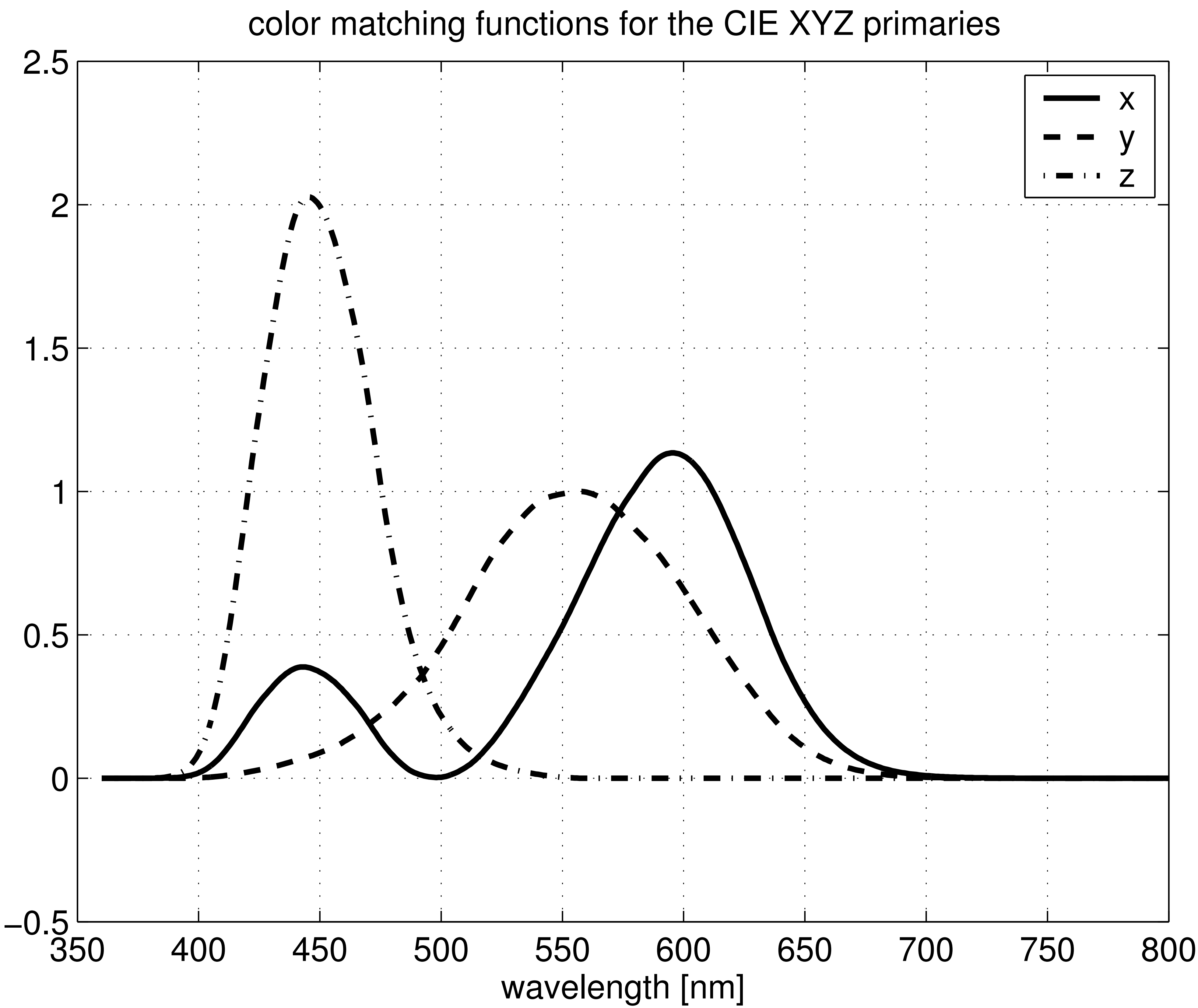


H
S
V

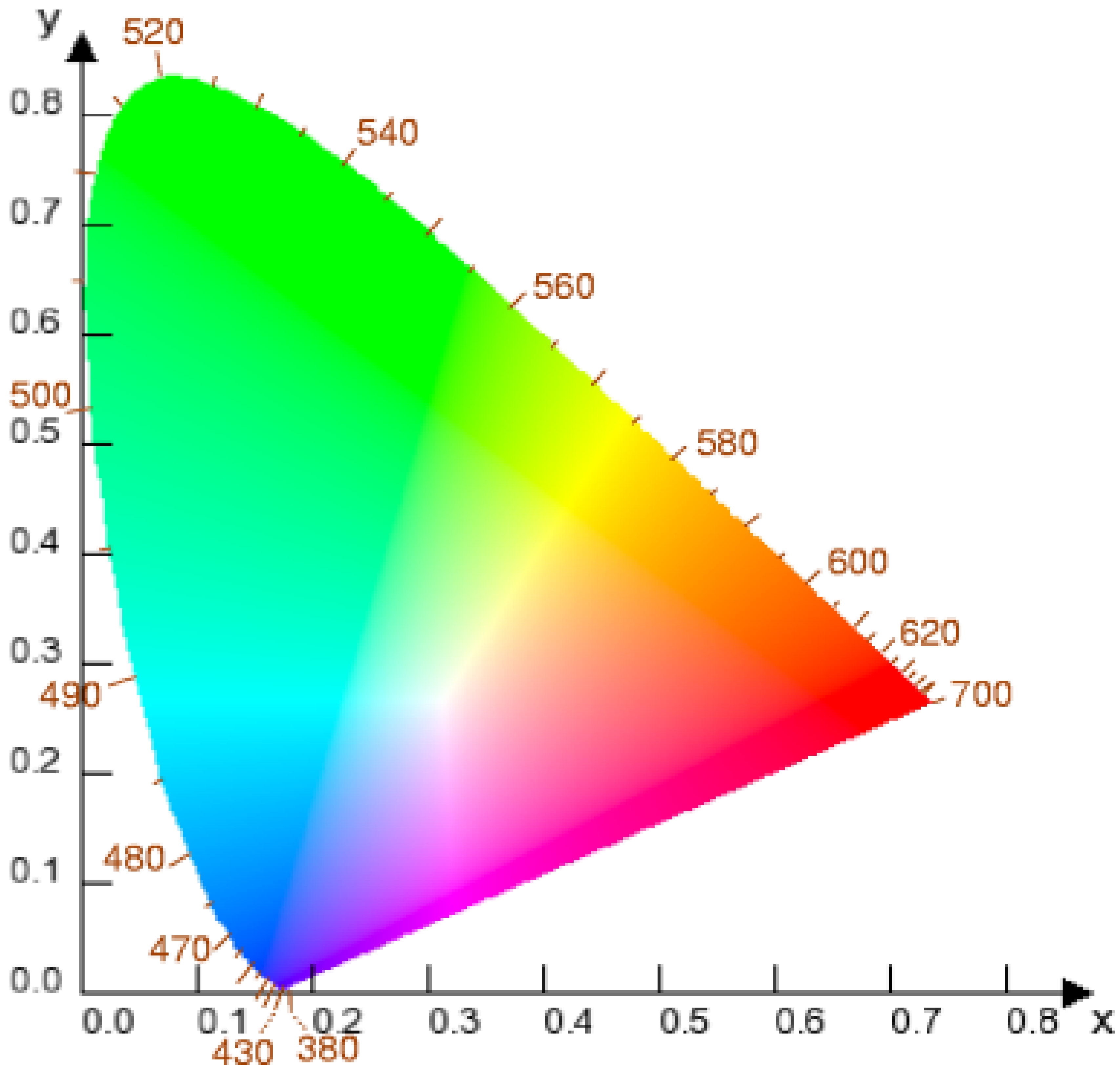


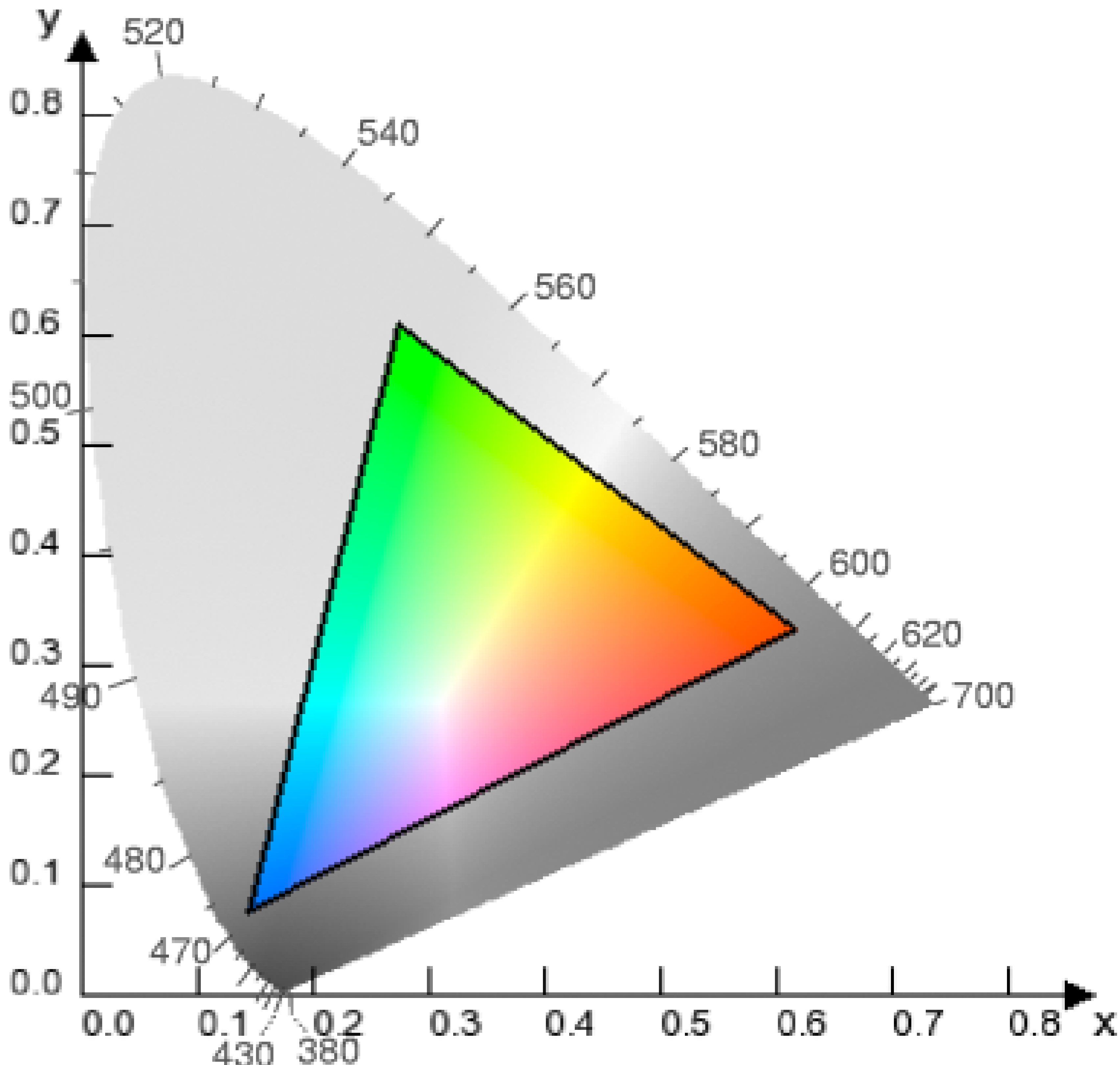
Color spectrum S(λ)
1
0.8
0.6
0.4
0.2
0
350
400
450
500
550
600
650
700
750
800
wavelength [nm]


color matching functions for the RGB primaries
r
g
b
3.5
3
2.5
2
1.5
1
0.5
0
−0.5
350
400
450
500
550
600
650
700
750
800
wavelength [nm]



color matching functions for the CIE XYZ primaries
2.5
2
1.5
1
0.5
0
−0.5
350
400
450
500
550
600
650
700
750
800
wavelength [nm]
x
y
z


y
x
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
520
540
560
580
600
620
700
500
490
480
470
430
380


y
520
540
560
580
600
620
700
500
490
480
470
430
380
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8
x