

# Mathematica para entender la aritmética del color

P. E. Moreira Galván

Tecnológico Nacional de México Campus Querétaro,  
Av Tecnológico S/N, Centro Histórico, Centro, 76000 Santiago de Querétaro, Qro.  
Universidad Anáhuac Querétaro,  
Cto. Universidades I, Fracción 2, 76246 Qro.

Received 9 November 2023; accepted 9 February 2024

El trabajo propone una manera visual de entender la aritmética del color usando el software Mathematica y las leyes de Grassmann. El uso de Mathematica permite visualizar estas leyes que de otra manera sería difícil. La representación vectorial de las leyes de Grassmann es una herramienta valiosa para la docencia y el aprendizaje de la teoría del color, debido a que permite a los estudiantes comprender cómo funciona la aritmética del color de una manera intuitiva.

La metodología propuesta utiliza Mathematica para crear una representación visual de las leyes de Grassmann. Comprender los conceptos matemáticos involucrados, permite que mediante el uso de software como Mathematica puede hacer que la aritmética del color sea más fácil de entender y usar.

*Descriptores:* Mezcla de colores; solución simbólica; Mathematica; aritmética cromática.

The work proposes a visual way to understand color arithmetic using the Mathematica software and the Grassmann laws. The use of Mathematica allows to visualize these laws in a way that would be difficult to do otherwise. The vector representation of the Grassmann laws is a valuable tool for teaching and learning color theory, as it allows students to understand how color arithmetic works in an intuitive way.

The proposed methodology uses Mathematica to create a visual representation of the Grassmann laws. Understanding the mathematical concepts involved allows the use of software such as Mathematica to make color arithmetic easier to understand and use.

*Keywords:* Mixture of colors; solution symbolic; Mathematica; chromatic arithmetic.

DOI: <https://doi.org/10.31349/RevMexFis.21.020218>

## 1. Introducción

Siguiendo las ideas mostradas en Ref. [1] en donde se muestra de manera experimental los colores resultantes de la mezcla de colores Rojo, Verde y Azul, y los no usuales Cian, Magenta y Amarillo. Se muestra mediante el software Mathematica (o en su modalidad libre WolframCloud) una manera digital de hacer combinaciones de colores y sus posibles aplicaciones, como se exhibe en Ref. [2]. Este trabajo busca relacionar las leyes de Grassman que describe de manera empírica el comportamiento de las mezclas de colores y los vectores usuales en el espacio tridimensional, permitiendo conectar los conceptos de manera práctica y visual, además de proponer y/o resolver problemas de la vida real.

La importancia de incorporar herramientas digitales es descrita en numerosos trabajos [3–6], entre los que destacan:

1. La tecnología permite visualizar un problema, lo cual permite un mejor entendimiento de este.
2. Se puede relacionar los conceptos vistos en clase en problemas contextualizados.
3. Estimula la curiosidad e interés de los estudiantes.

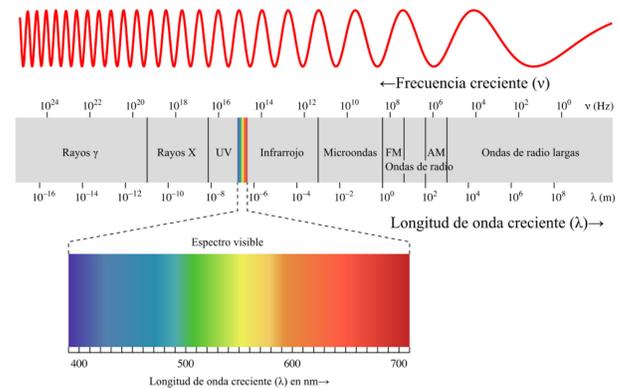


FIGURE 1. Longitud de ondas electromagnéticas.

## 2. Teoría

Mediante la teoría electromagnética se conoce que cada color es una onda de longitud determinada, las longitudes de ondas visibles oscilan entre 380 nm y 780 nm [7], aunque se sabe que existen más longitudes de onda imperceptibles para el ojo humano como se muestra en la Fig. 1.

## 3. Leyes de Grassmann

Grassmann enunció sus leyes en el siglo XIX, sintetizó la teoría de la mezcla aditiva del color, las cuales pueden ser consultadas en Refs. [7, 8].

### 1. Ley trivariación visual

Por síntesis aditiva del color es posible conseguir todos los colores percibidos mezclando tres franjas del espectro visible, siempre y cuando ninguno de los tres, pueda obtenerse de los otros dos.

Por ejemplo si usamos el sistema RGB (rojo, verde y azul), cualquier color puede ser obtenido como combinación de estos.

$$C = \alpha R + \beta G + \gamma B,$$

donde  $\alpha, \beta, \gamma \in \mathbb{R}$ .

### 2. Ley de aditividad

Al mezclar aditivamente dos radiaciones cualesquiera se forma un color que puede ser generado por síntesis aditiva de los colores primarios.

Es decir, si consideramos el sistema de colores descrito en 1, como  $e_1, e_2, e_3$  y  $C$  resulta de mezclar dos colores  $C_1, C_2$ . Entonces

$$C_1 = \alpha_1 e_1 + \beta_1 e_2 + \gamma_1 e_3$$

$$C_2 = \alpha_2 e_1 + \beta_2 e_2 + \gamma_2 e_3$$

Se tiene

$$C = (\alpha_1 + \alpha_2)e_1 + (\beta_1 + \beta_2)e_2 + (\gamma_1 + \gamma_2)e_3$$

### 3. Ley de proporcionalidad

Siempre que dos superficies produzcan igual sensación cromática es posible variar su luminancia manteniendo constante el matiz y la saturación, sin que altere la igualdad cromática, es decir: si  $C_1 = C_2$ . Entonces  $kC_1 = kC_2$ .

Estas leyes de Grassman pueden ser asociadas a vectores en el espacio, pues tenemos lo siguiente:

**Ley de trivariación**  $\longleftrightarrow$  Dimensión de espacio vectorial es 3.

**Ley de aditividad**  $\longleftrightarrow$  Suma de vectores.

**Ley de proporcionalidad**  $\longleftrightarrow$  Multiplicación de vectores por un escalar.

Por lo cual de manera natural identificamos el espacio de color con el espacio vectorial  $\mathbb{R}^3$  con suma y multiplicación por escalar usual. Para el desarrollo de este trabajo se considera lo siguiente:

- Sea  $S = \{e_1, e_2, e_3\}$  el conjunto descrito en 1, entonces se hará la siguiente asociación,

$$e_1 = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \quad e_2 = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \quad e_3 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}. \quad (1)$$

- En caso de operar con los vectores y obtener un vector

$$c = \begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix}$$

con algún  $a_i > 1$  (o  $a_i < 0$ ), entonces se considera  $a_i = 1$  (o  $a_i = 0$ ), es decir solo se consideran vectores con entradas con valores entre 0 y 1.

- Existen diferentes maneras de obtener colores como *mezcla aditiva* y *mezcla sustractiva*, las cuales tienen por colores básicos *Rojo*, *Verde* y *Azul* y el *Cian*, *Magenta* y *Amarillo* denotados por *RGB* y *CMY* respectivamente por sus siglas en inglés, aunque existen más sistemas de colores como *HUE*, *XYZ*, *CIELAB* y *CIELUV* (ver Ref. [8]).

Los sistemas *RGB* y *CMY*, satisfacen lo siguiente:

$$\text{Rojo} + \text{Verde} + \text{Azul} = \text{Blanco},$$

$$\text{Cian} + \text{Magenta} + \text{Amarillo} = \text{Negro},$$

$$0 * \text{Rojo} + 0 * \text{Verde} + 0 * \text{Azul} = \text{Negro},$$

$$0 * \text{Cian} + 0 * \text{Magenta} + 0 * \text{Amarillo} = \text{Blanco},$$

lo cual puede ser verificado experimentalmente como se muestra en Ref. [1], esto muestra por que se usan en diferentes contextos los sistemas de colores, pues el *RGB* se usa principalmente en pantallas de televisores, mientras que el *CMY* en impresión pues el papel donde se imprime es usualmente blanco y la tipografía es de color negro, aunque es posible obtener este color mediante el cian, magenta y amarillo, resulta muy costoso, es por esta razón que se agrega un cartucho negro a este sistema obteniendo *CMYK*, donde *K* denota el color negro.

De ahora en adelante solo trabajaremos con imágenes *.jpg* y el software *Mathematica*, que puede ser usado en su forma gratuita y en línea *WolframCloud*.

## 4. Sistema RGB

Consideremos el sistema de colores *RGB*, entonces

$$r = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \quad v = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \quad a = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}, \quad (2)$$

representan los colores rojo, verde y azul respectivamente. Los cuales se define en *Mathematica* como sigue:

$$\begin{aligned} r &= \{\{\{1, 0, 0\}\}\} \\ v &= \{\{\{0, 1, 0\}\}\} \\ a &= \{\{\{0, 0, 1\}\}\} \end{aligned}$$

Para tener la imagen que representa estos se usa el comando *Image*, especificando el sistema de colores a usar. El siguiente comando

```
Image[r, ColorSpace -> "RGB"]
Image[v, ColorSpace -> "RGB"]
Image[a, ColorSpace -> "RGB"]
```

produce lo siguiente:



Siguiendo lo descrito en Ref. [1], podemos obtener los colores *Cian* (*c*), *Magenta* (*m*) y *Amarillo* (*y*), estos se pueden obtener a partir de la resta del color *Blanco* (*b*):

$$c = b - r = a + v = \begin{pmatrix} 0 \\ 1 \\ 1 \end{pmatrix}, \quad (3)$$

$$m = b - v = r + a = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}, \quad (4)$$

$$y = b - a = r + v = \begin{pmatrix} 1 \\ 1 \\ 0 \end{pmatrix}, \quad (5)$$

los cuales usando el comando *Image* en cada una obtenemos los siguientes colores:



Con lo cual podemos hacer predicciones al mezclar colores al realizar la aritmética de vectores. Por ejemplo si mezclamos *Cian*, *Magenta* y *Amarillo* en la misma proporción tenemos lo siguiente:

$$c + m + y = (b - r) + (b - v) + (b - a), \quad (6)$$

$$= 3b - (r + v + a), \quad (7)$$

$$\equiv b - (r + v + a), \quad (8)$$

$$= b - b, \quad (9)$$

$$= 0, \quad (10)$$

lo cual significa que no hay reflejo de luz, es decir tenemos el color negro, como se puede observar experimentalmente en Ref. [1]. Observar que para pasar de (7) a (8) usamos que, al mezclar color blanco obtenemos el mismo color blanco, que fue la segunda consideración que hicimos.

```
In[605]:= ImageData[datos de imag] // MatrixForm
[forma de matriz]
```

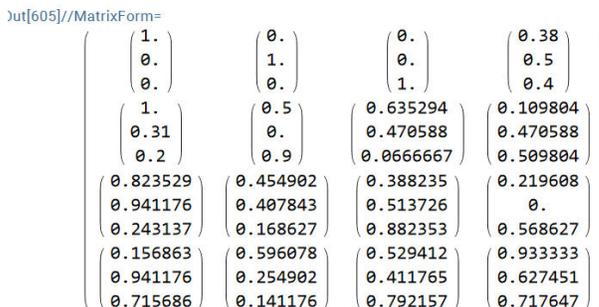


FIGURE 2. Representación matricial de imagen.

### 4.1. Aplicación

Entendiendo como se representa una imagen de manera matricial, podemos enfocarnos en ciertas características que nos interesen. El comando *ImageData*[imagen] tiene por argumento la imagen de la cual deseamos encontrar su representación matricial, como se muestra en la Fig. 2.

Se usó el comando *MatrixForm*, para que la salida sea en forma matricial usual, en caso de no ponerlo nos lo regresa en forma de lista. Así cada entrada de la matriz representa el color de la imagen, por ejemplo el vector que está en la posición  $a_{43}$ , representa el color lila de la imagen. Esto nos permite buscar ciertas características que nos interesan, por ejemplo, si alguna manzana está madura, pues el color que buscamos es la cantidad de color rojo en la imagen, es decir solo estamos interesados en la primera componente de cada color, para lo cual usaremos el producto punto que tiene predeterminado Mathematica,  $\{x, y\} \cdot \{\{a, b\}, \{c, d\}\} = \{ax + cy, bx + dy\}$ , el cual hace el producto punto de la primera lista con cada una de las listas. Entonces podemos obtener la proyección sobre el color rojo como sigue:

```
in[{{c1, c2, c3}}].{{1, 0, 0}, {0, 0, 0}, {0, 0, 0}}
out[{{c1, 0, 0}}].
```

De manera análoga se puede obtener para el color azul y verde. Como aplicación de lo visto, consideremos fotos de manzanas, la primera es una manzana verde mientras que la segunda es una manzana ya madura.

Para determinar que parte de la fruta falta de madurar, nos debemos de fijar en la cantidad de rojo de la imagen,

```
matrizmanzana=ImageData[manzana];
matrizmanzanaraja=ImageData[manzanaraja];
rojo = {{1, 0, 0}, {0, 0, 0}, {0, 0, 0}}.
```

Los primeros dos comandos nos definen la matriz que representan a las Figs. 3 y 4, mientras que la tercera línea nos pro-



FIGURE 3. Manzana verde.



FIGURE 4. Manzana roja.

porcionara con ayuda de producto punto, la cantidad de rojo de cada una de las componentes.

```
r = matrizmanzana.rojo;
r1 = matrizmanzanaraja.rojo;
```

Con estos comandos obtenemos la cantidad de rojo de cada una de las manzanas.

```
{Image[matrizmanzana-r,
      ColorSpace->"RGB"],
```

```
Image[matrizmanzanaraja-r1,
      ColorSpace->"RGB"]},
```

como hicimos la resta de la matriz original y la que contiene solo rojo, obtendremos como imagen el resaltado de los lugares donde no esta madura la manzana como se muestra en la Fig. 5.



FIGURE 5. Resultado de partes verdes.



FIGURE 6. Imagen a resaltar el color azul.

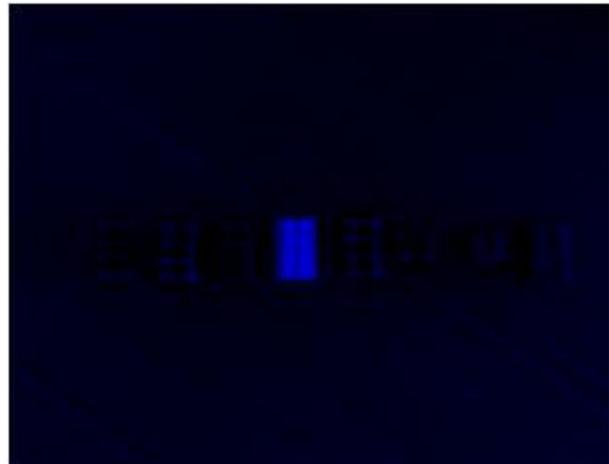


FIGURE 7. Imagen resaltando el color azul.

De manera similar podemos escoger un color que queremos resaltar, por ejemplo en la Fig. 6, solo queremos el color azul, por lo cual de manera similar definimos lo siguiente

$$\text{azul} = \{\{0,0,0\}, \{0,1,0\}, \{0,0,0\}\}.$$

Con el siguiente código:

```
matrizlego = ImageData[lego];
legoazul = Image[(matrizlego.azul),
                 ColorSpace->"RGB"].
```

Obtenemos la siguiente imagen (ver Fig. 7), mientras que si definimos lo siguiente:

```
magenta = \{\{0,0,0\}, \{0,0,0\}, \{1,0,1\}\};
legomagenta = Image[(matrizlego.azul)
                    .magenta,ColorSpace->"RGB"].
```

Obtendremos que todo lo azul cambia por color magenta, produciendo lo mostrado en la Fig. 8.

De manera similar podemos intercambiar colores a *cian*, *magenta* y *amarillo* usando las Ecs. (3)-(5).

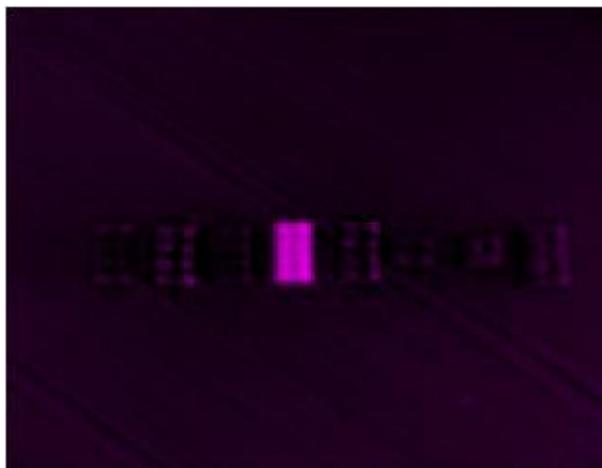


FIGURE 8. Imagen resaltando el color magenta.

## 5. Conclusión

Con ayuda del software Mathematica y entendiendo la aritmética de los colores, se desarrolló una manera sencilla de trabajar con imágenes digitales, relacionando las leyes de Grassmann con conceptos de álgebra lineal. Esto permitió lograr transformaciones sencillas pero interesantes para los estudiantes, como: sumar colores, invertir colores, obtener una tonalidad deseada, esta metodología es una herramienta útil para la enseñanza y el aprendizaje de la teoría del color, debido a que es sencilla de entender y aplicar.

## Agradecimientos

El autor quiere agradecer al SNI por el apoyo dado con la beca 336528 y también a la Universidad Anáhuac Querétaro, por las facilidades dadas para hacer el artículo.

- 
1. J. Medina Márquez, S.E. Balderas-Mata, y C. Zúñiga Islas, Understanding the arithmetic of color. *Rev. Mex. Fis. E*, **61** (2015) 28.
  2. G. P. E. Moreira y D.G. Arroyo, Procesamiento de Imágenes Digitales y Fundamentos de Teoría del Color para Enseñanza del Álgebra Lineal (Especialidad). CIIDET. (2019)
  3. M. Artigue, The Future of Teaching and Learning Mathematics with Digital Technologies. In C. Hoyles and J.-B. Lagrange, eds., *Mathematics Education and Technology-Rethinking the Terrain: The 17th ICMI Study* (Springer US, Boston, MA, 2010) pp. 463-475, [https://doi.org/10.1007/978-1-4419-0146-0\\_23](https://doi.org/10.1007/978-1-4419-0146-0_23)
  4. M. A. S. Parrot y K. E. Leong, Impact of Using Graphing Calculator in Problem Solving, *Int. Electron. J. Math. Educ.* **13** (2018) 139, <https://doi.org/10.12973/iejme/2704>.
  5. C. M. R. Caridade, R. Pedro, y N. Q. Nora, (s.f.). Applying image processing techniques to motivate students in linear algebra classes.
  6. R. M. Zbiek, M. K. Heid, y B. G. W. Research on Technology in Mathematics Education. In *Second Handbook of Research on Mathematics Teaching and Learning* (pp. 1169-1207), (2007).
  7. Zamora, F. G. O. Procesamiento morfológico de las imágenes en color. Aplicación a la reconstrucción geodésica (Disertación Doctoral). Universidad de Alicante, (2002).
  8. H.-C. Lee, *Introduction to Color Imaging Science* (Cambridge University Press, 2005), <https://doi.org/10.1017/CBO9780511614392>.