

## Ejercicio Opcional: Daltonización de Imágenes

### Introducción:

*Daltonizer* es un programa desarrollado en el lenguaje de programación *Processing*, como parte de la asignatura de Técnicas Gráficas 2010.

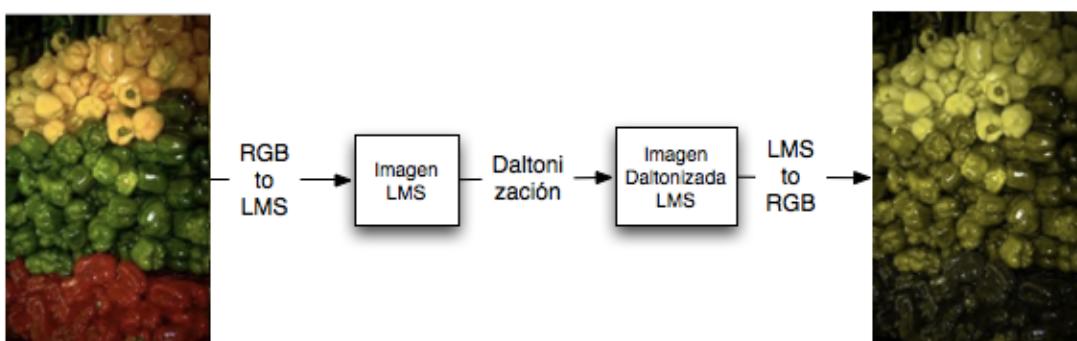
*Daltonizer* tiene por objetivo final la asistencia y ayuda a los diferentes tipos de daltónicos mediante el resaltado de la información que sus ojos no son capaces de captar.

El proceso se divide en tres etapas secuenciales: daltonización, cálculo del error y corrección del mismo.

### Daltonización:

Es el proceso de conversión que tomando una imagen cualquiera la transforma en una imagen tal y como la vería un daltónico. Habiendo distintos tipos de daltonismo serán necesarios distintos algoritmos.

El proceso es el siguiente:



1. Partimos de una imagen cualquiera en el espacio de colores RGB
2. La convertimos al espacio de colores LMS, dado que este representa fielmente la visión del ojo humano (al contrario que RGB), haciéndolo mucho más adecuado para cualquier algoritmo de daltonización. Para ello aplicamos una matriz de conversión pixel a pixel.

$$\begin{pmatrix} L \\ M \\ S \end{pmatrix} = ( \text{RGB\_to\_LMS} ) \begin{pmatrix} R \\ G \\ B \end{pmatrix} = \begin{pmatrix} 17.8824 & 43.5161 & 4.11935 \\ 3.45565 & 27.1554 & 3.86714 \\ 0.0299566 & 0.184309 & 1.46709 \end{pmatrix} \times \begin{pmatrix} R \\ G \\ B \end{pmatrix}$$

3. Le aplicamos el **algoritmo de daltonización** deseado, que no se trata más que de otra conversión matricial pixel a pixel. Disponemos de dos algoritmos de simulación distintos, para
  - a. protánnopes

$$\begin{pmatrix} L_p \\ M_p \\ S_p \end{pmatrix} = \begin{pmatrix} 0 & 2.02344 & -2.52581 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} L \\ M \\ S \end{pmatrix}$$

b. deuteránope

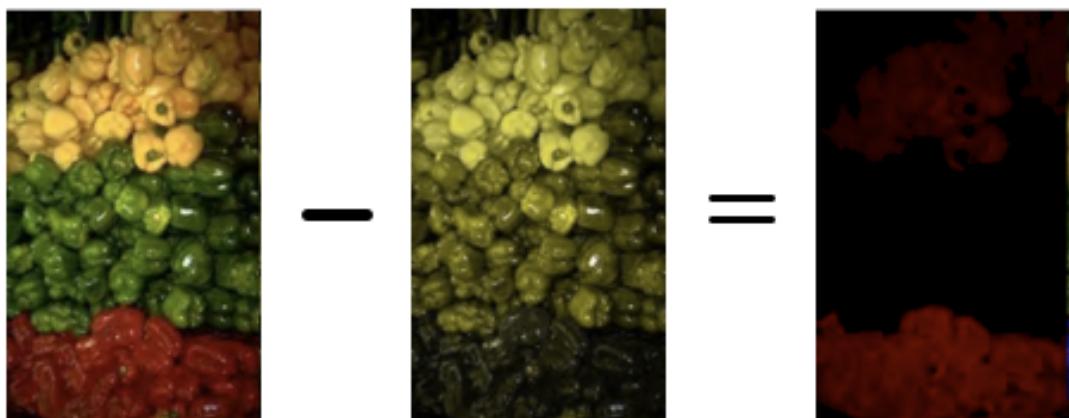
$$\begin{pmatrix} L_d \\ M_d \\ S_d \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 \\ 0.494207 & 0 & 1.24827 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} L \\ M \\ S \end{pmatrix}$$

4. y a continuación **volvemos al espacio RGB**:

$$\begin{pmatrix} R_d \\ G_d \\ B_d \end{pmatrix} = (\text{RGB\_to\_LMS})^{-1} \begin{pmatrix} L_d \\ M_d \\ S_d \end{pmatrix} = \begin{pmatrix} 0.080944 & -0.130504 & 0.116721 \\ -0.0102485 & 0.0540194 & -0.113615 \\ -0.000365294 & -0.00412163 & 0.693513 \end{pmatrix} \times \begin{pmatrix} L_d \\ M_d \\ S_d \end{pmatrix}$$

### Cálculo del error:

El cálculo del error consiste en realizar la resta pixel a pixel entre la imagen original y la imagen daltonizada para cada componente RGB. El resultado será una imagen con los colores (la información) que los ojos del daltónico no son capaces de ver.

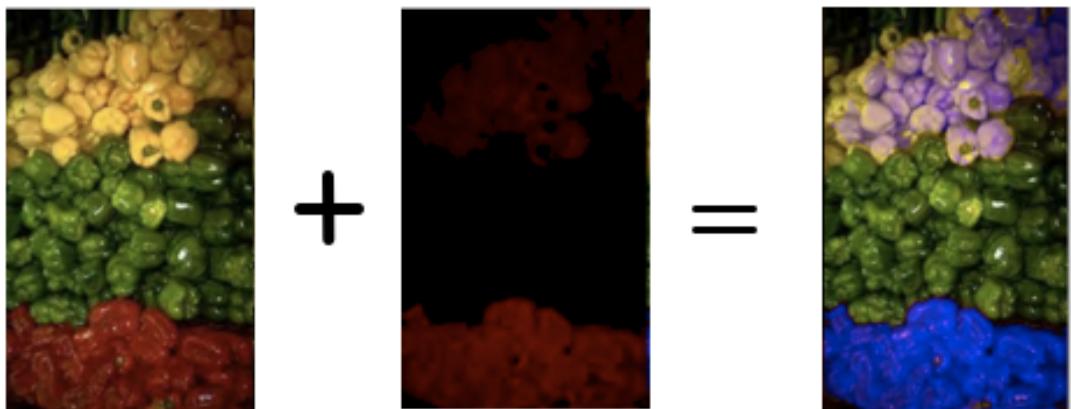


### Corrección del error:

La idea es conseguir que el daltónico sepa que esa información perdida está ahí y cual es. Evidentemente no podemos conseguir que un dicrómata vea del mismo modo que un tricrómata, pero si podemos ayudarle señalando donde se encuentra esa información que no es capaz de ver.

Supongamos que tratamos con un protánope. Este tipo de daltonismo tiene una cadencia de visión en la zona de los rojos, por tanto la imagen de error que obtendremos será mayormente roja. Para conseguir que nuestro protánope vea la información perdida podemos introducirla en una nueva imagen por el canal de los azules, donde no tiene ningún problema de visión.

Así si sumamos a la imagen original la imagen del error sólo por el canal de los azules:



### Pequeñas optimizaciones realizadas:

#### En el cálculo del error:

Después de haber calculado la imagen error, hayamos la media de intensidad para cada canal (rojo, verde y azul) y hacemos un filtrado. Todo pixel que no esté por encima de la media lo pintamos de negro.

Esto ayuda a afinar el cálculo de la imagen error en aquellas imágenes que sufren de una desviación hacia alguno de estos colores.

#### Ejemplo sin media:



#### Ejemplo con media:



**En la corrección del error:**

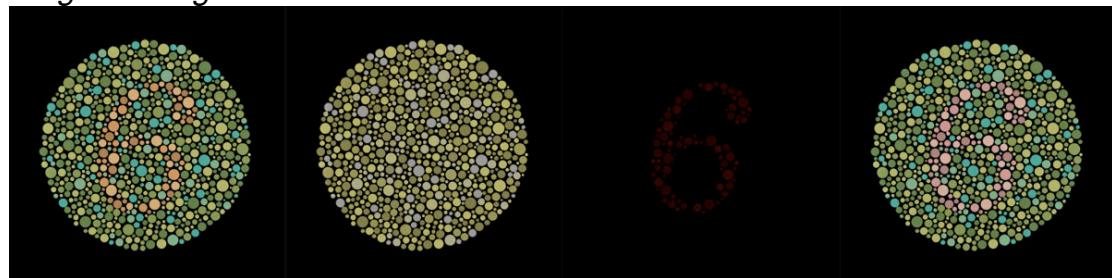
Resulta conveniente decrementar las componentes rojas y verdes proporcionalmente a lo que se incrementa la del azul, con el objetivo de enfatizar la diferencia.

También se puede incrementar la intensidad de la imagen error a la hora de sumarla con la misma intención.

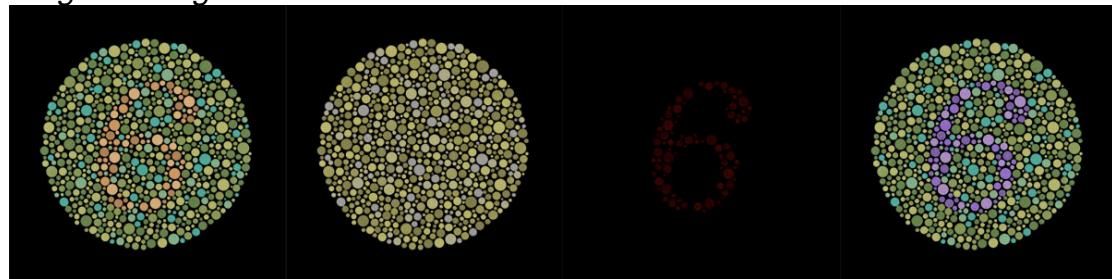
Recordemos que no pretendemos mostrar imágenes realistas sino resaltar la información perdida.

La diferencia es apreciable si cogemos por ejemplo una imagen típica de los tests de daltonismo:

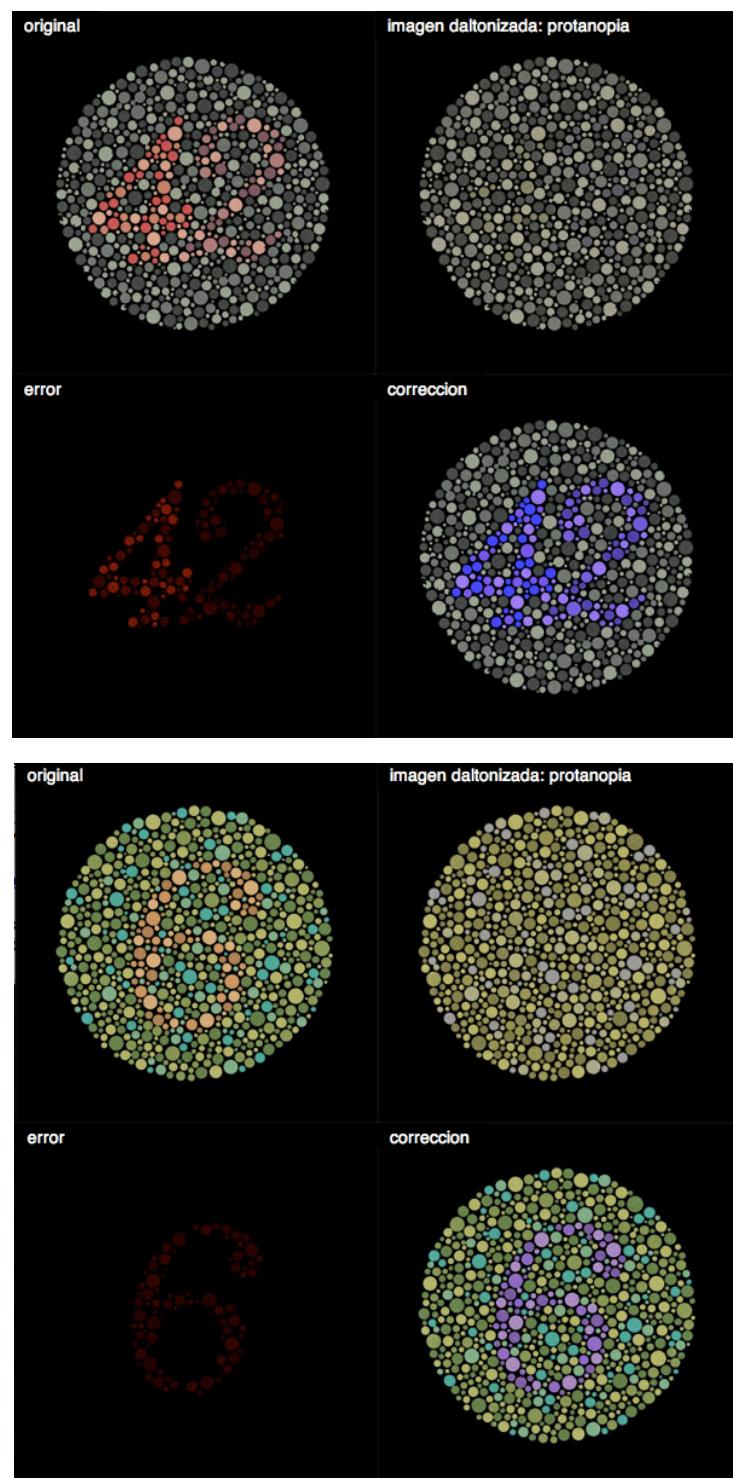
*Imagen corregida sin enfatizar:*

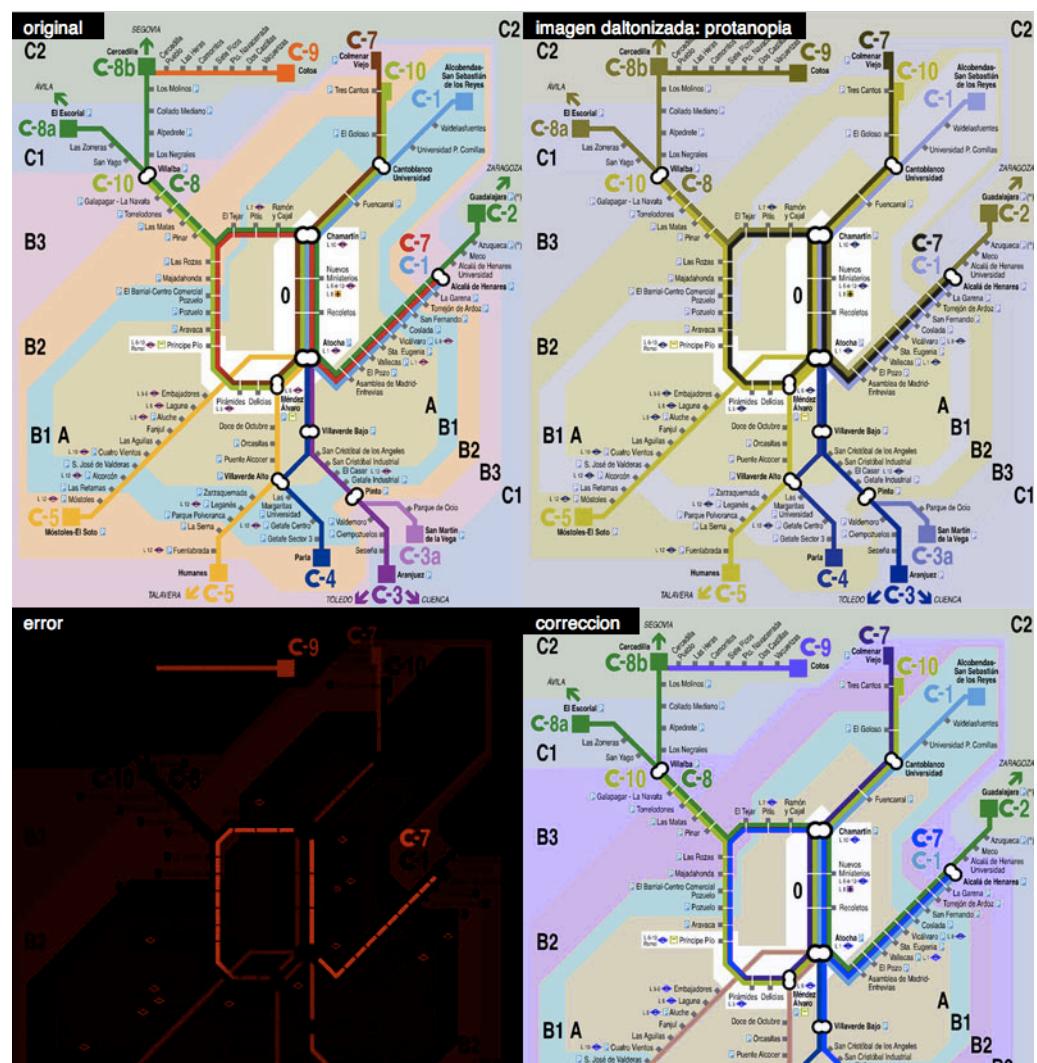
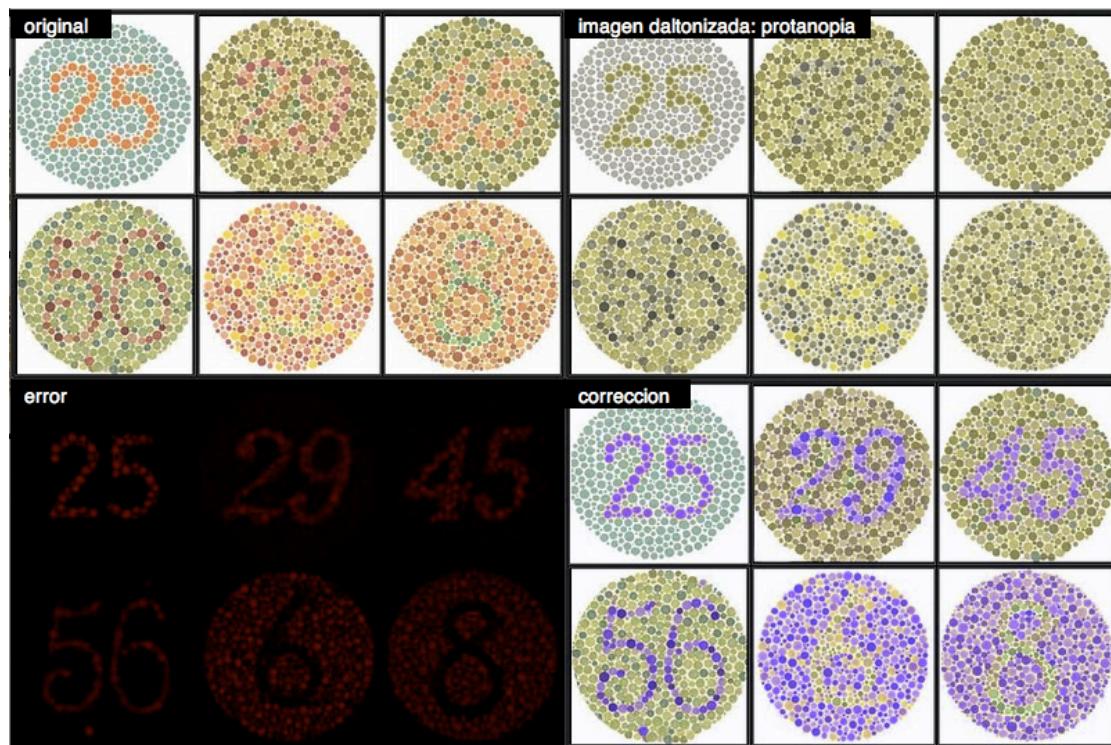


*Imagen corregida enfatizada:*



## Algunos ejemplos:





**Referencias:**

1. Onur Fidaner Poliang Lin Nevran Ozguven, Analysis of Color Blindness Project Writeup
2. Viénot F, Brettel H, Ott L, Ben M'Barek A, Mollon JD. What do colour-blind people see? *Nature* 1995;376:127–128.
3. Brettel H, Viénot F, Mollon JD. Computerized simulation of color appearance for dichromats. *J Opt Soc Am A* 1997;14:2647–2655.
4. Viénot F., Brettel H., and Mollon J. D., "Digital video colourmaps for checking the legibility of displays by dichromats", *Color Res. Application*, 24, pp243-252, 1999.  
<http://tsi.enst.fr/~brettel/colourmaps.html>, 1999.
5. Viénot F, Brettel H. Color display for dichromats. *Proc. SPIE* Vol. 4300 (2001) 199–207