

# *Ejemplo práctico de procesamiento de imágenes en color: efecto publicitario “rojo sobre grises”*

*Pablo Roncagliolo B.*

A continuación se presenta el desarrollo de un algoritmo de procesamiento de imágenes en color, con la finalidad de lograr un conocido efecto visual utilizado en la fotografía publicitaria que denominaremos “rojo sobre grises”.

En términos simples el efecto consiste en resaltar un objeto rojo (como por ejemplo una rosa), manteniendo su colorido original, por sobre el resto de la imagen transformada a escala de grises. De esta manera toda la imagen queda en gris excepto el objeto seleccionado que se mantiene en rojo.

## *ETAPA 1*

La primera etapa consiste simplemente en abrir la imagen original, la cual es almacenada en la variable “im”. Esta variable corresponde a una imagen RGB, por lo tanto posee dimensiones  $M \times N \times 3$ .

```
clear; home;
im=imread('niña_y_rosa.jpg');
figure(1);
subplot(2,3,1);
imshow(im);title('1. Imagen RGB original');
```



Figura 1

## *ETAPA 2*

La segunda etapa consiste en transformar la imagen RGB en una imagen en escala de grises. Para ello es posible utilizar las ventajas de Matlab en la manipulación de matrices. Así, la imagen en escala de grises se puede obtener promediando las 3 bandas R, G y B de la imagen original:  $gris=(im(:, :, 1)+im(:, :, 2)+im(:, :, 3))/3$ . Otra forma es utilizar la función de Matlab destinada para esta función:

```
gris=rgb2gray(im);
subplot(2,3,2);
imshow(gris);title('2. Transformación a escala de grises');
```

De esta forma se obtiene la imagen N°2 mostrada en la figura.

### ETAPA 3

En la tercera etapa se presenta una previsualización de la banda de interés en este ejemplo, que es la banda roja. Para esto es necesario descomponer la imagen RGB original en 3 imágenes simples que representan a cada banda.

```
imR=double(im(:,:,1));
imG=double(im(:,:,2));
imB=double(im(:,:,3));

subplot(2,3,3);
imshow(imR,gray);title('3. Visualización de Banda ROJA original');
```

Se realiza además una transformación al tipo de variable “double” para poder realizar operaciones aritméticas con las matrices (pues con imágenes uint8 no es posible). También se debe explicitar el “colormap” para visualizar una imagen *double* en la función *imshow*.

En la imagen N°3 de la figura 1, se observa la banda roja en escala de grises. Es posible observar que tanto el sombrero de la niña, como la rosa presentan un valor elevado, lo cual es consecuente con el color rojo de dichos objetos. Sin embargo, es posible observar que existen otras zonas de la imagen también con valores elevados (posiblemente cercanos a 255). Esto parece contradictorio pues el color rojo se encuentra presente sólo en el sombrero y en la rosa, pero este análisis no es correcto pues otros colores, como por ejemplo el blanco de la bufanda de la niña, también poseen elevados valores en la banda roja. A pesar de esto, se observa que la rosa y el sombrero de la niña en la imagen N°3, poseen un valor mayor que en la imagen N°2. Esto es correcto, pues la imagen gris corresponde a un promedio de las 3 bandas, en cambio en la imagen N°3 queda sólo el valor de la banda roja, que en este caso (rosa y sombrero) es el mayor de las 3 bandas.

Entonces, ¿qué es lo más representativo del color rojo?. No basta analizar lo que ocurre en la banda roja, por el contrario un objeto o un píxel rojo se caracteriza por tener un valor elevado en la banda roja, y valores menores en las bandas verde y azul. De esta forma el rojo ideal corresponde al valor RGB 255,0,0. Pero también es un rojo intenso un valor RGB 195,23,14.

En base a este razonamiento podemos afirmar que no será posible “aislar” la rosa roja analizando sólo la banda roja, para ello necesitamos imponer una condición lógica que considere las 3 bandas en sus conjunto.

### ETAPA 4

En esta etapa se busca determinar automáticamente las zonas predominantemente rojas. Para ello se aplica el siguiente criterio:

```
imR2=(imR-imG-imB);
masc=(imR2>20);
imR2=imR2.*masc;

subplot(2,3,4);
imR2=medfilt2(imR2);
imshow(imR2,gray);title('4. Filtro de mediana sobre banda ROJA');
```

En palabras simples el criterio utilizado es que la banda roja menos las bandas verde y azul debe ser a lo menos mayor a 20. Este valor, es un parámetro que puede ser ajustado para cada aplicación. Un punto RGB que cumple este criterio sería por ejemplo 255,155,79. (En este caso 255-155-79=21).

Obviamente todos los puntos donde la banda roja es más predominante también cumplen con el criterio: (255,10,5), (192,20,32), etc.

Todos los puntos que cumplen esta condición quedan con valor "1" y el resto con valor "0". La instrucción `mask=(imR2>20)` permite lograr una imagen binaria que representa a la "máscara" de todos los puntos rojos que cumplen la condición.

Antes de desplegar la imagen en pantalla se aplica un filtro de mediana para lograr homogeneidad entre los puntos vecinos. Esta imagen N°4 se observa en la siguiente figura.

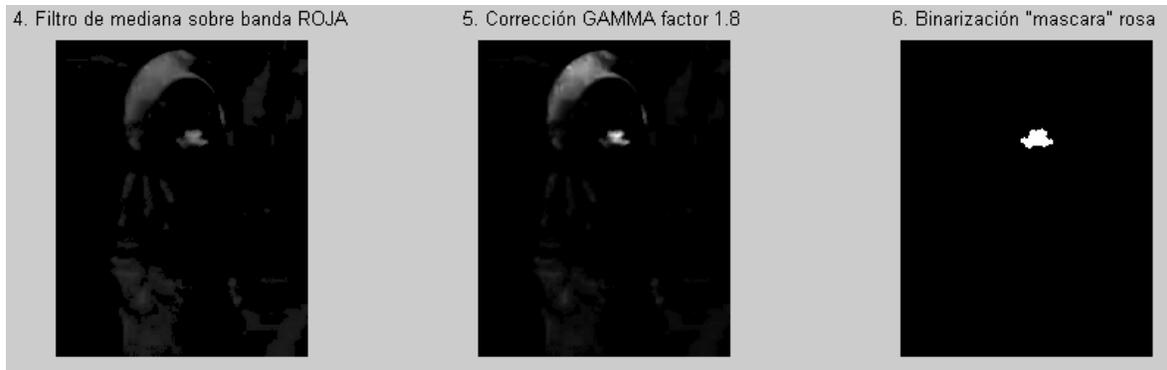


Figura 2

Es posible observar que a diferencia de la imagen N°2 y N°3, en esta imagen sólo se resaltan los objetos realmente "rojos".

### ETAPA 5

Esta etapa es sólo un refinamiento de la etapa anterior, que busca aislar la rosa respecto de su contorno.

```
imR2=imR2/255;
imR3=imadjust(imR2, [], [], 1.8);
subplot(2,3,5);
imshow(imR3, []); title('5. Corrección GAMMA factor 1.8');
```

Para resaltar aun más la diferencia entre los objetos rojos y el resto se aplica la función de corrección Gamma. El resultado se observa en la imagen N°5.

Utilizando esta imagen N°5 se pide al usuario que seleccione con el mouse el centro de la rosa. Para ello se utiliza la función "ginput".

```
disp('En la imagen 5 haga click en el centro de la rosa...');
[x y]=ginput(1);
y=round(y); x=round(x);
```

### ETAPA 6

Al conocer la ubicación de la rosa, es posible transformar la rosa en una máscara binaria con valor "1" para los píxeles rojos y "0" para el resto.

```
mask=0*imR3;
T=28;
mask(y-T:y+T, x-T:x+T)=mask(y-T:y+T, x-T:x+T)+1;
```

```

imR4=(double(imR3>0.01)).*masc;
subplot(2,3,6);
imshow(imR4,[1]);title('6. Binarización "mascara" rosa');

```

Luego de obtener el centro X,Y de la rosa, se extrae una vecindad cuadrada de diámetro 2T aplicando la máscara “masc” sobre todos los valores de la imagen imR3 mayores a 0.01(parámetro). De esta forma se puede “aislar” los valores representativos de la rosa.

```

masc2=1-(imR4);
imR=double(gris)/255;
imG=double(gris).*masc2/255;
imB=double(gris).*masc2/255;
figure(2);
imFinal=cat(3,imR,imG,imB);
imshow(imFinal);title('Efecto publicitario buscado...');

```

Finalmente para poder resaltar el color rojo de la rosa se obtiene una máscara inversa denominada “masc2” mediante la cual se asigna “0” a todos los píxeles de la rosa en la banda verde y azul. El resto de los píxeles de la imagen se mantienen igual. En el caso de la banda Roja se deja sin alteración respecto de la banda original. En otras palabras, para resaltar el rojo de la rosa no es necesario aumentar el valor de la banda roja en los píxeles correspondientes a la rosa, sino por el contrario es necesario disminuir los valores en las otras bandas, utilizando la imagen “gris” como punto de partida para cada banda RGB.

La figura 3 presenta el efecto publicitario “rojo sobre gris” resultante en base al algoritmo presentado en este ejemplo.



Figura 3