

# ***Ejemplo práctico de función de segmentación de color utilizando la distancia de Mahalanobis***

***Pablo Roncagliolo B.***

Este documento presenta una aplicación simple que permite calcular la distancia de Mahalanobis, generalmente utilizada para segmentación de color. A diferencia de la distancia euclidiana, la distancia de Mahalanobis corresponde a un elipsoide cuyo eje principal está en la orientación del matiz representativo de una muestra.

Se implementó un algoritmo que permite calcular la diferencia entre ambos tipos de filtros (euclidiano y mahalanobiano), utilizando como referencia la imagen de la fig.1



**Figura 1**

## ***ETAPA I: ANTECEDENTES***

La covarianza corresponde a una medición estadística que indica la "fuerza" de la correlación entre 2 o más variables aleatorias. Por ejemplo, la covarianza entre dos variables X e Y con N muestras cada una, se define como:

$$\text{cov}(X, Y) = \sum_{i=1}^N \frac{(x_i - \bar{x})(y_i - \bar{y})}{N} = E((X - E(X))(Y - E(Y))^T)$$

Si la covarianza es 0, indica que no existe correlación entre ellas. Si el valor es mayor que 0 indica que Y tiende a aumentar tal como X aumenta. Valores mejores de 0 indican que Y tiende a disminuir cuando X aumenta. Si X=Y entonces la covarianza se reduce a la formula de varianza.

Para un conjunto de más de 2 variables aleatorias se utiliza la "Matriz de Covarianza", que se define como:

$$V_{ij} = \text{cov}(X_i, X_j) = \sum_{k=1}^N \frac{(x_{i_k} - \bar{x}_i)(x_{j_k} - \bar{x}_j)}{N}$$

Donde  $V_{ij}$  corresponde a la entrada (i,j) de la matriz.

En el caso de imágenes RGB, se puede considerar que existen tres variables aleatorias R, G y B. Por lo tanto la matriz de covarianza sería:

i/j	R	G	B
R	cov(R,R)	cov(R,G)	cov(R,B)
G	cov(G,R)	cov(G,G)	cov(G,B)
B	cov(B,R)	cov(B,G)	cov(B,B)

Por ejemplo, sea ROI un conjunto de píxeles representativos de una región de interés:

```
ROI =
    209    31    17
    213    33    18
    217    37    14
    232    53    12
    199    26    10
    251    89    17
    252   105     1
    249    88    10
    212    34    20
    213    35    21
```

Para obtener la matriz de covarianza en Matlab se aplica el comando MC=cov(ROI);

```
MC =
    386.9000    562.8111   -59.6667
    562.8111    864.3222  -109.5556
   -59.6667  -109.5556     36.0000
```

La distancia de Mahalanobis utiliza la inversa de la matriz de covarianza, por lo tanto se debe aplicar el comando de Matlab "inv" para invertir la matriz: MCi=inv(MC);

```
MCi =
    0.0701   -0.0504   -0.0370
   -0.0504    0.0380    0.0323
   -0.0370    0.0323    0.0648
```

Generalmente se utiliza el valor normalizado de la matriz inversa de covarianza. Para ello se divide la matriz MCi por el máximo valor presente: MCi=MCi/max(max(MCi));

```
MCi =
    1.0000   -0.7181   -0.5278
   -0.7181    0.5425    0.4607
   -0.5278    0.4607    0.9234
```

## ETAPA 2: EJEMPLO DISTANCIA EUCLIDIANA

El filtro de distancia de euclidiana consiste en determinar la distancia de todos los píxeles de la imagen respecto de un píxel o color de referencia.

$$D = \|(X - Y)\|$$

$$D = \left[ (X - Y)^T (X - Y) \right]^{1/2}$$

$$D = \sqrt{(Xr - Yr)^2 + (Xg - Yg)^2 + (Xb - Yb)^2}$$

El algoritmo en Matlab que implementa este filtro es:

```
IM=imread('flor.jpg');
subplot(2,2,1);imshow(IM);
title('Imagen Original');
[nf nc c]=size(IM);

disp('Selecciones 1 punto de color para el filtro...');
[x y]=ginput(1);
x=round(x);
y=round(y);
colormediol=[IM(y,x,1); IM(y,x,2); IM(y,x,3)];
MC=[1 0 0;0 1 0;0 0 1];
MCi=inv(MC);
IM2=IM;
disp('Espere unos segundos mientras se aplica el filtro esférico de
color...');
for f=1:nf
    for c=1:nc
        z=[IM(f,c,1); IM(f,c,2); IM(f,c,3)];
        d2=DMaha(z,colormediol,MCi);
        if d2>0.12,
            IM2(f,c,:)= [0 0 0];
        end;
    end;
end;

subplot(2,2,2);imshow(IM2);
title('Filtro Distancia Euclidiana');
```

El programa solicita hacer click sobre un píxel representativo. El resultado se muestra en la imagen de la derecha.



Figura 2

Se puede concluir que el filtro funciona adecuadamente, sin embargo no es una buena segmentación de las "hojas" de la flor, por cuanto los colores que ellas contienen no pueden ser representados por un solo píxel de referencia.

### ETAPA 3: DISTANCIA DE MAHALANOBIS

La distancia de Mahalanobis permite calcular distancias entre colores ponderadas por la importancia del Matiz (color principal) y la varianza en cada componente (R, G y B). Incluyendo la inversa de la Matriz de Covarianza de un conjunto de píxeles representativos, se puede lograr esta "ponderación". La distancia de Mahalanobis es igual a la distancia Euclidiana si la matriz de covarianza  $C$  corresponde a la matriz identidad.

$$D = \sqrt{(X - Y)^T C^{-1} (X - Y)}$$

La figura 3 presenta la comparación entre ambos métodos. Luego de realizar el filtro euclidiano en base a 1 píxel de referencia, el programa diseñado solicita al usuario realizar 10 click sobre píxeles representativos de las hojas de la flor. El rectángulo blanco indica la zona sobre la cual se realizaron los 10 click. La imagen 3.c, muestra los colores de los 10 píxeles seleccionados. La imagen 3.d presenta el resultado del filtro de Mahalanobis. **Como pueden ver el resultado es espectacular!**



Figura 3

El código en Matlab de esta etapa es:

```
disp('Selecciones 10 puntos de color representativos para el  
filtro...');  
ROI=[];  
for f=1:10  
    [x y]=ginput(1);  
    x=round(x);  
    y=round(y);
```

```

        color=[IM(y,x,1) IM(y,x,2) IM(y,x,3)];
        ROI=[ROI; color];
    end;
    ROI=double(ROI);
    mapROI=ROI/255;

    R=1:10;
    R=[R;R;R;R;R;R;R;R;R;R];
    subplot(2,2,3);imshow(R,mapROI);
    title('Los 10 colores seleccionados visualizados como columnas');

    MC=cov(ROI);
    MCi=inv(MC);
    MCi=MCi/max(max(MCi));

    colormedio2=[mean(ROI(:,1)); mean(ROI(:,2)); mean(ROI(:,3))];

    IM3=IM;
    disp('Espere unos segundos mientras se aplica el filtro con distancia
    Mahalanobis...');
    for f=1:nf
        for c=1:nc
            z=[IM(f,c,1); IM(f,c,2); IM(f,c,3)];
            d2=DMaha(z,colormedio2,MCi);
            if d2>0.12,
                IM3(f,c,:)= [0 0 0];
            end;
        end;
    end;

    subplot(2,2,4);imshow(IM3);
    title('Filtro con distancia de Mahalanobis');

```

Finalmente el código de la función de distancia de Mahalanobis es:

```

function d2=DMaha(z,a,MCi)

z=double(z)/255;
a=double(a)/255;
%d2=distancia al cuadrado
d2=(z-a)'*MCi*(z-a);

```

Esta función es la misma que se utiliza para el filtro euclidiano, pero utilizando matriz de covarianza con "1" en la diagonal (identidad).