



## Laboratorio 7

### PROCESAMIENTO DE SEÑALES E IMÁGENES

Profesores:

- Violeta Chang C.
- Leonel E. Medina

Ayudante: Luis Corral

Alumno: John Serrano C.

## Actividades

1. Escribir una función para eliminar el ruido en una imagen binaria, usando operadores morfológicos. Utilizar la imagen 'formasSP.png' para probar la función escrita y guardar la imagen resultante con el nombre de 'formas-sinruido.jpg'.
2. Escribir una función en Matlab para detectar patrones en forma de 'T' en una imagen binaria, usando operadores morfológicos. La función debe devolver una imagen con los patrones 'T' marcados en ella. Utilizar la imagen 'patrones.png' para evaluar la función escrita.
3. Escribir una función en Matlab para obtener el borde de todos los componentes conexos de una imagen binaria, según el método revisado en clase. Utilizar la imagen 'patrones.png' para probar la función escrita.
4. Utilizar la imagen 'aerea.jpg', y escribir una función que lea la imagen, mejore el contraste mediante ecualización del histograma y guarde la imagen mejorada con el nombre de 'aerea-eq.jpg'. Comparar visualmente con resultado de aplicar función *histeq* de Matlab.

5. Utilizar la imagen 'indoor.jpg' y mejorar el contraste usando tanto *imadjust* y *histeq*. Observar visualmente los resultados de la aplicación de ambas funciones y analizar qué diferencia tienen en su funcionamiento y resultado.

## Tarea

Entregue el resultado de la actividad 5 en formato mlx mostrando las imágenes originales y resultantes en conjunto a sus histogramas e incluyendo el análisis solicitado. Se evalúa los conceptos en formato de texto, los comentarios dentro del código, la exactitud del algoritmo y la calidad de los gráficos generados. Muestre solo los valores más importantes.

## Desarrollo Tarea

En primer lugar, se debe cargar la imagen **indoor.jpg** utilizando la función `imread()` y luego transformarla a una escala de grises, utilizando la función `im2gray()`. Notese que la imagen corresponde a un museo, donde se pueden ver algunos tonos bastante oscuros y otros un poco mas claros, como por ejemplo la superficie del barco. Se obtiene el histograma de la imagen utilizando `imhist()` para ver la distribución de los tonos de grises en la imagen.

```
clearvars

% Carga de la imagen indoor.jpg
imagen = imread("indoor.jpg"); % Se carga la imagen indoor.jpg
imagen = im2gray(imagen);      % Se transforma a una escala de grises

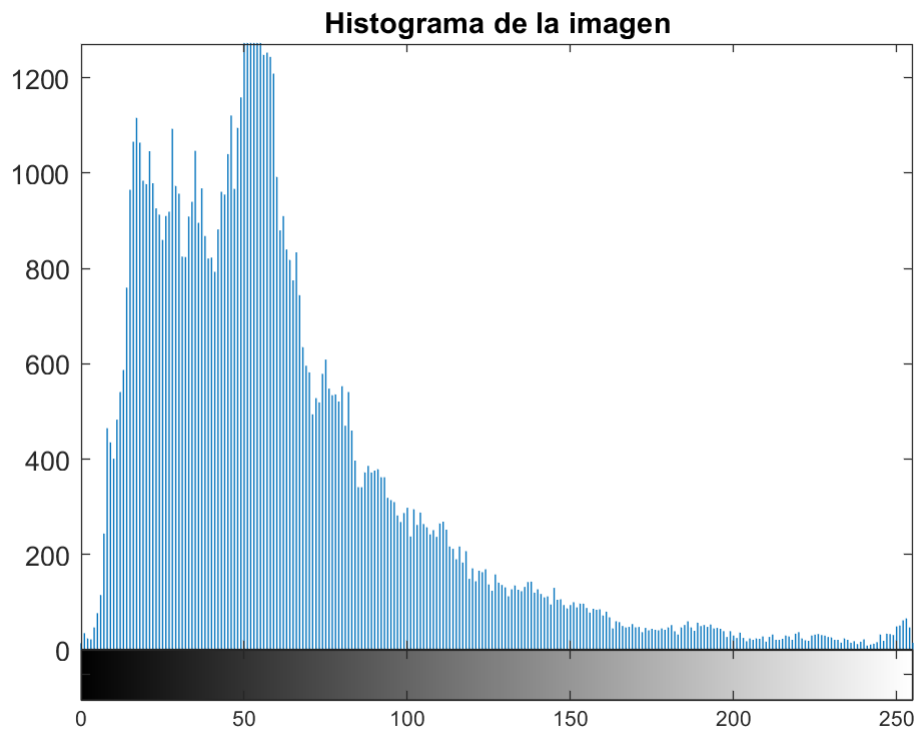
figure % Se inicializa una figura
% Se muestra la imagen original en escala de grises
imshow(imagen),title('Imagen Original');
```

Imagen Original



El histograma de la imagen anterior es:

```
figure % Se inicializa una figura
% Se muestra el histograma de la imagen original en escala de grises
imhist(imagen),title('Histograma de la imagen');
```



Si se observa el histograma de la imagen, se puede ver que sus valores están muy cercanos al lado izquierdo, es decir, hacia los tonos de grises más oscuros, lo cual es sinónimo de que la imagen no tiene muy bien contraste. Ahora, se busca mejorar el contraste de la imagen y para ello, se deben utilizar tanto las funciones **imadjust()**, la cual nos permite realizar la extensión del histograma, y **histeq()**, la cual permite realizar la **ecualización del histograma**. Por lo tanto, se puede realizar una comparación entre ambas funciones si se muestran las imágenes tras aplicarles las funciones junto con sus histogramas correspondientes.

Primero, se aplica **imadjust()**:

```
% Se aplica imadjust() a la imagen. Esto permite realizar el
% estiramiento (Stretching) del histograma.
ip1 = imadjust(imagen);

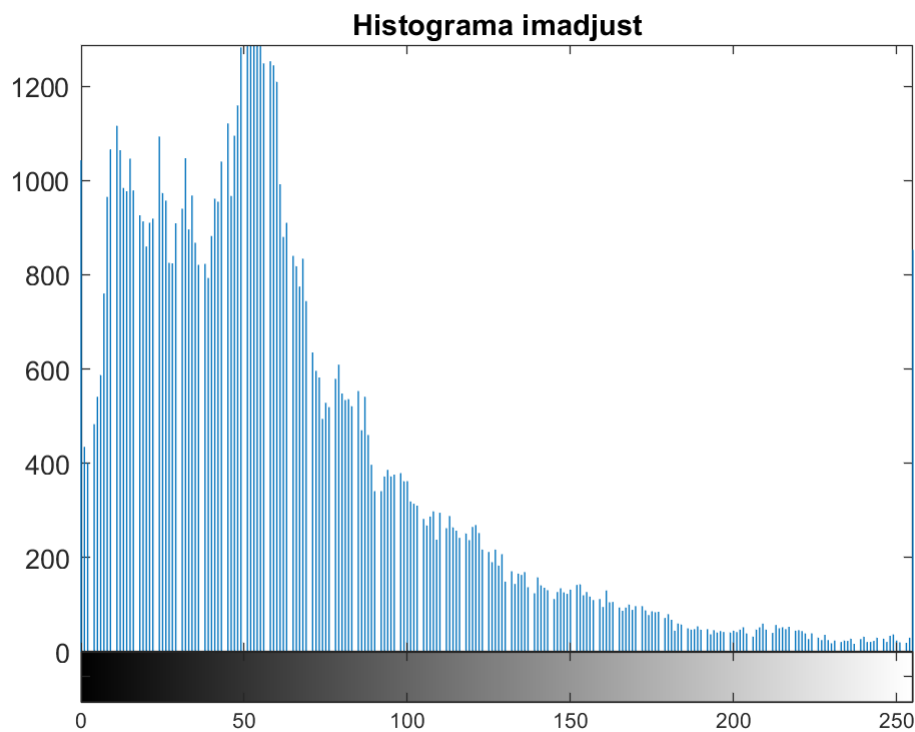
figure % Se inicializa una figura
% Se muestra la imagen tras aplicar la función imadjust()
imshow(ip1,[]),title('Función imadjust');
```

**Función imadjust**



El histograma de la imagen anterior es:

```
figure % Se inicializa una figura
% Se muestra el histograma de la imagen tras aplicar la función imadjust()
imhist(ip1),title('Histograma imadjust');
```



Se puede observar que el histograma de la imagen cambió, ahora **se alargó hacia los extremos, estando un poco más pegado a los tonos de grises oscuros y teniendo un poco mas de tono de grises claros**. Sin embargo, nótese que las líneas del histograma se mantienen, por lo que solo se extendió hacia los extremos, pero no hubo un intento de "aplanar" el histograma. Se puede realizar una comparación de las imágenes lado a lado para ver que tanto mejoró el contraste.

```
figure % Se inicializa una figura
% Se muestra la imagen original
subplot(1,2,1), imshow(imagen),title('Imagen Original');
```

```
% Se muestra la imagen tras aplicar la función imadjust()  
subplot(1,2,2),imshow(ip1,[]),title('Stretching (imadjust)');
```

Imagen Original



Stretching (imadjust)



Si se observa las superficies del barco, se puede notar que **ahora el tono de gris de las superficies son un poco más claras** y la iluminación en general mejoró bastante. Por lo que efectivamente se logra mejorar el contraste de la imagen. Ahora, se repite el mismo proceso con **histeq()**:

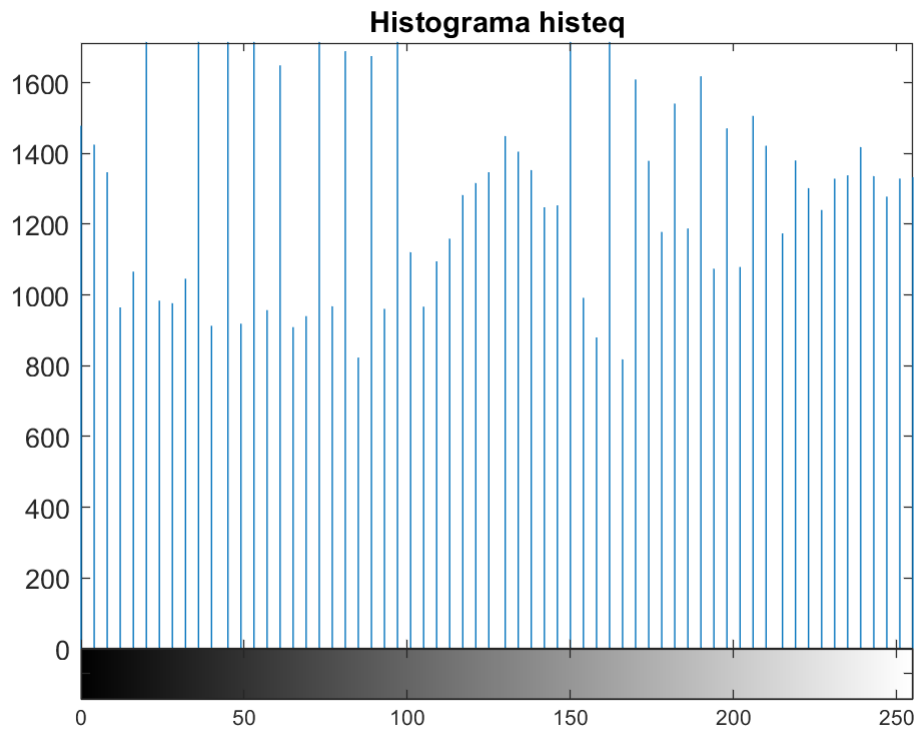
```
% Se aplica la ecualización del histograma mediante la  
% función histeq()  
ip2 = histeq(imagen);  
  
figure % Se inicializa una figura  
% Se muestra la imagen tras aplicar la función histeq()  
imshow(ip2,[]),title('Función histeq');
```

Función histeq



El histograma de la imagen anterior es:

```
figure % Se inicializa una figura
% Se muestra el histograma de la imagen tras aplicar la función histeq()
imhist(ip2),title('Histograma histeq');
```



Se puede ver que nuevamente se realiza un estiramiento del histograma, pero esta vez hubo **un cambio en las alturas del histograma**, donde se intentó aplanarlas lo más posible.

Nuevamente, realizando una comparación lado a lado con la imagen original y la obtenida anteriormente:

```
figure % Se inicializa una figura
% Se muestra la imagen original
subplot(1,2,1), imshow(imagen),title('Imagen Original');
% Se muestra la imagen tras aplicar la función histeq()
subplot(1,2,2), imshow(ip2,[]),title('Ecualización (histeq)');
```

Imagen Original



Ecualización (histeq)



Se puede observar que el contraste mejoró bastante, incluso **ahora los tonos de grises más oscuros se volvieron mucho más claros** y se pueden apreciar más detalles en la imagen. Por último, realizando una comparación entre los resultados de ambas funciones:

```
figure % Se inicializa una figura
% Se muestra la imagen tras aplicar la función imadjust()
subplot(1,2,1),imshow(ip1,[]),title('Stretching (imadjust)');
% Se muestra la imagen tras aplicar la función histeq()
subplot(1,2,2),imshow(ip2,[]),title('Ecualización (histeq)');
```

**Stretching (imadjust)**



**Ecualización (histeq)**



Claramente se puede ver que el contraste mejora mucho más con la ecualización del histograma realizado gracias a la función `histeq()`. Los tonos de grises de la segunda imagen son mucho más claros y se pueden observar muchos mas detalles en la segunda imagen comparado con la primera. Por lo tanto y en conclusión, se podría decir que la ecualización del histograma, para este caso, es la mejor forma de mejorar el contraste con **la imagen que estamos trabajando**.