

# 4 Técnicas de preprocesado

---

Las técnicas de procesamiento pretenden mejorar o realzar las propiedades de la imagen para facilitar las siguientes operaciones de la Visión Artificial, tales como las etapas de segmentación, extracción de las características y finalmente la interpretación automática de las imágenes. Recuérdese que el fin último de la Visión Artificial es la interpretación automática de la imagen o una mejora de la calidad de la imagen.

Las técnicas de preprocesado se basan bien en técnicas derivadas del procesamiento digital de señales, vistas en el capítulo anterior, o bien en un conjunto de procedimientos heurísticos que han dado resultados satisfactorios. En este tema será objeto de estudio el segundo proceder. Las técnicas heurísticas son una combinación de procedimientos basados en el procesamiento digital de señales y en otros tipos de manipulaciones matemáticas. Su estudio se iniciará con una primera clasificación. Estos algoritmos se pueden catalogar en función de las pretensiones de sus transformaciones en alguna de las siguientes facetas:

- Realce o aumento del contraste (*enhancement*).
- Suavizado o eliminación del ruido (*denoising*)
- Detección de bordes (*edges*)

El desarrollo de estas técnicas se realizará en la información de luminancia de las imágenes, esto es, sólo se analizarán sobre imágenes en niveles de grises. Muchas de

ellas pueden ser luego trasladadas a imágenes en color. Sin embargo, no se puede generalizar. Las técnicas de preprocesado sobre imágenes en color requieren un tratamiento diferenciado.

## 4.1 Realce o manipulación del contraste

---

Las técnicas de realce pretenden aumentar el contraste de las imágenes. No en el sentido estricto de aumentar la calidad radiométrica, sino de mejorar algunas de sus características visuales para las siguientes etapas del análisis automático de las imágenes. Las causas de aplicar estos algoritmos se deben bien a una falta de iluminación uniforme en la escena o bien al deseo de aumentar el contraste entre los objetos presentes en la imagen.

En el capítulo de formación de las imágenes se estudió las dificultades de la implementación del sistema de iluminación. Normalmente, la escena capturada por el sistema de visión se encuentra con unas fuentes de iluminación no colimadas y variables con el tiempo y con objetos que producen reflejos y sombras no deseadas. De otro lado, las ganancias de las cámaras no son lineales. El doble de intensidad radiada en un píxel, no se traduce en el doble de cuantificación en el nivel de gris de ese píxel. Todas estas imperfecciones deben ser corregidas tanto en la formación de las imágenes como con el uso de técnicas preprocesado de realce. Ambas deben ser utilizadas.

Estas técnicas de realce también se aplican cuando se pretende utilizar técnicas de segmentación. El objetivo de este procesado es facilitar las tareas de partición de la imagen. Con este fin se aumenta el contraste entre los objetos de la imagen.

Muchas de las aplicaciones de realzado se fundamentan en operaciones punto a punto, es decir, no tienen en cuenta el entorno de vecindad del píxel. Para su estudio se requiere introducir los conceptos de histograma, brillo y contraste. Los fundamentos de estas técnicas están en la adecuación del rango dinámico de la imagen y en la ecualización del histograma. Para acabar, también se explorarán algunos procedimientos basados en el uso de filtros paso alto.

### 4.1.1 Histograma, brillo y contraste

El histograma es una función discreta que contabiliza el número de ocurrencias de cada nivel de gris presenta en una imagen. Se representa en un diagrama donde en el eje de abscisa está el nivel de gris y en el de ordenadas la frecuencia de cada nivel de gris en la imagen. Si al histograma se le divide por el número de píxeles de la imagen se obtendrá la función de probabilidad de cada nivel de gris en la imagen:

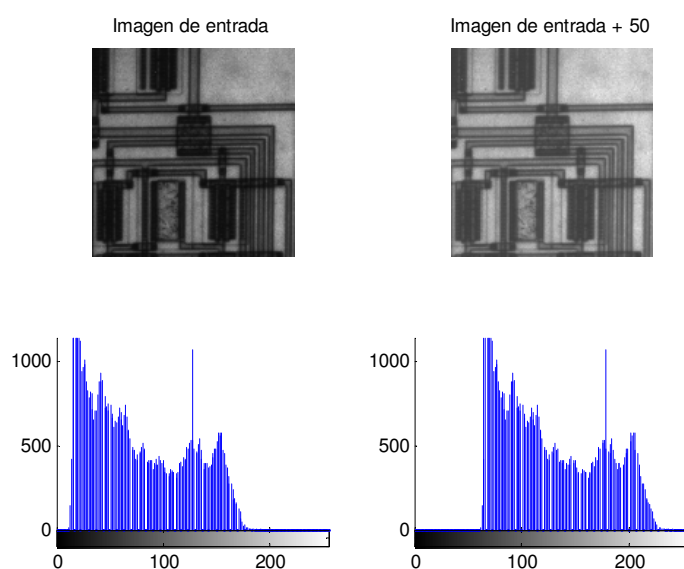
$$h(i) = \text{número de ocurrencia del nivel de gris } i \text{ en la imagen}$$
$$p(i) = \frac{h(i)}{M \cdot N} \quad (4.1)$$

Siendo  $M$  y  $N$  el número de filas y columnas de la imagen,  $h(i)$  es la función histograma y  $p(i)$  su probabilidad. Un histograma no dice nada sobre la disposición espacial de las distintas intensidades. Dos imágenes diferentes pueden tener igual histograma. Sin embargo, el histograma habla sobre el tipo de adquisición que ha sido realizado.

El histograma proporciona información estadística de la imagen. Esta información es bastante útil para conocer cómo se ha producido el proceso de formación de la imagen. Así, se define el brillo de la imagen como el valor medio de la imagen que coincide con el valor medio del histograma:

$$\text{Brillo} = \mu = \frac{1}{M \cdot N} \sum_{x=1}^M \sum_{y=1}^N f(x, y) = \sum_{i=0}^{I-1} i \cdot p(i) \quad (4.2)$$

Donde  $f(x,y)$  retorna el nivel de gris del píxel situado en las coordenadas  $(x,y)$ ,  $I$  es el número de grises que se ha empleado en el proceso de cuantificación de la imagen. Una imagen que presente un valor de brillo mayor que otro sobre la misma escena, indica que el nivel de radiación medio que ha llegado a la cámara es mayor en una imagen que en otra, bien por que se disminuyó el número  $F$  de diafragma o por que se varió el tiempo de integración. Este efecto se observa en el histograma como un desplazamiento neto hacia la derecha del eje de abscisa. A medida de que la imagen parezca más clara, el histograma se mueve hacia valores más altos de los niveles de grises. La variación del brillo se efectúa con la operación aritmética de suma o resta a cada píxel de una cantidad determinada.



**Figura 4. 1. Imagen con dos valores distintos de brillo a) 76 b) 126**

La varianza del histograma también es asociada al contraste de la imagen:

$$\begin{aligned} \text{Contraste} = \sigma^2 &= \frac{1}{M \cdot N} \sum_{x=1}^M \sum_{y=1}^N (f(x, y) - \mu)^2 = \\ &= \frac{1}{M \cdot N} \sum_{i=0}^{I-1} (i - \mu)^2 \cdot h(i) = \sum_{i=0}^{I-1} (i - \mu)^2 \cdot p(i) \end{aligned} \quad (4.3)$$

El contraste indica la dispersión de los niveles de grises en la imagen. Aunque se pueden encontrar otras definiciones matemáticas del contraste, todas son medidas de dispersión. En todo caso, una imagen con poco contraste indica que hay poca variabilidad de los niveles de grises en la imagen. Su efecto se muestra en un histograma muy concentrado, con un rango dinámico pequeño. El rango dinámico es la variación de niveles de grises en la imagen. Otro estadístico interesante es la función energía:

$$\text{Energía} = \sum_{i=0}^{I-1} p(i)^2 \quad (4.4)$$

Ésta indica el grado de dispersión de grises en la imagen. Si hubiese un único nivel de gris la energía sería máxima y de valor uno. A medida de que aumente el número de niveles de grises existentes en la imagen disminuirá la energía. La función inversa a la energía es la entropía:

$$\text{Entropía} = - \sum_{i=0}^{I-1} p(i) \cdot \log(p(i)) \quad (4.5)$$

A mayor entropía más desorden en la imagen, esto es, más niveles de grises participan en la imagen. La entropía será máxima cuando todos los niveles de grises sean equiprobables.

Se dice que una imagen está saturada cuando su histograma presenta valores muy altos en sus extremos del rango dinámico; el histograma tiene una forma en “U”. Las imágenes con bajo contraste o saturada tienen pérdidas de información en sus adquisiciones. La solución está en una nueva formación de la escena con valores diferentes de los parámetros de la captura. Las técnicas de preprocesado digital no aumentan el nivel de información, aunque acondicionan la imagen para las siguientes etapas de interpretación de las imágenes.

Para aumentar el contraste se emplean las funciones de transferencias del histograma. Éstas son transformaciones matemáticas que asocian a un nivel de gris de entrada otro nivel en la salida:

$$i' = T(i) \quad (4.6)$$

$T$  es la función matemática de transformación e  $i'$  es el nuevo nivel de gris asociado. La transformación más generalizada consiste en adecuar el rango dinámico de la imagen de entrada a todo el espectro de valores de cuantificación:

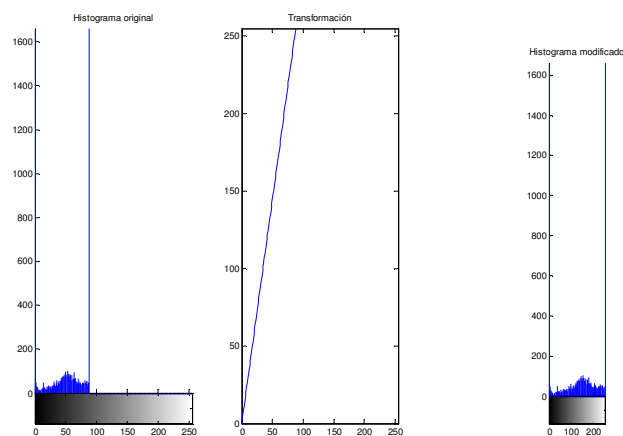
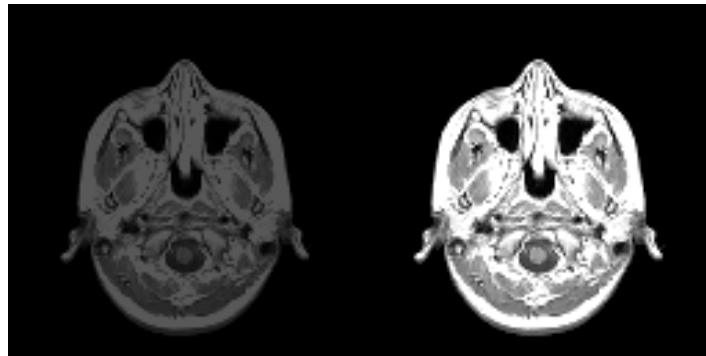
$$\frac{i_{out} - 0}{(I - 1) - 0} = \frac{(i_{in} - a_{min})}{a_{max} - a_{min}} \Rightarrow i_{out} = \frac{I - 1}{a_{max} - a_{min}} (i_{in} - a_{min}) \quad (4.7)$$

Siendo  $I$  el rango máximo de cuantificación y  $a_{min}$  y  $a_{max}$  el rango dinámico de la imagen de entrada.

### **Ejemplo 4.1**

Adecuar el rango dinámico de una imagen de resonancia magnética, cuyos valores varían entre el nivel de gris de 0 a 88. Nótese el carácter saturado de la adquisición.

$$i_{out} = \frac{255}{88} i_{in}$$



**Figura 4. 2 Imagen de resonancia magnética y con aumento del contraste a)  $\sigma^2=33$  b)  $\sigma^2=96$**

**Resolución Matlab**

```
>>imagen=imread('mri.tif');
>>imhist(imagen);
>>max(max(imagen))
>>imagen_sal=imadjust(imagen);
>>imhist(imagen_sal);
>>imshow([imagen,imagen_sal]);
>>std2(imagen),std2(imagen_sal)
```

Evidentemente, las transformaciones no necesariamente deben ser lineales. Por ejemplo, las siguientes operaciones consiguen el negativo de la imagen de entrada y variaciones diferentes del contraste:

$$T_1(i) = I - i \quad T_2(i) = \frac{i^2}{I} \quad T_3(i) = \sqrt{I \cdot i} \quad T_4(i) = i^\gamma \quad (4.8)$$

La última transformación se la llama la corrección de gamma. Su propósito es adecuar la luminancia a la sensibilidad de los monitores de vídeo de rayos catódicos, los cuales presentan no linealidad entre la intensidad y el voltaje de excitación de la pantalla. Si  $\gamma < 1$ , la imagen de salida será más clara y con  $\gamma > 1$  se volverá más oscura. Este tipo de transformación se realiza con valores normalizados de intensidad en el intervalo  $[0,1]$ . Por ejemplo, suponga que gamma fuese 2 y no estuviera el rango de intensidad normalizado, por ejemplo en el intervalo de  $[0 \ 255]$ . Se observa que el rango de salida sería mayor a 256 niveles de grises.

No obstante, como se ha comentado, estas transformaciones no introducen mayor información en las imágenes, pues estos resultados son sólo expansiones de los rangos dinámicos. En cambio, estas operaciones facilitan las siguientes etapas de extracción de información de mayor relevancia.

La implementación de estas transformaciones se realiza a través de una tabla que se la denomina LUT (*Look Up Table*). Para aumentar la eficiencia de la conversión, en la tabla se entra por el valor de entrada de nivel de gris y ésta retorna el nivel de salida. Comúnmente a la LUT se la denomina paleta o mapa de color.

#### 4.1.2 Ecualización del histograma

En cualquier función de densidad probabilística que es transformada a uniforme se maximiza la explotación de la información. Este fenómeno que rige a los procesos continuos también se puede aplicar sobre información discreta. La conversión de cualquier forma de histograma a uno uniforme, si bien es cierto que no aumentará la información, como ya se ha comentado anteriormente, sí mejorará la imagen para las siguientes etapas. Recuerde que el valor máximo de la entropía se dará cuando el histograma sea uniforme.

El fundamento se basa en realizar una conversión en el histograma de forma que la probabilidad de cualquier nivel de gris en la imagen sea idéntica. Desde el punto de vista del histograma, esta transformación producirá que el número de ocurrencias de un nivel de gris sea el mismo valor siempre,  $h(i) = \frac{M \cdot N}{I}$ .

Para obtener esta técnica de preprocesado se partirá de la función de distribución. Ésta se define, para variables aleatorias discretas, como el sumatorio de probabilidades desde el inicio de la variable hasta un valor prefijado. Por tanto, el sumatorio de la función en todo el espectro de la variable aleatoria independiente es la unidad:

$$\begin{aligned} F(r) &= \sum_{i=0}^r p(i) \\ F(I-1) &= \sum_{i=0}^{I-1} p(i) = 1 \end{aligned} \quad (4.9)$$

En este caso, la probabilidad de la variable aleatoria corresponderá con la probabilidad de que aparezca el nivel gris  $i$  en la imagen. La función de distribución de una imagen estará determinada por el histograma acumulado relativizado por el tamaño de la imagen.

Sea  $F(r)$  la función de distribución de una imagen cualquiera. Un histograma uniforme se caracterizará por tener:

$$F(r') = \sum_{i=0}^{r'} p(i) = (r' + 1) \frac{1}{I} \quad (4.10)$$

Siendo  $F(r')$  la función de distribución uniforme. La conversión del histograma se establece igualando las funciones de distribución que posee la imagen con la función de distribución deseada. Con esta relación se establece la función de transferencia del histograma, de  $r$  a  $r'$ :

$$F(r') = F(r) \Rightarrow (r' + 1) \frac{1}{I} = F(r) \Rightarrow r' = F(r) \cdot I - 1 \quad (4.11)$$

El carácter discreto de la variable supondrá una aproximación a una densidad uniforme.

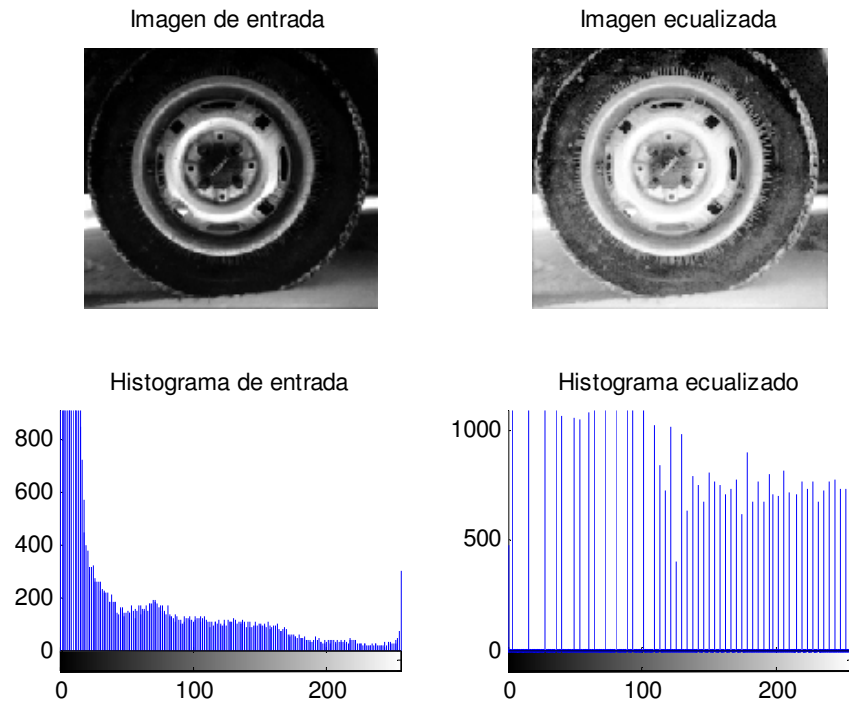


Figura 4. 3 Ejemplo de ecualización del histograma

### Resolución Matlabs

```
>>imagen=imread('tire.tif');
>>imhist(imagen);
>>imagen_sal=histeq (imagen);
>>imhist(imagen_sal);
```

### Ejemplo 4.2

Se tiene esta subimagen:

|   |   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|---|
| 5 | 5 | 1 | 3 | 1 | 0 | 2 | 4 | 2 | 6 |
| 8 | 4 | 3 | 1 | 3 | 0 | 4 | 6 | 1 | 3 |
| 4 | 6 | 5 | 4 | 2 | 2 | 0 | 0 | 3 | 8 |
| 2 | 0 | 2 | 5 | 6 | 2 | 3 | 3 | 8 | 2 |
| 1 | 5 | 1 | 1 | 2 | 1 | 2 | 4 | 5 | 7 |
| 1 | 0 | 5 | 6 | 3 | 2 | 2 | 5 | 4 | 1 |
| 0 | 5 | 5 | 5 | 1 | 0 | 1 | 1 | 4 | 5 |
| 4 | 5 | 4 | 3 | 3 | 6 | 8 | 2 | 2 | 1 |
| 4 | 1 | 0 | 5 | 3 | 2 | 1 | 0 | 6 | 5 |
| 1 | 4 | 3 | 0 | 5 | 5 | 6 | 1 | 1 | 4 |

Calcular:



1. Histograma
2. Brillo, contraste, energía y entropía
3. Ecualización del histograma.

|               |      |      |      |      |      |      |      |      |      |
|---------------|------|------|------|------|------|------|------|------|------|
| Nivel de gris | 0    | 1    | 2    | 3    | 4    | 5    | 6    | 7    | 8    |
| Ocurrencias   | 11   | 19   | 15   | 12   | 13   | 17   | 8    | 1    | 4    |
| Probabilidad  | 0.11 | 0.19 | 0.15 | 0.12 | 0.13 | 0.17 | 0.08 | 0.01 | 0.04 |

Brillo = 3, contraste = 4, energía = 0.139, entropía = 2.94

|                          |         |       |     |        |       |        |        |        |     |
|--------------------------|---------|-------|-----|--------|-------|--------|--------|--------|-----|
| Nivel de gris de entrada | 0       | 1     | 2   | 3      | 4     | 5      | 6      | 7      | 8   |
| Nivel de gris de salida  | (-0.01) | (1.7) | (3) | (4.13) | (5.3) | (6.83) | (7.55) | (7.64) | (8) |
|                          | 0       | 1 ó 2 | 3   | 4      | 5     | 6 ó 7  | 7 ó 8  | 7 ó 8  | 8   |
| Histograma ecualizado    | 11      | 19    | 15  | 12     | 13    | 17     | 13     | 13     | 13  |

#### 4.1.3 Realzado basado en el dominio frecuencial

Las componentes de alta frecuencia están relacionadas con los bordes de los objetos y con los pequeños detalles de la imagen. El aumento del contraste puede estar basado en una combinación lineal entre la imagen original y el resultado de un filtro paso alto. La imagen salida será:

$$\begin{aligned}
 \text{Imagen Salida} &= A * (\text{Imagen}) - (\text{Paso bajo de la Imagen}) \\
 &= (A-1) * (\text{Imagen}) + (\text{Imagen}) - (\text{Paso bajo de la Imagen}) \\
 &= (A-1) * (\text{Imagen}) + (\text{Paso alto de la Imagen})
 \end{aligned}$$

$A$  es un escalar; cuando es la unidad se tendrá el resultado del filtro paso alto, en cambio para  $A$  mayor a la unidad se estará realizando un énfasis sobre las componentes de alta frecuencia, produciendo el proceso de realzado y para  $A$  menor a la unidad aparece como el negativo de la imagen. A este algoritmo se le denomina *unsharp masking*. Desde el punto de vista de la implementación se suele utilizar la imagen menos el resultado de ésta con una máscara laplaciana; en un entorno de 3x3 queda definido como:

$$\begin{pmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{pmatrix} - \begin{pmatrix} 1 & 1 & 1 \\ 1 & -4 & 1 \\ 0 & 1 & 0 \end{pmatrix} \quad (4.12)$$



Figura 4. 4 Filtrado de realzado 'unsharp'

#### **Resolución Matlab**

```
>>imagen=imread('cameraman.tif');
>>H=fspecial('unsharp',0)
>>imagen_sal=imfilter (imagen, H);
>>imshow([imagen,imagen_sal]);
```

## 4.2 Técnicas de suavizado

Las técnicas de suavizado de las imágenes intentan eliminar el nivel de ruido presente en la imagen. En el proceso de formación de la imagen se ha generado ruido aleatorio que se ha sumado a la señal. Este ruido se explica como una variable aleatoria que sigue una función de densidad determinada. El origen del ruido es múltiple y existen tres tipos básicos: gaussiano, impulsional y multiplicativo.

El ruido más común se origina en el proceso de captación al convertir los fotones en carga eléctrica, debido a su carácter no determinista de esta transformación. Este tipo de ruido se explica mediante el modelo de función de densidad normal o gaussiana. El segundo tipo de perturbación se consigue como consecuencia de la saturación de carga que recibe un píxel ya sea por exceso o por defecto. A este tipo de ruido se le denomina de 'sal y pimienta'. La saturación se produce debido a la sensibilidad de las cámaras al infrarrojo y al encontrar sobre el escenario objetos que están calientes. Este segundo tipo de comportamiento se modela con una función de densidad de tipo impulsional. Por último, hay ruido generado por la falta de iluminación uniforme sobre la escena capturada. Este error es como si la imagen estuviera formada por la multiplicación de dos imágenes, por un lado la imagen que refleja la intensidad de

iluminación sobre los objetos y de otro la función de refracción de los cuerpos vistos en la escena. A esta categoría de ruido se le relaciona con los filtros homomórficos.

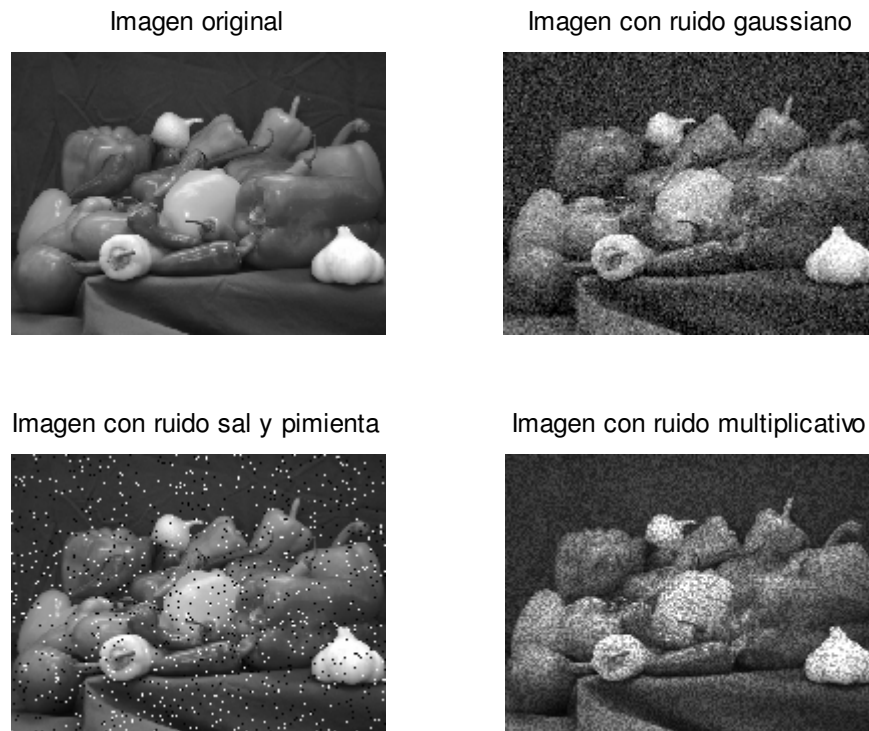


Figura 4. 5 Fuentes de error a) original b) gaussiano c)impulsional d)multiplicativo

#### 4.2.1 Mejora en la adquisición

La adquisición de las imágenes es un proceso no determinista. La señal de vídeo está definida por la interacción de distintas causas aleatorias. Se puede suponer que la imagen resultante,  $g(x,y)$ , puede ser explicada como la suma entre la imagen ideal,  $f(x,y)$  y el de ruido blanco,  $r(x,y)$ <sup>1</sup>:

$$g(x,y) = f(x,y) + r(x,y) \quad (4.13)$$

La ponderación de varias capturas sobre la escena producirá una disminución del ruido. Obviamente, se necesitará que durante el intervalo de tiempo de la adquisición de las imágenes, los objetos permanezcan quietos y la iluminación sea constante.

---

<sup>1</sup> Se define ruido blanco a una variable aleatoria con función de densidad normal, de media cero y varianza determina,  $N(0, \sigma)$ .



**Figura 4. 6. Dos imágenes adquiridas consecutivamente y la resta de ambas**

A medida de que se tomen más imágenes disminuirá la influencia del ruido. De hecho al aplicar el operador esperanza sobre la imagen real conducirá hacia la imagen ideal:

$$E(g(x, y)) = \lim_{n \rightarrow \infty} \frac{1}{n} \sum_{i=1}^n g_i(x, y) = f(x, y) \quad (4. 14)$$

El ruido queda cancelado con la ponderación de las imágenes. Este efecto se explica matemáticamente debido a que la suma de la variable aleatoria genera otra función de densidad probabilística de carácter normal, con media nula y varianza decreciente con el número de muestras:

$$N\left(0, \frac{\sigma}{\sqrt{n}}\right) \quad (4. 15)$$

El mayor inconveniente de esta técnica, en la práctica, consiste en mantener constante la iluminación durante la exposición, tener tiempo suficiente para adquirir las imágenes y que no aparezcan objetos en movimiento durante las capturas.

#### **4.2.2 Filtros paso bajo**

El espectro del ruido corresponde a componentes de alta frecuencia, por tanto, la aplicación de un filtro paso bajo significará la reducción del ruido sobre la imagen. El precio a pagar será una pérdida en la calidad de los detalles finos y de los bordes de la imagen. La aplicación del filtro paso bajo supone la cancelación parcial del ruido pero también pérdida de la propia información de la imagen.

Para la aplicación de los filtros se aplicará la relación fundamental de los sistemas discretos. No es necesario actuar sobre el espectro de la propia imagen para filtrarla frecuencialmente. Basta con aplicar una máscara de convolución cuya respuesta frecuencial corresponda a la cancelación del espectro deseado.

A razón de lo comentado en el anterior apartado, el primer filtro paso bajo sería tomar un promediado del entorno de vecindad:

$$f'(x, y) = \frac{1}{M \cdot N} \sum_{i=-M/2}^{M/2} \sum_{j=-N/2}^{N/2} f(x+i, y+j) \quad (4.16)$$

Por ejemplo, para una ventana de vecindad 3x3, la máscara de convolución sería:

$$\frac{1}{9} \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix} \quad (4.17)$$

Sin embargo, estos tipos de máscaras, las provenientes del promediado de vecindad, presentan lóbulos en su respuesta frecuencia, además de no tener la propiedad de ser de filtro de fase lineal, distorsionando gravemente la fase del espectro.

Las máscaras construidas mediante la función de densidad binomial discreta generan filtros paso bajo de fase lineal y no presentan lóbulos en el módulo. Sus coeficientes son obtenidos mediante el triángulo de Pascal:

| $n$ | $f$  | máscara   | $\sigma^2$ |
|-----|------|-----------|------------|
| 0   | 1    | 1         | 0          |
| 1   | 1/2  | 1 1       | 1/4        |
| 2   | 1/4  | 1 2 1     | 1/2        |
| 3   | 1/8  | 1 3 3 1   | 3/4        |
| 4   | 1/16 | 1 4 6 4 1 | 1          |

$n$  es el grado del filtro,  $f$  es el factor de escala y  $\sigma^2$  es la varianza efectiva de la máscara respecto al filtro de Gauss. Los filtros binomiales bidimensionales se construyen mediante la convolución discreta de un filtro binomial unidimensional con él mismo y su traspuesta:

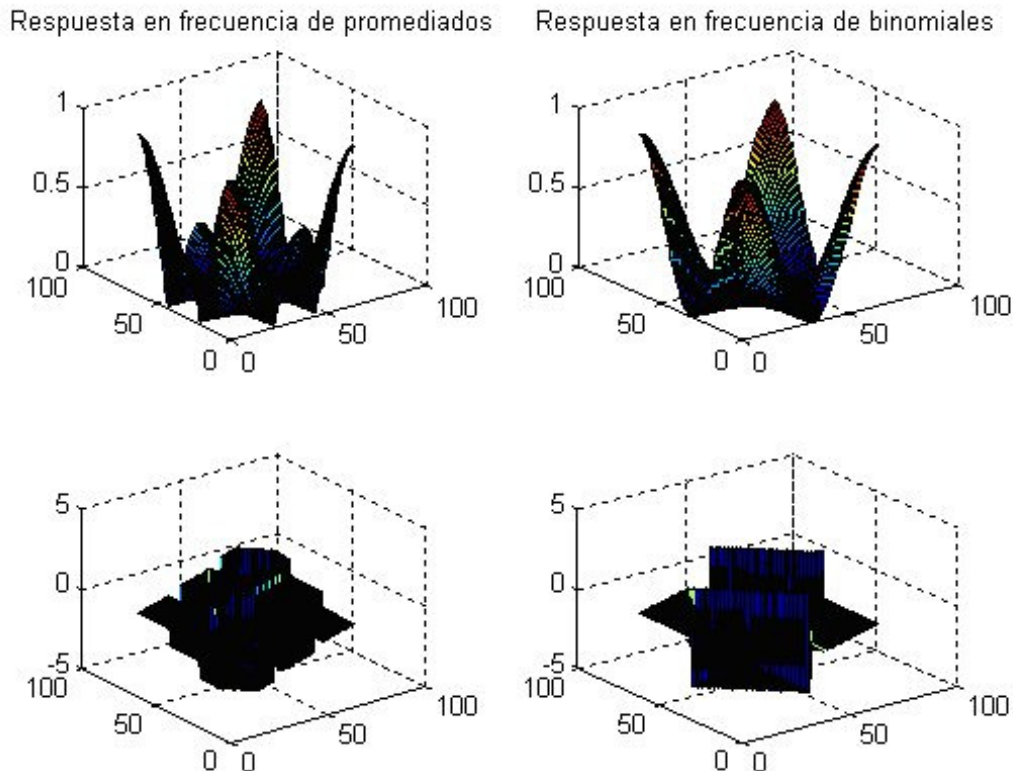
$$h_{2D}^n = h_x^n * h_y^n \quad (4.18)$$

Por ejemplo, el filtro 2D más pequeño, 3x3, se formará mediante la convolución de dos filtros 1D:

$$h_{2D}^2 = h_x^2 * h_y^2 = \frac{1}{4} \begin{pmatrix} 1 & 2 & 1 \end{pmatrix} * \frac{1}{4} \begin{pmatrix} 1 \\ 2 \\ 1 \end{pmatrix} = \frac{1}{16} \begin{pmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{pmatrix} \quad (4.19)$$

Esta propiedad de composición es posteriormente aplicada para el procesamiento de las imágenes, ahorrando tiempo en su ejecución: primero se pasa la máscara 1D por filas y luego por columnas. Desde el punto de vista espacial, nótese que la máscara presenta un promediado en la cual pondera la influencia de cada vecino en función de su

distancia respecto del píxel de pivotación. La figura adjunta muestra la respuesta en frecuencia de los filtros basados en la ponderación y los filtros bidimensionales.



### 4.2.3 Filtros gaussianos

Los filtros gaussianos son máscaras de convolución que emplean la discretización de las funciones de densidad normal de media cero y varianza dada,  $N(0, \sigma^2)$ :

$$h_{\sigma}(x, y) = e^{-\frac{(x^2 + y^2)}{2\sigma^2}} \quad (4.20)$$

Las respuestas en frecuencia de estas máscaras son filtros pasos cuyos anchos de banda dependen de sus varianzas. No presentan rizados o lóbulos secundarios en el módulo y son de fase lineal. Al igual que los filtros binomiales, las máscaras de Gauss bidimensionales se construyen a partir de las unidimensionales. Su ventaja, respecto a los filtros binomiales, está en su parametrización dependiendo de la varianza. A medida de que la varianza sea más elevada significará que tiene en cuenta a vecinos más alejados, por el contrario, si la varianza es más pequeña indicará que sólo ponderará con los más próximos. La varianza funciona como grado de ponderación en la vecindad. Ésta es la interpretación en el dominio espacial. Desde el punto de vista frecuencial, la varianza determina el ancho de banda del filtro paso bajo que supone la máscara de Gauss. El lóbulo principal del módulo de la respuesta frecuencia viene determinado por la varianza. A mayor varianza menor ancho de banda.

En el proceso de discretización de la máscara Gaussiana aparece el problema del número de coeficientes finitos a emular a la función continua. Para evitar la formación de lóbulos secundarios en la respuesta frecuencial de la máscara, se toma como regla que el tamaño del operador,  $w$ , debe ser al menos de:

$$w \geq 3c \quad (4.21)$$

Siendo  $c$  el tamaño del lóbulo central y dependiente de la varianza:

$$c = 2\sqrt{2 \cdot \sigma^2} \quad (4.22)$$

Esta regla indica que el 98% de los posibles casos de la variación de la variable aleatoria normal están comprendidos. Si la varianza es unitaria, la máscara debería de ser al menos  $9 \times 9$ . Nótese que se busca una máscara bidimensional que tenga simetría par. También hay que considerar que esta máscara tiene simetría radial, por lo que sólo es necesario de calcular un cuadrante de la máscara.

### **Ejemplo 4.3**

Determinar la máscara de Gauss para una varianza de 0.25 y 1.

El tamaño de la máscara para 0.25 será:

$$c = 2 \cdot \sqrt{\frac{2}{4}} = \sqrt{2} \Rightarrow w \approx 3$$

Los coeficientes de la máscara tendrá simetría radial y sólo se buscará para el radio igual a 1 y  $\sqrt{2}$ :

$$\frac{1}{1.6154} \begin{pmatrix} 0.0183 & 0.1353 & 0.0183 \\ 0.1353 & 1 & 0.1353 \\ 0.0183 & 0.1353 & 0.0183 \end{pmatrix}$$

Para el caso de varianza unitaria la máscara será de  $9 \times 9$ :

|        |        |        |        |        |        |        |        |        |
|--------|--------|--------|--------|--------|--------|--------|--------|--------|
| 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0001 | 0.0000 | 0.0000 | 0.0000 | 0.0000 |
| 0.0000 | 0.0000 | 0.0002 | 0.0011 | 0.0018 | 0.0011 | 0.0002 | 0.0000 | 0.0000 |
| 0.0000 | 0.0002 | 0.0029 | 0.0131 | 0.0215 | 0.0131 | 0.0029 | 0.0002 | 0.0000 |
| 0.0000 | 0.0011 | 0.0131 | 0.0586 | 0.0965 | 0.0586 | 0.0131 | 0.0011 | 0.0000 |
| 0.0001 | 0.0018 | 0.0215 | 0.0965 | 0.1592 | 0.0965 | 0.0215 | 0.0018 | 0.0001 |
| 0.0000 | 0.0011 | 0.0131 | 0.0586 | 0.0965 | 0.0586 | 0.0131 | 0.0011 | 0.0000 |
| 0.0000 | 0.0002 | 0.0029 | 0.0131 | 0.0215 | 0.0131 | 0.0029 | 0.0002 | 0.0000 |
| 0.0000 | 0.0000 | 0.0002 | 0.0011 | 0.0018 | 0.0011 | 0.0002 | 0.0000 | 0.0000 |
| 0.0000 | 0.0000 | 0.0000 | 0.0000 | 0.0001 | 0.0000 | 0.0000 | 0.0000 | 0.0000 |

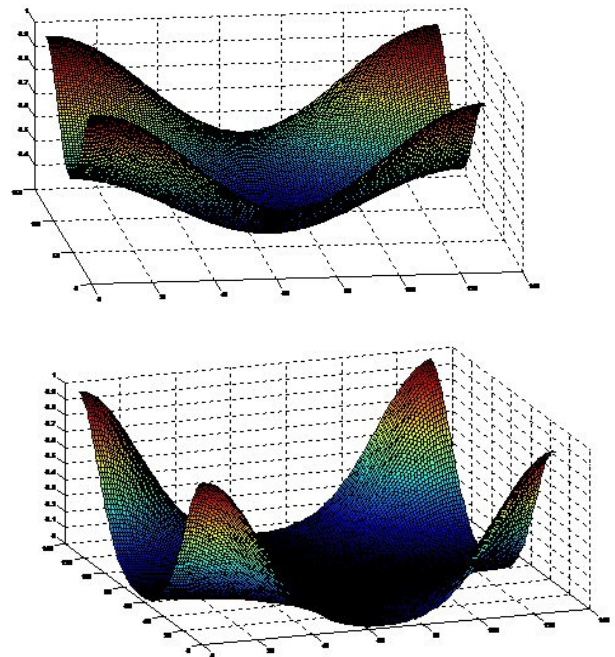
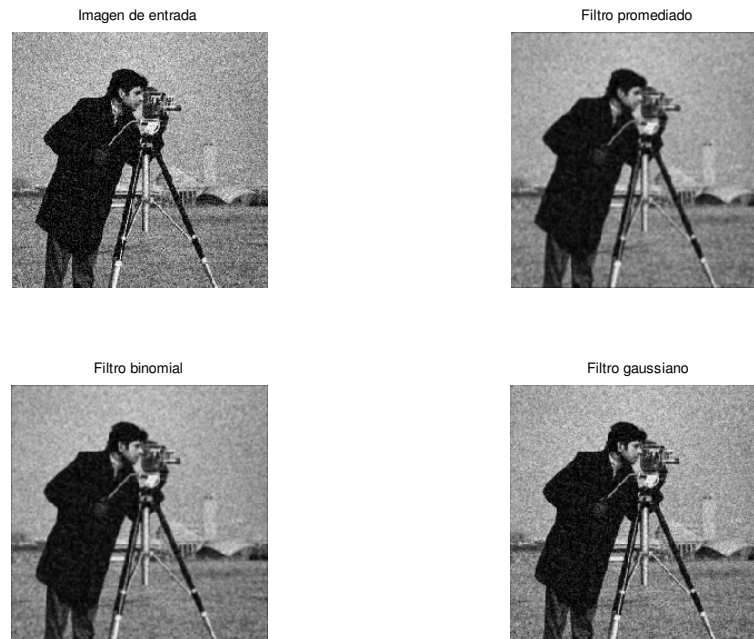


Figura 4. 7. Modulo de la respuesta en frecuencia de un filtro de Gauss a)  $\sigma^2=0.25$ , b)  $\sigma^2= 1$





**Figura 4. 8 Tipos de suavizado a) imagen original b)promediado c) binomial d) gaussiano con  $\sigma^2 = 0.25$**

#### **Resolución Matlab**

```
>>h1=fspecial('average')
>>h2=conv2([1 2 1],[1 2 1])/16
>>h3=fspecial('gaussian')
>>imagen= imread('cameraman.tif');imshow(imagen);
>>imagen1=imfilter(imagen,h1);imshow(imagen1);
>>imagen2=imfilter(imagen,h2);imshow(imagen2);
>>imagen3=imfilter(imagen,h3);imshow(imagen3);
```

Al igual que los filtros binomiales, los gaussianos bidimensionales también se construyen mediante la convolución discreta de un filtro gaussiano unidimensional con él mismo y su traspuesta. Esta propiedad es utilizada para disminuir el coste computacional.

#### **4.2.4 Filtros basados en la mediana**

Con el procesamiento lineal, empleando técnicas de filtrado paso bajo, se ha observado que al eliminar el ruido se han desdibujado los bordes y cancelado los pequeños detalles. Este efecto secundario suele ser reducido empleando el filtrado basado en la mediana del entorno de vecindad. Esta técnica de procesamiento es no lineal, con lo cual no se aplica mediante máscara de convolución. Su actuación se basa en elegir el valor de la mediana en el entorno de vecindad. Se define la mediana como el valor intermedio de la ordenación de los píxeles vecinos en función de su intensidad. Si los vecinos son  $2M+1$  píxeles, la mediana estará por encima de los  $M$  primeros valores



y por debajo de los  $M$  últimos. Por ejemplo, en un entorno de  $3 \times 3$ , la mediana será el nivel del colocado en quinta posición:

$$\begin{pmatrix} 89 & 87 & 14 \\ 92 & 75 & 56 \\ 99 & 78 & 90 \end{pmatrix}$$

$$\{89 \ 87 \ 14 \ 92 \ 75 \mid 56 \ 99 \ 78 \ 90\} \Rightarrow \{14 \ 56 \ 75 \ 78 \ 87 \mid 89 \ 90 \ 92 \ 99\}$$

En este ejemplo, la mediana será el nivel 87. En consecuencia, el ruido que tiene un valor atípico en el entorno quedará colocado en los extremos de la ordenación. Véase cómo el nivel 14 queda desplazado. La mediana se presenta como una buena técnica para el suavizado en general y en particular para la cancelación de ruido impulsional. El mayor inconveniente es su alto coste computacional. Cuando la aplicación de Visión tiene restricciones de tiempo suele ser prohibitivo el uso de la mediana, optando por el filtrado lineal.

Otro efecto adverso de la mediana es el desplazamiento de los bordes de la imagen. Si se desea precisión subpixel en la imagen, no es aconsejable esta técnica. Para evitar este inconveniente, se emplean entorno de vecindad  $N_4(p)$ . Si la vecindad está definida en las diagonales, preservará los bordes oblicuos. En caso de estar definida la vecindad en fila y columna, se mantendrá los bordes horizontales y verticales.

Comparación entre filtrado gaussiano y mediana con el ruido impulsional

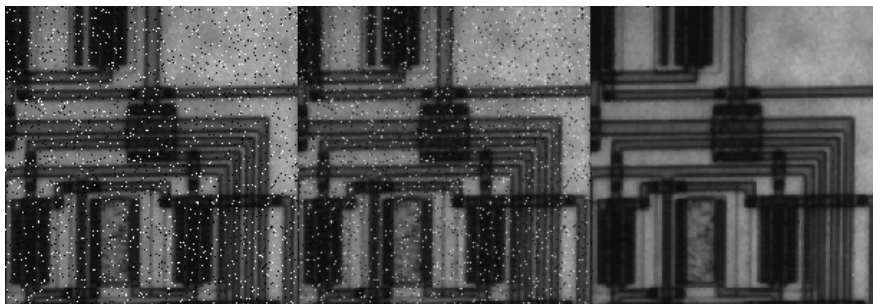


Figura 4. 9 a) imagen con ruido de sal y pimienta, b) filtrado gaussiano c) mediana

#### **Resolución Matlab**

```
>>h=fspecial('gaussian')
>>imagen= imnoise(imread('circuit.tif'),'salt & pepper');imshow(imagen);
>>imagen1=imfilter(imagen,h);imshow(imagen1);
>>imagen2=medfilt2(imagen);imshow(imagen2);
```

### **4.2.5 Filtros homomórficos**

El caso de iluminación no uniforme en la escena capturada es un ejemplo de ruido multiplicativo. La formación de la imagen depende básicamente de la iluminación sobre los objetos y del tipo de superficie de éstos. La intensidad en la imagen puede ser

modelada por la multiplicación entre la causa derivada de la iluminación y la causa proveniente del tipo de superficie:

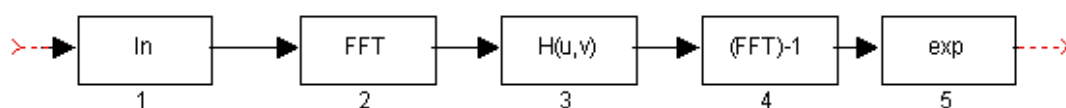
$$f(x, y) = i(x, y) \cdot o(x, y) \quad (4.23)$$

$f(x, y)$  será la intensidad del píxel,  $i(x, y)$  será el efecto de la iluminación y  $o(x, y)$  corresponderá a la función de reflectancia de la escena. Si se convolucionase la imagen con cualquier filtro, atacaría indistintamente a ambas componentes de iluminación y reflectancia. La forma de proceder es aplicar un sistema homomórfico capaz de separar los efectos y procesar por separado. Los sistemas homomórficos son operaciones de aplicar una función, procesar y aplicar su inversa. Por ejemplo, la función logarítmica y su inversa la exponencial.

Para el caso que ocupa, la forma de separar los efectos es mediante la función logarítmica. Al aplicarla, es posible obtener la respuesta frecuencial por separado de las componentes de iluminación y reflectancia:

$$\ln(f(x, y)) = \ln(i(x, y)) + \ln(o(x, y)) \Rightarrow F(\ln(f(x, y))) = F(\ln(i(x, y))) + F(\ln(o(x, y))) \quad (4.24)$$

El espectro del efecto de iluminación estará dominado por la baja frecuencia, ya que los cambios de iluminación en la escena son paulatinos, por el propio carácter continuo de la luz. Sin embargo, los bordes de las superficies, los contactos entre los objetos y sus intersecciones tendrán un perfil claramente de alta frecuencia. Por tanto, la aplicación de un filtro paso alto, atenuará la falta de iluminación uniforme y realzará los bordes entre los objetos. Esta algoritmia acaba empleando la función inversa, después de haber filtrado las dos componentes. En la figura se muestra el diagrama a bloques del algoritmo.



### 4.3 Detección de bordes

Estudios sobre el procesamiento visual humano indica que para la percepción de las escenas, el cortex visual utiliza información del color de los objetos, de sus texturas, de las sombras, pero sobre todo de los contornos o bordes de los objetos. Parece ser que existen neuronas, del sistema visual primario, especializadas en las discontinuidades de la intensidad lumínica o bordes. Según la teoría dominante, el cortex visual trabaja

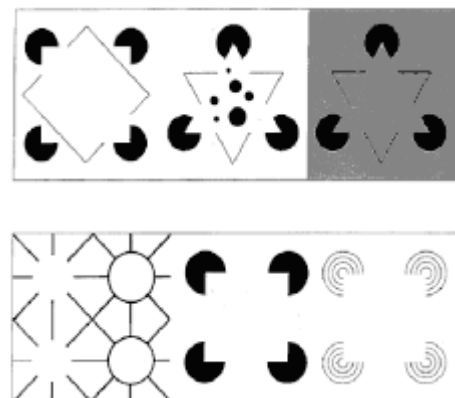


Figura 4. 10. Contornos subjetivos de Kanizsa

en una jerarquía de características visuales, donde agrupa los elementos primarios de los contornos en objetos geométricos más complejos hasta la interpretación de las figuras. Este subsistema biológico es capaz de completar bordes parcialmente ocluidos o explícitamente eliminados. En los contornos subjetivos de Kanizsa, figura 4.10, nótese como al visualizarlos quedan completos, aunque los objetos geométricos estén inacabados.

Otro experimento psicofisiológico es la capacidad de interpretar la imagen con sólo apreciar los bordes de la escena. En la figura adjunta se presenta la imagen con sólo los bordes. Es fácil de interpretar que es una escena de bodegón.

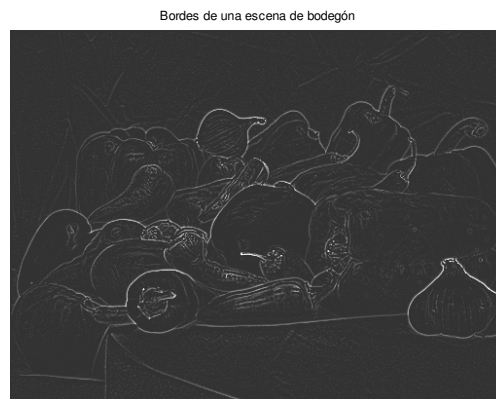


Figura 4.11. Imagen de contornos

Estos aspectos son aplicados y emulados en la Visión Artificial. La detección automática de bordes es una tarea esencial en muchas aplicaciones de interpretación de las imágenes. Suele ser una etapa muy empleada para las tareas de segmentación o para la búsqueda de objetos geométricos más complejos como líneas, elipses, esquinas, etc.

Se entiende como borde aquella región donde aparece una fuerte variación del nivel de intensidad en los píxeles adyacentes. Su causa principal es originada por la intersección de varios objetos, con diferentes niveles de reflectancia, que al ser proyectados sobre la cámara generan discontinuidades de intensidad en los píxeles correspondidos. Sin embargo, estas discontinuidades también aparecen de forma no deseada por la presencia del ruido, por el efecto de sombras sobre los propios objetos o por una iluminación no uniforme dentro escena.

El fundamento para la detección de los bordes está en la aplicación del operador derivada en un entorno de vecindad. Si se construye una imagen sintética con franjas de alto contraste y se adquiere una fila de la imagen, se observará una fuerte variación de la intensidad en el entorno de los bordes de las franjas. Al aplicar el operador derivada, se observa que ésta toma un valor de máximo o mínimo justamente cuando en la transición se pasa de cóncavo a convexo o viceversa, esto es, en el punto de inflexión del borde. Si en vez de emplear la primera derivada se realiza con la segunda, el punto de inflexión de la primera deriva coincidirá con un paso por cero (*zero-crossing*).

Ambos razonamiento son empleados para la detección de los bordes. Cuando

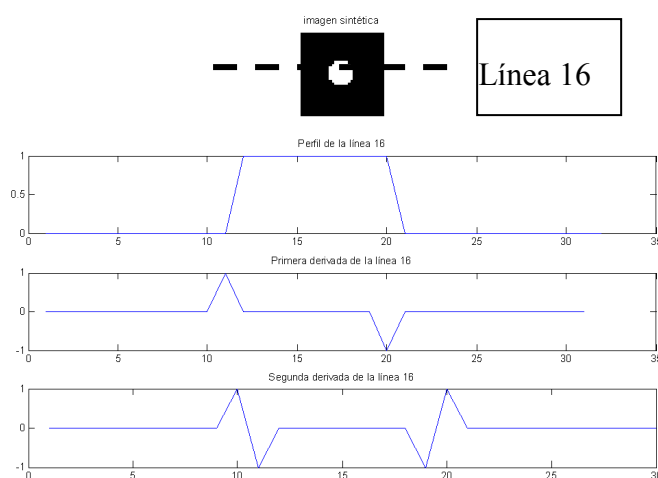


Figura 4.12. Bordes ideales

se aplica el operador gradiente en la imagen se localizará valores que tengan un gran valor, normalmente, en el módulo del gradiente. Por el contrario, al emplear la laplaciana se trata de detectar píxeles en la imagen que sean pasos por cero.

El problema reside en los bordes de las escenas reales, donde el modelo propuesto no está tan claramente definido y no se ajustan a la simplificación indicada. Las discrepancias entre el modelo de los bordes y lo almacenado en el computador, tras un proceso de formación de la imagen, está en:

- Las imágenes digitales son de carácter discreto y no continuo.
- La presencia de ruido en la imagen, la cual produce variaciones locales de intensidad, produce falsos bordes al aplicar los operadores derivadas. Por esta razón, la detección de bordes está ligada a etapas de eliminación del ruido.

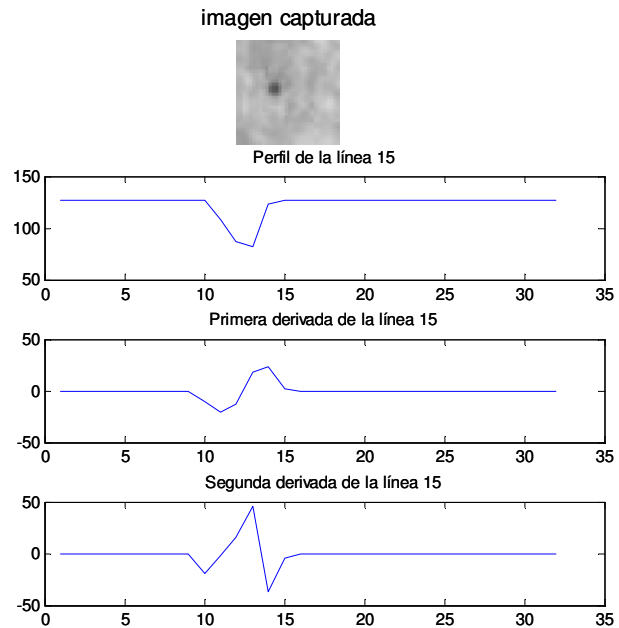


Figura 4. 13. Bordes reales

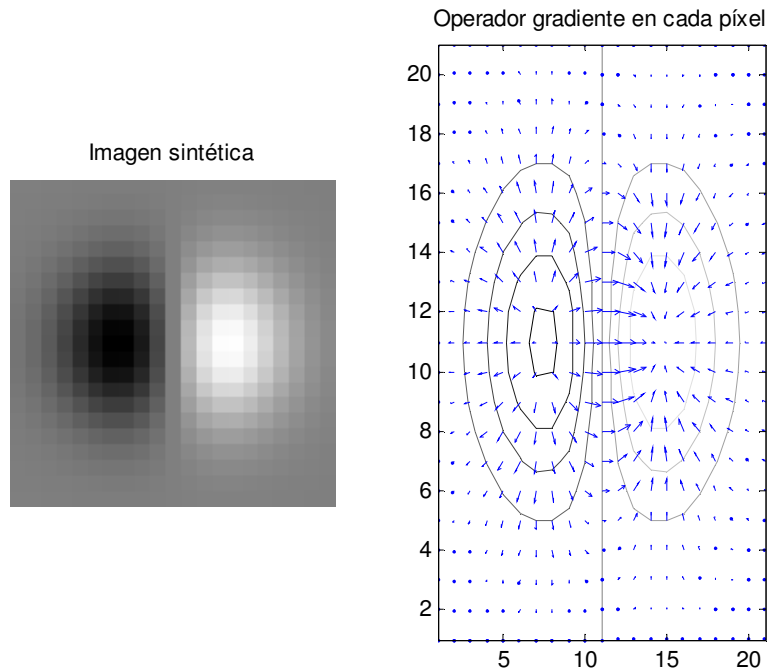
- Los orígenes diversos de los bordes, tales como oclusiones, superficies de diferentes orientaciones, cambios de texturas o de iluminación, reflejos, sombras, etc, hace que sea difícil la determinación de los bordes.

#### 4.3.1 Técnicas basadas en el operador gradiente

El operador gradiente aplicado sobre un píxel (x,y) de la imagen, retorna un vector que indica la dirección de máxima variabilidad de la intensidad luminosa y su nivel de variación:

$$\nabla f(x, y) = \begin{bmatrix} \frac{\partial f(x, y)}{\partial x} \\ \frac{\partial f(x, y)}{\partial y} \end{bmatrix} \Rightarrow \begin{cases} \|\nabla f(x, y)\| = \sqrt{\left(\frac{\partial f(x, y)}{\partial x}\right)^2 + \left(\frac{\partial f(x, y)}{\partial y}\right)^2} \\ \angle \nabla f(x, y) = \arctan\left(\frac{\frac{\partial f(x, y)}{\partial y}}{\frac{\partial f(x, y)}{\partial x}}\right) \end{cases} \quad (4.25)$$

Sobre una imagen sintética se ha aplicado el operador gradiente y se visualiza el vector resultante en cada píxel. Nótese cómo cada vector indica la dirección de máxima variabilidad en su entorno y su magnitud está relacionada con la intensidad de la variación. Al aplicar el operador divergencia sobre este campo vectorial retornará un campo escalar, de manera que los valores positivos indica la presencia de una fuente y los valores negativos un sumidero. Justamente la transición de una fuente a sumidero será la presencia del borde.



**Figura 4. 14. Efecto del operador derivada**

Un píxel se considerará que pertenece a un contorno si el módulo del gradiente supera un cierto umbral  $T$ :

$$g(x, y) = \begin{cases} 1 & |\nabla f(x, y)| > T \\ 0 & |\nabla f(x, y)| < T \end{cases} \quad (4. 26)$$

En la práctica, generalmente se establece como umbral,  $T$ , cuando el módulo del gradiente está por encima del 70% al 80% del valor máximo detectado en la imagen. Para el cálculo del módulo del gradiente se suele emplear la aproximación de la suma de sus derivadas parciales, mejorando el coste computacional:

$$|\nabla f(x, y)| \cong \left| \frac{\partial f(x, y)}{\partial x} \right| + \left| \frac{\partial f(x, y)}{\partial y} \right| \quad (4. 27)$$

La discretización del operador gradiente se basa en las diferencias de los niveles de grises en el entorno de vecindad. En el escenario numérico de aproximación de las

derivadas por las diferencias, éstas suelen emplear el truncamiento del desarrollo de Taylor obteniendo las *diferencias progresivas y regresivas*:

$$\begin{aligned}\frac{\partial f^+(x, y)}{\partial x} &\approx \frac{f(x + \Delta x, y) - f(x, y)}{\Delta x} & \frac{\partial f^+(x, y)}{\partial y} &\approx \frac{f(x, y + \Delta y) - f(x, y)}{\Delta y} \\ \frac{\partial f^-(x, y)}{\partial x} &\approx \frac{f(x, y) - f(x - \Delta x, y)}{\Delta x} & \frac{\partial f^-(x, y)}{\partial y} &\approx \frac{f(x, y) - f(x, y - \Delta y)}{\Delta y}\end{aligned}\quad (4.28)$$

Un promediado de ambas se consigue las *diferencias centradas*:

$$\frac{\partial f(x, y)}{\partial x} \approx \frac{f(x + \Delta x, y) - f(x - \Delta x, y)}{2\Delta x} \quad \frac{\partial f(x, y)}{\partial y} \approx \frac{f(x, y + \Delta y) - f(x, y - \Delta y)}{2\Delta y} \quad (4.29)$$

Existen diferentes máscaras de convolución que implementan estas diferenciaciones. En un entorno de 2x2 se usa el operador de Roberts y en una máscara de 3x3 se aproximan por el operador de Prewitt, Sobel y el de Fre-Chen. Considerando el caso de un entorno de vecindad 3x3, las derivadas parciales serían:

$$\begin{pmatrix} f_{-1,-1} & f_{-1,0} & f_{-1,1} \\ f_{0,-1} & f_{0,0} & f_{0,1} \\ f_{1,-1} & f_{1,0} & f_{1,1} \end{pmatrix} \rightarrow \begin{cases} \left. \frac{\partial f}{\partial x} \right|_{0,0} \equiv (f_{1,-1} + f_{1,0} + f_{1,1}) - (f_{-1,-1} + f_{-1,0} + f_{-1,1}) \\ \left. \frac{\partial f}{\partial y} \right|_{0,0} \equiv (f_{-1,1} + f_{0,1} + f_{1,1}) - (f_{-1,-1} + f_{0,-1} + f_{1,-1}) \end{cases} \quad (4.30)$$

Los incrementos de las variables independientes,  $\Delta x$  e  $\Delta y$ , no se consideran ya que son valores constantes correspondientes a la discretización espacial y en la detección de borde sólo importa el valor de módulo relativizado. De esta expresión se desprende las máscaras de Prewitt:

$$\begin{aligned} \begin{pmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{pmatrix} & \quad \begin{pmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{pmatrix} \\ \frac{\partial}{\partial x} & \quad \frac{\partial}{\partial y} \end{aligned} \quad (4.31)$$

El operador derivada resulta ser muy sensible con el ruido, ya que su respuesta frecuencial tiende a ser un filtro paso alto. Para reducir el realce de éste se propone una máscara que sea la convolución entre el operador derivada con un filtro binomial, obteniéndose las máscaras de Sobel:

$$S = D^T * B = \begin{bmatrix} -1 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} \Rightarrow \begin{bmatrix} \begin{pmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{pmatrix} & \begin{pmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{pmatrix} \\ \frac{\partial}{\partial x} & \frac{\partial}{\partial y} \end{bmatrix} \quad (4.32)$$

Mientras que Prewitt detecta mejor los bordes verticales, Sobel mejora su localización en los bordes diagonales. El operador isotrópico o de Frei-Chen intenta llegar a un compromiso entre ambos:

$$\begin{bmatrix} -1 & -\sqrt{2} & -1 \\ 0 & 0 & 0 \\ 1 & \sqrt{2} & 1 \end{bmatrix} \begin{bmatrix} -1 & 0 & 1 \\ -\sqrt{2} & 0 & \sqrt{2} \\ -1 & 0 & 1 \end{bmatrix} \\ \frac{\partial}{\partial x} \quad \frac{\partial}{\partial y} \quad (4.33)$$

Experimentalmente se ha encontrado que todos los operadores tienen resultados similares, y que la respuesta se va deteriorando de acuerdo con la cantidad de ruido presente en la imagen. Debido al menor coste computacional se suele elegir el operador de Sobel.

Máscaras de Prewitt y Sobel 3x3 en la dirección X



### **Resolución Matlab**

```
>>imagen= imread('cameraman.tif');imshow(imagen);
>>imagen1=imfilter(imagen,fspecial('prewitt'));imshow(imagen1);
>>imagen2= imfilter (imagen, fspecial('sobel'));imshow(imagen2);
```

### **4.3.2 Operadores basados en la laplaciana**

Para la detección de bordes, empleando la laplaciana, serán aquellos píxeles que sean paso por cero. El operador laplaciana se define en el plano continuo como:

$$\nabla^2 f(x, y) = \frac{\partial^2 f(x, y)}{\partial x^2} + \frac{\partial^2 f(x, y)}{\partial y^2} \quad (4.34)$$

Obsérvese que a diferencia del gradiente, la laplaciana no retorna una información vectorial sino un escalar. La laplaciana al aplicarla sobre una zona homogénea de intensidad, ésta será nula en toda la región. Por el contrario al aplicar este operador sobre un borde, aparecerán en sus alrededores valores positivos y negativos. Este comportamiento indicará la regla de actuación para la detección de los bordes. Se le aplicará a la imagen el operador laplaciana y en píxeles con entornos de valores positivos y negativos indicará la presencia de un borde. Por las propiedades de la segunda derivada, este operador es más exacto en la precisión del borde que el uso del gradiente (véase las figuras 4.12 y 4.13 de los bordes ideales y reales).

La discretización del operador laplaciana para un entorno de 3x3 estará constituida por la suma de las derivadas parciales de segundo orden:

$$\left. \begin{aligned} \frac{\partial^2 f(x, y)}{\partial x^2} &\cong \frac{f(x+1, y) - 2f(x, y) + f(x-1, y)}{\Delta x^2} \\ \frac{\partial^2 f(x, y)}{\partial y^2} &\cong \frac{f(x, y+1) - 2f(x, y) + f(x, y-1)}{\Delta y^2} \end{aligned} \right\} \Rightarrow \begin{pmatrix} 0 & -1 & 0 \\ -1 & 4 & -1 \\ 0 & -1 & 0 \end{pmatrix} \quad (4.35)$$

Esta laplaciana se conoce como vecindad a 4. Obsérvese que no se ha cambiado el signo y no se ha normalizado, ya que sólo interesa conocer píxeles que transitan en un entorno de positivos a negativos o viceversa. También se emplea la máscara a vecindad a 8:

$$\begin{pmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{pmatrix} \quad (4.36)$$

Al aplicarla, cualquiera de sus implementaciones, sobre una imagen real no habrá valores cero en los bordes sino que se aproximan a valores pequeños positivos o negativos, por el carácter discreto tanto de la señal como del operador. Eso sí, estos píxeles etiquetados de bordes, tendrán en su entorno valores positivos y negativos. Los bordes, con este operador, estarán formados por el ancho de un píxel a diferencia del método del gradiente. Además, el operador laplaciana es isotrópico a diferencia de las distintas máscaras que se implementan con el operador gradiente. Obsérvese, las diferencias entre la aplicación sobre una misma imagen del operador gradiente y del operador laplaciana. Los bordes detectados son más finos con el operador laplaciana y son de carácter isotrópico.





**Figura 4. 15 Diferencias entre el operador gradiente y el operador laplaciana**

#### **Resolución Matlab**

```
>>imagen= imread('cameraman.tif');imshow(imagen);
>>imagen1=imfilter(imagen,fspecial('sobel'));imshow(imagen1);
>>imagen2= imfilter (imagen, fspecial('log'));imshow(imagen2);
```

El mayor inconveniente del operador laplaciana es su sensibilidad al ruido. Para disminuir los falsos bordes, se hace interaccionar con un filtro paso bajo, dando paso al operador laplaciana de Gauss (LoG).

#### **4.3.2.1 Laplaciana de Gauss**

Marr al final de la década de los 70 sintetizó las diferentes teorías de visión biológica de los mamíferos. Según esta teoría existen diferentes fases en el proceso de la percepción y en la primera etapa se basa en la detección de los bordes. Se observó que este proceso estaba constituido por dos características:

1. Medida del gradiente de luminancia en diferentes escalas. Las diferentes causas de discontinuidades en la imagen son variadas, tal como se ha comentado al principio de este apartado (iluminación, texturas, sombras,...). Todos estos cambios se producen a distintas escalas espaciales, propias de las causas que las originan.
2. La localización de los bordes debe de hacerse con precisión. El ruido de la imagen desvirtúa la posición exacta, conllevando a un suavizado previo de la imagen. En el dominio frecuencia supone la delimitación del ancho de banda del detector de bordes. La actuación del operador debe estar delimitado en las frecuencias.

Estas condiciones de diferentes escalas en el dominio espacial y de ancho de banda delimitado en el dominio frecuencial, son contradictorias entre sí. Marr y Hildreth (1980) proponen una solución de compromiso con el operador laplaciana de Gauss. Éste es un filtro paso banda, mezcla del filtro paso bajo que supone el suavizado con un filtro de Gauss y su posterior procesamiento con el operador laplaciana. Desde el punto de vista formal, el operador queda definido como:

$$\nabla^2(f(x, y) * h_\sigma(x, y)) = f(x, y) * \nabla^2(h_\sigma(x, y)) = f(x, y) * LoG(x, y) \quad (4.37)$$

De la expresión se deduce la posibilidad de convolucionar la imagen de entrada con una sola máscara, originada ésta del proceso de discretización del operador *LoG*. En el dominio continuo este operador está definido como:

$$h_\sigma(x) = e^{-\frac{x^2}{2\sigma^2}} \quad \frac{dh_\sigma}{dx} = -\frac{x}{\sigma^2} e^{-\frac{x^2}{2\sigma^2}} \quad \frac{d^2h_\sigma}{dx^2} = \frac{x - \sigma^2}{\sigma^4} e^{-\frac{x^2}{2\sigma^2}} \quad (4.38)$$

La representación espacial del operador muestra una forma de ‘sombrero mejicano’. Notándose de la ecuación que tiene simetría radial, por tanto, sólo será necesario calcular un único cuadrante de los coeficientes. El resto son simétricos. Desde el punto de vista frecuencia, responde a un filtro paso banda bidimensional, cuyo ancho de banda depende de la varianza,  $\sigma^2$ .

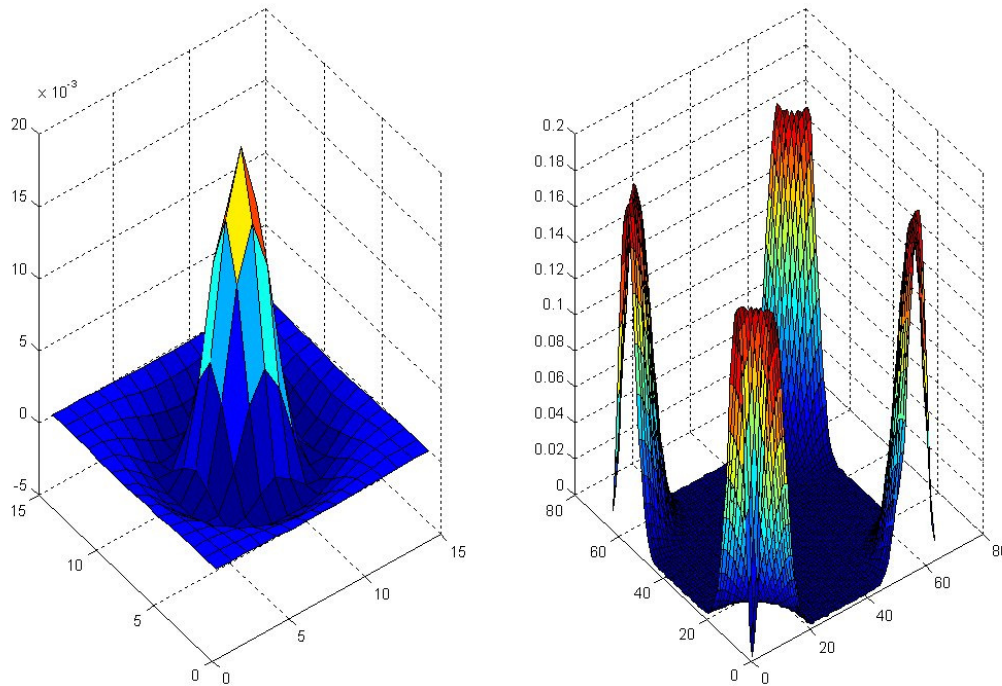


Figura 4. 16 a) máscara LoG 15x15 con varianza 2, b) Módulo de la respuesta en frecuencia

El mayor inconveniente de este operador es su alto coste computacional. Una alternativa para la reducción de tiempos es una aproximación basada en la diferencia de la imagen suavizada por dos varianzas distintas, DoG, muy relacionado con el aspecto de diferentes escalas espaciales:

$$f(x, y) * DoG(x, y) \cong (f(x, y) * h_{\sigma_1}(x, y)) - (f(x, y) * h_{\sigma_2}(x, y)) \quad (4.39)$$

Según Marr y Hildreth, este proceder DoG se asemeja al operador LoG, cuando las varianzas se encuentran en una relación de:

$$\frac{\sigma_2^2}{\sigma_1^2} = (1.6)^2 \quad (4.40)$$

En el proceso de discretización de la máscara de convolución del operador laplaciana de Gauss, ya sea mediante LoG o DoG, para evitar la aparición de lóbulos en la respuesta frecuencia de la máscara, el tamaño de ésta viene condicionado por la varianza:

$$w \geq 3c \quad c = 2\sqrt{2 \cdot \sigma^2} \quad (4.41)$$

La misma regla que ya se aplicó para el suavizado mediante máscaras gaussianas. No hay que olvidar que la varianza es la que delimita el ancho de banda del operador LoG.

### 4.3.3 Operador de Canny

Este algoritmo es ampliamente usado en la localización de contornos. Se caracteriza por evitar la ruptura de los bordes de los objetos. Su fundamento se basa en un proceso de optimización, teniendo en cuenta los siguientes objetivos a maximizar:

1. Aumentar la relación señal-ruido de la imagen.
2. Disminuir todo lo posible la distancia entre el borde detectado y el borde real.
3. No identificar un borde por un único píxel, sino por un conjunto de píxeles que tengan una cierta conectividad.

Según Canny, el operador óptimo está en la derivada de Gauss:

$$\nabla(f(x, y) * h_\sigma(x, y)) = f(x, y) * \nabla(h_\sigma(x, y)) = f(x, y) * DroG(x, y) \quad (4.42)$$

Donde el operador *DroG* es una combinación de suavizado y gradiente, haciendo de este operador menos vulnerable al ruido. Aquí también la varianza controlará el grado de suavizado deseado. El operador *DroG* quedará definido por:

$$DroG(x, y) = \begin{bmatrix} \frac{\partial h_{\sigma}(x, y)}{\partial x} \\ \frac{\partial h_{\sigma}(x, y)}{\partial y} \end{bmatrix} \quad (4. 43)$$

Pudiendo ser implementado en una sola máscara.

Los pasos que sigue el algoritmo de Canny son tres:

1. Calcular el módulo y el argumento del gradiente de una imagen suavizada aplicando el operador derivada de Gauss.
2. En la dirección del gradiente, eliminar puntos que no sean máximos locales del módulo (equivalente a encontrar el paso por cero en el operador LoG). Se eliminan los píxeles que no sean máximos locales, mejorándose la localización y evitando falsas detecciones.
3. Para la detección de los bordes se emplean dos umbrales de histéresis,  $T_1$  y  $T_2$ , siendo este último el mayor. Con  $T_2$  habría pocos píxeles de bordes y con  $T_1$  habría muchos píxeles de bordes.  $T_2$  es empleado para localizar las semillas de los bordes. A partir de los píxeles que superen el umbral  $T_2$ , se seguirá construyendo el borde mediante la adición de píxeles que sean mayor a  $T_1$  y que sean perpendiculares a la normal del borde.

En la figura adjunta, se ha pasado una imagen por cuatro operadores de bordes. Nótese cómo el operador de Canny presenta unos contornos cerrado, respecto al resto.

Detección de bordes de Prewitt, Sobel, LoG y Canny



Figura 4. 17 Detección de bordes a) Prewitt, b) Sobel, c)LoG y d) Canny

**Resolución Matlab**

```
>>imagen= imread('cameraman.tif');  
>>im1=edge(imagen, 'prewitt'); im1=edge(imagen, 'sobel');  
>>im1=edge(imagen, 'log'); im1=edge(imagen, 'canny');  
>> imshow([im3,im1;im2,im4])
```

## 4.4 Cuestiones

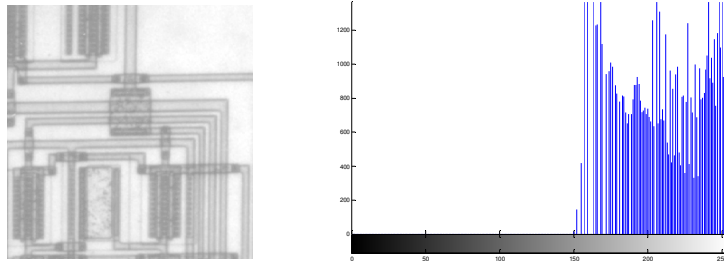
1. Determinar las posibles transformaciones para obtener la imagen negativa y para realizar corrección de gamma.
2. Tipos de ruido en una imagen digital.

3. Construir un filtro binomial de 5x5
4. Conseguir la máscara de convolución para realizar un énfasis de alta frecuencia, empleando un filtro binomial 3x3. La constante es A=1.5.
5. Realizar un filtro de Gauss con varianza unitaria.
6. Discrepancias entre los operadores basados en la primera derivada y en la segunda derivada.
7. Obtener la máscara de LoG para una varianza igual a 0.25.
8. Determinar las máscaras del operador derivada de Gauss para una varianza igual a 0.25.

## 4.5 Problemas

### Ejercicio 1

Dada la imagen de la figura, calcular la transformación mediante LUT para aumentar su contraste.



La imagen está saturada, por tanto, en una primera aproximación se podría una LUT de forma:

$$i_{out} = \frac{255}{255 - I_{min}} \cdot (i_{in} - I_{min})$$

Como dentro de la región es bastante uniforme  $\gamma$  sería 1.

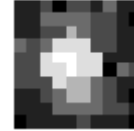
### Ejercicio 2

Para la siguiente figura, con 8 niveles de grises de cuantificación, obtener:

a) Histograma

b) Brillo y contraste

c) Ecualización del histograma

$$\begin{pmatrix} 1 & 1 & 2 & 0 & 2 & 2 & 1 & 3 & 1 & 1 \\ 1 & 1 & 3 & 3 & 3 & 3 & 2 & 2 & 1 & 0 \\ 1 & 1 & 2 & 2 & 5 & 3 & 2 & 2 & 1 & 1 \\ 3 & 2 & 3 & 6 & 6 & 6 & 2 & 2 & 1 & 1 \\ 2 & 2 & 6 & 7 & 7 & 6 & 6 & 1 & 1 & 1 \\ 2 & 2 & 6 & 6 & 7 & 6 & 6 & 0 & 3 & 2 \\ 2 & 2 & 2 & 6 & 5 & 5 & 3 & 2 & 1 & 1 \\ 1 & 1 & 2 & 3 & 5 & 5 & 3 & 2 & 1 & 0 \\ 1 & 1 & 1 & 3 & 3 & 3 & 3 & 2 & 2 & 1 \\ 0 & 1 & 1 & 1 & 1 & 2 & 1 & 1 & 1 & 0 \end{pmatrix}$$


| nivel(i) | 0    | 1    | 2    | 3    | 4 | 5    | 6    | 7    |
|----------|------|------|------|------|---|------|------|------|
| h(i)     | 6    | 33   | 26   | 16   | 0 | 5    | 11   | 3    |
| p(i)     | 0.06 | 0.33 | 0.26 | 0.16 | 0 | 0.05 | 0.11 | 0.03 |

$$\text{brillo} = \hat{\mu} = \sum_{i=0}^7 i \cdot p(i) = 2.45$$

$$\text{contraste} = \hat{\sigma}^2 = \sum_{i=0}^7 (i - \hat{\mu})^2 \cdot p(i) = 3.52$$

|                          |         |        |       |        |        |        |        |     |
|--------------------------|---------|--------|-------|--------|--------|--------|--------|-----|
| Nivel de gris de entrada | 0       | 1      | 2     | 3      | 4      | 5      | 6      | 7   |
| Nivel de gris de salida  | (-0.52) | (2.12) | (4.2) | (5.48) | (5.48) | (5.88) | (6.66) | (7) |
|                          | 0       | 2      | 4     | 5 ó 6  | 5 ó 6  | 6      | 7      | 7   |
| Histograma ecualizado    | 6       | 33     | 26    | 16     | 16     | 5      | 14     | 14  |

### Ejercicio 3

Dada la siguiente imagen, calcular para el píxel marcado el módulo y la dirección del gradiente, empleando los operadores de Sobel:

$$\begin{pmatrix} 1 & 1 & 10 & 10 & 10 \\ 1 & 1 & 10 \bullet & 10 & 10 \\ 1 & 1 & 10 & 10 & 10 \\ 1 & 1 & 10 & 10 & 10 \end{pmatrix}$$

Al aplicar el operador de Sobel sobre el píxel anotado, las derivadas parciales quedan definidas por:

$$\frac{\partial f(x, y)}{\partial x} = 0 \quad \frac{\partial f(x, y)}{\partial y} = 36$$

Por tanto, el módulo y dirección del gradiente será:

$$|\nabla f(x, y)| = 36 \quad \arg(\nabla f(x, y)) = \frac{\pi}{2}$$

### **Ejercicio 3**

Dada la siguiente imagen, calcular para el píxel marcado, la media y mediana en un entorno de 3x3:

$$\begin{pmatrix} 6 & 5 & 6 & 6 & 7 \\ 6 & 2 & 6 \bullet & 6 & 5 \\ 7 & 7 & 4 & 5 & 7 \\ 6 & 7 & 7 & 5 & 6 \end{pmatrix}$$

La media es 5.22 y la mediana corresponde a 6.

Derecho de Autor © 2008 Carlos Platero Dueñas.

Permiso para copiar, distribuir y/o modificar este documento bajo los términos de la Licencia de Documentación Libre GNU, Versión 1.1 o cualquier otra versión posterior publicada por la Free Software Foundation; sin secciones invariantes, sin texto de la Cubierta Frontal, así como el texto de la Cubierta Posterior. Una copia de la licencia es incluida en la sección titulada "Licencia de Documentación Libre GNU".

La Licencia de documentación libre GNU (GNU Free Documentation License) es una licencia con [copyleft](http://www.gnu.org/copyleft/fdl.html) para [contenidos abiertos](http://www.gnu.org/copyleft/fdl.html). Todos los contenidos de estos apuntes están cubiertos por esta licencia. La versión 1.1 se encuentra en <http://www.gnu.org/copyleft/fdl.html>. La traducción (no oficial) al castellano de la versión 1.1 se encuentra en <http://www.es.gnu.org/Licencias/fdles.html>