

Procesamiento Digital de Imágenes

Apuntes del curso impartido por el Dr. Boris Escalante R.

6 de Octubre de 2006

4. Realce de la imagen

Los procesos de realce de imágenes consisten de una serie de técnicas cuyo objetivo es mejorar la apariencia visual de una imagen, ya sea en contraste, ruido, escala de grises, distorsiones, luminosidad, falta de nitidez, etc., o bien convertir o mapear la imagen a una mejor forma para su análisis. El principal objetivo del realce de la imagen es procesar una imagen de tal manera que el resultado obtenido sea el apropiado para una aplicación específica. Los métodos de realce de imágenes los podemos dividir en dos categorías: los métodos de **realce en el dominio espacial** y los métodos de **realce en el dominio de la frecuencia**. Los métodos de la primera categoría consisten en la manipulación directa de los píxeles de la imagen mientras que los métodos de la segunda categoría corresponden a técnicas basadas en la representación de los píxeles, a través de una transformación hacia el dominio frecuencial y usa como operador de mapeo o transformación a la DFT.

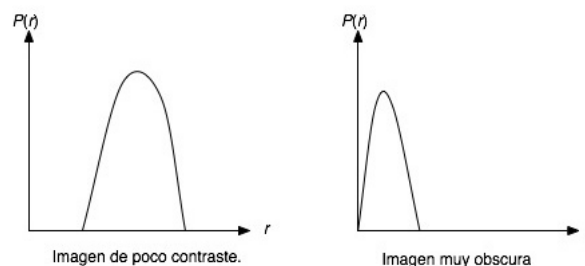
Dentro de los métodos de realce espacial podemos encontrar dos técnicas:

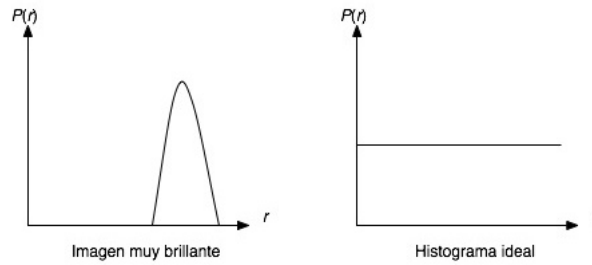
- El **realce radiométrico** donde las operaciones son efectuadas directamente sobre un píxel sin importar o tomar en cuenta a los píxeles vecinos. Sirve para mejorar condiciones de bajo contraste, baja luminosidad o demasiada oscuridad. Ejemplo: ecualización de histograma.
- El **realce con operaciones entre vecinos** que, a diferencia del realce radiométrico, las operaciones son efectuadas sobre un píxel pero tomando en cuenta a los píxeles que lo rodean. Sirve para eliminar ruido o para el mejoramiento de la nitidez. Ejemplo: kernel correspondiente a un filtro paso-bajas usando la convolución para realizar un **filtrado espacial**.

4.1. Realce punto a punto

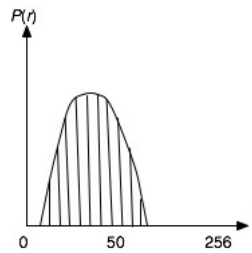
4.1.1. Histograma

El histograma de una imagen es la representación gráfica de la distribución que existe de las distintas tonalidades de grises con relación al número de píxeles o porcentaje de los mismos. La representación de un histograma ideal sería la de una recta horizontal, ya que eso nos indicaría que todos los posibles valores de grises están distribuidos de manera uniforme en nuestra imagen.





El histograma de una imagen digital con niveles de gris en el rango $[0, L - 1]$ es una función discreta $P(r) = n_r$, donde r representa un nivel de gris y n_r representa el número de pixeles que tienen ese valor de gris como se muestra en la siguiente figura:



Es muy frecuente normalizar un histograma dividiendo cada uno de sus valores por el número total de pixeles en la imagen, denotado por n . De tal modo que el histograma nos quedaría $P(r_k) = n_k/n$, para $k = 0, 1, \dots, L - 1$. De esta manera, $P(r_k)$ da una estimación de la probabilidad de la ocurrencia del nivel de gris r_k . Hay que notar que la suma de todos los componentes de un histograma normalizado es igual a 1.

4.1.2. Ecuación del histograma

La ecuación de un histograma consiste en encontrar una transformación con la cual el histograma tenga una representación uniforme, es decir, con la ecuación trataremos de igualar lo más posible el histograma de una imagen al histograma ideal. Consideremos por un momento funciones continuas. Sea r la variable que representa los niveles de gris de la imagen. Primeramente asumimos que nuestro histograma ha sido normalizado en el intervalo $[0, 1]$, donde $r = 0$ representa el nivel de gris negro y $r = 1$ el nivel de gris blanco.

Para cualquier r que satisfaga las condiciones anteriores, centramos nuestra atención en transformaciones de la forma:

$$s = T(r) \quad 0 \leq r \leq 1$$

la cual produce un nivel s para cada valor del pixel r en la imagen original. Sin embargo, para llevar a cabo esta ecuación es necesario establecer dos restricciones, ya que sin estas podríamos obtener resultados erróneos:

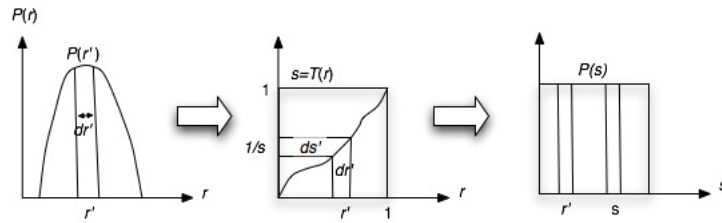
1. $T(r)$ debe ser **monotónicamente creciente**, esto indica que la pendiente de la función no puede ser negativa, ya que si fuera así, la imagen se vería afectada en su totalidad, por que lo que estaríamos haciendo sería invertirla en sus tonalidades.

$$\frac{ds}{dr} > 0$$

2. $T(r)$ debe tener inversa y debe ser **monotónicamente creciente**. Para resolver un problema como este se debe suponer que r y s son continuas y que además:

- r y s son variables aleatorias y continuas.
- $P(r)$ y $P(s)$ son funciones de densidad de probabilidad de r y s .

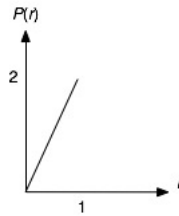
Figura 1: Ecuación de histograma



La ley de conservación de la probabilidad indica que

$$\begin{aligned}
 \int_0^1 P_r(r) dr &= 1 \\
 \Rightarrow \int_0^1 P_r(r) dr &= \int_0^1 P_s(s) ds \\
 P_s(s) ds &= P_r(r) dr |_{r=T^{-1}(s)} \\
 P_s(s) &= P_r(r) \frac{dr}{ds} |_{r=T^{-1}(s)=1} \\
 \frac{dr}{ds} &= \frac{1}{P_r(r)} \\
 \Rightarrow ds &= P_r(r) dr \\
 \Rightarrow s &= \int_0^r P_r(w) dw = T(r)
 \end{aligned} \tag{1}$$

Ejemplo: Encontrar $s = T(r)$ que ecualice $P_s(s)$:



Solución: $P_r(r) = 2r$ $0 \leq r \leq 1$

$$s = \int_0^r 2w dw = w^2 |_0^r = r^2$$

Comprobando

$$P_s(s) = P_r(r) \frac{dr}{ds} |_{r=T^{-1}(s)}$$

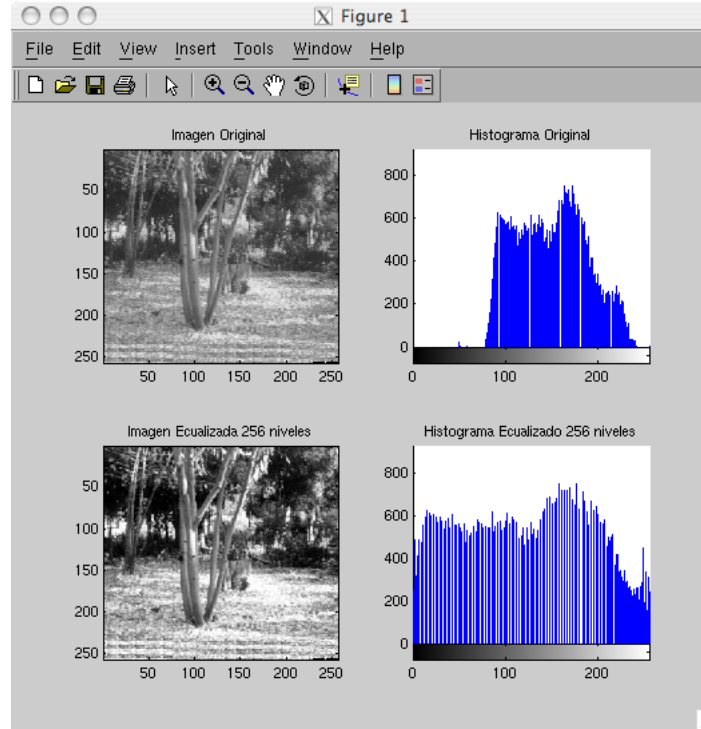
pero $\frac{ds}{dr} = 2r$ y $P_r(r) = 2r$ por lo tanto:

$$P_s(s) = 2r \frac{1}{2r} = 1$$

por lo que si se cumple.

La siguiente figura muestra un ejemplo de ecualización. Podemos observar del histograma de la imagen original que dicha imagen posee muchos niveles de gris altos, es decir, la imagen tiene mucho brillo. Después de la ecualización del histograma, podemos observar que los niveles de gris se distribuyen más uniformemente que en la imagen original.

Figura 2: Arriba: imagen antes de ecualizar y su histograma. Abajo: imagen después de ecualizar y su histograma



4.1.3. Especificación del histograma

Si ahora en lugar de ecualizar un histograma decidimos ajustarlo pero usando un histograma de otra imagen, es decir, se desea tener aproximadamente la misma distribución de una imagen con respecto a otra imagen que consideramos tiene un buen nivel de brillo. Esto tiene aplicaciones como por ejemplo unir varias imágenes que fueron tomadas en diferentes condiciones, como por ejemplo fotografía aérea.

Primeramente definimos las funciones de densidad de probabilidad $P_r(r)$ y $P_z(z)$ y las consideramos continuas como antes. Sean r y z los niveles de gris de la imagen de entrada y la imagen procesada respectivamente. $P_r(r)$ puede estimarse de la imagen de entrada dada mientras $P_z(z)$ es la función de densidad de probabilidad *especificada*. Como antes sea

$$s = \int_0^r P_r(w)dw = T(r) \quad (2)$$

la transformación para ecualizar la imagen de entrada para aproximar a un histograma ideal. Supongamos que definimos la variable z con la siguiente propiedad:

$$v = G(z) = \int_0^z P_z(t)dt \quad (3)$$

pero como deseamos que nuestro histograma se comporte o tenga una forma dada $s = v$, es decir, $T(r) = G(z)$ entonces z debe satisfacer las condición:

$$z = G^{-1}(s) = G^{-1}[T(r)] \quad (4)$$

La transformación $T(r)$ puede ser obtenida usando la Ec.(2) una vez que $P_r(r)$ ha sido estimada de la imagen de entrada y a su vez, la transformación $G(z)$ puede ser obtenida de la ecuación usando la Ec.(3) ya que $P_z(z)$ esta dada y es la especificación del histograma.

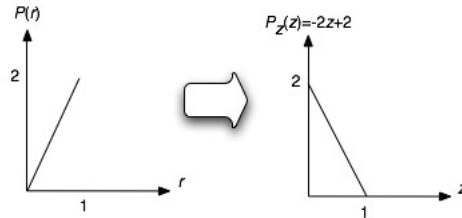
Asumiendo que G^{-1} existe y que satisface las condiciones 1 y 2 vistas en la subsección anterior, las Ecs. (2) y (3) muestran que una imagen con una función de distribución de probabilidad especificada puede ser obtenida usando el siguiente procedimiento:

1. Obtener $T(r)$ usando (2).

2. Usar la Ec. (3) para obtener la función de transformación $G(z)$.
3. Obtener la función de transformación inversa, $G^{-1}(z)$.
4. Aplicamos la Ec. (4) a todos los pixeles de la imagen de entrada.

El resultado de este procedimiento será una imagen cuyos niveles de gris z tendrán la función de densidad de probabilidad especificada, $P_z(z)$.

Ejemplo: Encontrar $z = F(r)$.



$$\begin{aligned}
 s &= \int_0^r P_r(w)dw \\
 &= \int_0^r 2w dw \\
 &= w^2 \Big|_0^r \\
 &= r^2
 \end{aligned}$$

y de la especificación tenemos:

$$\begin{aligned}
 v &= \int_0^z P_z(t)dt \\
 &= \int_0^z (-2t + 2)dt \\
 &= (-t^2 + 2t) \Big|_0^z \\
 &= -z^2 + 2z
 \end{aligned}$$

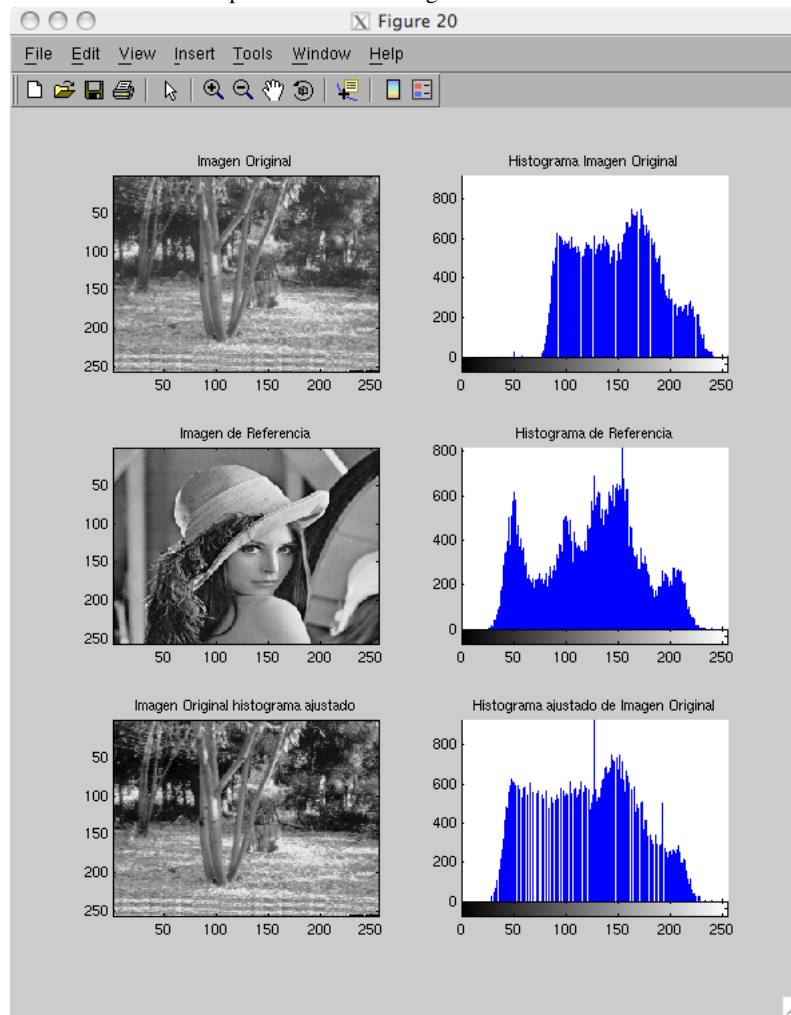
por lo que

$$\begin{aligned}
 -z^2 + 2z &= r^2 \\
 (z-1)^2 &= 1-r^2 \\
 z &= 1 \pm \sqrt{1-r^2}
 \end{aligned}$$

pero $z = 1 + \sqrt{1-r^2}$ no puede ser porque si evaluamos, cuando $r = 0$, queda $z = 2$ y ya no está dentro del intervalo donde está definida z , que es $0 \leq z \leq 1$. Por el contrario al evaluar r en $z = 1 - \sqrt{1-r^2}$, los valores para z si pertenecen al intervalo. Por lo tanto la respuesta es $z = 1 - \sqrt{1-r^2}$.

La siguiente figura muestra un ejemplo de especificación:

Figura 3: Arriba: imagen sin ajuste de histograma y su histograma. En el centro: imagen de referencia e histograma. Abajo: imagen ajustada usando la técnica de especificación de histograma.



4.2. Filtrado espacial

Los filtros espaciales tienen como objetivo modificar la contribución de determinados rangos de frecuencias de una imagen. El término espacial se refiere al hecho de que el filtro se aplica directamente a la imagen y no a una transformada de la misma, es decir, el nivel de gris de un pixel se obtiene directamente en función del valor de sus vecinos.

Los filtros espaciales pueden clasificarse basándose en su linealidad: filtros lineales y filtros no lineales. A su vez los filtros lineales pueden clasificarse según las frecuencias que dejen pasar: los filtros paso bajo atenúan o eliminan las componentes de alta frecuencia a la vez que dejan inalteradas las bajas frecuencias; los filtros paso alto atenúan o eliminan las componentes de baja frecuencia con lo que agudizan las componentes de alta frecuencia; los filtros paso banda eliminan regiones elegidas de frecuencias intermedias.

La convolución es una operación por la cual se lleva a cabo una acción de filtrado. Como veremos más adelante y como se ha visto en las propiedades de la Transformada de Fourier, existe una relación entre el filtrado espacial y el filtrado en el dominio de la frecuencia (con una restricción a tomar en cuenta). En general, el filtrado de señales posee tres categorías, según el resultado que se busque:

- Filtros Paso-Bajas: Son utilizados en la reducción de ruido; suavizan y aplanan un poco las imágenes y como

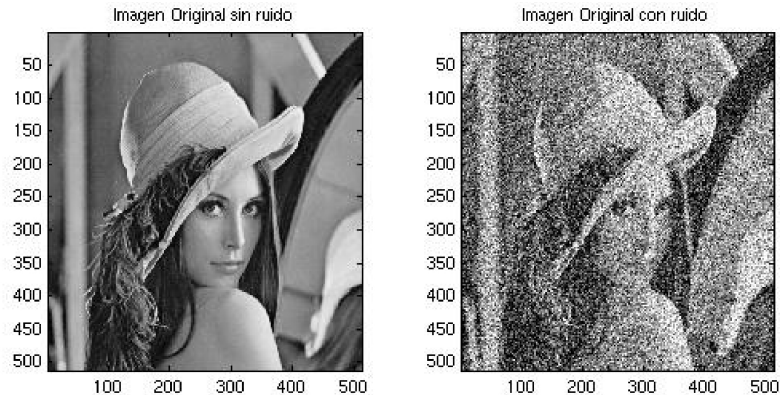
consecuencia se reduce o se pierde la nitidez. En inglés son conocidos como Smoothing Spatial Filters.

- Filtros Paso-Altas: Estos filtros son utilizados para detectar cambios de luminosidad. Son utilizados en la detección de patrones como bordes o para resaltar detalles finos de una imagen. Son conocidos como Sharpening Spatial Filters
- Filtros Paso-Banda: Son utilizados para detectar patrones de ruido. Ya que un filtro paso-banda generalmente elimina demasiado contenido de una imagen casi no son usados. Sin embargo, los filtros paso-banda son útiles para aislar los efectos de ciertas bandas de frecuencias seleccionadas sobre una imagen. De esta manera, estos filtros ayudan a simplificar el análisis de ruido, razonablemente independiente del contenido de la imagen.

4.2.1. Suavizado

Los filtros paso-bajas son utilizados para difuminar y reducir ruido en las imágenes, a este proceso se le conoce en inglés como *smoothing*. El difuminado (*blurring*) es usado en etapas de preprocesamiento desde la eliminación de pequeños detalles hasta la extracción de objetos y rellenado de pequeños huecos en líneas y curvas. La reducción de ruido puede ser completada por el difuminado usando filtros lineales o bien con un filtrado no lineal.

Figura 4: Imagen original e imagen con ruido.



La salida (respuesta) de un filtro paso-bajas lineal simplemente es un tipo de promedio de los pixeles contenidos en la vecindad de la mascara del filtro. Estos filtros son frecuentemente llamados filtros promediadores. La idea detrás de estos filtros es la de reemplazar cada pixel en la imagen por un promedio de de los niveles de gris de los vecinos definidos por la mascara del filtro.

Figura 5: Pérdida de nitidez.



A continuación se muestra un ejemplo de filtro paso-bajas de tamaño 3x3 llamados **filtros de bloque**:

$$\frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

Usando los filtros de bloque para la reducción del ruido, como por ejemplo en la imagen con ruido de la figura 4, se pueden obtener los resultados como los que se muestran en la figura 6.

En el diseño de los filtros es común que los kernels o máscaras sean generados a partir de ciertas funciones que tengan un comportamiento similar al deseado en el espacio continuo. En el caso de los filtros paso-bajas es común usar

Figura 6: Imagen con ruido suavizada usando un filtro de bloque de 7x7.



la función Gaussiana la cual es aproximada en su forma discreta a través de los **filtros binomiales** obtenidos a través de la función binomial o de los coeficientes del triángulo de Pascal. La función Gaussiana tiene la siguiente expresión :

$$f(x) = Ne^{-ax^2}$$

donde N es una constante de normalización que depende de a pero no de x . Hay varias posibilidades para el exponente a . En la forma estándar, $a = \frac{1}{2\sigma^2}$.

Los filtros binomiales se generan a partir de la función binomial o bien del triángulo de Pascal. La función binomial esta definida como:

$$f_N(x) = \binom{N}{x} = \frac{N!}{x!(N-x)!} \quad (5)$$

para $x = 0, 1, \dots, N$. Donde N es el orden del filtro deseado.

Ejemplo 1: se desea diseñar un filtro en una dimensión de orden $N = 2$.

$$f_2(x) = \frac{2!}{x!(2-x)!}$$

para $x = 0$, $f(0) = \frac{2!}{0!(2!)}$ = 1; para $x = 1$, $f(1) = \frac{2!}{1!(2-1)!} = \frac{2!}{1!(1)!} = 2$; y para $x = 2$, $f(2) = \frac{2!}{2!(0)!} = 1$. Por lo que el filtro en una dimensión queda:

$$f_2(x) = [1 \quad 2 \quad 1]$$

En el triángulo de Pascal tendríamos lo siguiente:

N					
1			1		
2		1	2	1	
3	1	3	3	1	
4	1	4	6	4	1

Se deja como ejercicio al lector comprobar que $f_3(x) = [1 \quad 3 \quad 3 \quad 1]$

■ Propiedades de los filtros binomiales:

- Son separables, es decir, en dos dimensiones podemos aplicar un filtro de un mismo orden primero en la dirección x y luego en la dirección y o viceversa.
- La convolución de un filtro de orden N consigo mismo produce un filtro de orden $2N$. Ejemplo:

$$f_2(x) * f_2(x) = f_4(x)$$

$$[1 \quad 2 \quad 1] * [1 \quad 2 \quad 1] = [1 \quad 4 \quad 6 \quad 4 \quad 1]$$

Los filtros binomiales en 2 dimensiones se obtienen de la siguiente manera:

$$[f_N(x)]^T \times [f_N(x)]$$

para el caso del filtro de orden $N = 2$ tendríamos:

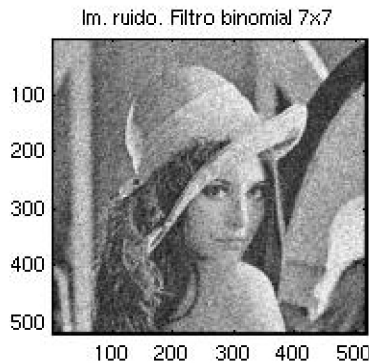
$$\begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} \times [1 \quad 2 \quad 1] = \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

Es común normalizar los filtros para no alterar la luminancia de la imagen, ya que el resultado es una suma y por lo tanto se puede llegar a saturar. En el caso de nuestro ejemplo tendríamos el filtro binomial paso-bajas de orden $N = 2$ normalizado:

$$\frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}$$

El resultado de aplicar un filtro suavizador binomial a una imagen con ruido como el de la figura 4 se muestra a continuación:

Figura 7: Imagen con ruido suavizada usando un filtro binomial de 7x7.



4.2.2. Filtros basados en derivadas de la función Gaussiana

Los filtros basados en derivadas de Gaussianas fueron en un principio obtenidos de manera heurística. Sin embargo estos filtros tienen un fundamento matemático y un comportamiento muy similar al sistema de visión humano (HVS); por ello son muy importantes ya que se especializan en la detección de cambios bruscos como los bordes.

Como se había visto, la función Gaussiana tiene la siguiente expresión :

$$f(x) = Ne^{-ax^2} \quad (6)$$

donde N es una constante de normalización que depende de a pero no de x . Hay varias posibilidades para el exponente a . En la forma estándar, $a = \frac{1}{2\sigma^2}$. La primera derivada de Gaussiana tiene la siguiente forma:

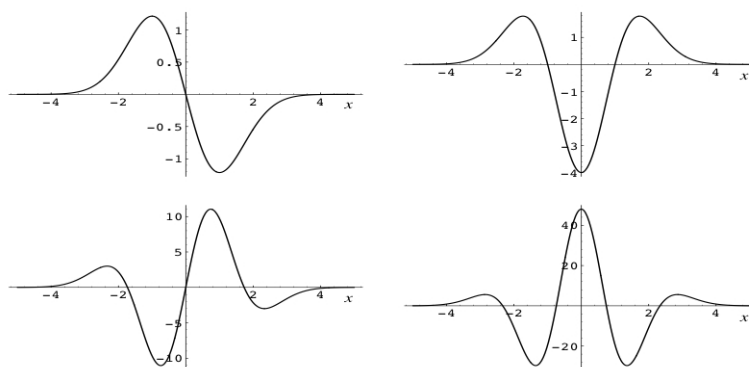
$$f(x) = -N2a(a)e^{-ax^2} \quad (7)$$

y la segunda derivada la expresión siguiente:

$$f(x) = N2a(2ax^2 - 1)e^{-ax^2} \quad (8)$$

Las siguientes figuras muestran las cuatro primeras derivadas de la función Gaussiana con $N = \sigma = 1$:

Figura 8: Primeras cuatro derivadas de Gaussiana.



Las derivadas de una función digital o discreta son definidas en términos de diferencias. La forma discreta de la primera derivada de Gaussiana se puede expresar de la siguiente manera:

$$f(x) - f(x-1)$$

Ahora bien, se desea diseñar un filtro espacial tal que $g(x) = h(x) * f(x)$ y $g(x) = f(x) - f(x-1)$. Como se había visto en clase,

$$f(x) = f(x) * \delta(x)$$

y

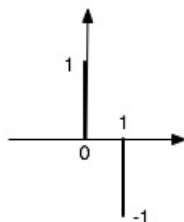
$$f(x-1) = f(x-1) * \delta(x-1)$$

por lo tanto, el filtro deseado $h(x)$ tiene la siguiente expresión:

$$h(x) = \delta(x) - \delta(x-1)$$

y tiene la siguiente representación en el plano

Figura 9: Filtro $h(x) = \delta(x) - \delta(x-1)$.



y puede escribirse como:

$$h(x) = [1 \ -1]$$

Si ahora agregamos un cero a nuestro filtro anterior, es decir:

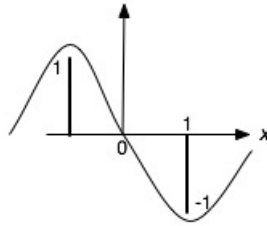
$$h(x) = [1 \ 0 \ -1]$$

podemos reescribir a $h(x)$ como:

$$h(x) = \delta(x+1) - \delta(x-1)$$

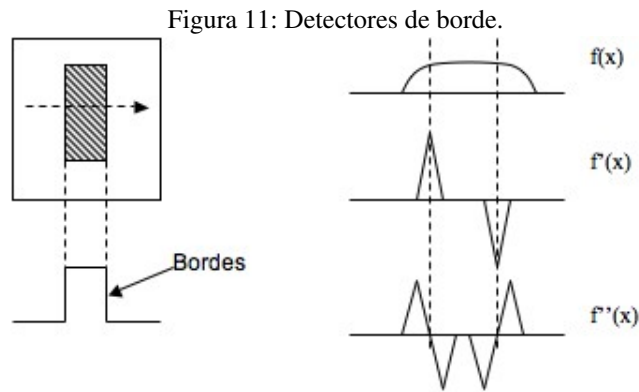
cuya representación en el plano sería:

Figura 10: Filtro $h(x) = \delta(x+1) - \delta(x-1)$.



el cual posee ya un comportamiento discreto de la primera derivada de Gaussiana en el plano continuo y estaríamos en la posibilidad de detectar cambios en la dirección x .

Estos filtros detectores de discontinuidades o cambios, basados en las derivadas de Gaussianas, tienen regularmente aplicación en tareas previas a la segmentación como detectores de bordes.



Para el caso de imágenes, la mascara utilizada sería de la siguiente forma:

$$\begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix}$$

y tendría la característica de detectar cambios en la dirección X de la imagen. El filtro anterior se conoce como **filtro Prewitt**.

Una variación del filtro anterior es el siguiente:

$$\begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}$$

Este filtro es conocido como **filtro Sobel** (en este caso también en la dirección X). La razón por la cual este filtro usa un peso de 2 al centro es para obtener algo de suavizado con lo cual se logra cierta reducción de ruido. Los filtros Sobel son también detectores de bordes. En un principio estos filtros se obtuvieron heurísticamente, pero posteriormente se verificó que su funcionamiento correspondía a la aproximación de una binomial de primer orden para una primera derivada de una Gaussiana. A continuación se muestran los filtros Prewitt y Sobel en la dirección Y :

$$\begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix} \quad \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

Al igual que en los filtros paso-bajas o de suavizado, también es posible aproximar a las derivadas de Gaussianas usando la función binomial. Los filtros generados a partir de estos se llaman también filtros binomiales. La forma

general de la función binomial para cualquier orden r de la derivada (o de diferencia) de Gaussiana y para cualquier dimensión o longitud N del filtro se muestra a continuación:

$$f_N^r(x) = \nabla^r \binom{N-r}{x} \quad (9)$$

donde

$$\nabla f(x) = f(x) - f(x-1) \quad (10)$$

y

$$f_N(x) = \binom{N}{x} = \frac{N!}{x!(N-x)!} \quad (11)$$

es un caso especial de una familia de los filtros binomiales cuando $r = 0$. Algo que hay que notar es que las primeras derivadas de Gaussiana son utilizadas para la detección de bordes mientras que las segundas derivadas de Gaussiana indican si un pixel perteneciente a un borde esta dentro del nivel de gris oscuro o claro, llamándose a esta propiedad, cruces por cero.

Ejemplo: Encontrar el filtro binomial de dimensión $N = 2$ cuya derivada es de orden $r = 0$.

$$f_N^0(x) = \nabla^0 \binom{N-0}{x} = \binom{N}{x}$$

si recordamos, la derivada de orden $r = 0$ corresponde a la función binomial vista en los filtros paso-bajas y entonces:

$$f_2(x) = \frac{2!}{x!(2-x)!}$$

con lo cual obtenemos: para $x = 0$, $f(0) = \frac{2!}{0!(2!)!} = 1$; para $x = 1$, $f(1) = \frac{2!}{1!(2-1)!} = \frac{2!}{1!(1)!} = 2$; y para $x = 2$, $f(2) = \frac{2!}{2!(0)!} = 1$. Por lo que el filtro en una dimensión queda:

$$f_2(x) = [1 \quad 2 \quad 1]$$

Ejemplo: Encontrar el filtro binomial de dimensión $N = 2$ cuya derivada es de orden $r = 1$.

$$f_N^1(x) = \nabla^1 \binom{N-1}{x} = \nabla \binom{N-1}{x}$$

usando $\nabla f(x) = f(x) - f(x-1)$ tenemos que:

$$f_N^1(x) = \nabla \binom{N-1}{x} = \binom{N-1}{x} - \binom{N-1}{x-1} = \binom{1}{x} - \binom{1}{x-1}$$

y entonces tenemos:

$$f_2^1(0) = \frac{1}{0!(1-0)!} - \frac{1}{-1!(1-(0-1))!} = 1$$

$$f_2^1(1) = \frac{1}{1!(1-1)!} - \frac{1}{0!(1-(1-1))!} = 0$$

$$f_2^1(2) = \frac{1}{2!(1-2)!} - \frac{1}{1!(1-(2-1))!} = -1$$

por lo que el filtro resultante sería:

$$h(x) = [1 \quad 0 \quad -1]$$

De la misma forma, con un triángulo de Pascal modificado se pueden encontrar los coeficientes del filtro que aproximan a la derivada de Gaussiana, en este caso para una primera derivada de Gaussiana:

$$\begin{array}{cccccc} & & & & & N \\ & & & & & 1 & -1 \\ & & & & & 2 & -2 \\ & & & & & 3 & -3 \\ & & & & & 4 & -4 \\ & & & & & 5 & -5 \\ & & & & & 6 & -6 \\ & & & & & 7 & -7 \\ & & & & & 8 & -8 \\ & & & & & 9 & -9 \\ & & & & & 10 & -10 \end{array}$$

También existe una forma recursiva de encontrar los coeficientes y está dada por:

$$f_N^{r+1} = -f_N^{r+1}(x-1) + f_N^r(x) - f_N^r(x-1) \quad (12)$$

donde el caso base indica que:

$$f_N^r(-1) = 0 \quad \text{y} \quad f_N^0(x) = \begin{pmatrix} N \\ x \end{pmatrix} \quad (13)$$

■ Estos filtros funcionan muy bien debido a que:

- Son separables.
- Están inspirados en la función Gaussiana.
- Tienen derivada de Gaussiana.
- Los perfiles coinciden con la forma en que trabaja el HVS.

Por último cabe señalar que, por ejemplo, para construir un filtro Sobel X se multiplica la aproximación de la Gaussiana por la aproximación de su primera derivada:

$$h_x(x,y) = \begin{bmatrix} 1 \\ 2 \\ 1 \end{bmatrix} \times [1 \quad 0 \quad -1] = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}$$

y de manera similar, para obtener el filtro Sobel Y se multiplica la aproximación de la Gaussiana de la primera derivada por la aproximación de la Gaussiana.

$$h_y(x,y) = \begin{bmatrix} 1 \\ 0 \\ -1 \end{bmatrix} \times [1 \quad 2 \quad 1] = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

Otros tipos de filtros también importantes son los **filtros laplacianos**. Los filtros laplacianos no existen para la primera derivada. El Laplaciano es un operador de tipo escalar formado por las segundas derivadas de ambas direcciones (X e Y). Por lo tanto es capaz de detectar bordes, pero no en máximos y mínimos (como el caso de los filtros Sobel) sino en los cruces por cero. Este filtro se desarrolló por completo empíricamente y posteriormente se le encontró parecido con la segunda derivada de la Gaussiana. La aproximación del Laplaciano es:

$$\begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix}$$

Las siguientes figuras muestran el resultado de aplicar algunos de los filtros discutidos:

Figura 12: Filtro Prewitt aplicado en la dirección X a una imagen sin ruido y a una imagen con ruido.

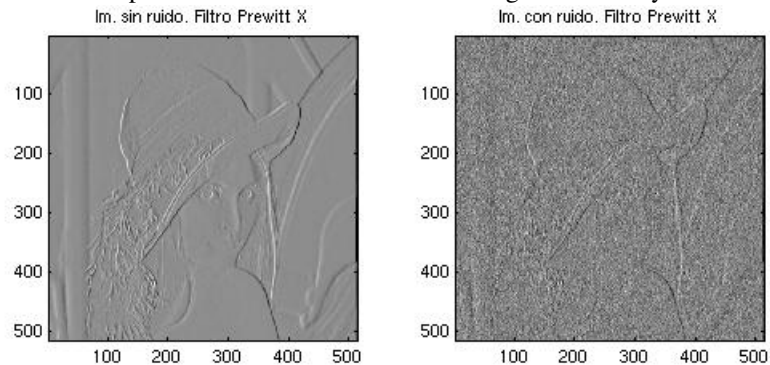


Figura 13: Filtro Sobel aplicado en la dirección X a una imagen sin ruido y a una imagen con ruido.

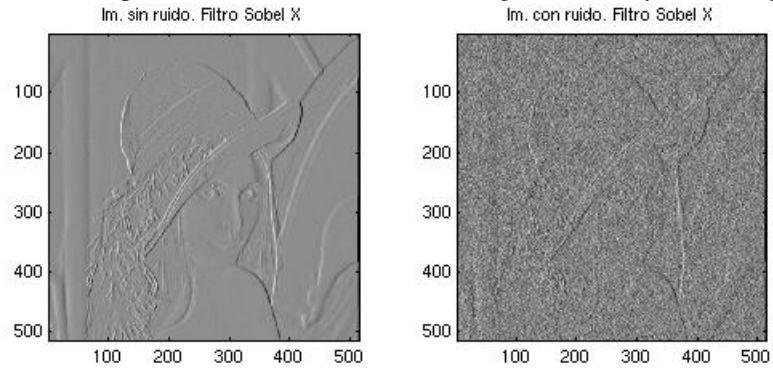
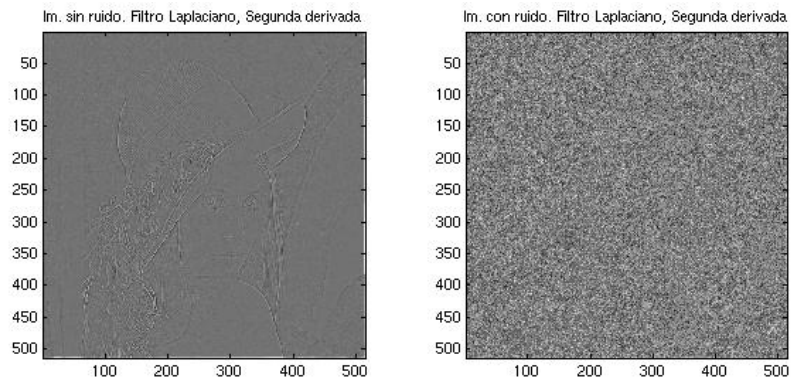


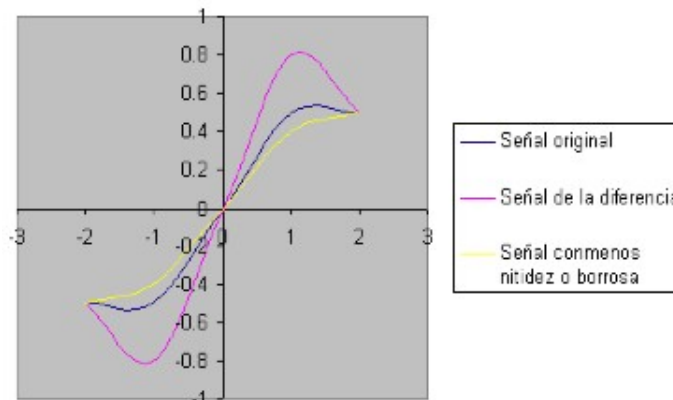
Figura 14: Filtro Laplaciano aplicado a una imagen sin ruido y a una imagen con ruido.



4.2.3. Mejoramiento de la nitidez

El mejoramiento de la nitidez o de la calidad visual de una imagen basado en los **filtros unsharp masking** tiene bastante importancia en el procesamiento digital de imágenes. Si se quisiera traducir al español, unsharp masking podría interpretarse como enmascaramiento (masking) de imagen borrosa (unsharp). Se basa en el hecho de que se tiene una imagen borrosa (con pendiente pequeña), y se le resta una pendiente aún más pequeña (f_{PB}). A su vez esto va multiplicado por un factor k el cual se recomienda tome valores entre 1 y 3.

Figura 15: Unsharp masking.



Después se suman la señal de la diferencia con la original para obtener una pendiente mayor y por lo tanto más resolución. Lo anterior se puede expresar como:

$$k[f(x,y) - f_{PB}(x,y)] + f(x,y) = f(x,y) * h_{UM}$$

donde $h_{UM}(x,y)$ está definido como:

$$h_{UM}(x,y) = \delta(x,y) + k\delta(x,y) - kh_{PB}(x,y)$$

con lo cual tendríamos la forma final para los filtros unsharp masking la siguiente expresión:

$$h_{UM}(x,y) = (1+k)\delta(x,y) - kh_{PB}(x,y)$$

De la ecuación anterior, se puede notar que si modificamos el filtro paso bajas f_{PB} entonces el filtro unsharp masking también cambiara. Esto implica que podemos tener varios filtros unsharp masking. Por ejemplo:

Si

$$h_{PB}(x,y) = \frac{1}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

entonces el filtro unsharp masking queda definido de la siguiente forma:

$$h_{UM}(x,y) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1+k & 0 \\ 0 & 0 & 0 \end{bmatrix} - \frac{k}{9} \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{bmatrix} = \begin{bmatrix} -\frac{k}{9} & -\frac{k}{9} & -\frac{k}{9} \\ -\frac{k}{9} & 1 + \frac{8}{9}k & -\frac{k}{9} \\ -\frac{k}{9} & -\frac{k}{9} & -\frac{k}{9} \end{bmatrix}$$

4.3. Filtrado en frecuencia

El matemático Francés Jean Baptiste Joseph Fourier nació en 1768 en el pueblo de Auxerre, entre París y Dijon. La contribución por la cual él es recordado fue publicada en 1822 en su libro, La Théorie Analytique de la Chaleur (La Teoría Analítica del Calor). La contribución de Fourier en este campo en particular básicamente establece que cualquier función que se repite periódicamente puede ser expresada como una suma de senos y/o cosenos de frecuencias diferentes, cada una multiplicada por un coeficiente diferente.

La correspondencia entre el filtrado espacial y el filtrado en el dominio de la frecuencia está establecida por el **teorema de la convolución**. El proceso en el cual nosotros movemos una máscara de un pixel a otro pixel sobre una imagen y calculamos una cantidad en cada pixel tiene su fundamento en dicho teorema. Formalmente, la convolución discreta de dos funciones $f(x,y)$ y $h(x,y)$ de tamaño $M \times N$ es denotada por $f(x,y) * h(x,y)$ y es definida por la expresión:

$$f(x,y) * h(x,y) = \frac{1}{MN} \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} f(m,n)h(x-m,y-n)$$

Sea $F(u,v)$ y $H(u,v)$ las transformadas de Fourier de $f(x,y)$ y $h(x,y)$ respectivamente. El teorema de la convolución indica que $f(x,y) * h(x,y)$ y $F(u,v)H(u,v)$ están relacionadas de la siguiente manera:

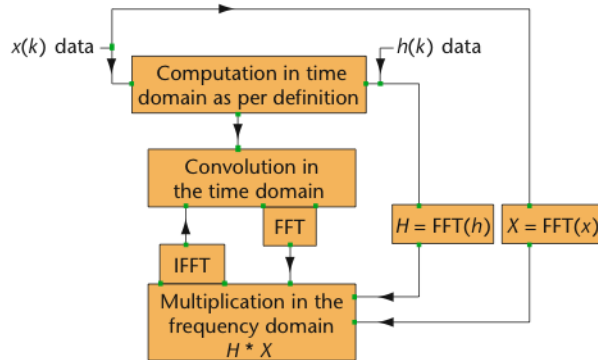
$$f(x,y) * h(x,y) \Leftrightarrow F(u,v)H(u,v)$$

y de manera análoga que:

$$f(x,y)h(x,y) \Leftrightarrow F(u,v) * H(u,v)$$

El proceso de filtrado en el dominio de la frecuencia se muestra en la siguiente figura:

Figura 16: Teorema de la convolución.

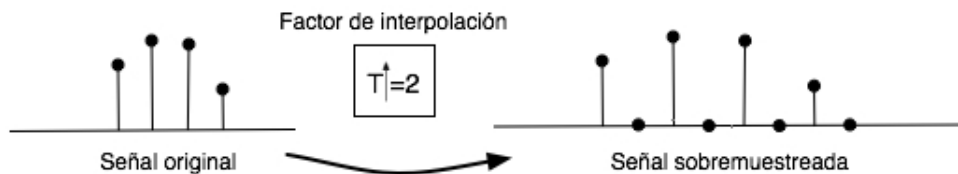


El filtrado en frecuencia nos va a servir principalmente para observar las características de los filtros al obtener su espectro y de esa manera saber si se trata de filtros paso-bajas, paso-altas, etc. Otro factor por el cual es preferible hacer filtrado en frecuencia es la aceleración del algoritmo para calcular la Transformada Discreta de Fourier, usando para esto las implementaciones de la Transformada Rápida de Fourier [2].

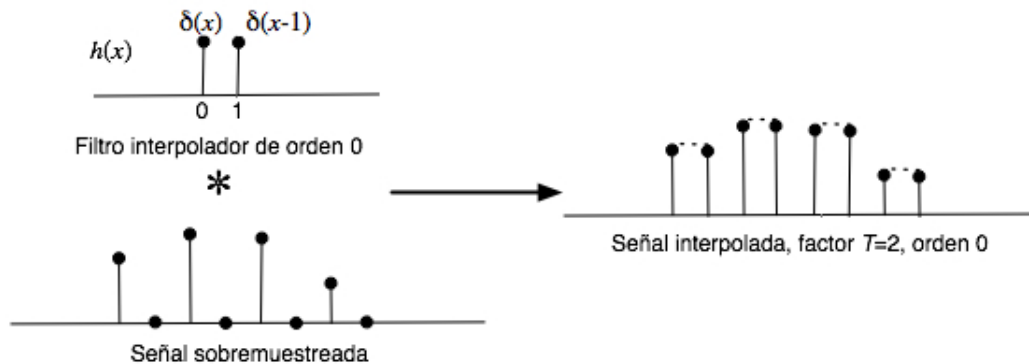
4.4. Interpolación

El proceso de interpolación está relacionado con el hecho de desear cambiar la tasa de muestreo de una señal discreta a una tasa menor o mayor con la que fue muestreada originalmente. El proceso con el cual se disminuye la tasa de muestreo de una señal discreta por un factor D se denomina decimación o submuestreo (*downsampling*) y el proceso con el cual se aumenta la tasa de muestreo de una señal discreta por un factor T se denomina interpolación (*upsampling*).

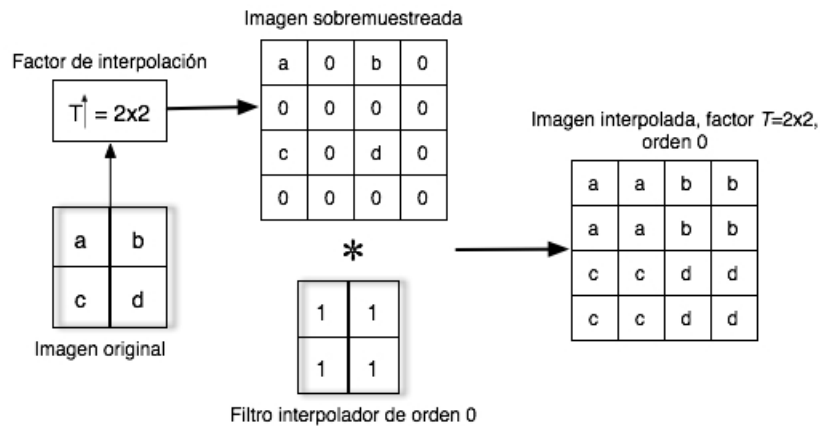
Una de las más simples formas de interpolación es elegir la amplitud de un pixel de salida como la amplitud de su vecino más cercano. Está interpolación es llamada interpolación del vecino más cercano o bien interpolación de orden cero. El factor de interpolación T indicará cuantos pixeles se desean interpolar. Por ejemplo, en una dimensión podríamos ver lo siguiente, se tiene una señal compuesta por cuatro muestras y se desea interpolar con un filtro $h(x)$ de orden cero con un factor de interpolación $T = 2$ como se muestra en la siguiente figura:



La señal interpolada será el resultado de hacer la convolución de la señal original sobremuestreada (agregando ceros) con el filtro interpolador:



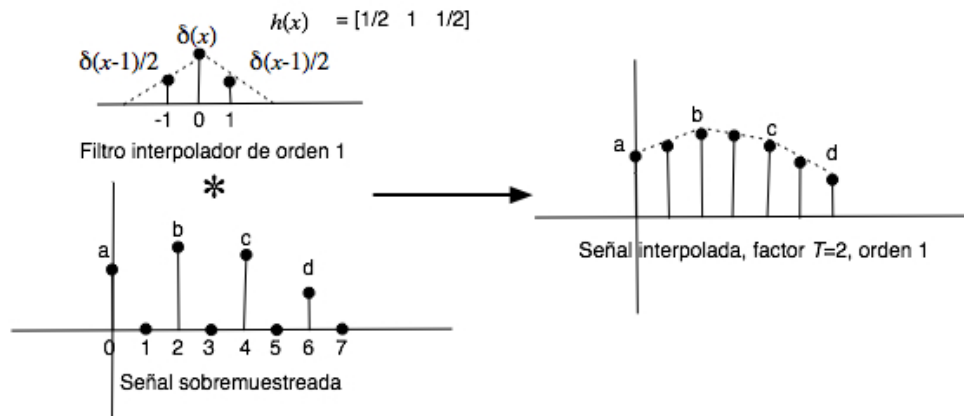
Como se puede ver, el factor de interpolación indicará cuantos elementos se desean interpolar. En nuestro estudio de imágenes, el factor de interpolación indicará cuanto se quiere amplificar una imagen. En el caso de una imagen tendríamos que hacer un proceso muy similar al efectuado en una dimensión:



El orden del filtro indicará como se aproxima a través de un polinomio a la función representada por las muestras. Existen muchos filtros interpoladores y en el caso de imágenes estos se obtienen discretizando, en los valores deseados, los polinomios interpoladores. Los polinomios interpoladores utilizados frecuentemente son: líneas, cúbicos o bicúbicos. Un interpolador de orden cero básicamente efectúa un zoom duplicado. Un interpolador de orden lineal efectúa un zoom asignando el promedio de dos muestras a la muestra interpolada.

Ejemplo: interpolar una imagen por un factor de $T = 2 \times 2$ usando un filtro de orden lineal:

Como primer paso tenemos que encontrar al filtro interpolador, podemos encontrarlo en 1 dimensión y posteriormente extenderlo a 2 dimensiones. Luego se sobremuestra la señal por el factor T deseado, en este caso $T = 2 \times 2$. En 1 dimensión y tomando como señal original la presentada anteriormente se tendría lo siguiente:



Como se puede ver, el comportamiento de la señal interpolada con un filtro de orden lineal es más suave que la señal interpolada de orden cero. Este comportamiento se puede mejorar si incrementamos el orden del polinomio.

Existe también otro método de interpolación y es que se realiza en el dominio de la frecuencia. Uno de los principales fundamentos de las señales discretas es que el “zero padding” en un dominio resulta en un incremento de la tasa de muestreo en el otro dominio.

Referencias

- [1] Gonzalez, R. C. , and Woods, P., *Digital Image Processing*, Addison Wesley, 2002.
- [2] FFTW, www.fftw.org