

Capítulo 1

Imágenes Digitales

1.1 Conceptos Básicos

El Procesamiento de Imágenes es un área del conocimiento que se ocupa de los algoritmos que permiten realizar transformaciones sobre una imagen digital. La visión Computarizada, también llamada Visión Artificial o Visión Técnica, va mas allá, haciendo uso de estos algoritmos como base, integrándolos con sistemas apropiados de hardware, tales como sensores de distintas configuraciones (cámaras), objetivos y filtros ópticos de todo tipo, procesadores de datos, dispositivos de iluminación, y algoritmos de reconocimiento (lo que se discuten en la sección sobre reconocimiento de patrones o redes neurales).

Como corolario a esta integración de tecnologías, el área de Visión Computarizada introduce sus propios algoritmos. Estos últimos constituyen una frontera de la ciencia y podríamos intentar clasificarlos en dos tipos diferentes: Algoritmos de Visión orientados a aplicaciones prácticas y Algoritmos de Visión cuyas perspectivas de aplicación a corto plazo son menores y que están especialmente sujetos a actividades de investigación que buscan alcanzar capacidades de visión avanzadas ¹.

Esta sección se orienta a explicar los fundamentos de los algoritmos que se aplican en estas áreas. Es importante notar que los campos de Procesamiento de Imágenes y Visión, nos proveen de elementos sobre los cuales es posible experimentar los procedimientos de una buena parte de la IA.

Para introducirnos en nuestro tema, debemos tener presente que nuestros ojos ven los objetos en imágenes analógicas. La intensidad de una imagen analógica puede ser considerada una función continua $f(x,y)$ en un espacio continuo de dos dimensiones. Para que esta imagen pueda ser procesada por un computador debe ser previamente digitalizada. La digitalización es resuelta mediante dos operaciones principales:

¹Los tipos de procesos a aplicar sobre imágenes suelen clasificarse como procesamiento de puntos, de áreas, de imágenes (frames), de transformaciones geométricas y de extracción de características. Esto último se refiere básicamente a los fundamentos de un sistema de reconocimiento bajo condiciones de entorno no estructuradas. Esto significa que el sistema no posee información a priori acerca de las características de posición espacial, iluminación, movimientos etc. de objetos que el sistema pretende reconocer. Si debe, desde luego, poseer algún tipo de información acerca de ciertas propiedades visuales del objeto.

- Muestreo de intensidades correspondiente a una matriz de $R \times C$ puntos. El módulo del circuito que efectúa esta operación es denominado *Sample and Hold*.
- Cuantización de los niveles de intensidad continuos en niveles de gris por el convertidor A/D. Siendo k el número de bits por muestra de la imagen. Cada punto muestreado se denomina pixel (de las palabras en inglés *Picture Element*).

La imagen digital queda así representada por $R \times C$ pixels, cada uno codificado con k bits. En nuestro estudio nos referiremos en general a imágenes de intensidad -sin considerar información de color-, llamadas Imágenes de Nivel de Gris.

El Formato PGM

Como ejemplo de valores de parámetros de una imagen digital podemos considerar los siguientes:

- Número de renglones y columnas: 640 y 480.
- Niveles de gris: 256 (Donde por ejemplo $g = 0$ = negro y $g = 255$ = blanco)
- Bits por nivel de gris = 8 -256 niveles diferentes- Pueden emplearse valores mayores, o bytes adicionales para almacenar la información de color.

Existe un formato de imagen digital diseñado para ser extremadamente fácil de comprender y manipular por cualquier lenguaje. Se denomina PGM (Portable Gray Map) y se lo emplea usualmente con fines didácticos o experimentales. Una imagen PGM representa una imagen de escala de grises.

A continuación se da un ejemplo del formato PGM a los efectos de facilitar la comprensión de los temas que se ven en el capítulo³.

Cada entero representa un nivel de gris, es decir es proporcional a la intensidad de la imagen en ese punto. Las líneas que comienzan con “#” son ignoradas.

```
P2                # Código del tipo de formato PGM
# feep.pgm        # nombre comentado
24 7              # ancho por alto de la imagen en pixels
15                # Niveles de grises
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 3 3 3 3 0 0 7 7 7 7 0 0 11 11 11 11 0 0 15 15 15 0
0 3 0 0 0 0 0 7 0 0 0 0 0 11 0 0 0 0 0 15 0 0 15 0
0 3 3 3 0 0 0 7 7 7 0 0 0 11 11 11 0 0 0 15 15 15 0
...
```

³ Suele tenerse por mas sencillo comprender este formato mediante un ejemplo que por sus especificaciones.

1.1.1. Vecindad de un pixel

Para poder comprender los procedimientos propios de los sistemas de visión artificial, se requiere poder transformar las imágenes mediante una gran cantidad de métodos actualmente existentes para este fin. Estos métodos efectúan diversas transformaciones a partir de el calculo de relaciones matemáticas entre un pixel y los que lo rodean. Biológicamente el análisis o procesamiento de un subconjunto de una imagen por medio de neuronas suele asociarse al término *campo receptivo*, el cual estaría formado por un conjunto de neuronas que procesan esa porción de información. Esto nos permite establecer algún paralelismo básico entre los mecanismos biológicos y los computacionales.

En el caso de los procedimientos computacionales, es necesario definir con precisión cual es el entorno o vecindad de un pixel. Una vez definido este grupo de pixels adyacentes podremos discutir como se aplican los procesos de transformación. Se define como la *vecindad de un pixel* o simplemente *vecindad* al conjunto de pixels adyacentes a un pixel dado. Existen distintas convenciones para definir esta vecindad. En aplicaciones prácticas la convención a adoptar depende generalmente del problema a resolver. Las vecindades mas usuales (Fig. 1.1) son:

- Vecindad 4: Los 4 pixels adyacentes tienen un borde común con el pixel central. La vecindad tiene forma de cruz.
- Vecindad 8: Los 8 pixels tienen un borde común con el pixel central. Se incluyen los pixels situados en las diagonales. Los códigos que identifican a los pixels coinciden con el sistema usual de medición de ángulos y son de importancia práctica a la hora de seguir la codificación de un programa o la descripción de un algoritmo.
- Vecindad 6: A la vecindad 4 se agregan dos pixels opuestos sobre una de las diagonales. Puede definirse con diferentes orientaciones.

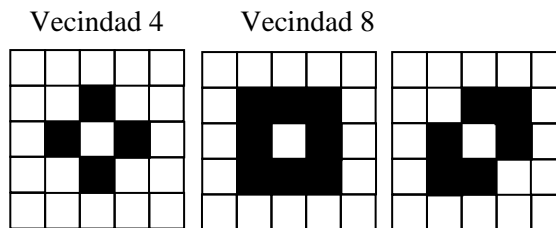


Figura 1.1: Vecindades 4, 6 y 8.

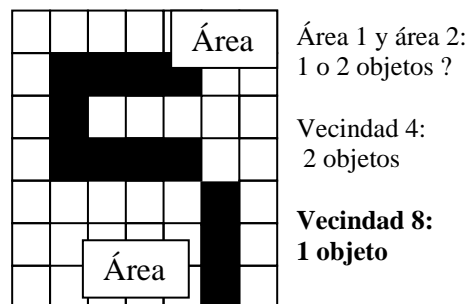


Figura 1.2: La cantidad de objetos queda determinada en ciertos casos por la vecindad elegida

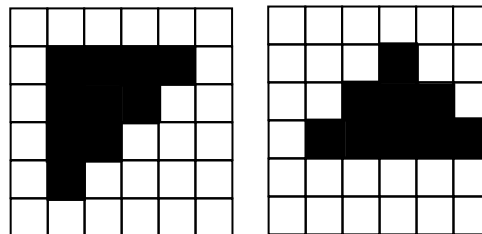


Figura 1.3: El calculo de parámetros puede depender por ejemplo de la rotación de la figura .En este caso el perímetro y el área.

La vecindad adoptada en un caso particular es determinante para considerar válido el resultado que se espera obtener, por ejemplo la vecindad adoptada puede significar que dos áreas estén separadas o conectadas (Fig. 1.2). Es común que, en aplicaciones de

Procesamiento de Imágenes y Visión, haya que fijar otras pautas de antemano. Por ejemplo, dada una imagen binaria –de dos tonos, usualmente blanco y negro- podemos necesitar definir si la línea de borde de una región en una imagen esta construida con pixels 0 o 1. También la forma, el borde y el perímetro de un objeto pueden variar al rotarlo (Fig 1.3). Adoptar una vecindad es frecuentemente un criterio inicial de diseño de un sistema.

1.1.1. Histograma

Algunos métodos de Procesamiento de Imágenes y Visión dependen de ciertas propiedades estadísticas de la imagen. Por ejemplo de la frecuencia relativa de niveles de gris. Denominamos a esta distribución de frecuencias de intensidad Histograma de Imagen. La escala de niveles de gris puede ser representada en forma normalizada variando entre 0 y 1, o bien con valores enteros.

El histograma de niveles de gris de una imagen digital corresponde a la distribución de probabilidad de intensidad de la imagen.

La cantidad de información almacenada en el histograma resulta solo una fracción de la imagen original. El histograma es empleado entre otras cosas para cálculo de coeficientes estadísticos tales como el valor medio, desviación estándar y entropía. También es empleado para aplicar un procedimiento de mejoramiento de la imagen denominado ecualización o igualación de Histograma

El valor medio de niveles de gris se expresa

$$m_g = \sum_{g=0}^{255} g * h(g)$$

es decir la suma de cada nivel de gris g por la cantidad de pixels con ese nivel de gris $h(g)$ - distribución de frecuencia normalizada- La desviación estándar:

$$\sigma^2 = \sum_{g=0}^{255} (g - m_g)^2 * h(g)$$

o sea la suma de todas las desviaciones. Para reducir cálculos esta fórmula suma los productos de la desviación de cada nivel de gris g multiplicados por la cantidad de veces que aparecen -cantidad de pixels $h(g)$ -

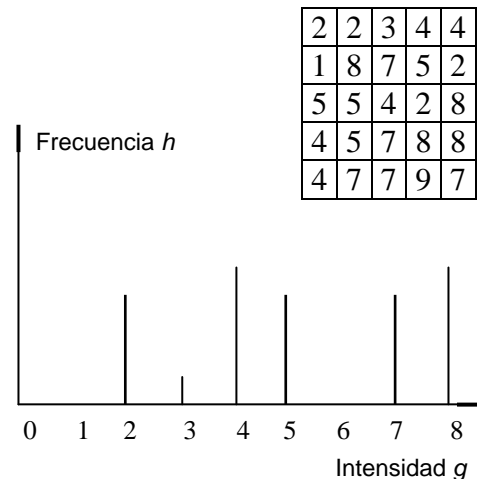


Figura 1.4: Valores de intensidad correspondientes a una parte de una imagen y su histograma correspondiente.

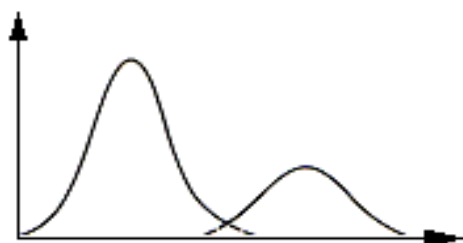
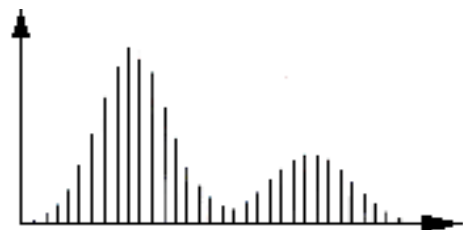


Figura 1.5: Histograma de una imagen de dos tonos predominantes

Ejemplo de comparación de dos imágenes $g_1(x, y)$ y $g_2(x, y)$ mediante el valor medio y la desviación de los niveles de gris

$g_1(x, y) = 127$; nivel de gris uniforme, sin contraste

$g_2(x, y)$ = Patrón correspondiente a un tablero de ajedrez, los niveles de gris se alternan de 0 a 255 de pixel a pixel, alto contraste.

Obviamente los valores medios resultan iguales $m_{g_1} = m_{g_2} = 127$

Desviación estándar: $\sigma_{g_1} = 0$ -Ausencia de contraste- $\sigma_{g_2} = 127^2 = 16129$ -

Alto contraste-

1.1.2. Binarización. Detección de Umbral Óptimo.

Se denomina imagen binaria a una imagen que solo posee dos tonos, 0 y 1, o simplemente blanco y negro. Estas se obtienen en principio fácilmente “gatillando” los valores de intensidad de la imagen con respecto a un valor umbral.

Existe un caso particular de histograma denominado Histograma bimodal. Este tipo de histograma muestra dos máximos locales claramente separados, y corresponde a una imagen con áreas claras y oscuras muy bien definidas, con pocos o ningún nivel de gris intermedio.

De este tipo de imagen y a partir de su histograma puede obtenerse fácilmente la imagen binaria, para esto se emplea como valor de umbral el valor mínimo dado por el histograma entre los picos.

Este sistema permite asimismo efectuar automáticamente el ajuste del umbral para cada imagen (Fig. 1.7).

Otras adaptaciones de esta idea básica permiten procesar las imágenes por vecindades de modo de optimizar localmente el umbral de gatillado para cada zona de la imagen.

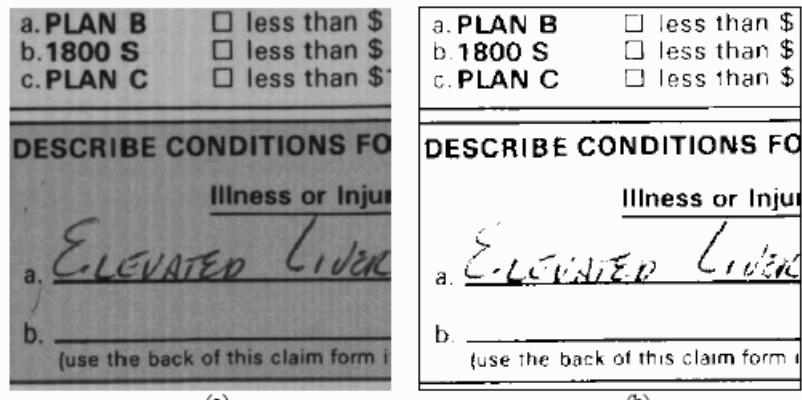


Figura 1.6: Imagen de gris (izquierda) y su correspondiente imagen binaria (derecha)

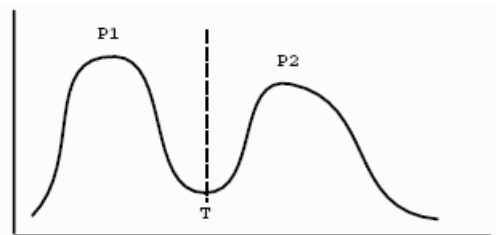


Figura 1.7: Un histograma bimodal permite la detección del umbral óptimo.

1.1.3. Frecuencia espacial

Es importante tener en claro el significado de las palabras Frecuencia Espacial en una imagen, ya que su uso es de uso frecuente en este campo. Básicamente se introduce el concepto haciendo referencia a una imagen en la que el nivel de gris varíe uniformemente en una dirección. Tomemos por simplicidad un imagen donde el nivel de gris varíe en forma abrupta, por ejemplo, una imagen de barras verticales de igual ancho e igualmente espaciadas. La frecuencia determinada al recorrer horizontalmente una secuencia de pixels, es en este caso la frecuencia espacial. Sin embargo, dado que este es un caso muy favorable para nuestra definición, podríamos pensar en una imagen en la que las transiciones de gris no son ya abruptas. En este caso también entendemos de igual forma la frecuencia espacial. Una tercera complicación sería que las barras de transición suave no fueran todas del mismo ancho. Para la transición de una de ellas podríamos mantener nuestro concepto imaginando que las otras fueran iguales. Así tendríamos una definición de frecuencia espacial válida solo para algunos puntos de la imagen. Finalmente en una imagen cualquiera no tenemos por lo general ninguna barra, y entonces la frecuencia espacial es en realidad una referencia al gradiente de intensidad en una dirección en ese punto.

1.1.4. Contraste

Contraste como función de la frecuencia espacial

Se entiende por contraste a la sensibilidad del ojo a los diferentes niveles de intensidad, en áreas de imágenes adyacentes. El contraste se describe frecuentemente de dos maneras:

- * Como función de la frecuencia espacial, considerando la denominada función de sensibilidad al contraste (FSC).

- * Como función de la intensidad, a través de la denominada "Constante de Weber".

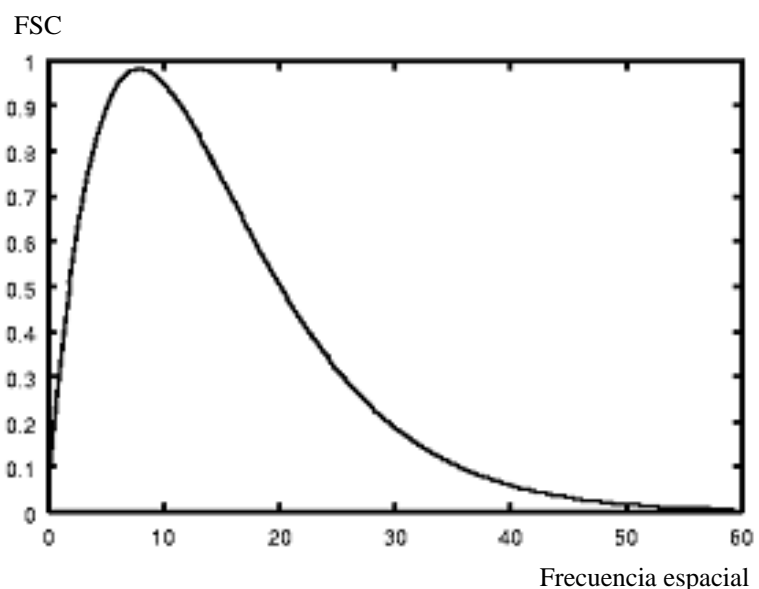


Figura 1.8: Función de Sensibilidad al Contraste

Veamos como se describe el contraste en función de la frecuencia espacial. La función de sensibilidad al contraste nos indica cuan sensitivos somos al estímulo visual de distintas frecuencias. Si la frecuencia del estímulo visual es demasiado grande no es posible reconocer el patrón del estímulo. Imaginemos una imagen consistente en bandas verticales negras y blancas. Si las bandas son muy delgadas (i.e. miles por milímetro) no nos será posible distinguirlas individualmente. Todo lo que vemos es una imagen gris.

Si las bandas se hicieran cada vez mas anchas, habrá un espesor umbral, a partir del cual es posible distinguir las bandas.

Se han propuesto diversas funciones de sensibilidad al contraste. Una de estas se muestra en la figura.

Contraste e Intensidad. Constante de Weber

La respuesta del ojo al contraste no es lineal. La mínima diferencia de intensidad que puede detectar el ojo es pequeña para valores pequeños de f y crece proporcionalmente con f para valores altos de intensidad.

Definimos el contraste en función de la intensidad como la relación entre la intensidad en un punto de la imagen y la variación de intensidad con respecto a un punto adyacente. Para valores intermedios de intensidad esta relación se suele considerar constante y queda definida por un valor denominado constante de Weber.

Tal como esta planteada la función en la fig. 1.9, podemos decir que para observar un objeto sobre el fondo, necesitaremos mayor contraste si las intensidades son muy altas (imágenes y objetos brillantes). En el caso de que las intensidades sean muy bajas (imágenes y objetos oscuros), una variación modesta de intensidad, hará que el cociente $\Delta i/i$ tome un valor importante, lo que nos dice que también a bajas intensidades necesitamos mayor contraste para detectar los objetos.

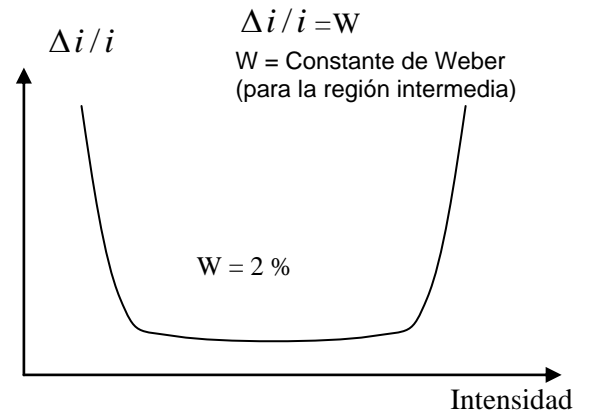


Figura 1.9: Contraste $\Delta i/i$ como función de la intensidad

1.1.5. Cuantización

La intensidad de una imagen analógica se cuantifica en niveles de gris, el valor total de niveles de gris puede ser por ejemplo 256, lo que permite describirlos a todos con un solo byte de información.

La cuantización es usualmente lineal, pero para resaltar el contraste en rangos de la escala de grises donde este es débil puede emplearse una cuantización no lineal.

Para reducir los requerimientos de almacenamiento de la imagen de una manera drástica puede cuantificarse la imagen con menos de 8 bits. Esto puede hacerse perdiendo poca información debido a que:

- La pérdida de información es reconocida a partir de los 5 bits
- Los bits menos significativos 0, 1 y eventualmente 2 generalmente contienen ruido.

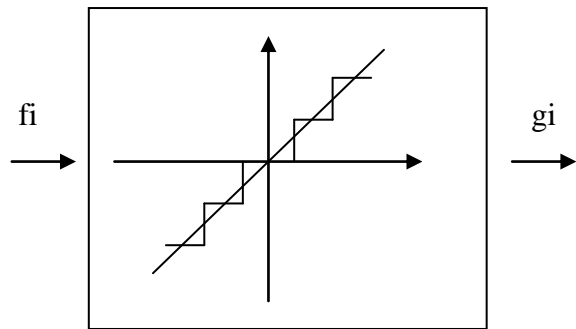


Figura 1.10: Característica de cuantización lineal. Intensidad continua y discreta

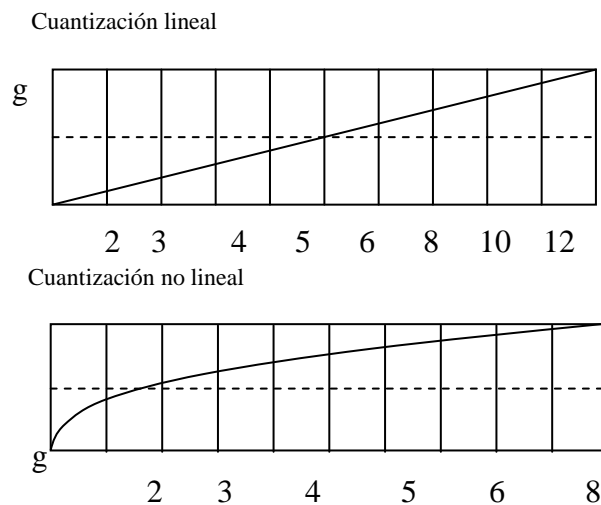


Figura 1.11: Cuantización lineal y no lineal.

1.2. Procesamiento

1.2.1 Ecualización de histogramas en función de la distribución de frecuencias

Este método se explica mediante un ejemplo. Se parte del histograma mostrado en la Fig. 1.12, en el cual, por simplicidad se han supuesto 40 niveles de gris. Se expresan las densidades aplicando un factor constante 4 a los efectos de hacer notorias las diferencias. Puede observarse que sin aplicar este factor (4) la suma de las frecuencias relativas normalizadas dan 1. Es decir sumando todos los valores de la columna g de la tabla resulta 4, por haberse aplicado un factor 4 a las frecuencias normalizadas.

Este método justifica su aplicación en imágenes de bajo contraste con áreas grandes de niveles medios de gris. Esto se cumple para el histograma ejemplo propuesto.

En dicho ejemplo en las partes de bajo y alto nivel de gris del histograma, la frecuencia $h(g)$ es igual a cero. En el área central algunas líneas representativas de $h(g)$ son bastante mayores que otras, lo que equivale a decir que sus intensidades ocurren mas frecuentemente. Se observa además que el intervalo entre líneas $h(g)$ es constante y se le asigna el valor $\Delta g = 1$.

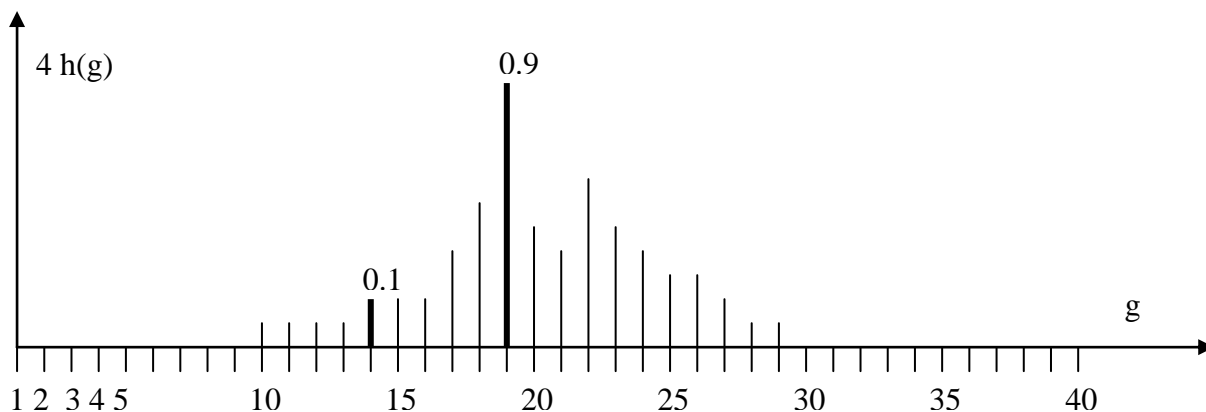


Figura 1.12: Histograma de partida para ejemplo de ecualización

El método de igualación de histogramas que se describe a continuación consiste en una transformación de niveles de gris $g \rightarrow g'$. Se reacomodan las líneas del histograma con nuevos intervalos Δg de manera de obtener niveles de densidades correspondientes entre todos los valores de g y g' .

Se tiene entonces que:

$$h(g') / \Delta g' = \text{constante} \quad \text{y} \quad g' = \text{niveles de gris de la imagen transformada} \quad (1)$$

Para el histograma de 40 líneas esto significa:

- En $h(g)$, las líneas $h(g)=0.9$ y 0.1 tienen el mismo intervalo asociado $\Delta g=1$
- Puede calcularse una tabla para la transformación $g \rightarrow g'$. El nuevo histograma es $h(g')$
- En $h(g')$, la línea $h(g')=0.9$ requiere un intervalo $\Delta g'$ que es 9 veces mayor que en $h(g)$
donde se tenía un intervalo $\Delta g=1$

De la eq. (1) se tiene que:

$\Delta g' = K \times 4 \times h(g')$ donde 4 es un valor de proporcionalidad ya explicado.

Por otra parte:

$\Sigma \Delta g' = Q = 40$ (Suma de todos los intervalos)

Por definición, las frecuencias normalizadas del histograma deben sumar 1, $\Sigma h(g) = 1$

por lo cual; $\Sigma \Delta g' = 40 = \Sigma K \times 4 \times h(g') = K \Rightarrow K = 10$

Ahora pueden calcularse los intervalos según se muestran en la tabla siguiente. El valor g' de una fila es obtenido del $\Delta g'$ de la misma fila sumado al g' de la fila previa. Sea por ejemplo la fila actual la fila $g=18$; se tiene $g' = g'(\text{previo}) + \Delta g'(\text{actual}) = 7 + 4 = 11$.

g	4 h(g)	$\Delta g' = 10 \times 4 \times h(g)$	$g'(g)$	$4 \times h(g')$
<10	0	0	0	
10	0.05	0.5	0	0.05
11	0.05	0.5	1	0.05
12	0.05	0.5	1	0.05
13	0.05	0.5	2	0.05
14	0.1	1	3	0.1
15	0.1	1	4	0.1
16	0.1	1	5	0.1
17	0.2	2	7	0.2
18	0.4	4	11	0.4
19	0.9	9	20	0.9
20	0.3	3	23	0.3
21	0.2	2	25	0.2
22	0.5	5	30	0.5
23	0.3	3	33	0.3
24	0.2	2	35	0.2
25	0.15	1.5	37	0.15
26	0.15	1.5	38	0.15
27	0.1	1	39	0.1
28	0.05	0.5	39	0.05
29	0.05	0.5	40	0.05
>29	0	0	40	

Como puede verse del ejemplo, efectivamente las frecuencias no cambian, sino que se distribuyen los niveles de gris, es decir se modifican los intervalos.

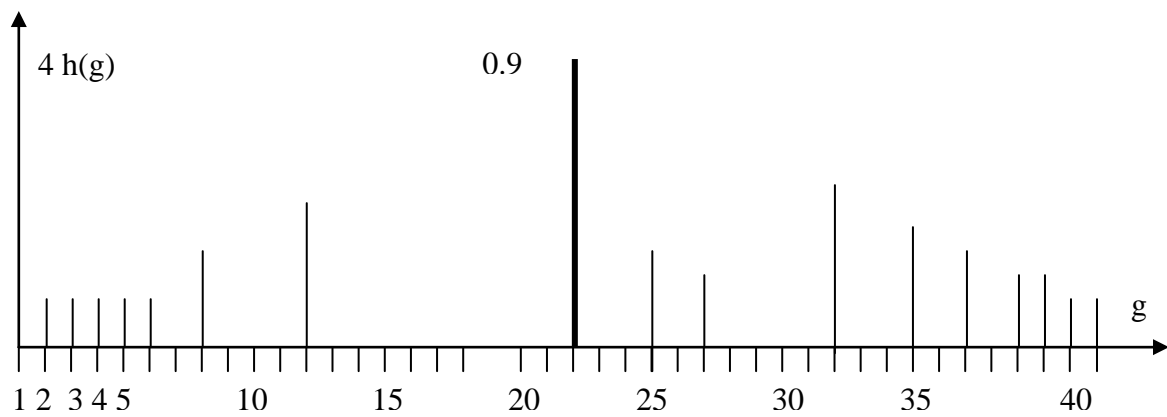


Figura 1.13: Histograma resultante

A continuación se expone otro ejemplo muy simplificado del procedimiento de ecualización. Supongamos tener 10 niveles de gris, y 10 pixels distribuidos en los niveles 2, 3 y 4 en cantidades 3, 5 y 2 respectivamente. Normalizando con respecto al total de muestras obtenemos el siguiente histograma de imagen:

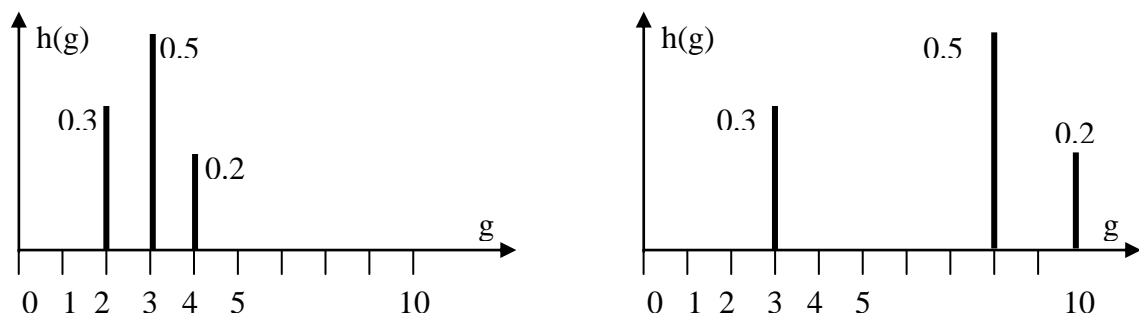


Figura 1.14: Histograma previo a la ecualización (izquierda) e histograma resultante (derecha)

La suma de la distribución de $h(g)$ normalizada resulta: $\sum h(g) = 0.3 + 0.5 + 0.2 = 1$

La constante $K = \text{niveles de gris} / \sum h(g) = 10$

Considerando que $g_{\text{nuevo}} = g_{\text{previo}} + k \cdot h(g)$

se obtiene:

g_{previo}	$h(g)$	$k \cdot h(g)$	g_{nuevo}
1	0	0	0
2	0.3	3	3
3	0.5	5	8
4	0.2	2	10
5	0	0	10

1.2.2. Invariantes geométricos

Área: Es el invariante quizás mas sencillo. Sumamos los valores de todos los pixels de una región $A = \sum p_{ij}$. Como ejemplo, en la Fig. 1.15 se muestra un fragmento de una imagen binaria, en la que esta claramente definida una región marcada con valores uno (1). El área será simplemente la suma de todos los unos, es decir 5.

```

0 0 0 0 0
0 1 0 0 0
0 1 1 1 0
0 1 0 0 0
0 0 0 0 0

```

Figura 1.15: Fragmento de imagen binarizada conteniendo una región

Baricentro (centro de Gravedad): Para una superficie definida por valores discretos calculamos sus dos coordenadas como un promedio de la suma de las coordenadas de cada píxel que pertenece a la región. Para el fragmento de imagen anterior se tiene:

$$i_0 = (1/A) \sum \sum i p_{ij} \quad j_0 = (1/A) \sum \sum j p_{ij}$$

Por ser una imagen binaria $p_{ij} = 1$. En la figura el área es 5, y la coordenada i

```

0 0 0 0 0
0 1 0 0 0
0 1 1 1 0
0 1 0 0 0
0 0 0 0 0

```

Figura 1.16: Fragmento de imagen binarizada y baricentro

$i_0 = (1/5) \times (2 \times 1 + 3 \times 3 + 4 \times 1)$ -suma de productos de cada fila por cantidad de pixels en esa fila-

$$i_0 = (1/5) \times 15 = 3 \quad y \quad j_0 = (1/5) \times (2 \times 3 + 3 \times 1 + 4 \times 1) = 13/5 = 2.6$$

tomamos $j_0 = 3$ como valor discreto mas próximo. El píxel correspondiente al baricentro calculado se muestra en la Fig. 1.16.

Perímetro: Una forma de calcularlo es encontrando los pixel de contorno y sumándolos. Un método sencillo para detectar los pixel de contorno se explica mas adelante. Se debe definir primero si se trabaja con los pixeles de borde de la región (unos en el ejemplo) o con los externos a ella (ceros en el ejemplo).

```

0 0 0 0 0
0 1 1 1 0
0 1 0 1 0
0 1 1 1 0
0 0 0 0 0

```

Figura 1.17: Región hueca a la que corresponde numero de Euler cero

Número de Euler: Dados los patrones $\begin{matrix} 01 & 00 \\ 11 & 01 \end{matrix}$ y

El número de Euler es el número obtenido al calcular las ocurrencias del primer patrón menos el segundo. Indica cuantos “espacios internos” tiene una región. Por ejemplo, una región con forma de seis tiene un número de Euler igual a cero, con forma de ocho, igual a uno, etc. Por ejemplo en la Fig. 1.17 vemos una región con un “agujero”. La diferencia de las ocurrencias entre ambos patrones es cero, y si el cero central fuera un uno, es decir si no hubiera “agujeros”, el valor obtenido para el número de Euler sería (1) o (-1). El signo depende de adoptar la convención respecto al orden en que se restan las ocurrencias de los patrones de referencia, es decir: Patrones con un cero en la esquina inferior derecha menos Patrones con un uno en la esquina inferior derecha o viceversa

Ejes principales de inercia:

Los ejes principales de inercia de un objeto físico son propiedades que dependen de la forma del objeto. Si esto se piensa al revés, conociendo los ejes principales de inercia podemos intentar sacar algunas conclusiones sobre el objeto en cuestión.

Este concepto físico se emplea para caracterizar formas (regiones) en imágenes, extendiendo el concepto al campo del procesamiento de imágenes. La aplicación de conceptos físicos a problemas de IA es muy frecuente.

En el caso de la Visión Computarizada estos ejes son importantes por varias razones, por ejemplo permiten conocer la orientación de una región en el plano lo que puede representar directamente la orientación de un objeto. De este modo un manipulador puede tomar un objeto por el c. g. Orientando su efector (garra) en la dirección apropiada (como lo hace inconscientemente una persona, que toma un objeto longitudinal aproximadamente pro su baricentro y orienta la mano para tomarlo de la mejor manera, por ejemplo una escoba).

El cálculo de los mismos requiere de manipulaciones matemáticas que aquí trataremos de abreviar.

El eje de inercia esta definido por el momento de inercia que las partes de un objeto ejercen con respecto a ese eje. El momento de inercia de una parte (diferencial de volumen) del objeto esta dada por un producto que es la masa de ese volumen por el cuadrado de la distancia al eje. Tratemos de llevar esta idea al campo de las imágenes discretas y observemos la figura. Un punto del objeto contribuye a la inercia del objeto con respecto al eje AB, con su masa multiplicada por la distancia al eje r . En una imagen discreta, la primera simplificación que encontramos es que estamos en el plano y no en el espacio como ocurre con un objeto físico real que tiene un volumen. La segunda observación es que la contribución de cada píxel no es una “Masa” sino el nivel de gris de cada píxel. En una imagen binaria el nivel de gris será solo cero o uno, por lo que consideramos solo los píxel con el valor digamos uno, como los que aportan a la “inercia” de la región. Así, todos los píxel de la región multiplicados por la distancia al eje, sumados estos valores nos darán la inercia de la región respecto de ese eje.

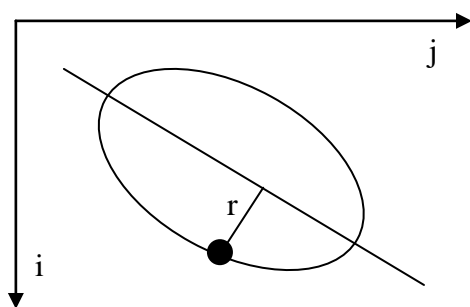
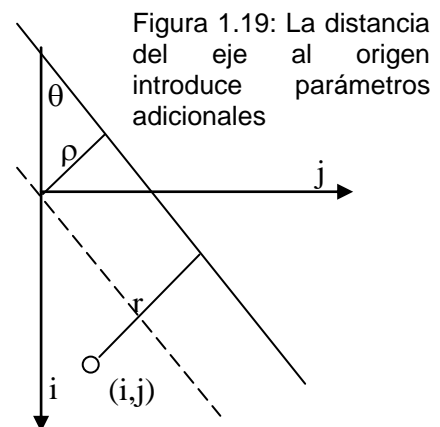


Figura 1.18: Cada punto contribuye según su distancia al eje de referencia



Para el cálculo que nos interesa, se hace la primer consideración de que en primer lugar, un eje de inercia de cualquier objeto (en nuestro caso región) pasa por el baricentro, por lo que sus expresiones están dadas en forma relativa a las coordenadas del c. g.

$$j' = j - j_0 \quad y \quad i' = i - i_0$$

Necesitamos entonces poder calcular la inercia de un pixel respecto al eje, para lo cual debemos poder calcular el cuadrado de la distancia del pixel al eje.

De la figura, la distancia se puede calcular como:

$$i' \sin \theta - j' \cos \theta + \rho$$

siendo el ángulo la inclinación respecto al eje vertical y ρ la distancia perpendicular al eje desde el origen del sistema de coordenadas.

Bajo estas condiciones la inercia de la región respecto de la línea resulta:

$$I = \sum \sum p_{ij} (i' \sin \theta - j' \cos \theta)^2 = a \sin^2 \theta - 2b \sin \theta \cos \theta + c \cos^2 \theta$$

Donde:

$$\begin{aligned} a &= \sum \sum (i')^2 p_{ij} \\ c &= \sum \sum (j')^2 p_{ij} \\ b &= \sum \sum (i' j') p_{ij} \end{aligned}$$

De donde planteamos un sistema para calcular los ejes de mayor y menor inercia (principales):

$$\begin{aligned} I_{\max} + I_{\min} &= a + c \\ I_{\max} - I_{\min} &= \sqrt{b^2 - (a-c)^2} \end{aligned}$$

Orientación:

Es uno de los métodos mas robustos para determinar ángulos. Su cálculo se hace a partir del momento de inercia mencionado

$$\tan 2\theta_0 = f(b, a, c) = 2b/(a-c)$$

Spread:

Es el cociente entre la suma de los momentos max y min y el cuadrado del área. Nos dice cuan irregular es el contorno de una región. Es decir si el contorno es “macizo” o “disperso”

$$(I_{\max} + I_{\min}) / A^2 = (a + c) / A^2$$

1.2.3. Filtro de la Mediana y del Valor Medio

Estos operadores calculan un pixel para la imagen transformada, en función de la vecindad local en la imagen original. Debemos tener presente que los filtros pasa bajo lineales reducen el ruido, pero al mismo tiempo reducen la resolución de los detalles pequeños.

El Filtro de la Mediana se explica mediante un sencillo ejemplo. Se considera una ventana 1*5 (Fila)

$$g(y) = [80 \ 90 \ 200 \ 110 \ 120]$$

El valor central “200” también llamado el *mediano* es evidentemente un “*extraño*” (*outlier*) o un pixel de ruido en la secuencia ascendente de valores de gris. En la secuencia ordenada de estos cinco números

80 90 110 120 200

El valor mediano es 110, es decir que mediante el ordenamiento el valor 200 ha sido desplazado (shifted) hacia la derecha.

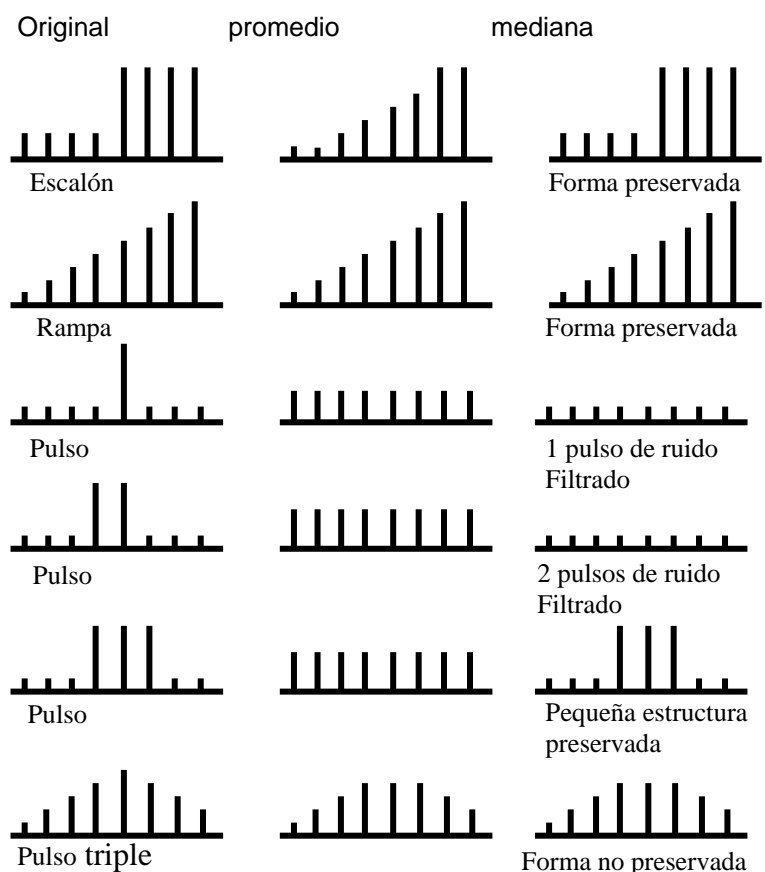
Para obtener el valor transformado libre de distorsión el valor original es reemplazado por el mediano en la secuencia original quedando:

$$g(y) = [80 \ 90 \ 110 \ 110 \ 120]$$

En la siguiente figura se muestra el resultado de aplicar un filtro de la mediana a algunos casos típicos. Su efecto es comparado con un filtro del valor medio. En todos los casos se emplea una ventana 1*5.

Cuando este filtro se aplica a la imagen digital, se consideran todos los pixels de la vecindad, se ordenan, y se selecciona el del medio cuyo valor reemplaza al original en la imagen transformada

Figura 1.20: Comparación de filtros (mediana y promedio)



El filtro del valor medio es similar pero se reemplaza el valor del medio por el promedio del conjunto.

1.2.4. Detección de contorno en imágenes binarias

Se suponen objetos binarios blancos ($g = 1$) en fondo negro ($g = 0$). A partir de un pixel inicial del objeto, el borde del objeto es buscado siguiendo a sus vecinos. En cada pixel se hace un giro de 90 grados del siguiente modo:

Giro a izquierda (+90) si $g(x,y)=0$,

Giro a derecha (-90) si $g(x,y)=1$,

Se elige para el resultado el borde contenido en el objeto blanco o el borde contenido en el fondo negro. Dependiendo de la orientación los pixel de contorno se adecúan mejor a la vecindad-8 -bordes de orientación diagonal- o a la vecindad-4 -bordes de orientación horizontal-. Para obtener una línea de contorno que correspondiese a vecindad uniforme se omiten o agregan pixels en los lugares apropiados.

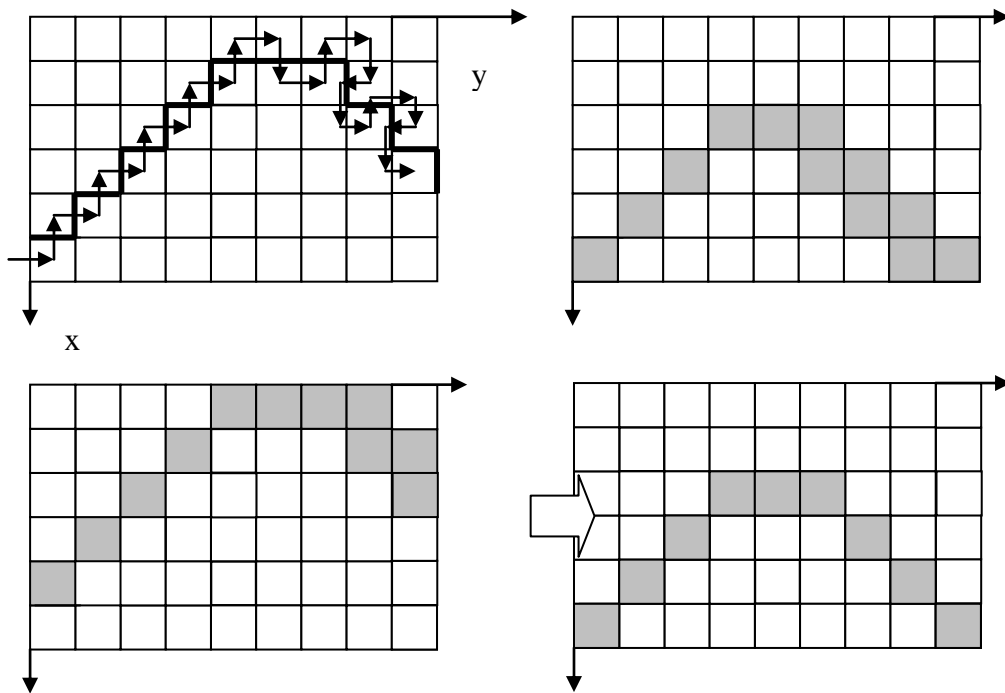



Figura 1.21: Representación del algoritmo de detección de contorno

1.2.5. Convolución en Imágenes Digitales

Este método es quizás el de aplicación mas general para procesamiento de imágenes, en el sentido de que permite una importante cantidad de transformaciones de una imagen a partir de un mismo procedimiento que es llamado precisamente Convolución.

La convolución de funciones es un concepto matemático que se introduce en su forma general para funciones matemáticas. Se puede considerar su aplicación en forma discreta o continua. Dado el carácter discreto de las imágenes digitales se aplica en ellas la convolución discreta. Esta aplicación se restringe a la vecindad de un pixel. El método consiste en definir una máscara, cuya dimensión sea igual a la dimensión de la vecindad elegida para procesar la imagen, por ejemplo, supongamos una máscara de 3x3 elementos:

Considerando que los vecinos 3x3 de un pixel sean los mostrados como $f(x)$ en la figura, se calculan los nuevos valores de cada pixel en función de los vecinos y de la máscara. El valor obtenido reemplaza al pixel de referencia (P) en la imagen transformada. Generalmente la imagen es recorrida por filas para aplicar la convolución a cada pixel. Es bastante evidente también, que los pixels de borde quedan fuera de este proceso (en este caso de vecindad 8 quedarían sin procesar los pixels de la columna 0, la fila 0, la columna final y la fila final).

$$g(x, y) = \begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix} \quad f(x, y) = \begin{bmatrix} d & c & b \\ e & P & a \\ f & g & h \end{bmatrix}$$


$$f(x) = \begin{bmatrix} d & c & b \\ e & P' & a \\ f & g & h \end{bmatrix}$$

$$P' = \sum \sum g(x, y) f(x, y)$$

Figura 1.22: El valor nuevo P' se computa en función de la convolución discreta entre $g(x)$ y $f(x)$

La convolución discreta consiste en nuestro caso simplemente en realizar la suma de los productos entre elementos de la misma posición en la imagen, y reemplazar el pixel de referencia de la vecindad (central) por el nuevo valor. (o en algunos casos por algún valor que es función de este).

Para el ejemplo de la figura deberíamos calcular para cada pixel

$$P' = 1 \times d + 0 \times c + (-1) \times b + 1 \times e + 0 \times P + (-1) \times a + 1 \times f + 0 \times g + (-1) \times h$$

Y reemplazar P por P' en la imagen transformada.

En otras mascaras es necesario dividir P por la sumatoria de las componentes de la máscara, a los efectos de que P tenga un valor en el rango de niveles de gris disponibles.

1.2.6. Detección de bordes

Una importante característica actualmente aceptada acerca del proceso de visión humana, es que los seres humanos son particularmente receptivos a los saltos abruptos de intensidad. Estos a menudo son llamados como bordes (edges) y frecuentemente corresponden a partes del contorno de los objetos. Se considera que juegan un rol primordial en las etapas tempranas de visión de los sistemas biológicos.

Afortunadamente la detección artificial de los mismos resulta sencilla, dado que estos corresponden a zonas de alta frecuencia espacial o alto gradiente de niveles de gris. La forma en que se obtienen es por convolución. Partiendo de la expresión matemática del gradiente (y otras funciones según el tipo de detector de bordes que se desea) se deducen las máscaras que deben aplicarse por convolución para obtener los bordes.

Entre los mas conocidos tenemos el de Prewitt, Sobel, Gradiente, etc. De estos a su vez se han derivado algunos mas complejos a medida que la potencia de procesamiento de los computadores los ha vuelto funcionales. Algunos son multidireccionales, es decir pueden detectar bordes en cualquier dirección (direcciones cuantificadas) en tanto que los mas simples solo pueden hacerlo para bordes horizontales o verticales.

El operador de Prewitt vertical es el visto en el apartado anterior. Los operadores de borde suelen emplear máscaras de 3x3, 5x5, 7x7 o 11x11.

$$g(x, y) = \begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix}$$

$$g(x, y) = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix}$$

Figura 1.23: Detector de Prewitt y de Sobel

Detector de bordes de Sobel

Este operador realiza una medición del gradiente en una imagen y de este modo permite realzar zonas de alta frecuencia espacial es decir detectar bordes. En su modelo mas elemental, consiste en un par de máscaras de convolución 3x3 (Una de ellas se muestra en la figura 1.23). Una rotada con respecto a la otra 90°.

Como se intuye, estas mascaras están diseñadas para detectar bordes verticales y horizontales. Las mascaras pueden aplicarse por separado para obtener los bordes en ambas direcciones independientemente (sean G_x y G_y).

Pueden combinarse para obtener el valor del módulo:

$$|G| = \sqrt{G_x^2 + G_y^2}$$

Lo que se suele aproximar (mas rápido de procesar)

$$\text{como: } |G| = |G_x| + |G_y|$$

La orientación del borde esta dada por:

$$\theta = \arctan(G_y / G_x)$$



Figura 1.24: Resultado de aplicar los operadores de Sobel

Posprocesamiento de bordes

Como ya se explicó, a partir de la experiencia humana se acepta como particularmente importante para el reconocimiento de objetos en imágenes el reconocimiento de bordes, y por lo tanto que las intensidades resultan ser de menor importancia. Los bordes pueden extraerse por los métodos ya vistos.

En ciertos casos las líneas de borde obtenidas deben ser posprocesadas. Un caso que ocurre con frecuencia es que se desean detectar con cierta certeza, los bordes que corresponden a un objeto dado.

Esto no resulta sencillo pues los bordes reales de los objetos, al ser detectados con un detector de bordes resultan en un conjunto de bordes aislados y con ramificaciones. Por este motivo las ramificaciones y las interrupciones deben ser eliminadas. Además, según el algoritmo de detección de bordes elegido y el tipo de imágenes a procesar puede requerirse un proceso de adelgazamiento posterior para que todos los bordes tengan un pixel de ancho.

En el caso particular de una imagen de valores de grises, ocurre que generalmente los bordes de niveles de gris:

- Tienen valores máximos fluctuantes y
- Poseen interrupciones

Lo que da origen al problema antes mencionado

Búsqueda en arco del pixel sucesor

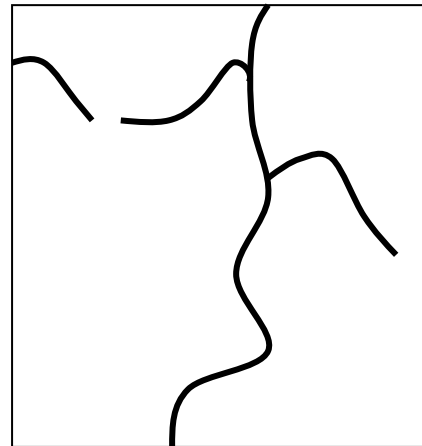


Figura 1.25: Ramificaciones e interrupciones en líneas de borde.

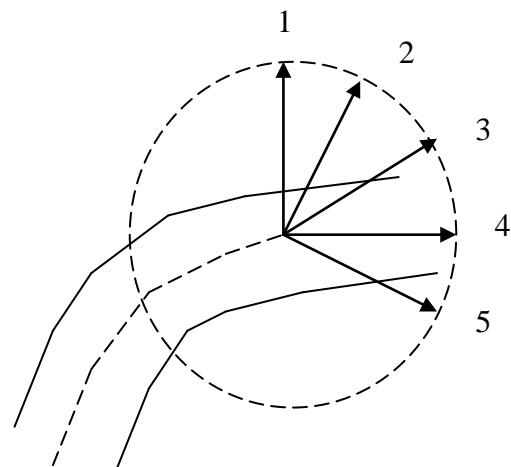


Figura 1.26: Pixels de contorno y direcciones de búsqueda

Una regla simple para eliminar interrupciones entre los bordes, obtenidos a partir de imágenes de tonos de gris es elegir como pixel sucesor al vecino con mayor nivel de gris.

El problema en este caso es que debido al ruido la línea puede oscilar dentro de un amplio margen de valores de gris. La solución puede ser la denominada búsqueda en arco que consiste en efectuar una búsqueda en una vecindad ampliada (mayor que la vecindad ocho).

Se define un espacio de búsqueda como un sector limitado a ambos lados de la dirección principal del borde en su extremo.

El pixel sucesor queda determinado por el rayo de búsqueda con el mayor valor medio. Como resultado de esto último las fluctuaciones de niveles de gris en el borde son relativamente neutralizadas.

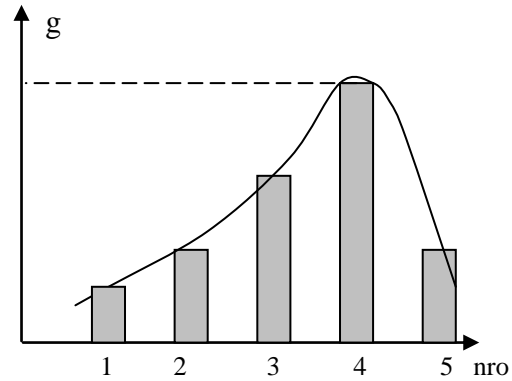


Figura 1.27: Valores de gris de los rayos de búsqueda

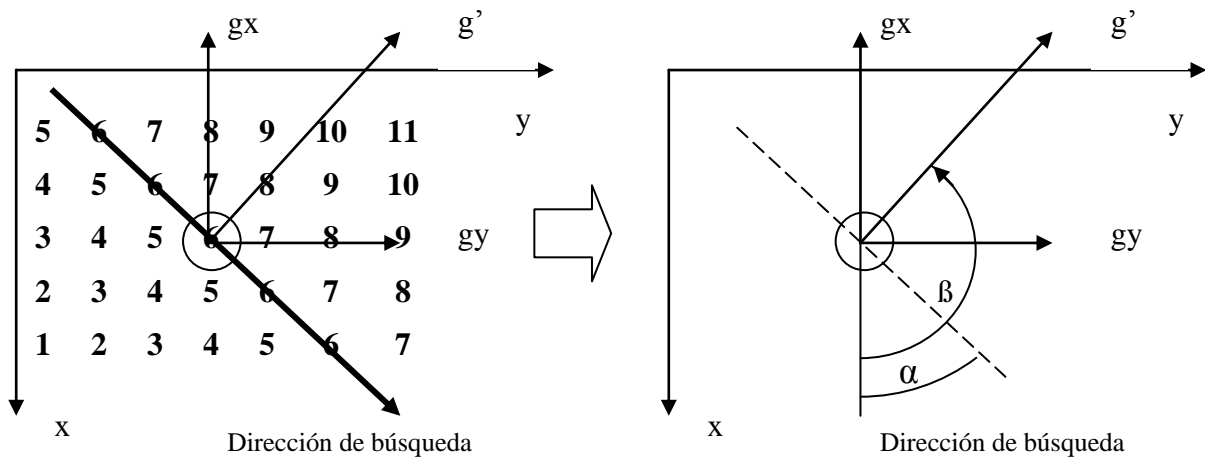


Figura 1.28: Dirección de búsqueda para los operadores de gradiente ortogonales

La línea de contorno una vez hallada puede ser almacenada como una lista de pixels o como una lista de vectores de contorno (una cadena de vectores consecutivos).

Búsqueda con dirección controlada por gradiente

El pixel inicial debe de pixel de borde.

La dirección principal de búsqueda puede calcularse de los gradientes de un detector de bordes local en el pixel investigado. Esta dirección puede describirse por el ángulo relativo al eje x. El pixel sucesor queda determinado por este vector de búsqueda. Este proceso es rápido debido al reducido espacio de búsqueda. El primer pixel, debe cumplir la condición de estar sobre el borde y puede obviamente encontrarse también mediante un detector de bordes.

1.2.7. Segmentación

En procesamiento de imágenes se denomina segmentación al proceso de extraer (es decir identificar y aislar) regiones de interés en una imagen. Se obtiene de este modo una lista de regiones compuestas por los pixels correspondientes a las mismas. Si la imagen no es binaria, obtendremos también la intensidad de los pixels que componen la región. Esto implica entre otras cosas la posibilidad de reducir los requerimientos de memoria de la imagen. Pero más importante que la posibilidad de comprimir la imagen, nos resulta aquí la posibilidad de aislar regiones en una imagen las que a su vez pueden corresponder, a imágenes de objetos de interés que aparecen en la imagen. Así por ejemplo podremos diferenciar objetos en una imagen, o bien decidir una acción a ejecutar en función de estos objetos y sus características.

Estas regiones u objetos de interés pueden ser descritos principalmente mediante sus características, que pueden obtenerse por ejemplo calculando los invariantes ya vistos para cada región y conformando así un vector de características.

Dos tipos clásicos de segmentación corresponden a:

- Determinación de áreas que corresponden a objetos de interés
- Determinación de perímetros de objetos de interés

Desde este punto de vista los procesos de segmentación se pueden clasificar en dos categorías principales: (Existen otros métodos empleados con menor frecuencia tales como la segmentación basada en texturas)

- Orientados a regiones (áreas)
- Orientados a perímetros (líneas)

Métodos orientados a regiones: Los pixels adyacentes a una región son asignados a esa región en tanto sean homogéneos con respecto a los pixels de la región, es decir tengan similares valores de intensidad.

Métodos orientados a bordes: En este caso no se obtiene generalmente un borde cerrado que define a una región sino que debe efectuarse un postproceso para obtener dicho contorno, este postproceso comprende operaciones tales como:

- Seguimiento de contorno
- Conexión de bordes abiertos
- Eliminación de bordes sobrantes

Métodos de segmentación orientados a regiones

Segmentación binaria: La imagen de grises es binarizada (explicado anteriormente) y luego dividida en subregiones a través de un algoritmo de segmentación binaria. Una etiqueta particular (1,2,3,...,n) es asignada a cada pixel de cada nueva región. Esto puede ser visto como un operador de pixels. Un algoritmo simple de Segmentación binaria se explica a continuación.

1.2.8. Región Aggregation

Se trata de elegir un conjunto de “semillas”, es decir pixels espaciados en la imagen, que son primeramente evaluados para determinar si pertenecen o no a una región (por ejemplo en una imagen binaria si su valor es cero o uno).

Estrategia de agregación: Analizada una semilla, si esta no pertenece a una región, se descarta, en caso contrario se etiqueta con un número de región y posteriormente se inspeccionan los ocho vecinos de este pixel, por ejemplo en sentido antihorario. A su vez, cada uno de estos 8

vecinos, si pertenece a la misma región, se etiqueta con el mismo valor de etiqueta que la semilla y se emplea para explorar sus propios ocho vecinos y así sucesivamente.

Si existieran semillas distintas pertenecientes a la misma región se resuelve en el curso del proceso.

Finalmente se obtiene adicionalmente a la imagen de grises una "Imagen etiquetada" la cual puede ser visualizada haciendo corresponder distintos tonos de grises o pseudo colores a las regiones.

La lista de los pixels a examinar de cada etiqueta pueden mantenerse actualizados en una estructura de pila, sin embargo debe tenerse presente que en muchos casos la pila es simulada dentro de un vector inicializado antes del proceso a los efectos de optimizar la velocidad.

Un sencillo algoritmo basado en una estructura de tipo pila se describe en la Fig. 1.30. En la Fig. 1.32. se muestra el avance del proceso en un conjunto reducido de pixels.

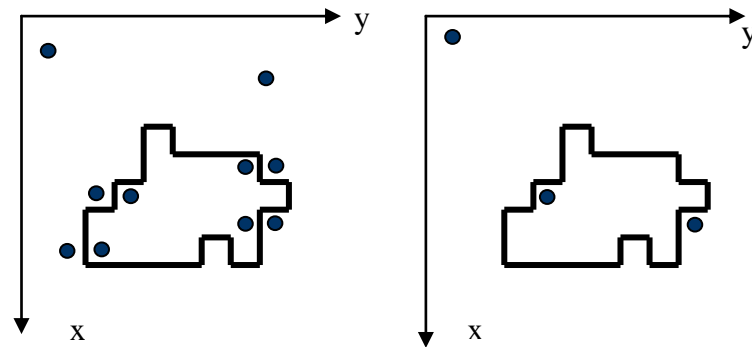


Figura 1.29. Selección de las semillas, estado inicial y final

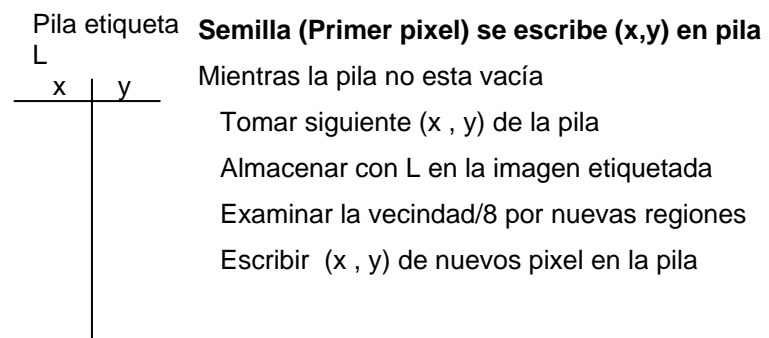


Figura 1.30: Región L, pila de búsqueda y almacenamiento de pixels

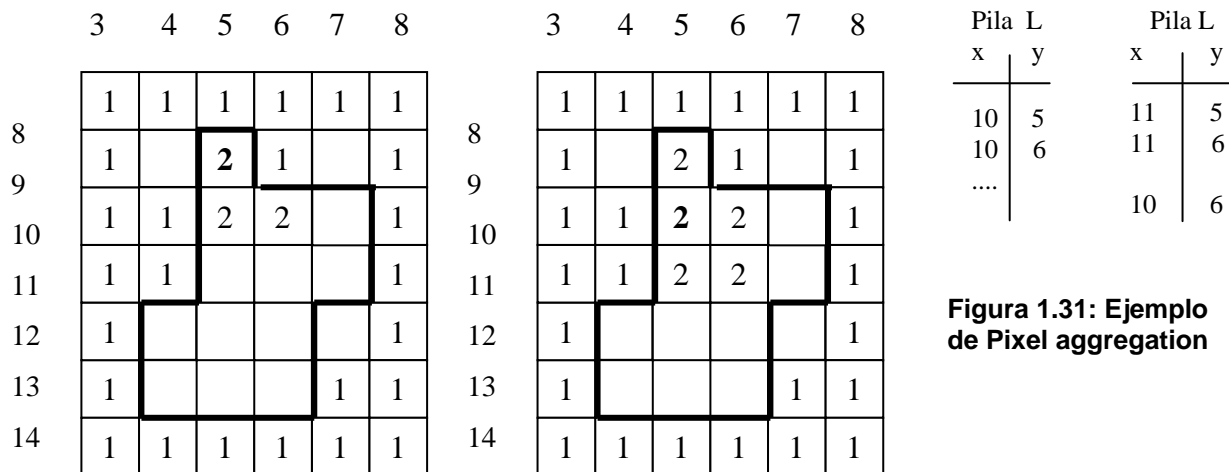


Figura 1.31: Ejemplo de Pixel aggregation

1.2.9. Transformada de Hough

A menudo es de interés detectar ciertas morfologías simples de los bordes, como por ejemplo bordes rectilíneos y círculos. Particularmente en estos casos es aplicable un método muy especial denominado transformada de Hough.

Este método permite identificar un conjunto de puntos relevante que pertenecen a la recta o curva y de este modo se pueden obtener los parámetros que definen a ese objeto en la imagen

Para la detección de rectas se obtienen los parámetros de la forma normal Hessiana (r, φ) y el número de pixels dispuestos sobre la recta. En cierto modo se trata de completar una Look-up-Table (LUT). Como puede verse en la Fig. 1.32 los parámetros (r, φ) satisfacen la ecuación $r/r_A = \cos(\varphi - \varphi_A)$, donde (r_A, φ_A) describen la ubicación de A en el plano (x, y) en coordenadas polares.

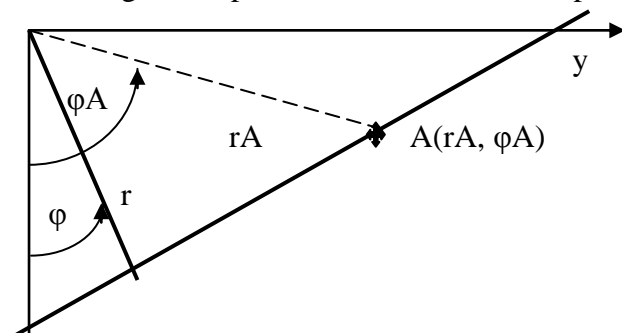


Figura 1.32 Línea expresada mediante los parámetros de la forma hessiana

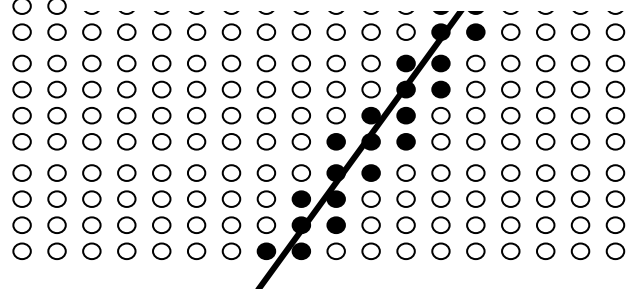


Figura 1.33: Recta con un cluster en el espacio (r, φ)

Siendo A y B pixels de un borde, se desea encontrar la línea de contorno que pasa por A y B.

Se considera el racimo de todas las rectas que pasan por A y B. Cada una de estas puede ser descripta por el ángulo φ y el módulo r de la normal. De acuerdo a la

ecuación citada los parámetros de estas líneas que pasan por A, pasan a través de una curva seno $r \geq 0$ con el punto máximo (r, φ) .

Para obtener un número limitado L de líneas de racimo, se cuantifican los parámetros (r, φ) por ejemplo $\Delta r = 5$ y $\Delta \varphi = 15^\circ$. El racimo se cuantifica de acuerdo a un conjunto de puntos (r_i, φ_i) (por ej. $L = 18/15 = 12$) de acumulación donde cada punto (r_i, φ_i) corresponde a una línea en el racimo.

A causa de esta cuantificación cada recta del racimo no pasa exactamente por A. Para el racimo a través de B se obtiene un punto similar de acumulación (r_B, φ_B) .

Pasos de la transformada de Hough

1. Dados los pixels de una imagen de binaria, cuyos bordes aparecen como líneas blancas gruesas en fondo negro
2. Elegir los parámetros de cuantización Δr y $\Delta \varphi$; por ejemplo $\Delta r = 5$ pixels y $\Delta \varphi = 15^\circ$. Un racimo consiste de $L = 180 / \Delta \varphi = 180 / 15 = 12$ líneas rectas, y cada curva de racimo tiene cuantificados 12 puntos (r, φ) .
3. Para cada pixel P con coordenadas polares (r_p, φ_p) en la imagen B, se calculan las posiciones de L puntos de racimo cuantificados de la curva. $r / r_p = \cos(\varphi - \varphi_p)$ e incrementar la correspondiente cuenta de pixels $z(r_p, \varphi_p)$

$$z(r_p, \varphi_p) = z(r_p, \varphi_p) + 1$$
4. Encontrar los pixels (r, φ) con altas acumulaciones $z(r, \varphi) \gg 1$. Generalmente estos puntos aparecen en clusters los cuales indican que se han detectado varias líneas con similares parámetros (r, φ) . Un cluster de este tipo podría ser causado por ejemplo por el objeto de la figura.

Una notable característica de la transformada de Hough es su robustez con respecto a las distorsiones. Las líneas rectas son detectadas aún si están interrumpidas o no todos los pixels yacen exactamente sobre la línea.