

CAPÍTULO 7

DESCRIPCIÓN DE TEXTURA EN IMÁGENES UTILIZANDO LOCAL BINARY PATTERN (LBP)

Oscar GARCÍA-OLALLA¹, Enrique ALEGRE¹

¹ Universidad de León, Departamento de Ingeniería eléctrica y de Sistemas y Automática, Escuela de Ingenierías, León, España.

En este capítulo vamos a estudiar uno de los descriptores de textura más utilizados en la actualidad, llamado Local Binary Pattern (LBP) o patrón binario local, que tiene numerosas aplicaciones en el análisis de imágenes digitales y, en general, en la visión por computador. Previamente, se introducirá el concepto de textura en una imagen digital y los tipos de descriptores que se utilizan para caracterizarla. Posteriormente se explicará con detalle qué es y cómo se obtiene un Patrón Binario Local (LBP), algunas variaciones en su cálculo que fueron propuestas como extensiones al descriptor básico y tres métodos derivados del LBP como son el ALBP (Adaptive Local Binary Pattern), el LBPV (Local Binary Pattern Variance) y el CLBP (Completed Local Binary Pattern). Finalmente se comentan algunas aplicaciones tanto del LBP como de sus métodos derivados.

7.1. Introducción

7.1.1. Concepto de textura

A pesar de que su estudio es un campo de investigación importante en visión por computador, no hay una descripción formal de lo que es una textura. El principal motivo para que la definición de textura no sea universal, es que las características propias de una textura a menudo son opuestas en diferentes tipos de imágenes: Regularidad frente a aleatoriedad, uniformidad frente a heterogeneidad,... A lo largo de los últimos años, distintos investigadores han tratado de definir la textura desde una perspectiva concreta de su naturaleza. Haralick (1979), considera una textura como una *“región organizada que puede ser descompuesta en primitivas teniendo unas distribuciones espaciales concretas”*. Esta definición, conocida como la aproximación estructural, se basa en la idea de que cada textura está compuesta por una serie de elementos básicos (puntos, aristas, etc.), similar a

la que utiliza el ser humano. Alternativamente, Cross y Jain (1983) definieron textura, desde un punto de vista estocástico, como “*una región de la imagen bidimensional, aleatoria y posiblemente periódica*”. Además de las definiciones clásicas, se pueden encontrar otras más orientadas al análisis de imágenes, en las que se define una textura como “*la distribución espacial de color o intensidad en una imagen o en una región de la misma*”. (Zhou, 2006)

7.1.2. Tipos de textura

En la actualidad, las texturas se pueden clasificar en diferentes categorías dependiendo del problema que se desee resolver (Pietikäinen y Zhao, 2009), como son:

- Microtexturas o Macrotexturas.
- Irregulares o regulares.
- Repetitivas o no repetitivas.
- Con alto contraste y no direccionales frente a bajo contraste y direccionales
- Granulada y de baja complejidad frente a no granuladas y de alta complejidad.

Estas tres últimas categorías pertenecen a las tres dimensiones ortogonales de las texturas propuestas por Rao y Lohse en 1993.

7.1.3. Características de los descriptores de textura

Debido a la alta complejidad y variabilidad de las texturas, es muy difícil crear un descriptor que se adapte a todas ellas y sea capaz de identificarlas a la perfección. Sin embargo, un buen descriptor de textura debe cumplir los siguientes requisitos:

1. Eficiente, diferenciando entre múltiples tipos de texturas.
2. Robusto frente a variaciones en el escalado y la pose.
3. Robusto frente a variaciones en la iluminación.
4. Robusto frente a la falta de uniformidad espacial.
5. Debe funcionar para imágenes de pequeño tamaño.
6. Debe tener baja complejidad computacional.

Sin embargo, es muy difícil encontrar técnicas de descripción de texturas que cumplan con todas estas características. Es por ello que en la actualidad hay gran actividad investigadora en este campo y cada año aparecen nuevas técnicas que intentan satisfacer los requisitos anteriores.

7.1.4. Tipos de descriptores de textura

Los descriptores de textura pueden clasificarse en diferentes tipos en función de las propiedades de la imagen que se tienen en cuenta a la hora de obtener sus características. Según Tuceryan y Jain (1993) los descriptores de textura se dividen en las siguientes categorías:

- **Estructurales:** Representan la textura por medio de una jerarquía y primitivas bien definidas. Para ello primero hay que especificar las primitivas y luego las reglas de posicionamiento. Estos métodos tratan de expresar de manera rigurosa la estructura de la región, por lo que funcionan mejor en texturas regulares y repetitivas, mientras que se comportan peor cuando se tratan de texturas naturales debido a la alta variabilidad que presentan.
- **Estadísticos:** Estas técnicas no pretenden describir la estructura jerárquica de la textura, sino que la representan mediante propiedades basadas en la distribución y la relación entre los niveles de gris de la imagen. Los métodos basados en estadísticas de segundo orden han demostrado obtener buenos resultados siendo el más conocido la matriz de co-ocurrencia (Haralick 1979).
- **Basados en modelos:** Estos descriptores utilizan fractales y modelos aleatorios para describir la estructura de la imagen, ajustando los parámetros de dichos modelos hasta encontrar los que mejor representan a la textura. La estimación de los parámetros óptimos conlleva una complejidad computacional elevada, por lo que no son apropiados en sistemas donde el tiempo es un factor clave.
- **Basados en transformaciones:** Representan la imagen en un espacio cuyo sistema de coordenadas tiene una interpretación que está relacionada con las características propias de la textura como la frecuencia o el tamaño. Técnicas como la transformada de Fourier (Rosenfeld y Weszka, 1980), Gabor (Daugman, 1985) o Wavelet (Mallat, 1989) han sido frecuentemente utilizadas, obteniendo esta última resultados muy interesantes, sobre todo para segmentación basada en textura.

7.2. Métodos de descripción de textura

En Haralick y col. (1973) se propuso un método de descripción de textura basado en la matriz de co-ocurrencia. Esta matriz refleja la distribución de los niveles de gris para dos píxeles situados a una determinada distancia y ángulo. Utilizando como base esta matriz, Haralick extrae un conjunto de 14 momentos para describir la textura. Es la técnica que más se ha utilizado para describir texturas desde que se desarrolló y funciona muy bien con texturas estocásticas, sin embargo su coste computacional es elevado y requiere de una normalización previa del nivel de gris.

En Laws (1980), Laws desarrolló como proyecto de tesis doctoral un descriptor basado en la utilización de nueve máscaras de tamaño 3x3 para realzar características de la imagen como pueden ser bordes, media del nivel de gris, crestas, etc. Las máscaras se obtienen multiplicando entre sí tres vectores:

- Level: $L3 = [1 \ 2 \ 1]$
- Edge: $E3 = [-1 \ 0 \ 1]$
- Spot : $S3 = [-1 \ 2 \ -1]$

El mayor inconveniente de los descriptores de Laws es el tiempo computacional que requiere realizar la convolución de las máscaras seleccionadas con la imagen original, por lo que este método tampoco sería apropiado para situaciones en las que el tiempo sea una restricción.

Durante más de 40 años, el espectro de Fourier ha sido utilizado para múltiples aplicaciones en el campo del tratamiento digital de imágenes. Diferentes trabajos han demostrado su gran capacidad a la hora de describir patrones bidimensionales periódicos en una imagen, obteniendo resultados que son muy difíciles de lograr con técnicas locales debido a su propia naturaleza. Sin embargo, el uso de esta técnica en la mayoría de texturas ofrece un rendimiento bajo debido a su falta de localización espacial. Los filtros de Gabor proporcionan herramientas para mejorar esta localización espacial. Su respuesta es similar a la del cortex visual del ser humano. Sin embargo, su coste computacional es muy alto y en casos donde las texturas son muy irregulares el rendimiento disminuye. Wavelet es una de las técnicas basadas en transformadas que mejor funciona en el reconocimiento de texturas (Stavros 2009). Esta transformada mapea la imagen en una sub-imagen de baja resolución, normalmente llamada “imagen tendencia” y tres imágenes de detalles para distintas orientaciones. Esta imagen tendencia se obtiene aplicando filtrado paso bajo, lo que produce un efecto de procesos de desenfoque a la imagen original y las imágenes detalladas se obtienen extrayendo la información que se pierde al realizar este proceso. La energía o la desviación típica de las imágenes detalladas proporcionan información muy valiosa a la hora de detectar texturas, que finalmente formarán el descriptor de éstas. La principal ventaja que otorga el uso de Wavelet para describir texturas es que permite realizar análisis multiresolución de la imagen, facilitando la descripción de todo tipo texturas.

7.3 Local Binary Pattern (LBP)

7.3.1. Concepto

Local Binary Pattern (LBP) (Ojala y col. 1994) es un operador de textura simple y eficiente que etiqueta cada píxel de la imagen analizando su vecindario, estudiando si el nivel de gris de cada píxel supera un determinado umbral y codificando dicha comparación mediante un número binario. Debido a su bajo coste computacional y al gran poder discriminativo que ha demostrado tener, LBP se ha convertido en los últimos años en una de las soluciones más utilizadas en numerosas aplicaciones relacionadas con textura. La característica más importante de LBP en aplicaciones reales es la robustez que ofrece frente a variaciones en la intensidad del nivel de gris, causado, entre otras muchas cosas, por diferencias en la iluminación.

Para calcular LBP sobre una imagen en escala de grises se utiliza la ecuación 7.1.

$$LBP_{P,R} = \sum_{p=0}^{P-1} s(g_p - g_c) 2^p, \quad s(x) = \begin{cases} 1 & \text{si } x \geq 0 \\ 0 & \text{si } x < 0 \end{cases} \quad (7.1)$$

Donde P es el número de vecinos que se van a considerar, R es el tamaño del vecindario y, g_c y g_p son los valores de gris del píxel central y cada uno de los p píxeles del vecindario respectivamente. En la figura 7.1 podemos observar un ejemplo del cálculo del LBP de manera gráfica sobre un píxel cualquiera de una imagen para los parámetros $P=8$ y $R=1$.

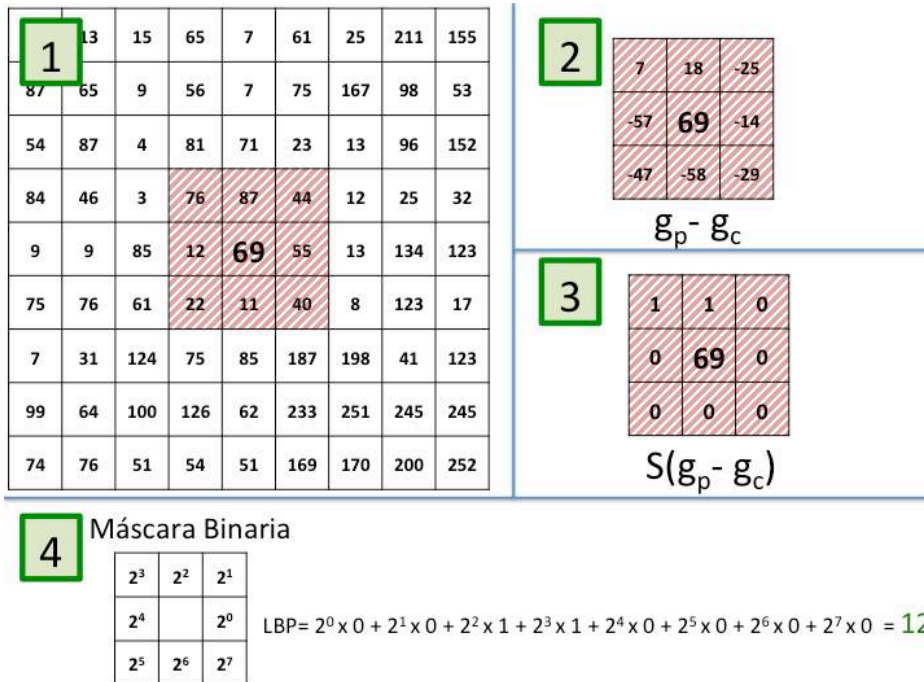


Figura 7.1. Esquema de los pasos necesarios para obtener el valor de LBP de un píxel concreto.

En 2002, Ojala y col. (2002) propusieron una mejora del método, incorporando vecindarios circulares, donde los vecinos se ubicaban igualmente espaciados, realizando una interpolación bilineal en caso de que éstos no coincidieran con el centro de un píxel concreto de la imagen. En el caso de la figura 7.1, el proceso sería idéntico salvo que los valores de los vecinos colocados en las esquinas no se corresponderían con 44, 76, 22 y 40 para los vecinos 1, 3, 5 y 7 respectivamente, ya que serían el resultado de una interpolación entre ese valor de gris y los adyacentes a éstos. En lo sucesivo, cuando se calculen valores de LBP en este capítulo, se hará teniendo en cuenta el vecindario cuadrado debido a que facilita mucho la comprensión por parte del lector.

Una vez obtenido el valor del LBP para cada uno de los píxeles de la imagen, se obtiene el histograma de la imagen, dando como resultado un vector de 256 elementos.

Ejemplo 7.1. Obtener el histograma LBP para el fragmento de la imagen que se muestra en la figura 7.2 utilizando 13, 24 y 10 bins (agrupaciones) con radio 1 y 8 vecinos (sin considerar interpolaciones).

		1	2	3	4	5	6	7
1	0	0	0	0	0	0	0	0
2	0	30	30	20	30	30	0	0
3	0	30	20	20	20	30	0	0
4	0	20	10	10	10	20	0	0
5	0	5	10	10	10	5	0	0
6	0	5	10	10	10	5	0	0
7	0	5	10	10	10	5	0	0

Figura 7.2. Fragmento de imagen en escala de grises.

Solución: El primer paso es determinar la región sobre la que se va a calcular LBP. Al ser un vecindario de radio 1, se debe eliminar del cálculo 1 píxel de cada borde de la imagen. El resultado se puede ver en la figura 7.3.

		1	2	3	4	5	6	7
1								
2			30	30	20	30	30	
3			30	20	20	20	30	
4			20	10	10	10	20	
5			5	10	10	10	5	
6			5	10	10	10	5	
7								

Figura 7.3. Área de la imagen sobre la que se debe calcular LBP

A continuación, en la figura 7.4, se muestran ejemplos de la extracción del valor de LBP para 4 píxeles de la imagen.

							1	2	4	8	16	32	64	128		
Píxel (1,1)	0	0	0	0	0	0	0	0	0	1	0	1	0	0		
	0	30	30	0	30	1	0	0	0	8	0	32	0	0		
	0	30	20	0	1	0	LBP8,1 (1,1) =	40	0	0	0	0	0	0		
Píxel (4,4)	20	20	20	1	1	1										
	10	10	10	1	10	1										
	10	10	10	1	1	1	LBP8,1 (4,4) =	255	1	2	4	8	16	32	64	128
Píxel (6,6)	10	5	0	1	1	0										
	10	5	0	1	5	0										
	10	5	0	1	1	0	LBP8,1 (6,6) =	227	1	1	0	0	0	1	1	1
Píxel (4,6)	20	30	0	1	1	0										
	10	20	0	0	20	0										
	10	5	0	0	0	0	LBP8,1 (4,6) =	3	1	2	0	0	0	0	0	0

Figura 7.4. Cálculo detallado de LBP para 4 píxeles de la imagen.

Realizando esta operación para todos los píxeles obtendríamos los valores que se muestran en la figura 7.5:

	Gris	LBP	1	2	4	8	16	32	64	128		Gris	LBP	1	2	4	8	16	32	64	128	
(2,2)	30	40	0	0	0	1	0	1	0	0	(3,2)	5	62	0	1	1	1	1	1	0	0	
(2,3)	30	192	0	0	0	0	0	0	1	1	(3,3)	10	63	1	1	1	1	1	1	0	0	
(2,4)	20	248	0	0	0	1	1	1	1	1	(3,4)	10	255	1	1	1	1	1	1	1	1	
(2,5)	30	24	0	0	0	1	1	0	0	0	(3,5)	10	231	1	1	1	0	0	1	1	1	
(2,6)	30	160	0	0	0	0	0	1	0	1	(3,6)	5	195	1	1	0	0	0	0	1	1	
(3,2)	30	6	0	1	1	0	0	0	0	0	(4,2)	5	62	0	1	1	1	1	1	0	0	
(3,3)	20	207	1	1	1	1	0	0	1	1	(4,3)	10	62	0	1	1	1	1	1	0	0	
(3,4)	20	143	1	1	1	1	0	0	0	1	(4,4)	10	255	1	1	1	1	1	1	1	1	
(3,5)	20	159	1	1	1	1	1	0	0	1	(4,5)	10	227	1	1	0	0	0	1	1	1	
(3,6)	30	3	1	1	0	0	0	0	0	0	(4,6)	5	227	1	1	0	0	0	1	1	1	
(4,2)	20	6	0	1	1	0	0	0	0	0												
(4,3)	10	191	1	1	1	1	1	1	0	1												
(4,4)	10	255	1	1	1	1	1	1	1	1												
(4,5)	10	239	1	1	1	1	0	1	1	1												
(4,6)	20	3	1	1	0	0	0	0	0	0												

Figura 7.5. Cálculo detallado de LBP para todos los píxeles de la región. En la figura 7.6 se puede ver como quedaría el mapa de valores LBP.

	1	2	3	4	5	6	7
1	0	0	0	0	0	0	0
2	0	40	192	248	24	160	0
3	0	6	207	143	159	3	0
4	0	6	191	255	239	3	0
5	0	62	63	255	231	195	0
6	0	62	62	255	227	227	0
7	0	0	0	0	0	0	0

Figura 7.6. Representación visual de los valores LBP con 8 vecinos y radio 1.

A continuación, tenemos que calcular la frecuencia de los valores para obtener el histograma. En el primer caso, se pide utilizar 13 bins, por lo que se divide el espacio de posibles valores [0,255] en 13 regiones iguales. En este caso no es posible por lo que la última región contará con tan solo 15 posibles valores, mientras que el resto abarcará 20. En la figura 7.7 se puede ver la representación gráfica del histograma.

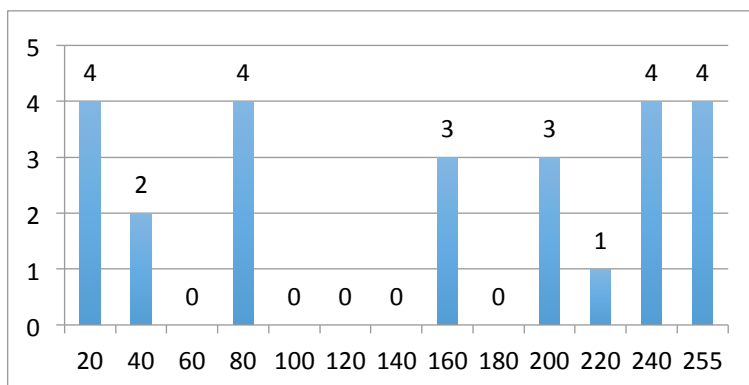


Figura 7.7. Histograma utilizando 13 bins del LBP de la región de la imagen de la figura 7.2

Realizando el mismo proceso se obtienen los histogramas para los dos casos restantes. En la figura 7.8 para 24 bins y en la figura 7.9 para 10 bins.

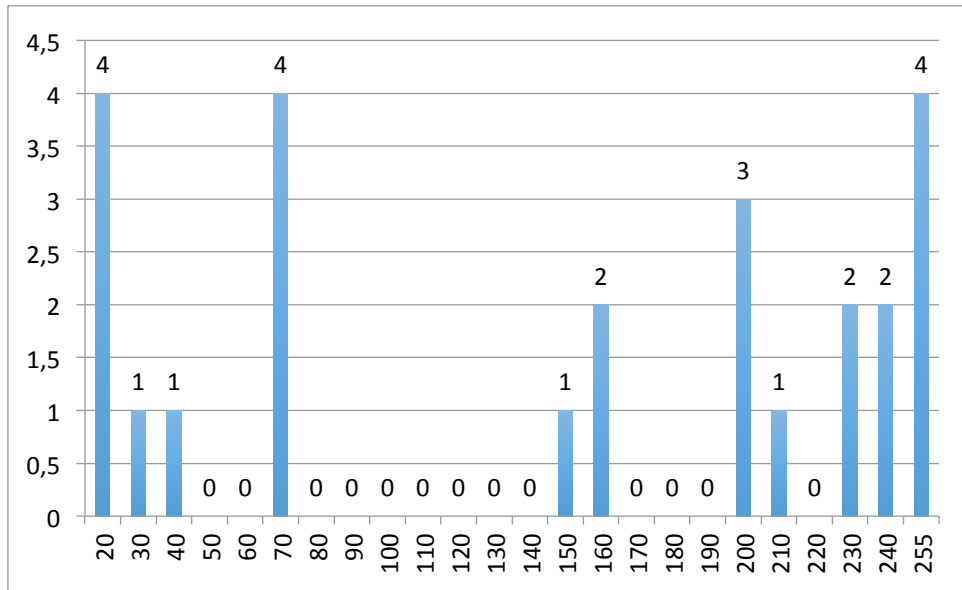


Figura 7.8. Frecuencias e histograma utilizando 24 bins del LBP de la región de la imagen.

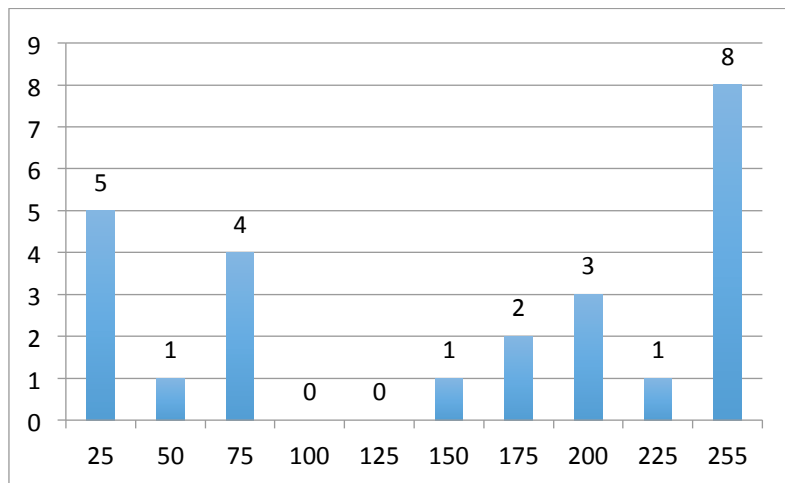


Figura 7.9. Histograma utilizando 10 bins del LBP de la región de la imagen.

7.3.2. Variaciones

7.3.2.1 Invariancia a la rotación

La primera mejora que se propuso, de gran importancia para el campo de la descripción de textura, fue la invariancia a la rotación. Atendiendo a la definición de Local Binary Pattern, es fácil darse cuenta que el método no es invariante a la rotación. En la figura 7.10 podemos ver un ejemplo de un patrón una vez aplicado el umbral a la diferencia de valor de gris, y el mismo rotado 90 grados, viendo que el resultado final es diferente

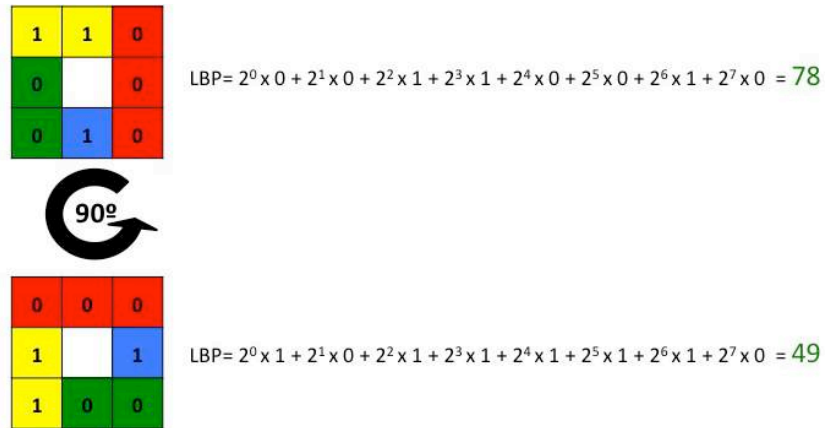


Figura 7.10 Ejemplo de dos patrones iguales pero con diferente orientación.

Para corregir esta falta de robustez, se propuso un método basado en asignar a cada píxel el menor valor resultante de LBP de entre todas las posibles rotaciones del patrón. En la figura 7.11 se pueden ver todas las posibles combinaciones de un patrón concreto y el valor asignado, correspondiente a la rotación 7.

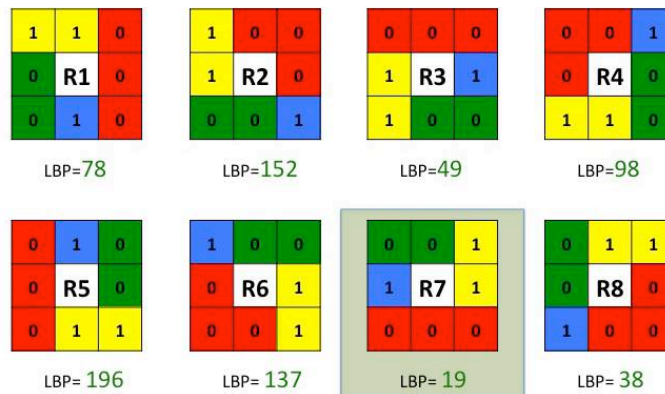


Figura 7.11. Ejemplo de todas las posibles rotaciones de un vector y su valor de LBP. En el método invariante a la rotación, se selecciona el valor mínimo para representar al píxel central.

Utilizando la ecuación 7.2 se puede obtener una descripción invariante a la rotación aplicando la idea que acabamos de mencionar:

$$LBP_{P,R}^{ri} = \min\{ROR(LBP_{P,R},i) | i = 0, \dots, P-1\} \tag{7.2}$$

donde $ROR(x,i)$ es un desplazamiento hacia la derecha de i píxeles para el patrón x .

7.3.2.2 Uniformidad

En 2002 se realizó una nueva propuesta por parte de varios de los autores del método original (Ojala y col. 2002), al darse cuenta de que la mayoría de la información relevante de la textura era descrita mediante un patrón uniforme. Se puede definir patrón uniforme como aquel patrón que tiene como mucho dos saltos entre 0 y 1 o entre 1 y 0 en su cadena, asumiéndose que es un patrón circular,

por lo que es necesario comprobar también si hay transición entre el último elemento y el primero. En la figura 7.12 podemos observar los distintos patrones uniformes que existen y algunos ejemplos de aquellos que no lo son.

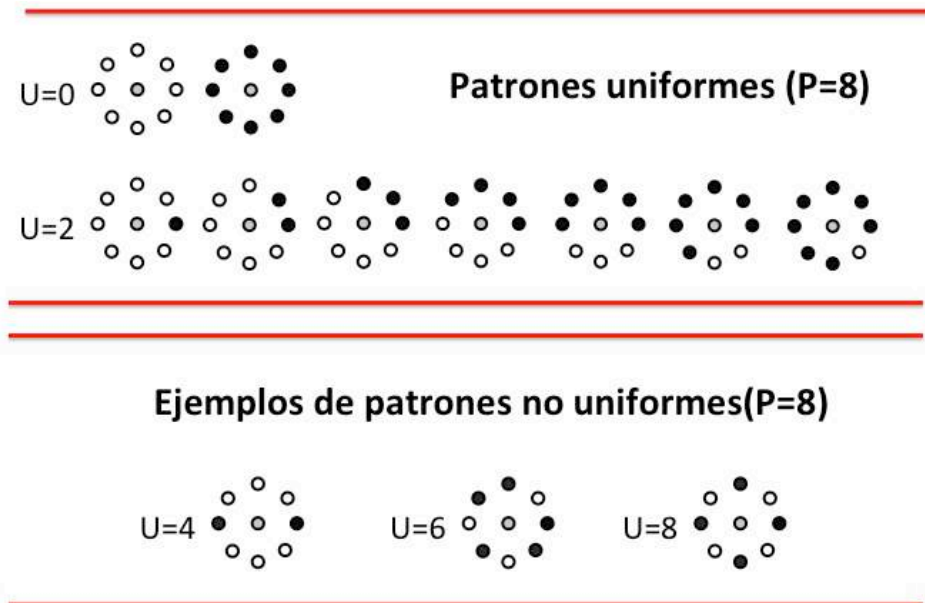


Figura 7.12. Patrones uniformes y no uniformes en LBP.

En la figura 7.13 podemos ver las primitivas que representan los valores de algunos de los patrones uniformes.

Teniendo en cuenta únicamente estos patrones, cualquier píxel de la imagen se puede representar mediante $P+2$ valores diferentes, asignando desde 0 hasta P el valor que corresponde con el número de píxeles cuyo valor es 1, y el valor $P+1$ a cualquier patrón no uniforme.



Figura 7.13. Patrones uniformes y no uniformes en LBP.

En la ecuación 7.3 se puede ver de manera matemática el cálculo del LBP uniforme.

$$LBP_{P,R}^{riu2} = \begin{cases} \sum_{p=0}^{P-1} s(g_p - g_c) & \text{si } U(LBP_{P,R}) \leq 2 \\ P + 1 & \text{otherwise} \end{cases} \quad (7.3)$$

Donde $U(LBP)$ se define como indica la ecuación 7.4

$$U(LBP) = |s(g_{P-1} - g_c) - s(g_0 - g_c)| + \sum_{p=1}^{P-1} s(g_p - g_c) - s(g_{p-1} - g_c) \quad (7.4)$$

Si el valor de U para un determinado patrón del LBP es menor o igual que 2, se designa como patrón uniforme. Los patrones que tienen un valor de U mayor que 2 son catalogados como no uniformes. El patrón $L1=00001000$ es uniforme porque $U(L1)=2$, mientras que el patrón $L2=01100100$ es no uniforme porque $U=4$.

Ejemplo 7.2. Obtener el histograma LBP Uniforme para el fragmento de la imagen que se muestra en la figura 7.14 con radio 1 y 8 vecinos (sin considerar interpolaciones).

		1	2	3	4	5	6	7
1	0	0	0	0	0	0	0	0
2	0	30	30	20	30	30	0	0
3	0	30	20	20	20	30	0	0
4	0	20	10	10	10	20	0	0
5	0	5	10	10	10	5	0	0
6	0	5	10	10	10	5	0	0
7	0	5	10	10	10	5	0	0

Figura 7.14. Fragmento de imagen en escala de grises.

Solución: El primer paso es determinar la región sobre la que se va a calcular LBP Uniforme. Al ser un vecindario de radio 1, se debe eliminar del cálculo 1 píxel de cada borde de la imagen. El resultado se puede ver en la figura 7.15. Este paso es igual que el realizado a la hora de calcular el método simple de LBP.

		1	2	3	4	5	6	7
1								
2			30	30	20	30	30	
3			30	20	20	20	30	
4			20	10	10	10	20	
5			5	10	10	10	5	
6			5	10	10	10	5	
7								

Figura 7.15. Área de la imagen sobre la que se debe calcular LBP Uniforme

A continuación, en la figura 7.16, se muestran ejemplos de la extracción del valor de LBP Uniforme para 4 píxeles de la imagen.

<table border="1"> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>30</td><td>30</td></tr> <tr><td>0</td><td>30</td><td>20</td></tr> </table>	0	0	0	0	30	30	0	30	20	<table border="1"> <tr><td>0</td><td>0</td><td>0</td></tr> <tr><td>0</td><td>30</td><td>1</td></tr> <tr><td>0</td><td>1</td><td>0</td></tr> </table>	0	0	0	0	30	1	0	1	0	<table border="1"> <tr><td>1</td><td>2</td><td>4</td><td>8</td><td>16</td><td>32</td><td>64</td><td>128</td></tr> </table>	1	2	4	8	16	32	64	128	LBP _{8,1} (1,1) = 9 0 0 0 1 0 1 0 0 NO UNIFORME
0	0	0																											
0	30	30																											
0	30	20																											
0	0	0																											
0	30	1																											
0	1	0																											
1	2	4	8	16	32	64	128																						
<table border="1"> <tr><td>20</td><td>20</td><td>20</td></tr> <tr><td>10</td><td>10</td><td>10</td></tr> <tr><td>10</td><td>10</td><td>10</td></tr> </table>	20	20	20	10	10	10	10	10	10	<table border="1"> <tr><td>1</td><td>1</td><td>1</td></tr> <tr><td>1</td><td>10</td><td>1</td></tr> <tr><td>1</td><td>1</td><td>1</td></tr> </table>	1	1	1	1	10	1	1	1	1		LBP _{8,1} (4,4) = 8 1 1 1 1 1 1 1 UNIFORME								
20	20	20																											
10	10	10																											
10	10	10																											
1	1	1																											
1	10	1																											
1	1	1																											
<table border="1"> <tr><td>10</td><td>5</td><td>0</td></tr> <tr><td>10</td><td>5</td><td>0</td></tr> <tr><td>10</td><td>5</td><td>0</td></tr> </table>	10	5	0	10	5	0	10	5	0	<table border="1"> <tr><td>1</td><td>1</td><td>0</td></tr> <tr><td>1</td><td>5</td><td>0</td></tr> <tr><td>1</td><td>1</td><td>0</td></tr> </table>	1	1	0	1	5	0	1	1	0		LBP _{8,1} (6,6) = 5 1 1 0 0 0 1 1 ESQUINA								
10	5	0																											
10	5	0																											
10	5	0																											
1	1	0																											
1	5	0																											
1	1	0																											
<table border="1"> <tr><td>20</td><td>30</td><td>0</td></tr> <tr><td>10</td><td>20</td><td>0</td></tr> <tr><td>10</td><td>5</td><td>0</td></tr> </table>	20	30	0	10	20	0	10	5	0	<table border="1"> <tr><td>1</td><td>1</td><td>0</td></tr> <tr><td>0</td><td>20</td><td>0</td></tr> <tr><td>0</td><td>0</td><td>0</td></tr> </table>	1	1	0	0	20	0	0	0	0		LBP _{8,1} (4,6) = 2 1 1 0 0 0 0 0 FIN DE LÍNEA								
20	30	0																											
10	20	0																											
10	5	0																											
1	1	0																											
0	20	0																											
0	0	0																											

Figura 7.16. Cálculo detallado de LBP Uniforme para 4 píxeles de la imagen.

Aplicando la fórmula de LBP uniforme para todos los píxeles obtendríamos los valores que se muestran en la figura 7.17:

	Gris	LBP	1	2	4	8	16	32	64	128		Gris	LBP	1	2	4	8	16	32	64	128
(2,2)	30	9	0	0	0	1	0	1	0	0	(3,2)	5	5	0	1	1	1	1	1	0	0
(2,3)	30	2	0	0	0	0	0	0	1	1	(3,3)	10	6	1	1	1	1	1	1	0	0
(2,4)	20	5	0	0	0	1	1	1	1	1	(3,4)	10	8	1	1	1	1	1	1	1	1
(2,5)	30	2	0	0	0	1	1	0	0	0	(3,5)	10	6	1	1	1	0	0	1	1	1
(2,6)	30	9	0	0	0	0	0	1	0	1	(3,6)	5	4	1	1	0	0	0	0	1	1
(3,2)	30	2	0	1	1	0	0	0	0	0	(4,2)	5	5	0	1	1	1	1	1	0	0
(3,3)	20	6	1	1	1	1	0	0	1	1	(4,3)	10	5	0	1	1	1	1	1	0	0
(3,4)	20	5	1	1	1	1	0	0	0	1	(4,4)	10	8	1	1	1	1	1	1	1	1
(3,5)	20	6	1	1	1	1	1	0	0	1	(4,5)	10	5	1	1	0	0	0	1	1	1
(3,6)	30	2	1	1	0	0	0	0	0	0	(4,6)	5	5	1	1	0	0	0	1	1	1
(4,2)	20	2	0	1	1	0	0	0	0	0											
(4,3)	10	7	1	1	1	1	1	1	0	1											
(4,4)	10	8	1	1	1	1	1	1	1	1											
(4,5)	10	7	1	1	1	1	0	1	1	1											
(4,6)	20	2	1	1	0	0	0	0	0	0											

Figura 7.17. Cálculo detallado de LBP Uniforme para todos los píxeles de la región.

En la figura 7.18 se puede ver cómo quedaría el mapa de valores LBP Uniforme distribuidos en la propia región.

	1	2	3	4	5	6	7
1	0	0	0	0	0	0	0
2	0	9	2	5	2	9	0
3	0	2	6	5	6	2	0
4	0	2	7	8	7	2	0
5	0	5	6	8	6	4	0
6	0	5	5	8	5	5	0
7	0	0	0	0	0	0	0

Figura 7.18. Representación visual de los valores LBP Uniforme con 8 vecinos y radio 1.

A continuación, tenemos que calcular la frecuencia de los valores para obtener el histograma. En este caso como tan solo hay 10 posibles valores, el histograma lo generaremos con 10 bins, uno para cada valor. En la figura 7.19 se puede ver las distintas frecuencias para cada valor y el histograma representado gráficamente.

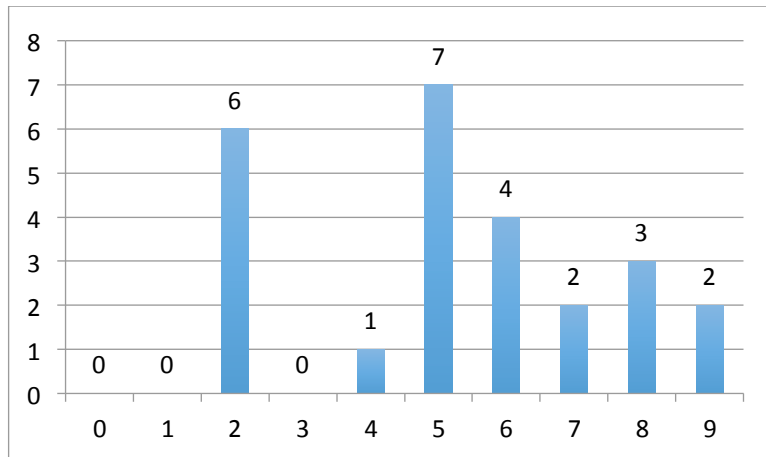


Figura 7.19. Histograma utilizando 10 bins del LBP Uniforme de la región de la imagen de la figura 7.2

7.4. Métodos derivados de LBP

7.4.1. ALBP

En Guo y col. (2010) se propuso un nuevo descriptor de textura basado en LBP, centrado en incorporar información acerca de la orientación de las texturas. El método trata de minimizar las variaciones de la media y la desviación estándar de las diferencias direccionales (diferencias del valor de gris del píxel central y cada uno de los vecinos dependiendo de su posición respecto al centro). Para ello, se propuso añadir un parámetro extra w en la ecuación $|g_c - w_p * g_p|$. La función objetivo es definida según la ecuación (7.5).

$$w_p = \arg_w \min \left\{ \sum_{i=1}^N \sum_{j=1}^M |g_c(i,j) - w \cdot g_p(i,j)|^2 \right\} \quad (7.5)$$

donde w_p es el elemento utilizado para minimizar la diferencia de la dirección p y, N y M son el número de filas y de columnas de la imagen. Una vez dicho esto, la ecuación final del Adaptive Local binary Pattern (ALBP) se define como:

$$ALBP_{p,R} = \sum_{p=0}^{P-1} s(g_p - w_p \cdot g_c) 2^p, s(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ 0 & \text{if } x < 0 \end{cases} \quad (7.6)$$

7.4.2. LBPV

Local Binary Pattern Variance fue propuesto por Guo y su grupo de investigación como una combinación de LBP y un método de distribución de contraste (Guo y col. 2010). Este método utiliza la varianza de la imagen como peso adaptativo para ajustar la contribución de cada valor de LBP en el cálculo del histograma. LBPV se calcula de la siguiente manera:

$$LBPV_{p,R} = \sum_{i=1}^N \sum_{j=1}^M w(LBP_{p,R}(i,j),k), k \in [0, K] \quad (7.7)$$

Donde k es cada uno de los valores del histograma, K es el valor máximo de LBP y w es definido como:

$$w(LBP_{p,R}(i,j),k) = \begin{cases} VAR_{p,R}(i,j), & LBP_{p,R}(i,j) = k \\ 0 & otherwise \end{cases} \quad (7.8)$$

Cuya varianza del vecindario $VAR_{p,R}$ se define como:

$$VAR_{p,R} = \frac{1}{P} \sum_{p=0}^{P-1} (g_p - \mu)^2 \quad (7.9)$$

donde μ es la media del vecindario.

7.4.3. CLBP

El mismo grupo de investigación que desarrolló LBPV y ALBP, presentó otro método llamado CLBP (Compound Local Binary Pattern) que trata de generalizar y completar la información que aporta el LBP clásico (Guo y col. 2010). En este método, una región local de la imagen es representada por su píxel central y una transformada local de diferencias signo-magnitud llamada LDSMT por sus iniciales en inglés. LDSMT descompone estructura local de la imagen en dos componentes complementarios: la diferencia de signos, que se corresponde con el clásico LBP y lo llaman CLBP_S y la diferencia de magnitudes CLBP_M que se define utilizando la ecuación:

$$CLBP_{-M_{p,R}} = \sum_{p=0}^{P-1} t(g_p - g_c, c) 2^p, t(x, c) = \begin{cases} 1 & \text{si } x \geq c \\ 0 & \text{si } x < c \end{cases} \quad (7.10)$$

siendo c un umbral determinado adaptativamente. Normalmente se utiliza el valor medio de todas las diferencias de magnitudes de la imagen.

7.5. Aplicaciones de LBP

LBP se ha convertido en un método muy utilizado en visión por computador debido a su alto poder discriminante, la tolerancia frente a cambios de iluminación y su simplicidad computacional.

Se ha utilizado LBP para realizar segmentación no supervisada basada en características de textura de la imagen, Ojala y Pietikäinen (1999). Para ello primero se divide la imagen de manera jerárquica en un número de regiones escogido por el usuario. A continuación se realiza una fusión de aquellas texturas contiguas que tengan un descriptor similar dejando para el final refinar las regiones, eliminando las aristas producidas en el paso de la división jerárquica, obteniendo una segmentación más ajustada y precisa.

En el ámbito del reconocimiento de rostros, LBP ha sido utilizado obteniendo buenos resultados. Yang y Chen (2013), realizaron un estudio evaluando diferentes técnicas basadas en LBP para el reconocimiento de rostros demostrando, entre otras cosas, que al dividir el rostro en subregiones para posteriormente describirlas usando LBP incrementaba notablemente la precisión de los sistemas de clasificación frente a la descripción del rostro por completo.

También se ha utilizado LBP para la detección de rostros en imágenes. En este campo se pueden utilizar diferentes estrategias para detectar rostros, por ejemplo en el trabajo de Jiayu y Chengjun (2013), se centran en detectar los ojos de las personas para estimar la posición de la cara teniendo en cuenta que el ojo es una región de la imagen con una alta variabilidad con respecto a las zonas adyacentes del mismo (resto de la cara).

Aunque el campo de la detección de objetos está muy focalizado en técnicas de características locales invariantes tipo SIFT o SURF, es posible encontrar trabajos que también han utilizado estrategias basadas en LBP para la detección de objetos. En el trabajo de Dornaika y col. (2014), se propone una técnica basada en la construcción de un grafo adaptativo que utiliza esta técnica (LBP) para la descripción de las regiones. En este trabajo, se centran en la idea de que el descriptor de una imagen está formado por una combinación del resto de descriptores existentes en la base de datos a los que se les asigna unos pesos concretos. De esta manera, si una imagen contiene un objeto concreto, el peso asociado a ese objeto existente en la base de datos será alto, permitiendo así su detección.

7.6. Bibliografía

- Bo, Y.; Songcan C. (2013) A comparative study on local binary pattern (LBP) based face recognition: LBP histogram versus LBP image, *Neurocomputing*, 120, 365-379.
- Cross, G.R.; Jain, A.K. (1983) Markov Random Field Texture Models. *TransPAMI*, 5, 25-39.
- Daugman, J. (1985) Uncertainty relation for resolution in space, spatial frequency and orientation optimised by two-dimensional visual cortical filters. *Journal of the Optical Society of America*, 2,1160-1169.
- Dornaika, F.; Bosaghzadeh, A.; Salmane, H.; Ruichek, Y. (2014) Graph-based semi-supervised learning with Local Binary Patterns for holistic object categorization, *Expert Systems with Applications*, 41, 7744-7753.
- Jiayu, G.; Chengjun, L.; Feature local binary patterns with application to eye detection, *Neurocomputing*, 113, 138-152.
- Guo, Z.; Zhang, L.; Zhang, D.; Zhang, S. (2010) Rotation invariant texture classification using adaptive LBP with directional statistical features, *Image Processing (ICIP)*, 285 –288.
- Guo, Z.; Zhang, L.; Zhang, D. (2010) Rotation invariant texture classification using lbp variance (LBPV) with global matching, *Pattern Recognition*, 43 (3), 706–719.
- Guo, Z.; Zhang, L.; Zhang, D. (2010) A completed modeling of local binary pattern operator for texture classification, *Image Processing, IEEE Transactions on*, 19(6), 1657–1663.
- Haralick, R. M.; Shanmugam, K.; Dinstein, I. (1973) Textural features for image classification. *IEEE Transactions on Systems, Man and Cybernetics*, 3(6), 610–621.
- Haralick, R.M. (1979) Statistical and structural approaches to texture. *Proc IEEE*, 67(5), 786-804.
- Laws, K.I. (1980) Textured image segmentation. *Ph.D. dissertation*, Univ. Southern California, Los Angeles.
- Mallat, S. (1989) Multifrequency channel decomposition of images and wavelet models. *IEEE Trans. Acoustic, Speech and Signal Processing*, 37(12), 2091-2110.
- Ojala, T.; Pietikäinen, M.; Harwood, D. (1994) Performance evaluation of texture measures with classification based on kullback discrimination of distributions. *Proceedings of the 12th IAPR International Conference on Pattern Recognition (ICPR 1994)*.
- Ojala, T.; Pietikäinen M.; Mäenpää T. (2002) Multiresolution gray-scale and rotation invariant texture classification with Local Binary Patterns, *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24 (7).
- Rao, A.; Lohse, G.L. (1993). Identifying High level features of texture perception. *CVGIP: Graphical Models and Image Processing*, 55(3),218-233.
- Rosenfeld, A.; Weszka, J. (1980) Picture Recognition. *Digital Pattern Recognition*.135-166.
- Tsantis, S.; Dimitropoulos, N.; Cavouras, D.; Nikiforidis, G. (2009) Morphological and wavelet features towards sonographic thyroid nodules evaluation. *Comput Med Imaging Graph*, 33(2), 91-99.
- Turcercyan, M.; Jain, A.K. (1993) *Texture Analysis*, chapter 2, World scientific publishing Co.,235-276.
- Zhou, D. (2006) Texture Analysis and Synthesis using a Generic Markov-Gibbs Image Model. *Master theses*. University of Auckland.