

UNIVERSITÀ DEGLI STUDI DI FIRENZE

Laurea Magistrale in Ingegneria Informatica

Corso di Analisi di Immagini e Video

# Implementazione Multiscale Local Binary Pattern

A. Rizzo, M. Bruni

---

Anno accademico 2012/2013

# Indice

<b>1</b>	<b>Introduzione</b>	<b>2</b>
1.1	Description of region features . . . . .	2
<b>2</b>	<b>Multi-scale Local Binary Pattern</b>	<b>3</b>
2.1	Local Binary Pattern . . . . .	3
2.2	Uniform Local Binary Pattern . . . . .	4
2.2.1	Descrittore . . . . .	6
2.3	Multiscale Local Binary Pattern . . . . .	6
2.3.1	Descrittore . . . . .	6
2.4	Filtro di smoothing gaussiano . . . . .	7
<b>3</b>	<b>Implementazione</b>	<b>10</b>
3.1	Segmentazione . . . . .	10
3.2	Misura di similarità . . . . .	10
<b>4</b>	<b>Conclusioni</b>	<b>14</b>
	<b>Acronimi</b>	<b>15</b>
	<b>Bibliografia</b>	<b>16</b>

# 1 Introduzione

## 1.1 Description of region features

Texture is a very general notion that can be attributed to almost everything in nature. For a human, the texture relates mostly to: A specific, spatially repetitive (micro)structure of surfaces formed by repeating a particular element or several elements in different relative spatial positions. Generally, the repetition involves local variations of scale, orientation, or other geometric and optical features of the elements. Texture is a property of regions: the texture of a point is undefined. Texture involves the spatial distribution of gray levels or colors. Texture in an image can be perceived at different scales or levels of resolution. A region is perceived to have texture when the number of primitive elements in the region is large. If only a few primitive elements are present, then a group of countable objects is perceived instead of a textured image. Texture. It is almost impossible to provide a complete description of textures in words. However, words capture some informal qualitative features that can help discriminate different textures: fineness - coarseness, smoothness, granularity, lineation, directionality, regularity - randomness.

Texture models can be of interest for three main purposes: Segmentation: partition the input image into regions of uniform texture.

Classification: produce a classification map of the input image where each uniform textured region is identified with the texture class it belongs to.

Synthesis: generate an image with a texture that looks like a requested texture class.

## 2 Multi-scale Local Binary Pattern

In questo capitolo verranno presentati alcuni concetti teorici e gli strumenti utilizzati per lo sviluppo dell'applicazione. Il primo paragrafo introdurrà il descrittore di feature Local Binary Pattern (LBP). Verrà analizzata un'estensione del LBP, ovvero Uniform LBP, che viene utilizzato per ridurre la lunghezza del vettore di feature. Analizzeremo il Multi-scale Local Binary Pattern (MLBP) e per concludere parleremo del filtro di smoothing gaussiano utilizzato per il preprocessing dell'immagine.

### 2.1 Local Binary Pattern

Local Binary Pattern (LBP) is a type of feature used for classification in computer vision. LBP is the particular case of the Texture Spectrum model proposed in 1990.[1][2] LBP was first described in 1994.[3][4] It has since been found to be a powerful feature for texture classification; it has further been determined that when LBP is combined with the Histogram of oriented gradients (HOG) classifier, it improves the detection performance considerably on some datasets.[5]

Local Binary Pattern (LBP) is a simple yet very efficient texture operator which labels the pixels of an image by thresholding the neighborhood of each pixel and considers the result as a binary number. Due to its discriminative power and computational simplicity, LBP texture operator has become a popular approach in various applications. It can be seen as a unifying approach to the traditionally divergent statistical and structural models of texture analysis. Perhaps the most important property of the LBP operator in real-world applications is its robustness to monotonic gray-scale changes caused, for example, by illumination variations. Another important property is its computational simplicity, which makes it possible to analyze images in challenging real-time settings.

The basic idea for developing the LBP operator was that two-dimensional surface textures can be described by two complementary measures: local spatial patterns and gray scale contrast. The original LBP operator (Ojala et al. 1996) forms labels for the image pixels by thresholding the 3 x 3 neighborhood of each pixel with the center value and considering the result as a binary number. The histogram of these  $2^3 = 8$  different labels can then be used as a texture descriptor. This operator used jointly with a simple local contrast measure provided very good performance in unsupervised texture segmentation (Ojala and Pietikäinen 1999). After this, many related approaches have been developed for texture and color texture segmentation.

The LBP operator was extended to use neighborhoods of different sizes (Ojala et al. 2002). Using a circular neighborhood and bilinearly interpolating values at non-integer pixel coordinates allow any radius and number of pixels in the neighborhood. The gray scale variance of the local neighborhood can be used as the complementary contrast measure. In the following, the notation (P,R) will be used for pixel neighborhoods which means P sampling points on a circle of radius of R. See Fig. 2 for an example of LBP computation.

An illustration of the basic LBP operator is shown in Figure 3.2 and the corresponding equation is shown below.

R raggio P pixel in the neighborhood

$$LBP_{P,R}(x,y) = \sum_{p=0}^{P-1} s(g_p - g_c) 2^p \quad (1)$$

$$s(x) = \begin{cases} 1, & \text{se } x \geq 0 \\ 0, & \text{se } x < 0 \end{cases} \quad (2)$$

## 2.2 Uniform Local Binary Pattern

Another extension to the original operator is the definition of so called uniform patterns, which can be used to reduce the length of the feature vector and im-

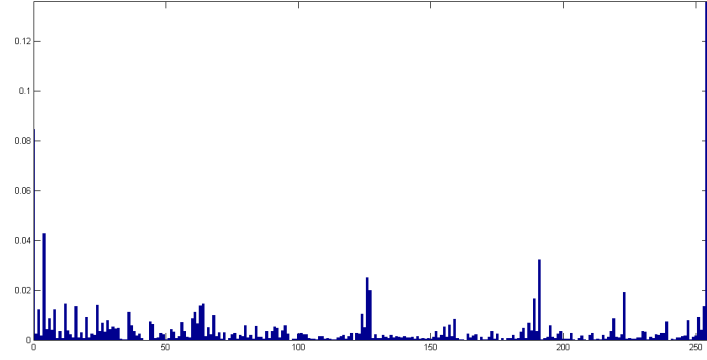


Figura 1: Istogramma immagine LBP Base

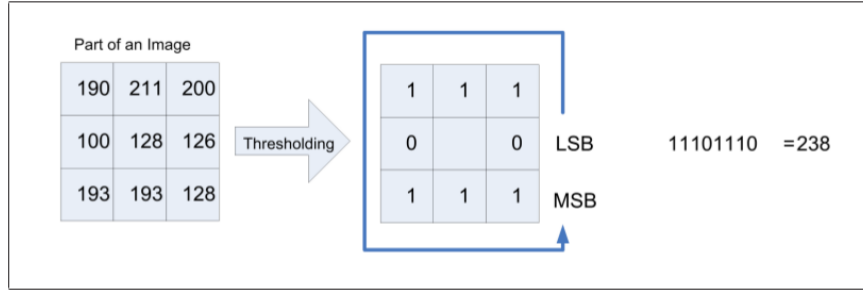


Figura 2: Codice LBP

plement a simple rotation-invariant descriptor. This extension was inspired by the fact that some binary patterns occur more commonly in texture images than others. A local binary pattern is called uniform if the binary pattern contains at most two bitwise transitions from 0 to 1 or vice versa when the bit pattern is traversed circularly. For example, the patterns 00000000 (0 transitions), 01110000 (2 transitions) and 11001111 (2 transitions) are uniform whereas the patterns 11001001 (4 transitions) and 01010010 (6 transitions) are not. In the computation of the LBP labels, uniform patterns are used so that there is a separate label for each uniform pattern and all the non-uniform patterns are labeled with a single label. For example, when using (8,R) neighborhood, there are a total of 256 patterns, 58 of which are uniform, which yields in 59 different labels.

Ojala et al. (2002) noticed in their experiments with texture images that uniform patterns account for a little less than 90% of all patterns when using the (8,1) neighborhood and for around 70% in the (16,2) neighborhood. Each bin (LBP code) can be regarded as a micro-texton. Local primitives which are codified by these bins include different types of curved edges, spots, flat areas etc.

$$LBP_{P,R}^{u2}(x,y) \begin{cases} I(LBP_{P,R}(x,y)), & \text{se } U(LBP_{P,R} \leq 2, I(z) \in [0, (P-1)P+2]) \\ (P-1)P+2, & \text{altrimenti} \end{cases} \quad (3)$$

$$U(LBP_{P,R}) = |s(g_{P-1} - g_c) - s(g_0 - g_c)| + \sum_{p=1}^P |s(g_p - g_c) - s(g_{P-1} - g_c)| \quad (4)$$

### 2.2.1 Descrittore

$$h(i) = \sum_{x,y} B(LBP_{P,R}(x,y)) = i, i \in [0, 2^P - 1] \quad (5)$$

$$B(v) = \begin{cases} 1, & v = \text{true} \\ 0, & \text{altrimenti} \end{cases} \quad (6)$$

## 2.3 Multiscale Local Binary Pattern

### 2.3.1 Descrittore

j = regione r = raggio P = num neighbor

$$h_{P,r,j}(i) = \sum_{x',y' \in M_j} B(LBP_{P,r}(x',y')) = i, i \in [0, L-1], r \in [1, R], j \in [0, J-1] \quad (7)$$

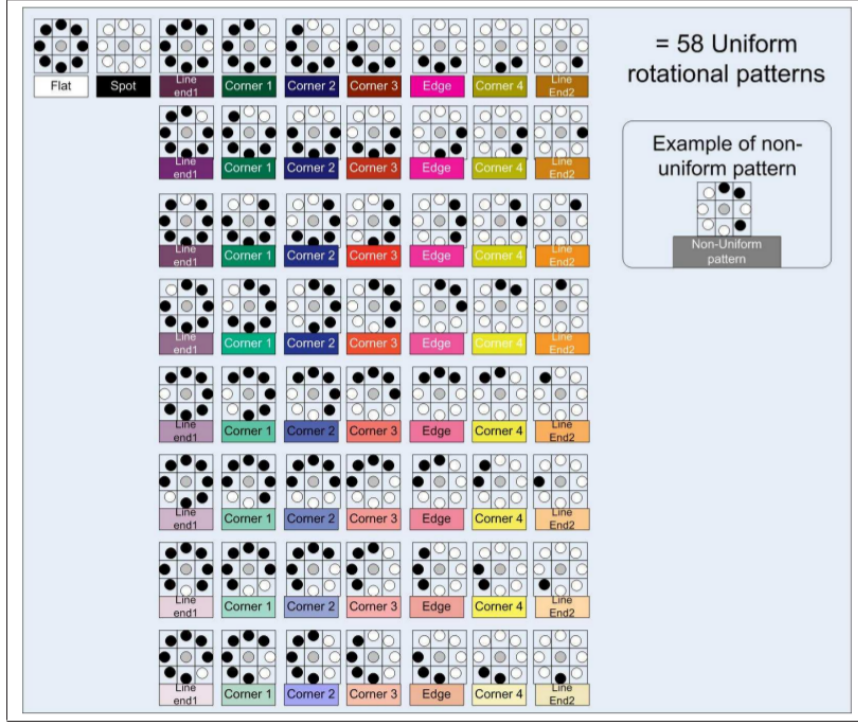


Figura 3: Uniform LBP

$$B(v) = \begin{cases} 1, & v = \text{true} \\ 0, & \text{altrimenti} \end{cases} \quad (8)$$

$$f_j = [h_{P,1,j}, h_{P,2,j}, \dots, h_{P,R,j}] \quad (9)$$

## 2.4 Filtro di smoothing gaussiano

I filtri di *smoothing* vengono utilizzati principalmente per ridurre il rumore presente nell'immagine. Sono anche detti *filtri di media*. Infatti facendo scorrere lungo l'immagine la maschera del filtro, detta anche *kernel mask*, il valore di ogni pixel viene sostituito con la media pesata dei livelli di grigio dei pixel interni alla regione della maschera. Il processo di smoothing permette di ridurre i dettagli meno significativi dell'immagine e mettere in risalto le caratteristi-



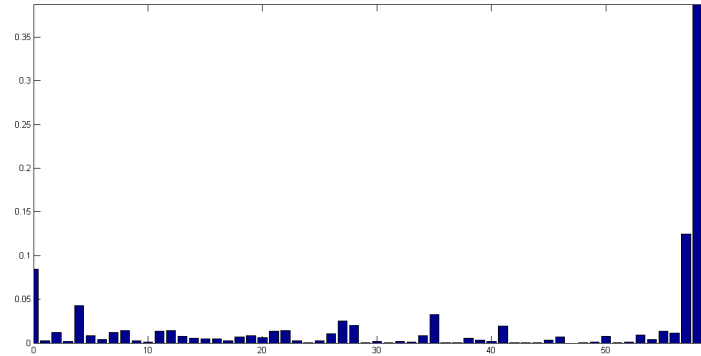


Figura 4: Istogramma immagine LBP Base

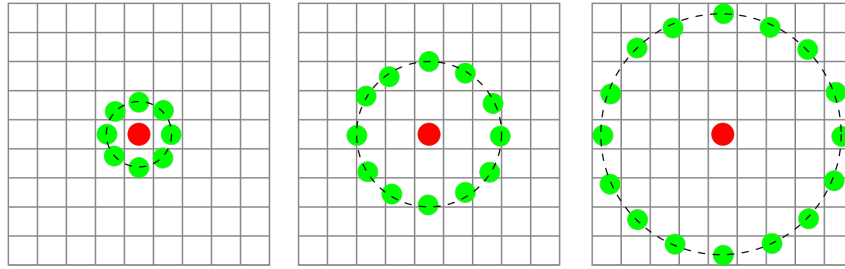


Figura 5: Raggio LBP

che strutturali della stessa. Un effetto indesiderato dello smoothing è quello di produrre un'immagine sfocata, soprattutto se applicato iterativamente. La figura 6 mostra un filtro di smoothing generico 3x3. La maschera del filtro verrà moltiplicata per un coefficiente di normalizzazione in modo tale che la somma degli elementi della maschera faccia uno. I coefficienti della maschera vengono scesi secondo il seguente principio: il peso associato al pixel centrale assume un valore superiore rispetto agli altri. I pesi associati agli altri pixel assumono valori decrescenti all'aumentare della distanza dal pixel centrale. In altre parole si vuole fare in modo che il pixel centrale abbia una maggiore importanza nel calcolo della media. Viceversa i pixel più lontani da quello centrale peseranno di meno nel calcolo della media. Questo permette di ridurre l'effetto indesiderato del processo di smoothing.

$\omega_{1,1}$	$\omega_{1,2}$	$\omega_{1,3}$
$\omega_{2,1}$	$\omega_{2,2}$	$\omega_{2,3}$
$\omega_{3,1}$	$\omega_{3,2}$	$\omega_{3,3}$

Figura 6: Esempio di una kernel mask di dimensioni 3x3, dove  $\omega_{i,j}$  rappresenta il peso applicato al valore del livello di grigio del corrispettivo pixel nell'immagine

Dunque, per ridurre l'effetto indesiderato dello smoothing si è scelto di utilizzare il filtro di smoothing gaussiano. I coefficienti della maschera sono calcolati utilizzando la *2D Gaussian Smoothing Operator*  $G(x, y)$ , cui formula è riportata qui di seguito:

$$G(x, y) = e^{-\frac{(x^2+y^2)}{2\sigma^2}} \quad (10)$$

La dimensione  $n$  della kernel mask è legata al valore di  $\sigma$  dalla seguente formula:

$$3\sigma - 1 = n \quad (11)$$

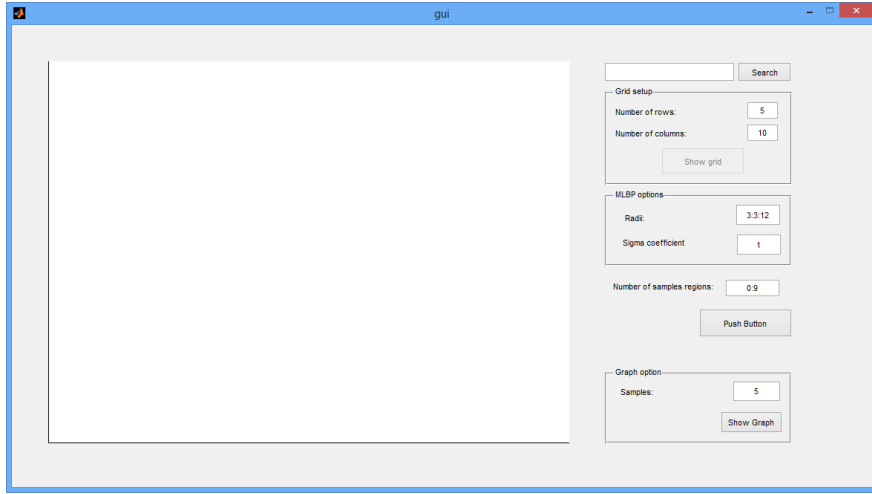


Figura 7: GUI

### 3 Implementazione

#### 3.1 Segmentazione

#### 3.2 Misura di similarità

Per misurare la similarità tra i descrittori di due regioni  $I$  e  $J$ , possono essere utilizzati vari criteri. Abbiamo implementato le seguenti misure di similarità  $Sim(I, J)$ :

- Chi-square criterion:

$$Sim(I, J) = - \sum_i \frac{(f_I(i) - f_J(i))^2}{f_I(i) + f_J(i)} \quad (12)$$

- Histogram intersection:

$$Sim(I, J) = \sum_i \min(f_I(i), f_J(i)) \quad (13)$$

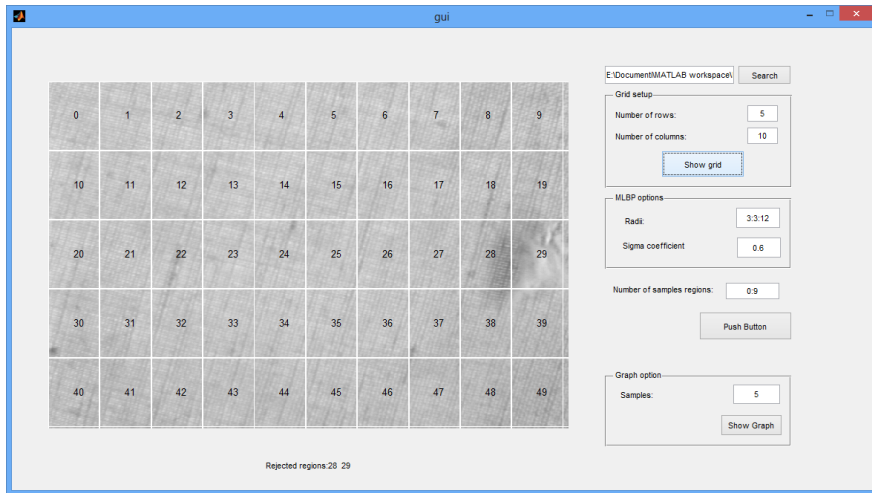


Figura 8: GUI pre LBP

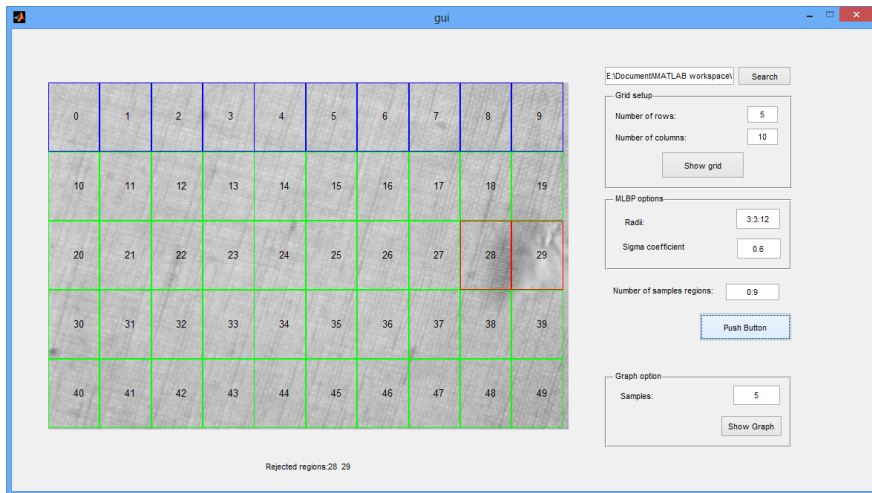


Figura 9: GUI post LBP

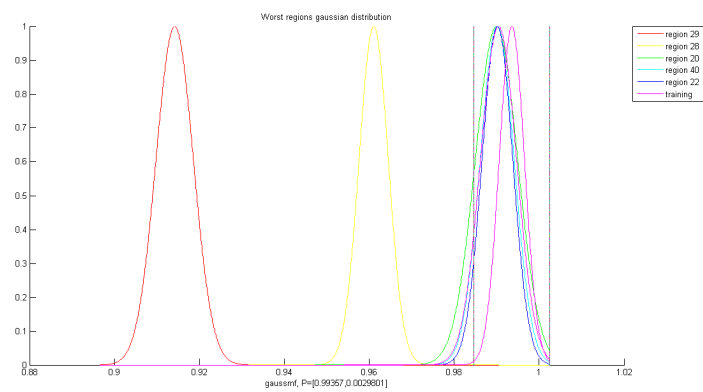


Figura 10: Caso peggiore

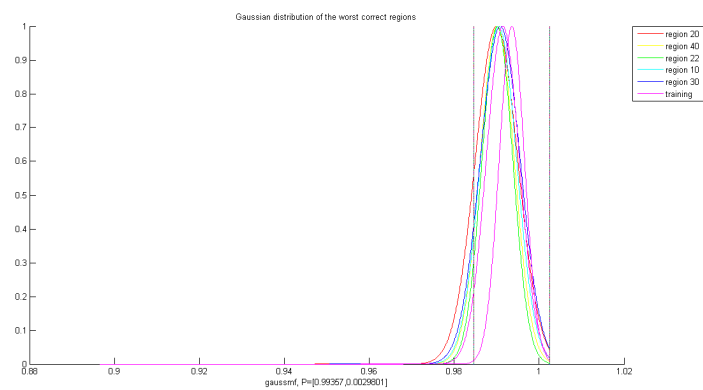


Figura 11: I peggiori tra i migliori

- Log-likelihood ratio (Kullack-Leibler divergence):

$$Sim(I, J) = - \sum_i f_I(i) \log(f_J(i)) \quad (14)$$

- Normalize Correlation:

$$Sim(I, J) = \frac{f_I f_J'}{||f_I|| ||f_J||} \quad (15)$$

- Normalize Cross-correlation:

$$Sim(I, J) = \frac{\sigma_{f_I f_J}}{\sigma_{f_I} \sigma_{f_J}} \quad (16)$$

## 4 Conclusioni

Gli autori

M. Bruni  
A. Rizzo

## **Acronimi**

**MLBP** Multi-scale Local Binary Pattern

**LBP** Local Binary Pattern



## Elenco delle figure

1	Istogramma immagine LBP Base . . . . .	5
2	Codice LBP . . . . .	5
3	Uniform LBP . . . . .	7
4	Istogramma immagine LBP Base . . . . .	8
5	Multiscale LBP . . . . .	8
6	Esempio di una kernel mask di dimensioni 3x3, dove $\omega_{i,j}$ rappresenta il peso applicato al valore del livello di grigio del corrispondente pixel nell'immagine . . . . .	9
7	GUI . . . . .	10
8	GUI pre LBP . . . . .	11
9	GUI post LBP . . . . .	11
10	Caso peggiore . . . . .	12
11	I peggiori tra i migliori . . . . .	12

## Elenco delle tabelle