Chapter 2
Local Binary Patterns for Still Images

The local binary pattern operator is an image operator which transforms an image into an array or image of integer labels describing small-scale appearance of the image. These labels or their statistics, most commonly the histogram, are then used for further image analysis. The most widely used versions of the operator are designed for monochrome still images but it has been extended also for color (multi channel) images as well as videos and volumetric data. This chapter covers the different versions of the actual LBP operator in spatial domain [42, 45, 53], while Chap. 3 deals with spatiotemporal LBP [88]. Parts II to IV of this book discuss how the labels are then used in different computer vision tasks.

2.1 Basic LBP

The basic local binary pattern operator, introduced by Ojala et al. [52], was based on the assumption that texture has locally two complementary aspects, a pattern and its strength. In that work, the LBP was proposed as a two-level version of the texture unit [74] to describe the local textural patterns.

The original version of the local binary pattern operator works in a 3×3 pixel block of an image. The pixels in this block are thresholded by its center pixel value, multiplied by powers of two and then summed to obtain a label for the center pixel. As the neighborhood consists of 8 pixels, a total of $2^8 = 256$ different labels can be obtained depending on the relative gray values of the center and the pixels in the neighborhood. See Fig. 1.1 for an illustration of the basic LBP operator. An example of an LBP image and histogram are shown in Fig. 2.1.

2.2 Derivation of the Generic LBP Operator

Several years after its original publication, the local binary pattern operator was presented in a more generic revised form by Ojala et al. [53]. In contrast to the basic

M. Pietikäinen et al., *Computer Vision Using Local Binary Patterns*,
Computational Imaging and Vision 40,
DOI 10.1007/978-0-85729-748-8_2, © Springer-Verlag London Limited 2011

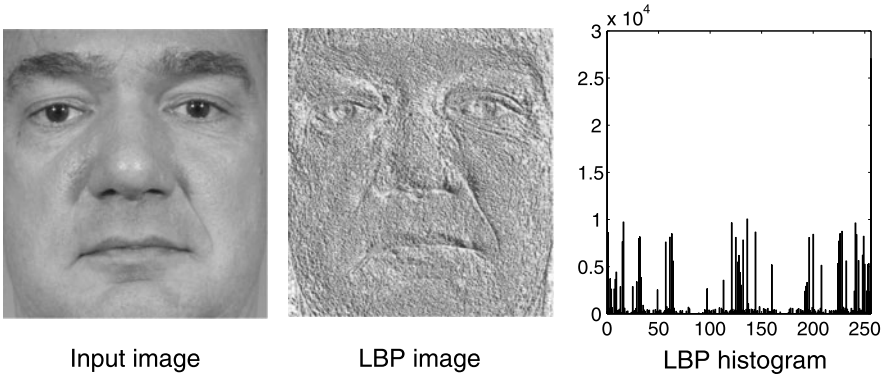

Fig. 2.1 Example of an input image, the corresponding LBP image and histogram

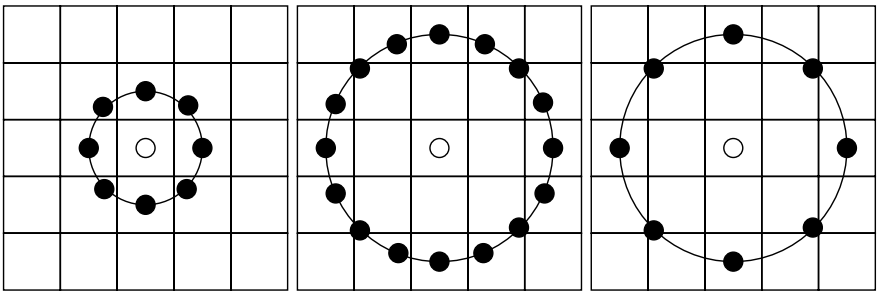

Fig. 2.2 The circular $(8, 1)$, $(16, 2)$ and $(8, 2)$ neighborhoods. The pixel values are bilinearly interpolated whenever the sampling point is not in the center of a pixel

LBP using 8 pixels in a 3×3 pixel block, this generic formulation of the operator puts no limitations to the size of the neighborhood or to the number of sampling points. The derivation of the generic LBP presented below follows that of [42, 45, 53].

Consider a monochrome image $I(x, y)$ and let g_c denote the gray level of an arbitrary pixel (x, y), i.e. $g_c = I(x, y)$.

Moreover, let g_p denote the gray value of a sampling point in an evenly spaced circular neighborhood of P sampling points and radius R around point (x, y):

$$g_p = I(x_p, y_p), \quad p = 0, \ldots, P-1 \quad \text{and} \tag{2.1}$$

$$x_p = x + R\cos(2\pi p/P), \tag{2.2}$$

$$y_p = y - R\sin(2\pi p/P). \tag{2.3}$$

See Fig. 2.2 for examples of local circular neighborhoods.

Assuming that the local texture of the image $I(x, y)$ is characterized by the joint distribution of gray values of $P + 1$ ($P > 0$) pixels:

$$T = t(g_c, g_0, g_1, \ldots, g_{P-1}). \tag{2.4}$$

Without loss of information, the center pixel value can be subtracted from the neighborhood:

$$T = t(g_c, g_0 - g_c, g_1 - g_c, \ldots, g_{P-1} - g_c). \tag{2.5}$$

In the next step the joint distribution is approximated by assuming the center pixel to be statistically independent of the differences, which allows for factorization of the distribution:

$$T \approx t(g_c)t(g_0 - g_c, g_1 - g_c, \ldots, g_{P-1} - g_c). \tag{2.6}$$

Now the first factor $t(g_c)$ is the intensity distribution over $I(x, y)$. From the point of view of analyzing local textural patterns, it contains no useful information. Instead the joint distribution of differences

$$t(g_0 - g_c, g_1 - g_c, \ldots, g_{P-1} - g_c) \tag{2.7}$$

can be used to model the local texture. However, reliable estimation of this multidimensional distribution from image data can be difficult. One solution to this problem, proposed by Ojala et al. in [54], is to apply vector quantization. They used learning vector quantization with a codebook of 384 codewords to reduce the dimensionality of the high dimensional feature space. The indices of the 384 codewords correspond to the 384 bins in the histogram. Thus, this powerful operator based on signed gray-level differences can be regarded as a texton operator, resembling some more recent methods based on image patch exemplars (e.g. [73]).

The learning vector quantization based approach still has certain unfortunate properties that make its use difficult. First, the differences $g_p - g_c$ are invariant to changes of the mean gray value of the image but not to other changes in gray levels. Second, in order to use it for texture classification the codebook must be trained similar to the other texton-based methods. In order to alleviate these challenges, only the signs of the differences are considered:

$$t(s(g_0 - g_c), s(g_1 - g_c), \ldots, s(g_{P-1} - g_c)), \tag{2.8}$$

where $s(z)$ is the thresholding (step) function

$$s(z) = \begin{cases} 1, & z \geq 0 \\ 0, & z < 0. \end{cases} \tag{2.9}$$

The generic local binary pattern operator is derived from this joint distribution. As in the case of basic LBP, it is obtained by summing the thresholded differences

weighted by powers of two. The $\mathrm{LBP}_{P,R}$ operator is defined as

$$\mathrm{LBP}_{P,R}(x_c, y_c) = \sum_{p=0}^{P-1} s(g_p - g_c)2^p. \tag{2.10}$$

In practice, Eq. 2.10 means that the signs of the differences in a neighborhood are interpreted as a P-bit binary number, resulting in 2^P distinct values for the LBP code. The local gray-scale distribution, i.e. texture, can thus be approximately described with a 2^P-bin discrete distribution of LBP codes:

$$T \approx t(\mathrm{LBP}_{P,R}(x_c, y_c)). \tag{2.11}$$

In calculating the $\mathrm{LBP}_{P,R}$ distribution (feature vector) for a given $N \times M$ image sample ($x_c \in \{0, \ldots, N-1\}$, $y_c \in \{0, \ldots, M-1\}$), the central part is only considered because a sufficiently large neighborhood cannot be used on the borders. The LBP code is calculated for each pixel in the cropped portion of the image, and the distribution of the codes is used as a feature vector, denoted by S:

$$\begin{aligned} &S = t(\mathrm{LBP}_{P,R}(x, y)), \\ &\quad x \in \{\lceil R \rceil, \ldots, N - 1 - \lceil R \rceil\}, y \in \{\lceil R \rceil, \ldots, M - 1 - \lceil R \rceil\}. \end{aligned} \tag{2.12}$$

The original LBP (Fig. 1.1) is very similar to $\mathrm{LBP}_{8,1}$, with two differences. First, the neighborhood in the general definition is indexed circularly, making it easier to derive rotation invariant texture descriptors. Second, the diagonal pixels in the 3×3 neighborhood are interpolated in $\mathrm{LBP}_{8,1}$.

2.3 Mappings of the LBP Labels: Uniform Patterns

In many texture analysis applications it is desirable to have features that are invariant or robust to rotations of the input image. As the $\mathrm{LBP}_{P,R}$ patterns are obtained by circularly sampling around the center pixel, rotation of the input image has two effects: each local neighborhood is rotated into other pixel location, and within each neighborhood, the sampling points on the circle surrounding the center point are rotated into a different orientation.

Another extension to the original operator uses so called *uniform patterns* [53]. For this, a uniformity measure of a pattern is used: U ("pattern") is the number of bitwise transitions from 0 to 1 or vice versa when the bit pattern is considered circular. A local binary pattern is called uniform if its uniformity measure is at most 2. For example, the patterns 00000000 (0 transitions), 01110000 (2 transitions) and 11001111 (2 transitions) are uniform whereas the patterns 11001001 (4 transitions) and 01010011 (6 transitions) are not. In uniform LBP mapping there is a separate output label for each uniform pattern and all the non-uniform patterns are assigned to a single label. Thus, the number of different output labels for mapping for patterns

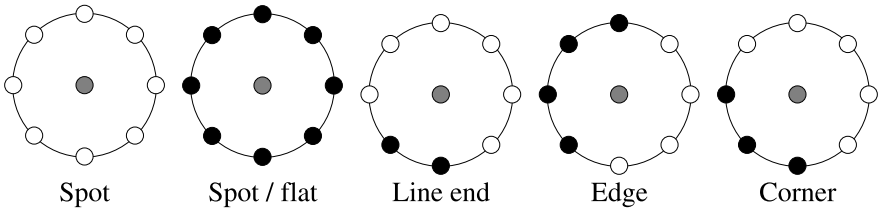

Fig. 2.3 Different texture primitives detected by the LBP

of P bits is $P(P-1)+3$. For instance, the uniform mapping produces 59 output labels for neighborhoods of 8 sampling points, and 243 labels for neighborhoods of 16 sampling points.

The reasons for omitting the non-uniform patterns are twofold. First, most of the local binary patterns in natural images are uniform. Ojala et al. noticed that in their experiments with texture images, uniform patterns account for a bit less than 90% of all patterns when using the $(8,1)$ neighborhood and for around 70% in the $(16,2)$ neighborhood. In experiments with facial images [4] it was found that 90.6% of the patterns in the $(8,1)$ neighborhood and 85.2% of the patterns in the $(8,2)$ neighborhood are uniform.

The second reason for considering uniform patterns is the statistical robustness. Using uniform patterns instead of all the possible patterns has produced better recognition results in many applications. On one hand, there are indications that uniform patterns themselves are more stable, i.e. less prone to noise and on the other hand, considering only uniform patterns makes the number of possible LBP labels significantly lower and reliable estimation of their distribution requires fewer samples.

The uniform patterns allows to see the LBP method as a unifying approach to the traditionally divergent statistical and structural models of texture analysis [45]. Each pixel is labeled with the code of the texture primitive that best matches the local neighborhood. Thus each LBP code can be regarded as a micro-texton. Local primitives detected by the LBP include spots, flat areas, edges, edge ends, curves and so on. Some examples are shown in Fig. 2.3 with the $\mathrm{LBP}_{8,R}$ operator. In the figure, ones are represented as black circles, and zeros are white.

The combination of the structural and statistical approaches stems from the fact that the distribution of micro-textons can be seen as statistical placement rules. The LBP distribution therefore has both of the properties of a structural analysis method: texture primitives and placement rules. On the other hand, the distribution is just a statistic of a non-linearly filtered image, clearly making the method a statistical one. For these reasons, the LBP distribution can be successfully used in recognizing a wide variety of different textures, to which statistical and structural methods have normally been applied separately.

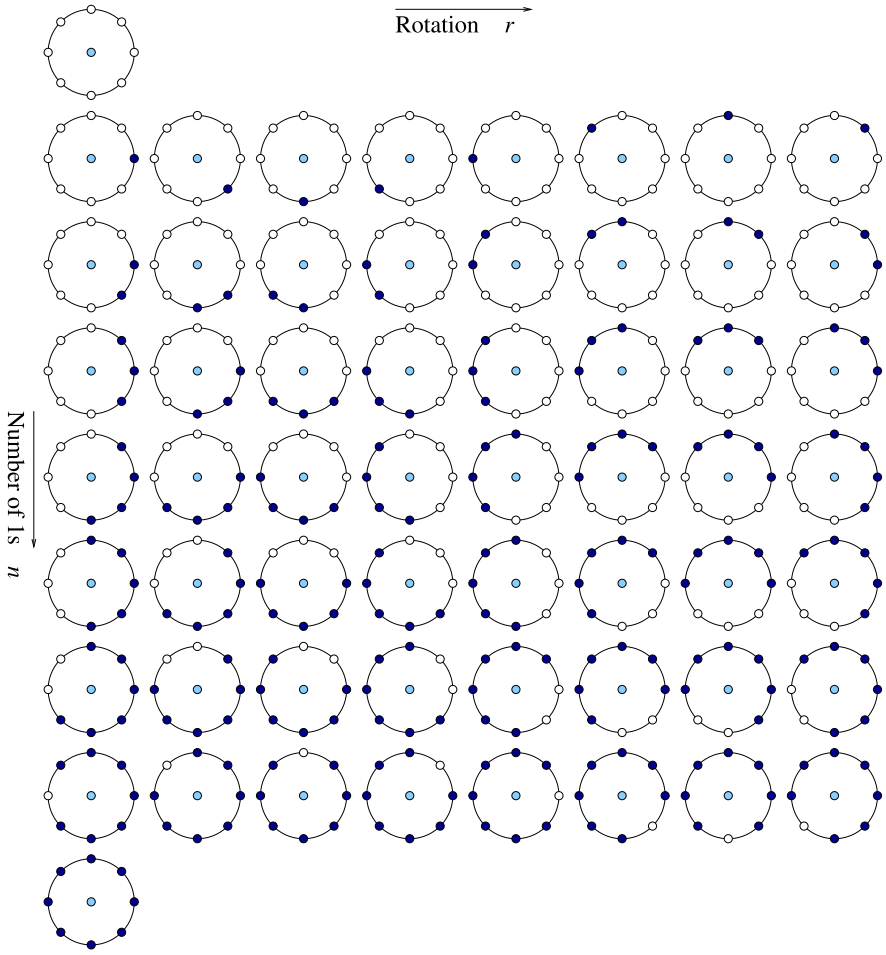

Fig. 2.4 The 58 different uniform patterns in $(8, R)$ neighborhood

2.4 Rotational Invariance

Let $U_P(n, r)$ denote a specific uniform LBP pattern. The pair (n, r) specifies a uniform pattern so that n is the number of 1-bits in the pattern (corresponds to row number in Fig. 2.4) and r is the rotation of the pattern (column number in Fig. 2.4) [6].

Now if the neighborhood has P sampling points, n gets values from 0 to $P + 1$, where $n = P + 1$ is the special label marking all the non-uniform patterns. Furthermore, when $1 \leq n \leq P - 1$, the rotation of the pattern is in the range $0 \leq r \leq P - 1$.

Let $I^{\alpha^\circ}(x, y)$ denote the rotation of image $I(x, y)$ by α degrees. Under this rotation, point (x, y) is rotated to location (x', y'). A circular sampling neighborhood on points $I(x, y)$ and $I^{\alpha^\circ}(x', y')$ also rotates by α°. See Fig. 2.5 [6].

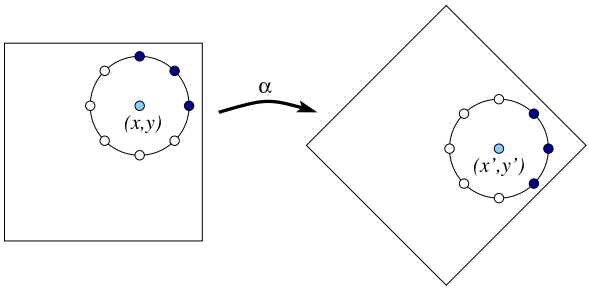

Fig. 2.5 Effect of image rotation on points in circular neighborhoods

If the rotations are limited to integer multiples of the angle between two sampling points, i.e. $\alpha = a\frac{360^\circ}{P}$, $a = 0, 1, \ldots, P-1$, this rotates the sampling neighborhood by exactly a discrete steps. Therefore the uniform pattern $U_P(n, r)$ at point (x, y) is replaced by uniform pattern $U_P(n, r + a \bmod P)$ at point (x', y') of the rotated image.

From this observation, the original rotation invariant LBPs introduced in [53] and newer, histogram transformation based rotation invariant features described in [6] can be derived. These are discussed in the following.

2.4.1 Rotation Invariant LBP

As observed in the preceding discussion, rotations of a textured input image cause the LBP patterns to translate into a different location and to rotate about their origin. Computing the histogram of LBP codes normalizes for translation, and normalization for rotation is achieved by rotation invariant mapping. In this mapping, each LBP binary code is circularly rotated into its minimum value

$$\mathrm{LBP}^{ri}_{P,R} = \min_i \mathrm{ROR}(\mathrm{LBP}_{P,R}, i), \tag{2.13}$$

where $\mathrm{ROR}(x, i)$ denotes the circular bitwise right rotation of bit sequence x by i steps. For instance, 8-bit LBP codes 10000010b, 00101000b, and 00000101b all map to the minimum code 00000101b.

Omitting sampling artifacts, the histogram of $\mathrm{LBP}^{ri}_{P,R}$ codes is invariant only to rotations of input image by angles $a\frac{360^\circ}{P}$, $a = 0, 1, \ldots, P-1$. However classification experiments show that this descriptor is very robust to in-plane rotations of images by any angle.

2.4.2 Rotation Invariance Using Histogram Transformations

The rotation invariant LBP descriptor discussed above defined a mapping for individual LBP codes so that the histogram of the mapped codes is rotation invariant. In this section, a family of histogram transformations is presented that can be used to compute rotation invariant features from a uniform LBP histogram.

Consider the uniform LBP histograms $h_I(U_P(n,r))$. The histogram value h_I at bin $U_P(n,r)$ is the number of occurrences of uniform pattern $U_P(n,r)$ in image I.

If the image I is rotated by $\alpha = a\frac{360^\circ}{P}$, this rotation of the input image causes a cyclic shift in the histogram along each of the rows,

$$h_{I^{\alpha^\circ}}(U_P(n,r+a)) = h_I(U_P(n,r)). \tag{2.14}$$

For example, in the case of 8 neighbor LBP, when the input image is rotated by 45°, the value from histogram bin $U_8(1,0) = 000000001b$ moves to bin $U_8(1,1) = 00000010b$, the value from bin $U_8(1,1)$ to bin $U_8(1,2)$, etc. Therefore, to achieve invariance to rotations of input image, features computed along the input histogram rows and are invariant to cyclic shifts can be used.

Discrete Fourier Transform is used to construct these features. Let $H(n,\cdot)$ be the DFT of nth row of the histogram $h_I(U_P(n,r))$, i.e.

$$H(n,u) = \sum_{r=0}^{P-1} h_I(U_P(n,r))e^{-i2\pi ur/P}. \tag{2.15}$$

In [6] it was shown that the Fourier magnitude spectrum

$$|H(n,u)| = \sqrt{H(n,u)\overline{H(n,u)}} \tag{2.16}$$

of the histogram rows results in features that are invariant to rotations of the input image.

Based on this property, an LBP-HF feature vector consisting of three LBP histogram values (all zeros, all ones, non-uniform) and Fourier magnitude spectrum values was defined. The feature vectors have the following form:

$$\begin{aligned} fv_{\text{LBP-HF}} = [&|H(1,0)|,\ldots,|H(1,P/2)|,\\ &\ldots,\\ &|H(P-1,0)|,\ldots,|H(P-1,P/2)|,\\ &h(U_P(0,0)),h(U_P(P,0)),h(U_P(P+1,0))]_{1\times((P-1)(P/2+1)+3)}. \end{aligned}$$

It should also be noted that the Fourier magnitude spectrum contains rotation-invariant uniform pattern features LBP^{riu2} as a subset, since

$$|H(n,0)| = \sum_{r=0}^{P-1} h_I(U_P(n,r)) = h_{\text{LBP}^{riu2}}(n). \tag{2.17}$$

An illustration of these features is in Fig. 2.6 [6].

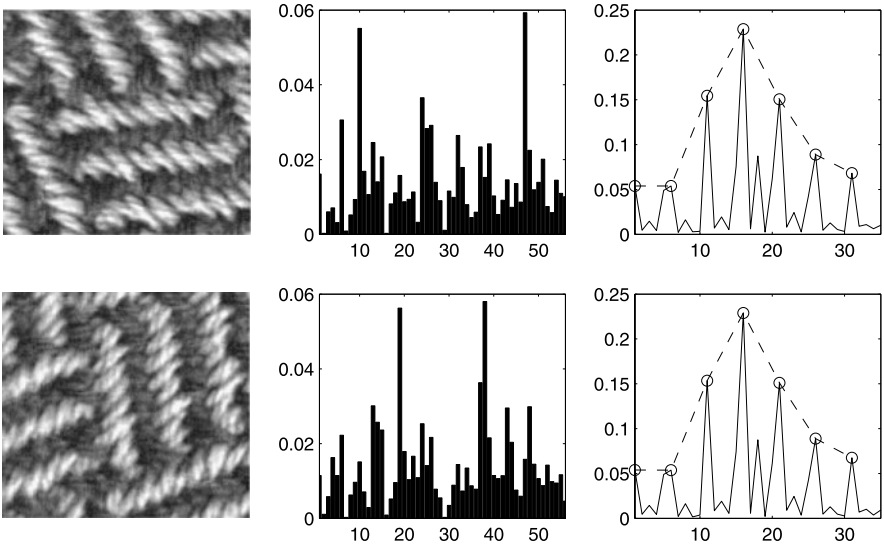

Fig. 2.6 *1st column*: Texture image at orientations 0° and 90°. *2nd column*: bins 1–56 of the corresponding LBP^{u2} histograms. *3rd column*: Rotation invariant features $|H(n,u)|$, $1 \leq n \leq 7$, $0 \leq u \leq 5$, (*solid line*) and LBP^{riu2} (*circles*, *dashed line*). Note that the LBP^{u2} histograms for the two images are markedly different, but the $|H(n,u)|$ features are nearly equal

2.5 Complementary Contrast Measure

Contrast is a property of texture usually regarded as a very important cue for human vision, but the LBP operator by itself totally ignores the magnitude of gray level differences. In many applications, for example in industrial visual inspection, illumination can be accurately controlled. In such cases, a purely gray-scale invariant texture operator may waste useful information, and adding gray-scale dependent information may enhance the accuracy of the method. Furthermore, in applications such as image segmentation, gradual changes in illumination may not require the use of a gray-scale invariant method [42, 51].

In a more general view, texture is distinguished not only by texture patterns but also the strength of the patterns. Texture can thus be regarded as a two-dimensional phenomenon characterized by two orthogonal properties: spatial structure (patterns) and contrast (the strength of the patterns). Pattern information is independent of the gray scale, whereas contrast is not. On the other hand, contrast is not affected by rotation, but patterns are, by default. These two measures supplement each other in a very useful way. The LBP operator was originally designed just for this purpose: to complement a gray-scale dependent measure of the "amount" of texture. In [52], the joint distribution of LBP codes and a local contrast measure (LBP/C, see Fig. 1.1) is used as a texture descriptor.

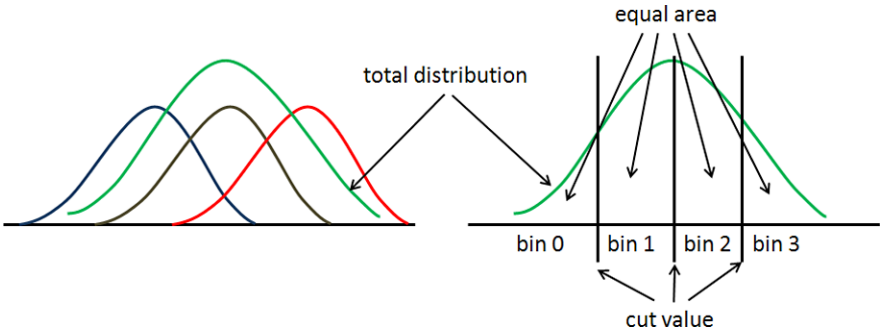

Fig. 2.7 Quantization of the feature space, when four bins are requested

Rotation invariant local contrast can be measured in a circularly symmetric neighbor set just like the LBP:

$$\mathrm{VAR}_{P,R} = \frac{1}{P}\sum_{p=0}^{P-1}(g_p - \mu)^2, \quad \text{where } \mu = \frac{1}{P}\sum_{p=0}^{P-1} g_p. \tag{2.18}$$

$\mathrm{VAR}_{P,R}$ is, by definition, invariant against shifts in the gray scale. Since contrast is measured locally, the measure can resist even intra-image illumination variation as long as the absolute gray value differences are not much affected. A rotation invariant description of texture in terms of texture patterns and their strength is obtained with the joint distribution of LBP and local variance, denoted as $\mathrm{LBP}^{riu2}_{P_1,R_1}/\mathrm{VAR}_{P_2,R_2}$. Typically, the neighborhood parameters are chosen so that $P_1 = P_2$ and $R_1 = R_2$, although nothing prevents one from choosing different values.

Variance measure has a continuous-valued output; hence, quantization of its feature space is needed. This can be done effectively by adding together feature distributions for every single model image in a total distribution, which is divided into B bins having an equal number of entries. Hence, the cut values of the bins of the histograms correspond to the $(100/B)$ percentile of the combined data. Deriving the cut values from the total distribution and allocating every bin the same amount of the combined data guarantees that the highest resolution of quantization is used where the number of entries is largest and vice versa. The number of bins used in the quantization of the feature space is of some importance as histograms with a too small number of bins fail to provide enough discriminative information about the distributions. On the other hand, since the distributions have a finite number of entries, a too large number of bins may lead to sparse and unstable histograms. As a rule of thumb, statistics literature often proposes that an average number of 10 entries per bin should be sufficient. In the experiments presented in this book, the value of B has been set so that this condition is satisfied. Figure 2.7 illustrates quantization of the feature space, when four bins are requested.

2.6 Non-parametric Classification Principle

In classification, the dissimilarity between a sample and a model LBP distribution is measured with a non-parametric statistical test. This approach has the advantage that no assumptions about the feature distributions need to be made. Originally, the statistical test chosen for this purpose was the cross-entropy principle [32, 52]. Later, Sokal and Rohlf [65] have called this measure the G statistic:

$$G(S, M) = 2 \sum_{b=1}^{B} S_b \log \frac{S_b}{M_b} = 2 \sum_{b=1}^{B} \left[S_b \log S_b - S_b \log M_b \right], \tag{2.19}$$

where S and M denote (discrete) sample and model distributions, respectively. S_b and M_b correspond to the probability of bin b in the sample and model distributions. B is the number of bins in the distributions [45].

For classification purposes, this measure can be simplified. First, the constant scaling factor 2 has no effect on the classification result. Furthermore, the term $\sum_{b=1}^{B} [S_b \log S_b]$ is constant for a given S, rendering it useless too. Thus the G statistic can be used in classification in a modified form:

$$L(S, M) = -\sum_{b=1}^{B} S_b \log M_b. \tag{2.20}$$

Model textures can be treated as random processes whose properties are captured by their LBP distributions. In a simple classification setting, each class is represented with a single model distribution M^i. Similarly, an unidentified sample texture can be described by the distribution S. L is a pseudo-metric that measures the likelihood that the sample S is from class i. The most likely class C of an unknown sample can thus be described by a simple nearest-neighbor rule:

$$C = \arg\min_i \; L(S, M^i). \tag{2.21}$$

Apart from a log-likelihood statistic, L can also be seen as a dissimilarity measure. Therefore, it can be used in conjunction with many classifiers, like the k-NN classifier, the self-organizing map (SOM) or the Support Vector Machine. The log-likelihood measure works well in many situations, but may be unstable with small sample sizes. The reason is that with small samples, the histogram is likely to contain many zeros, for which the logarithm is undefined. With small samples, the chi square distance usually works better [3]:

$$\chi^2(S, M) = \sum_{b=1}^{B} \frac{(S_b - M_b)^2}{S_b + M_b}. \tag{2.22}$$

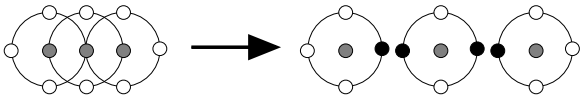

Fig. 2.8 Three adjacent $\mathrm{LBP}_{4,R}$ neighborhoods and an impossible combination of codes. A *black disk* means the gray level of a sample is lower than that of the center

Almost equivalent accuracy can be achieved with the histogram intersection [67], with a significantly smaller computational overhead:

$$H(S, M) = \sum_{b=1}^{B} \min(S_b, M_b). \tag{2.23}$$

2.7 Multiscale LBP

A significant limitation of the original LBP operator is its small spatial support area. Features calculated in a local 3×3 neighborhood cannot capture large-scale structures that may be the dominant features of some textures. However, adjacent LBP codes are not totally independent of each other. Figure 2.8 displays three adjacent four-bit LBP codes [42]. Assuming that the first bit in the leftmost code is zero, the third bit in the code to the right of it must be one. Similarly, the first bit in the code in the center and the third bit of the rightmost one must be either different or both equal to one. The right half of the figure shows an impossible combination of the codes. Each LBP code thus limits the set of possible codes adjacent to it, making the "effective area" of a single code actually slightly larger than 3×3 pixels. Nevertheless, the operator is not very robust against local changes in the texture, caused, for example, by varying viewpoints or illumination directions. An operator with a larger spatial support area is therefore often needed.

A straightforward way of enlarging the spatial support area is to combine the information provided by N LBP operators with varying P and R values. This way, each pixel in an image gets N different LBP codes. The most accurate information would be obtained by using the joint distribution of these codes. However, such a distribution would be overwhelmingly sparse with any reasonable image size. For example, the joint distribution of $\mathrm{LBP}_{8,1}$, $\mathrm{LBP}^{u2}_{16,3}$, and $\mathrm{LBP}^{u2}_{24,5}$ would contain $256 \times 243 \times 555 \approx 3.5 \times 10^7$ bins. Therefore, only the marginal distributions of the different operators are considered even though the statistical independence of the outputs of the different LBP operators at a pixel cannot be warranted. For example, a feature histogram obtained by concatenating histograms produced by rotation-invariant uniform pattern operators at three scales (1, 3 and 5) is denoted as: $\mathrm{LBP}^{riu2}_{8,1+16,3+24,5}$.

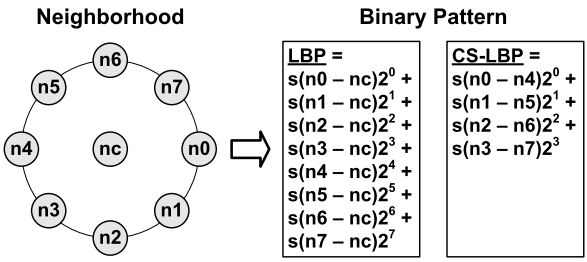

Fig. 2.9 LBP and CS-LBP features for a neighborhood of 8 pixels

The aggregate dissimilarity between a sample and a model can be calculated as a sum of the dissimilarities between the marginal distributions

$$L_N = \sum_{n=1}^{N} L(S^n, M^n), \tag{2.24}$$

where S^n and M^n correspond to the sample and model distributions extracted by the nth operator [53]. Of course, the chi square distance or histogram intersection can also be used instead of the log-likelihood measure.

Even though the LBP codes at different radii are not statistically independent in the typical case, using multi-resolution analysis often enhances the discriminative power of the resulting features. With most applications, this straightforward way of building a multi-scale LBP operator has resulted in very good accuracy.

2.8 Center-Symmetric LBP

Center-Symmetric Local Binary Patterns (CS-LBP) [23] were developed for interest region description. CS-LBP aims for smaller number of LBP labels to produce shorter histograms that are better suited to be used in region descriptors. Also, CS-LBP was designed to have higher stability in flat image regions.

In CS-LBP, pixel values are not compared to the center pixel but to the opposing pixel symmetrically with respect to the center pixel. See Fig. 2.9 for an illustration with eight neighbors [23].

Furthermore, to increase the operator's robustness in flat areas, the differences are thresholded at a typically non-zero threshold T. CS-LBP operator is thus defined as

$$\text{CS-LBP}_{R,P,T}(x, y) = \sum_{p=0}^{(P/2)-1} s(g_p - g_{p+(P/2)} - T)2^p, \quad s(z) = \begin{cases} 1 & z \geq 0 \\ 0 & \text{otherwise,} \end{cases} \tag{2.25}$$

where n_i and $n_{i+(N/2)}$ correspond to the gray values of center-symmetric pairs of pixels of N equally spaced pixels on a circle of radius R. It should be noticed that the CS-LBP is closely related to gradient operator, because like some gradient operators it considers gray level differences between pairs of opposite pixels in a neighborhood.

Based on the CS-LBP operator, Heikkilä et al. proposed a complete CS-LBP descriptor for interest regions. The steps of descriptor construction are summarized in the following. For more details, see [23] and Chap. 5.

1. Assuming that interest region with a known size and orientation has been detected, the region is normalized to a fixed size and orientation. In [23], 41×41 pixels was proposed as the size of the normalized region.
2. CS-LBP operator is applied to the normalized region.
3. The region is divided into cells. Authors suggest 3×3 or 4×4 Cartesian grids.
4. Histogram of the CS-LBP labels is constructed within each cell. To avoid boundary effects, bilinear interpolation is used to share the weight of each label between four nearest cells.
5. The histograms are concatenated to obtain the descriptor. The descriptor is then normalized to unit length, values above a pre-set threshold are clipped and finally the descriptor is re-normalized to unit length.

2.9 Other LBP Variants

The success of LBP methods in various computer vision problems and applications has inspired much new research on different variants. Due to its flexibility the LBP method can be easily modified to make it suitable for the needs of different types of problems. The basic LBP has also some problems that need to be addressed. Therefore, several extensions and modifications of LBP have been proposed with an aim to increase its robustness and discriminative power. In this section different variants are divided into such categories that describe their roles in feature extraction. Some of the variants could belong to more than one category, but in such cases only the most obvious category was chosen. A summary of the variants is presented in Table 2.1. The choice of a proper method for a given application depends on many factors, such as the discriminative power, computational efficiency, robustness to illumination and other variations, and the imaging system used. Therefore the LBP (and LBP with contrast) operators presented in the previous sections will usually provide a very good starting point when trying to find the optimal variant for a given application.

2.9.1 Preprocessing

In many applications, it is useful to preprocess the input image prior to LBP feature extraction. Especially multi-scale Gabor filtering and edge detection have been used for this purpose.

Gabor filtering has been widely used before LBP computation in face recognition. A motivation for this is that methods based on Gabor filtering and LBP provide complementary information: LBP captures small and fine details, while Gabor filters encode appearance information over a broader range of scales. For example,

Table 2.1 Summary of different LBP variants divided into such categories that describe their roles in feature extraction

Categories	LBP variants	Description	Ref.	Applications
Preprocessing	Local Gabor Binary Patterns (LGBP)	Gabor filtering before LBP computation	[85]	Face recognition
	Preprocessing chain	Gamma correction, DoG filtering, masking (optional), equalization of variation	[69]	Face recognition
	Local Edge Patterns (LEP)	Edge detection before LBP computation	[79]	Color texture retrieval
	Heat Kernel Local Binary Pattern (HKLBP)	Multiscale Heat kernel matrices are created before LBP computation	[37]	Face recognition
Neighborhood topology	Elliptical Binary Patterns (EBP)	Elliptical neighborhood is used	[38]	Face recognition
	Elongated Quinary Patterns (EQP)	Quinary encoding in elliptical neighborhood is used	[50]	Medical image analysis
	Local Line Binary Patterns (LLBP)	Lines in vertical and horizontal directions are used	[56]	Face recognition
	Three-Patch Local Binary Patterns (TPLBP)	Patch-based descriptor inspired by CS-LBP	[76]	Face analysis
	Four-Patch Local Binary Patterns (FPLBP)	Patch-based descriptor inspired by CS-LBP	[76]	Face analysis
Thresholding & encoding	Median Binary Patterns (MBP)	The median value within the neighborhood is used for thresholding	[19]	Texture classification
	Improved LBP (ILBP)	The mean of the local neighborhood is used for thresholding	[30]	Face detection
	Robust LBP	Differences thresholded at a non-zero threshold	[22]	Background subtraction
	Local Ternary Patterns (LTP)	Three values (1, 0 or -1) are used for encoding	[69]	Face recognition
	Elongated Quinary Patterns (EQP)	Five values $(-2, -1, 0, 1, 2)$ are used for encoding	[50]	Medical image analysis
	Elongated Ternary Patterns (ELTP)	Three values are used for encoding	[49]	Medical image analysis
	Soft/Fuzzy Local Binary Patterns	Thresholding is replaced by a fuzzy membership function	[1, 29]	Texture analysis
	Probabilistic LBP	Thresholding is replaced by a probabilistic function	[68]	Face verification
	Scale Invariant Local Ternary Pattern (SILTP)	Extension of LTP to handle illumination variations	[40]	Background subtraction

Table 2.1 (Continued)

Categories	LBP variants	Description	Ref.	Applications
	Transition coded LBP (tLBP)	Encoding relation between neighboring pixels	[71]	Car detection
	Direction coded LBP (dLBP)	Related to CS-LBP, but uses also center pixel for encoding	[71]	Gender classification
	Centralized binary patterns (CBP)	Also related to CS-LBP, using center pixel for encoding	[14]	Facial expressions
	Semantic Local Binary Patterns (S-LBP)	Adding semantic consistency to LBP	[48]	Human detection
	Fourier Local Binary Patterns (F-LBP)	Adding semantic consistency to LBP	[48]	Human detection
	Local Derivative Patterns (LDP)	Encoding high order derivative patterns	[81]	Face recognition
	Bayesian Local Binary Patterns (BLBP)	Labeling is modeled as a probability and optimization process	[20]	Texture retrieval
Multiscale analysis	Gaussian filtering	Multiscale low-pass filtering before feature extraction	[43]	Texture classification
	Cellular automata	Compactly encoding several LBP operators at different scales	[43]	Texture classification
	Multiscale Block LBP (MB-LBP)	Compare average pixel values within small blocks	[41]	Face recognition
	Pyramid-based multi-structure LBP	Apply LBP on different layers of image pyramid	[21, 72]	Texture analysis
	Sparse multiscale LBP	Exploit the discriminative capacity of multiscale features	[59]	Texture/face recognition
	Multiresolution uniform patterns	Multiscale sampling points ordered according to sampling angle	[31]	Gait recognition
Handling rotation	Adaptive LBP (ALBP)	Directional statistical information is incorporated	[18]	Texture classification
	LBP variance (LBPV)	Build rotation variant LBP histogram and then apply a global matching	[17]	Texture classification
	Monogenic-LBP (M-LBP)	Integrates LBP with two other rotation-invariant measures	[84]	Texture classification
Handling color	Opponent color LBP	Each color channel and pairs of color channels are used	[44]	Texture analysis
	Separate color and texture	Texture and color features are computed separately	[44]	Texture analysis

Table 2.1 (Continued)

Categories	LBP variants	Description	Ref.	Applications
	Multiscale color LBPs	LBP values computed from different color channels	[92]	Object classes recognition
	Color vector LBP	Color LBP images computed considering pixels as color vectors	[58]	Color-texture classification
	3D histograms of LBP	LBP values computed from LBP images of three channels	[11]	Image indexing
Complementary descriptors	Completed LBP (CLBP)	Use local difference sign-magnitude transform	[16]	Texture classification
	Extended LBP	Encode both local gray level differences and ordinary LBP patterns	[25, 27]	Face recognition
	LBP and Gabor	Use of LBP and Gabor methods jointly	[70, 85]	Face recognition
	HOG-LBP	Combining LBP with the Histogram of Oriented Gradients operator	[75]	Human detection
	HOG-LBP-LTP	Combining LBP, HOG and LTP operators	[28]	Visual object detection
	CS-LBP descriptor	Combining the strengths of SIFT and LBP	[23]	Interest region description
	Haar-LBP	Combining ideas from Haar and LBP features	[60, 77]	Face detection
Feature selection and Learning	Dominant Local Binary Patterns (DLBP)	Make use of the most frequently occurred patterns of LBP	[39]	Texture classification
	Extended LBP	Analyzes the structure & occurrence probability of nonuniform patterns	[91]	Texture analysis
	LBP with Hamming distance	Nonuniform patterns incorporated into uniform patterns	[78]	Face recognition
	FSC-LBP	Fisher separation criterion is used to learn the most prominent pattern types	[15]	Texture classification
	Beam Search	Use beam search for selecting a subset of LBP patterns	[46]	Texture analysis
	Fast correlation-based filtering	Use of fast correlation-based filtering to select LBP patterns	[64]	Facial expression analysis

Table 2.1 (Continued)

Categories	LBP variants	Description	Ref.	Applications
	Symmetry patterns	Select patterns that contain high number of ones or zeros	[33]	Face recognition
	Decision tree LBP	Use decision tree algorithms to learn discriminative LBP-like patterns	[47]	Face recognition
	Boosting LBP bins	AdaBoost is used for learning discriminative LBP histogram bins	[61]	Facial expression recognition
	Boosting LBP regions	AdaBoost is used for selecting local regions and LBP settings	[82]	Face recognition
	Linear Discriminant Analysis	Use of LDA to project Multi-Scale LBP features to compact space	[7, 8]	Face recognition
	Ensemble of piecewise FDA	Construct ensemble of piecewise FDA for building compact LGBP feature space	[63]	Face recognition
	Kernel Discriminative Common vectors	Applied to Gabor wavelets and LBP features after PCA projection	[70]	Face recognition
	AdaBoost-LDA	Select most discriminative LBP features from a large pool of multiscale features	[24]	Face recognition
	Dual-Space LDA	Select discriminative LBP features	[86]	Face recognition
	Laplacian PCA	Select discriminative LBP features	[87]	Face recognition
	Partial Least Squares	PLS dimensionality reduction for selecting discriminative features	[28]	Visual object detector
	Locality Preserving Projections	Applied Locality Preserving Projections (LPP) on LBP features	[62]	Facial expression analysis
Other methods inspired by LBP	Weber Law Descriptor (WLD)	Codifies differential excitation and orientation components	[10]	Texture analysis
	Local Phase Quantization (LPQ)	Quantizing the Fourier transform phase in local neighborhoods	[55]	Texture classification
	GMM-based density estimator	Avoids the quantization errors of LBP	[34]	Texture analysis

Zhang et al. [85] proposed the extraction of LBP features from images obtained by filtering a facial image with 40 Gabor filters of different scales and orientations. The extracted features are called Local Gabor Binary Patterns (LGBP). Due to its high performance, the LGBP operator has been used as a reference method, together with the basic LBP method, in many recent face recognition studies. A downside of the method is the high dimensionality of the LGBP representation.

Tan and Triggs [69] developed a very effective preprocessing chain for compensating illumination variations in face images. It is composed of gamma correction, difference of Gaussian (DoG) filtering, masking (optional) and equalization of variation. This approach has been very successful in LBP-based face recognition under varying illumination conditions (see Chap. 10). When using it for the original LBP, the last step (i.e. equalization of variations) can be omitted due to LBP's invariance to monotonic gray scale changes.

In some studies edge detection has been used prior to LBP computation to enhance the gradient information. Yao and Chen [79] proposed local edge patterns (LEP) to be used with color features for color texture retrieval. In LEP, the Sobel edge detection and thresholding are used to find strong edges, and then LBP-like computation is used to derive the LEP patterns. In their method for shape localization Huang el al. [26] proposed an approach in which gradient magnitude images and original images are used to describe the local appearance pattern of each facial keypoint. A derivative-based LBP is used by applying LBP computation to the gradient magnitude image obtained by a Sobel operator. The Sobel-LBP later proposed by Zhao et al. [90] uses the same idea for facial image representation. First the Sobel edge detector is used and the LBPs are computed from the gradient magnitude images. They also applied Sobel-LBP on both the real and imaginary features of the Gabor filtered images.

Li et al. [37] proposed an approach based on capturing the intrinsic structural information of face appearances with multi-scale heat kernel matrices. Heat kernels perform well in characterizing the topological structural information of face appearance. Histograms of local binary patterns computed for non-overlapping blocks are then used for face description.

Also other types of preprocessing have been applied prior to LBP feature extraction. For example, computing LBPs from curvelet transformed images provided very promising performance in medical image analysis problems [35].

2.9.2 Neighborhood Topology

One important factor which makes the LBP approach so flexible to different types of problems is that the topology of the neighborhood from which the LBP features are computed can be different, depending on the needs of the given application.

The extraction of LBP features is usually done in a circular or square neighborhood. A circular neighborhood is important especially for rotation-invariant operators. However, in some applications, such as face recognition, rotation invariance

is not required, but anisotropic information may be important. To exploit this, Liao and Chung used an elliptical neighborhood definition, calling their LBP variant an elliptical binary pattern (EBP). EBP, and EBP combined with a local gradient (contrast) measure, provided improved results in face recognition experiments compared to the ordinary LBP [38]. Nanni et al. investigated the use of different neighborhood topologies (circle, ellipse, parabola, hyperbola and Archimedean spiral) and encodings in their research on LBP variants for medical image analysis [50]. An operator using quinary encoding in an elliptic neighborhood (EQP) provided the best performance.

Petpon and Srisuk proposed to use lines in vertical and horizontal directions for LBP computations [56]. The Local Line Binary Pattern (LLBP) method then computes magnitudes by calculating the square root from the sum of the squared responses in these orthogonal directions.

Wolf et al. [76] considered different ways of using bit strings to encode the similarities between patches of pixels, which could capture complementary information to pixel-based descriptors. They proposed a Three-Patch LBP (TPLBP) and Four-Patch-LBP (FPLBP), which have borrowed some ideas from the Center-Symmetric LBP (CS-LBP) described earlier. For each pixel in TPLBP, for example, a $w \times w$ patch centered at the pixel and S additional patches distributed uniformly in a ring of radius r around it are considered. Then, the values for pairs of patches located on the circle at a specified distance apart are compared with those of the central patch. The value of a single bit is set according to which of the two patches is more similar to the central patch. The code produced will have S bits per pixel. In FPLBP, two rings centered on the pixel were used instead of one ring in TPLBP.

2.9.3 Thresholding and Encoding

Instead of using the value of the center pixel for thresholding in the local neighborhood, other techniques have also been considered. Hafiane et al. proposed Median Binary Pattern (MBP) operator by thresholding the local pixel values, including the center pixel, against the median (MBP) within the neighborhood [19]. The so-called Improved LBP, on the other hand, compares the values of the neighboring pixels against the mean gray level of the local neighborhood [13, 30]. A negative side is that the histograms for the methods using median or mean values for thresholding have 512 bins instead of 256 bins of the basic LBP. In fact, the use of the mean value of the local neighborhood was also considered, but not reported, when developing the original LBP in late 1992.

A drawback of the LBP method, as well as of all local descriptors that apply vector quantization, is that they are not robust in the sense that a small change in the input image would always cause a small change in the output. LBP may not work properly for noisy images or on flat image areas of constant gray level. This is due to the thresholding scheme of the operator.

In order to make the LBP more robust against these negligible changes in pixel values, the thresholding scheme of the operator was modified in [22] by replacing

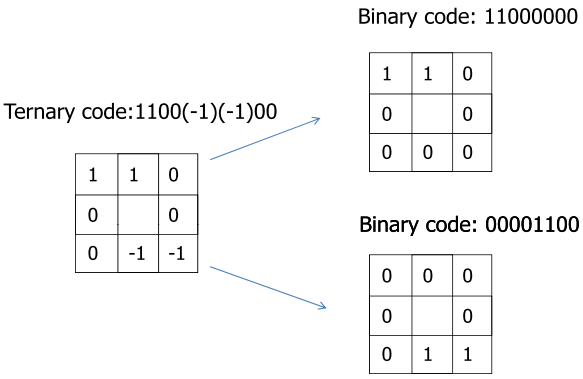

Fig. 2.10 Local ternary pattern operator

the term $s(g_p - g_c)$ in Eq. 2.10 with the term $s(g_p - g_c + a)$. The bigger the value of $|a|$ is, the bigger changes in pixel values are allowed without affecting the thresholding results. In order to retain the discriminative power of the LBP operator, a relatively small value should be used. In the experiments a was given a value of 3. An advantage of this robust LBP compared to the three-valued LBPs described below is that the feature vector length remains the same as in the ordinary LBP. A similar thresholding approach was also adopted to improve the robustness of CS-LBP as described in Sect. 2.8.

Tan and Triggs proposed a three-level operator called local ternary patterns (LTP) e.g. to deal with problems on near constant image areas [69]. In ternary encoding the difference between the center pixel and a neighboring pixel is encoded by three values (1, 0 or -1) according to a threshold T. The ternary pattern is divided into two binary patterns taking into account its positive and negative components. The histograms from these components computed over a region are then concatenated. Figure 2.10 depicts an example of splitting a ternary code into positive and negative codes. Note that LTP resembles the texture spectrum operator [74], which also used a three-valued output instead of two.

Nanni et al. [50] studied the effects of different encodings of the local gray-scale differences, using binary (B), ternary (T) and a quinary (Q) encodings. In binary coding, the difference between a neighboring pixel and the center pixel is encoded by two values (0 and 1) like in LBP, in ternary encoding it is encoded by three values as in LTP, and in quinary encoding by five values $(-2, -1, 0, 1, 2)$ according to two thresholds (T1 and T2). A quinary code can be split into four binary LBP codes. In experiments with three different types of medical images the elongated quinary patterns (EQP) using elliptical neighborhoods provided the best overall performance. In their another study dealing with classification of pain states from facial expressions, the best results were obtained with elongated ternary patterns (ELTP) [49].

A soft three-valued LBP using fuzzy membership functions was proposed to improve the robustness in [1]. In soft LBP, one pixel typically contributes to more than one bin in the histogram. Fuzzy local binary patterns were also proposed by Iakovdis et al., with an application in ultrasound texture characterization [29]. A probabilistic

LBP (PLBP) was developed by Tan et al. [68], allowing to encode the magnitude of the difference between a neighboring pixel and the center pixel. A disadvantage of the fuzzy and probabilistic methods is their increased computational cost.

Liao et al. [40] noticed that adding a small offset value (T) for comparison in LTP is not invariant under scaling of intensity values. The intensity scale invariant property of a local comparison operator is very important for example in background modeling, because illumination variations, either global or local, often cause sudden changes of gray scale intensities of neighboring pixels simultaneously, which would approximately be a scale transform with a constant factor. Therefore, a Scale Invariant Local Ternary Pattern (SILTP) operator was developed for dealing with the gray scale intensity changes in complex background. Given any pixel location (x_c, y_c), SILTP encodes it as

$$\text{SILTP}^{\tau}_{N,R}(x_c, y_c) = \bigoplus_{k=0}^{N-1} s_\tau(I_c, I_k), \tag{2.26}$$

where I_c is the gray intensity value of the center pixel, I_k are that of its N neighborhood pixels equally spaced on a circle of radius R, $\bigoplus$ denotes concatenation operator of binary strings, τ is a scale factor indicating the comparing range, and s_τ is a piecewise function defined as

$$s_\tau(I_c, I_k) = \begin{cases} 01, & \text{if } I_k > (1+\tau)I_c, \\ 10, & \text{if } I_k < (1-\tau)I_c, \\ 00, & \text{otherwise.} \end{cases} \tag{2.27}$$

Since each comparison can result in one of three values, SILTP encodes it with two bits (with "11" undefined). The scale invariance of SILTP operator can be easily verified. The advantage of SILTP operator is in three fold. First, it is computationally efficient, which causes only one more comparison than LBP for each neighbor. Second, by introducing a tolerative range like LTP, the SILTP operator is robust to local image noise within a range. Especially in the shadowed area, the region is darker and contains more noise, in which SILTP is tolerable while local comparison results of LBP would be affected more. Finally, the scale invariance property makes SILTP robust to illumination changes. Assuming linear camera response, the SILTP feature is invariant if the illumination is suddenly changed from darker to brighter or vice versa. Besides, SILTP is robust when a soft shadow covers a background region, because the soft cast shadow reserves the background texture information but tends to be darker than the local background region with a scale factor.

A downside of the methods mentioned above using one or two thresholds is that the methods are not strictly invariant to local monotonic gray level changes as the original LBP. The feature vector lengths of these operators are also longer.

Trefny and Matas [71] proposed two new encoding schemes, which are complementary to the standard LBPs and also invariant to monotonic intensity transformations. The binary value transition coded LBP (tLBP) is composed of neighbor pixel comparisons in clockwise direction for all pixels except the central, encoding relation between neighboring pixels. Direction coded LBP (dLBP) is related

to CS-LBP operator, but uses also center pixel information for encoding. Intensity variation along each of the four basic directions is coded into two bits. The first bit encodes whether the center pixel is an extrema and the second bit encodes whether difference of border pixels compared to the center pixel grows or falls. Experiments with face detection, car detection and gender recognition problems showed the efficiency of their approach. Another operator related to CS-LBP is centralized binary pattern (CBP) proposed by Fu and Wei [14] for facial expression recognition. CBP considers the contribution of the center pixel by comparing its value to the average of all nine pixels in the neighborhood, and encodes this bit with the largest weight.

Mu et al. developed LBP variants with an application in human detection in personal album [48]. They found that the original LBP does not suit so well for this problem due to its relatively high complexity and lack of semantic consistency. Therefore they proposed two variants of LBP: Semantic-LBP (S-LBP) and Fourier-LBP (F-LBP). First, a binarization of a pixel neighborhood is done on a color space like CIE-LAB. In S-LBP, several continuous "1" bits on the sampling circle form an arch, which can be represented with its principal direction and arch length. Non-uniform ones (with more than one arches) are abandoned. A two-dimensional histogram descriptor (arch angle vs. arch length) can be obtained for a given image region. In F-LBP, real valued color distance between the k-th samples and central pixel are computed and transformed into frequency domain. Low-frequency coefficients are then used to capture salient local structures around current pixel.

Inspired by LBP, higher order local derivative patterns (LDP) were proposed by Zhang et al., with applications in face recognition [81]. The basic LBP represents the first-order circular derivative pattern of images, a micropattern generated by the concatenation of the binary gradient directions as was shown in [2]. The higher order derivative patterns extracted by LDP will provide more detailed information, but may also be more sensitive to noise than in LBP.

Aiming at reducing the sensitivity of the image descriptor to illumination changes, a Bayesian LBP (BLBP) was developed by He et al. [20]. This operator is formulated in a Filtering, Labeling and Statistic (FLS) framework for texture descriptors. In the framework, the local labeling procedure, which is a part of many popular descriptors such as LBP and SIFT, can be modeled as a probability and optimization process. This enables the use of more reliable prior and likelihood information, and reduces the sensitivity to noise. The BLBP operator pursues a label image, when given the filtered vector image, by maximizing the joint probability of two images.

2.9.4 Multiscale Analysis

From a signal processing point of view, the sparse sampling used by multiscale LBP operators may not result in an adequate representation of the signal, resulting in aliasing effects [43]. Due to this some low-pass filtering would be needed to make the operator more robust. From the statistical point of view, however, even sparse

sampling is acceptable provided that the number of samples is large enough. The sparse sampling is commonly used for example with the methods based on gray scale difference or co-occurrence statistics. Mäenpää and Pietikäinen proposed two alternative ways to multiscale analysis. In the first method Gaussian low-pass filters are used in collecting texture information from an larger area than the original single pixel. The filters and sampling position were designed to cope the neighborhood as well as possible while minimizing the redundant information. With this approach, the radii of the LBP operators used in the multiresolution version grow exponentially [43]. They also proposed another way of encoding arbitrarily large neighborhoods with cellular automata. It was used in compactly encoding even 12-scale LBP operators. A feature vector containing marginal distributions of LBP codes and cellular automation rules was used as a texture descriptor. In experiments, however, no clear improvement was obtained compared to the basic multi-scale approach.

Another extension of multiscale LBP operator is the multiscale block local binary pattern (MB-LBP) [41] which has gained popularity especially in facial image analysis. The key idea of MB-LBP is to compare average pixel values within small blocks instead of comparing pixel values. The operator always considers 8 neighbors, producing labels from 0 to 255. For instance, if the block size is 3×3 pixels, the corresponding MB-LBP operator compares the average gray value of the center block to the average values of the 8 neighboring blocks of the same size, thus the effective area of the operator is 9×9 pixels. Instead of the fixed uniform pattern mapping, MB-LBP has been proposed to be used with a mapping that is dynamically learned from a training data. In this mapping, the N most often occurring MB-LBP patterns receive labels $0, \ldots, N-1$, and all the remaining patterns share a single label. The number of labels, and consequently the length of the MB-LBP histogram is a parameter the user can set.

A straightforward way for multiscale analysis is to utilize a pyramid of the input image computed at different resolutions, and then concatenate LBP distributions computed from different levels of the pyramid. In their research on contextual analysis of textured scene images Turtinen and Pietikäinen [72] combined this kind of idea with the original multiscale LBP approach: image patches at three different scales were resized to the same size and then LBP features were computed using LBPs with three different radii. Figure 2.11 illustrates the approach.

He et al. [21] developed a pyramid-based multistructure LBP for texture classification. It is obtained by executing the LBP on different layers of image pyramid, allowing to extract both micro and macro structures from textures. Five templates are used for creating the pyramid. The first one is a 2D Gaussian function used to smooth the image. Other four anisotropic filters are used to create anisotropic subimages of the pyramid in four directions. Good results are reported for the Outex textures, but the processing time is much higher than in the original LBP.

Raja and Gong proposed sparse multiscale local binary patterns to better exploit the discriminative capacity of multiscale features available [59]. A pairwise-coupled reformulation of LBP-type classification was used which involves selecting single-point features for each pair of classes to form compact, contextually-relevant multiscale predicates known as Multiscale Selected Local Binary Features (MSLBF).

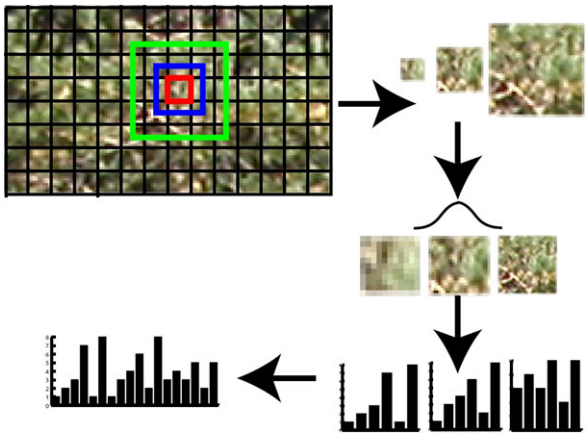

Fig. 2.11 Multiscale feature extraction

By the definition, uniform patterns are codes that consist of at most two bitwise transitions from 0 to 1 or vice versa when the binary string is considered circular. Therefore they can be considered as sectors on a sampling circle. When multiscale sampling points are ordered according to the sampling angle, they will also produce codes that satisfy the bit transition condition. Based on this observation, Kellokumpu et al. [31] proposed a new coding for multiresolution uniform patterns, obtaining improved results in gait recognition experiments.

2.9.5 Handling Rotation

LBPs have been used for rotation invariant texture recognition since late 1990s [57]. The most widely used version was proposed in [53] (Sect. 2.4.1), where the neighboring n binary bits around a pixel are clockwise rotated n times that a maximal number of the most significant bits is used to express this pixel.

Recently, in addition the method presented in Sect. 2.4.2, some other LBP variants for dealing with rotation have also been proposed.

Guo et al. developed an adaptive LBP (ALBP) [18] by incorporating the directional statistical information for rotation invariant texture classification. The directional statistical features, specifically the mean and standard deviation of the local absolute difference are extracted and used to improve the LBP classification efficiency. In addition, the least square estimation is used to adaptively minimize the local difference for more stable directional statistical features.

In [17], LBP variance (LBPV) was proposed as a rotation invariant descriptor. For LBPV there are three stages:

(1) putting the local contrast information into the one-dimensional LBP histogram; the variance $\mathrm{VAR}_{P,R}$ was used as an adaptive weight to adjust the contribution

of the LBP code in histogram calculation. LBPV histogram is computed as:

$$\mathrm{LBPV}_{P,R}(k) = \sum_{i=1}^{N}\sum_{j=1}^{M} w(\mathrm{LBP}_{P,R}(i,j),k), \quad k \in [0,K], \tag{2.28}$$

where

$$w(\mathrm{LBP}_{P,R}(i,j),k) = \begin{cases} \mathrm{VAR}_{P,R}(i,j), & \mathrm{LBP}_{P,R}(i,j) = k, \\ 0, & \text{otherwise;} \end{cases}$$

(2) learning the principal directions; the extracted LBPV features are used to estimate the principal orientations, and then the features are aligned to the principal orientations, and
(3) determining the non-dominant patterns and thus by reducing them, feature dimension reduction was achieved.

Zhang et al. [84] proposed Monogenic-LBP (M-LBP), which integrates the traditional rotation-invariant LBP operator with two other rotation-invariant measures: the local phase and local surface type computed by the first and second order Riesz transforms, respectively. The local phase corresponds to a qualitative measure of local structure (step, peak etc.), whereas the monogenic curvature tensor extracts local surface type information. In experiments with CUReT textures the method performed better than comparative methods (LBP, MR8, Joint), especially when the training set was small and not comprehensive.

2.9.6 Handling Color

LBP operator was originally developed for monochrome images. There are many possible ways for handling color with LBPs.

To describe color and texture jointly, opponent color LBP (OCLBP) [44] was defined. In opponent color LBP, the operator is used on each color channel independently, and then for pairs of color channels so that the center pixel is taken from one channel and the neighboring pixels from the other. Opposing pairs, such as R-G and G-R are highly redundant, so either of them can be used in the analysis. In total, six histograms (out of nine) are utilized (R, G, B, R-G, R-B, G-B), making the descriptor six times longer than the monochrome LBP histogram. Figure 2.12 illustrates the three situations in which the center pixel is taken from the red channel [42].

The OCLBP descriptor fares well in comparison to other color texture descriptors. It has been later used successfully e.g. for face recognition by Chan et al. [7]. However, the authors of [44] do not recommend joint color and texture description as in their experiments "all joint color texture descriptors and all methods of combining color and texture on a higher level are outperformed by either color or gray-scale texture alone". This approach of handling color and texture separately has been used in many recent studies.

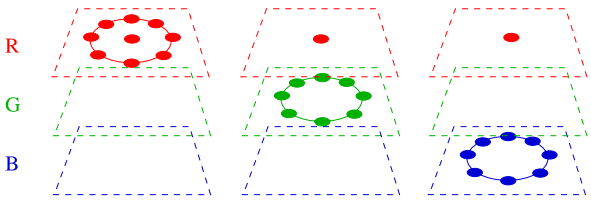

Fig. 2.12 Opponent color LBP for a *red center*. The three planes illustrate color channels

Instead of comparing the color components of pixels, Porebski et al. [58] considered color pixels represented by a vector when comparing neighboring pixels to the center pixel. Because there is no total order between vectors, they use a partial color order relation based on Euclidean distances for comparing their rank. As the result a single color LBP image is obtained instead of 6–9 provided by the OCLBP.

Another popular way is to apply the ordinary LBP to different color channels separately. Instead of the original R, G and B channels, other more discriminative and invariant color features derived from them can be used for LBP feature extraction as well. Along this line, Zhu et al. [92] proposed multiscale color LBPs for visual object classes recognition. Six operators were defined applying multiscale LBP on different types of channels and then concatenating the results together. From these the Hue-LBP (computed from the hue channel of the HSV color space), Opponent-LBP (computed over all three channels of the opponent color space) and nOpponent-LBP (computed over two channels of the normalized opponent space) provided the best performance on the well-known PASCAL Visual Object Classes Challenge 2007 benchmark (www.pascallin.ecs.soton.ac.uk/challenges/VOC/voc2007/).

Connah and Finlayson, on the other hand, used 3D histograms of LBP values computed from LBP images of three channels in their research on color constant image indexing [11]. They conclude that the good performance of the joint LBP histograms is a function of both their illumination invariance and their ability to encode additional information about the interaction between the color when using the three channels separately, whereas $10 \times 10 \times 10 = 1000$ bins are needed for a joint histogram.

2.9.7 Feature Selection and Learning

It has been shown by many studies that the dimensionality of the LBP distribution can be effectively decreased by reducing the number of neighboring pixels or by selecting a subset of bins available. In many cases a properly chosen subset of LBP patterns can perform better than the whole set of patterns.

Already the early studies on LBP indicated that in some problems considering only four neighbors of the center pixel (i.e. 16 bins) can provide almost as good results as eight neighbors (256 bins). Mäenpää et al. [46] showed that a major part of the discriminative power lies in a small properly selected subset of patterns. In addition to the uniform patterns (Sect. 2.3) they also considered a method based on beam search in which, starting from one, the size of the pattern set is iteratively

increased up to a specified dimension D, and the best B pattern sets produced so far are always considered. In their experiments the method based on feature selection by beam search performed better than the whole set of patterns or the uniform patterns when classifying tilted textures and using nontilted samples for training. Thus the feature selection procedure was able to find those patterns that were able to survive the tilting best. In [64], Smith and Windeatt used the fast correlation-based filtering (FCBF) algorithm [80] to select the most discriminative LBP patterns. FCBF operates by repeatedly choosing the feature that is most correlated with a given class (e.g. person identity in case of face recognition), excluding those features already chosen or rejected, and rejecting any features that are more correlated with it than with the class. As a measure of correlation, the information-theoretic concept of symmetric uncertainty is used. When applied to the LBP features, FCBF reduced their number from 107,000 down to 120.

Lahdenoja et al. [33] defined a discrimination concept of the uniform LBP patterns called symmetry to reduce the feature vector length for LBP-based face description. Patterns are assigned different levels of symmetry based on the number of ones or zeros they contain. By definition these symmetry levels are rotation invariant. The patterns with a high level of symmetry were shown to be the most discriminative in experiments.

Liao et al. [39] introduced dominant local binary patterns (DLBP) which make use of the most frequently occurred patterns of LBP to improve the recognition accuracy compared to the original uniform patterns. The method has also rotation invariant characteristics. Zhou et al. considered that the LBP operator based on uniform patterns discards some important texture information and is sensitive to noise [91]. They proposed an extended LBP operator, which classifies and combines the nonuniform local patterns based on analyzing their structure and occurrence probability. Yang and Wang [78] also found that the nonuniform patterns contain useful information, incorporating these patterns into uniform patterns by minimizing the Hamming distance between them.

Guo et al. proposed a learning framework for image descriptor design [15]. The Fisher separation criterion (FSC) is used to learn the most reliable and robust dominant pattern types considering intraclass similarity and inter-class distance. Image structures are thus be described by a new FSC-based learning (FBL) encoding method. The learning framework includes three stages: (1) The learning stage. Determine most reliable dominant types for each class. Then, all the learnt dominant types of each class are merged and form the global dominant types for the whole database; (2) Extract global dominant types learnt in stage (1) of the training set; (3) Extract the global dominant types learnt in stage (1) of the testing set. Finally, features obtained in stages (2) and (3) are served as inputs to the classifier. FBL-LBP outperformed many other methods, including DLBP, in the experiments on three texture databases.

From the observation that LBP is equivalent to the application of a fixed binary decision tree, Maturana et al. [47] proposed a new method for learning discriminative LBP-like patterns from training data using decision tree induction algorithms. For each local image region, a binary decision tree is constructed from training data,

thus obtaining an adaptive tree whose main branches are specially tuned to encode discriminative patterns in each region. Face recognition experiments on FERET and CAS-PEAL-R1 databases showed good performance compared to many traditional LBP-like approaches. Among the drawbacks of the proposed decision tree LBP (DT-LBP) is the high cost of constructing and storage of the decision trees especially when large pixel neighborhoods are used.

Boosting has become a very popular approach for feature selection. It has been widely adopted for LBP feature selection in various tasks e.g. 3D face recognition [36], face detection [83], gender classification [66] etc. AdaBoost is commonly used for selecting optimal LBP settings (such as the size and the location of local regions, the number of neighboring pixels etc.) or for selecting the most discriminative bins of an LBP histogram. For instance, Zhang et al. [82] used AdaBoost learning for selecting an optimal set for local regions and their weights for face recognition (see Chap. 10). Since then, many related approaches have been used at region level for LBP-based face analysis. Shan and Gritti [61], on the other hand, used AdaBoost for learning discriminative LBP histogram bins, with an application to facial expression recognition.

Another approach for deriving compact and discriminative LBP-based feature vectors consist of applying subspace methods for learning and projecting the LBP features from the original high-dimensional space into a lower dimensional space. For instance, Chan et al. used Linear Discriminant Analysis (LDA) to project high-dimensional Multi-Scale LBP features into a discriminant space [7, 8], yielding very promising results. To deal with the small sample size problem of LDA, Shan et al. [63] constructed ensemble of piecewise Fisher Discriminant Analysis (EPFDA) classifiers, each of which is designed based on one segment of the high-dimensional histogram of local Gabor binary pattern (LGBP) features. Their approach was shown to be more effective than applying LDA to high-dimensional holistic feature vectors.

Tan and Triggs [70] combined Gabor wavelets and LBP features and projected them to PCA space. Then, the Kernel Discriminative Common Vectors (KDCV) is applied to extract discriminant nonlinear compact features for face recognition. In [24], an AdaBoost-LDA learning algorithm was proposed to select the most discriminative LBP features from a large pool of multiscale features generated by shifting and scaling a subwindow over the image. Dual-Space LDA was also adopted to select discriminative LBP features in [86]. In another work, the authors applied Laplacian PCA (LPCA) for LBP feature selection and pointed out the superiority of LPCA over PCA and KPCA for feature selection [87]. In [28], the authors exploited the complementarity of three sets of features namely HOG features, local binary patterns (LBP) and local ternary patterns (LTP), and adopted Partial Least Squares (PLS) dimensionality reduction for selecting the most discriminative features, yielding fast and efficient visual object detector. In [62], Locality Preserving Projections (LPP) was applied on LBP features for embedding image sequences of facial expression from the high dimensional appearance feature space into a low dimensional manifold.

2.9.8 Complementary Descriptors

A current trend in the development of new effective local image and video descriptors is to combine the strengths of complementary descriptors. From the beginning the LBP operator was designed as a complementary measure of local image contrast. In many recent studies proposing new texture descriptors the role of the LBP contrast has not been considered when comparing LBP to the new descriptor. The use of LBP (or its simple robust version using a non-zero threshold [22], Sect. 2.9.3), can still be the method of choice for many applications, and should be considered when selecting a texture operator to be used. An interesting alternative for putting the local contrast into the one-dimensional LBP histogram was proposed by Guo et al. [17] (see Sect. 2.9.5).

In [16], a completed modeling of the LBP operator was proposed and an associated completed LBP (CLBP) scheme was developed for texture classification. The image local differences are decomposed into two complementary components: the signs and the magnitudes and two operators, CLBP-Sign (CLBP_S, also the original LBP) and CLBP-Magnitude (CLBP_M) were proposed to code them. As well, the center pixels represent the image gray level and they are converted into a binary code (CLBP_C) by global thresholding. The CLBP_M and CLBP_C were combined with CLBP_S as complementary information to improve the texture classification. Earlier, a related Extended LBP (ELBP) was proposed which also encodes the local gray level differences in addition to the ordinary LBP computation [25, 27]. LBP codes are computed at multiple layers to encode the gray level differences between the center pixel and its neighbors.

Magnitude-LBP contains supplementary information to LBP. It was embedded to the histogram Fourier framework [89] and concatenated to LBPHF features as complementary descriptors to improve the description power for dealing with rotation variations.

In addition to applying LBP to Gabor-filtered face images, the use of LBP and Gabor methods jointly has provided excellent results in face recognition [70, 85]. The HOG-LBP, combining LBP with the Histogram of Oriented Gradients operator [12], has performed very well in human detection with partial occlusion handling [75]. Combining ideas from Haar and LBP features have given excellent results in accurate and illumination invariant face detection [60, 77]. A CS-LBP method for combining the strengths of SIFT and LBP in interest region description has also been developed [23] (Chap. 5).

2.9.9 Other Methods Inspired by LBP

LBP has also inspired the development of new effective local image descriptors.

The Weber Law Descriptor (WLD) is based on the fact that human perception of a pattern depends not only on the change of a stimulus (such as sound, lighting) but also on the original intensity of the stimulus [10]. Specifically, WLD consists of

two components: differential excitation and orientation. The differential excitation component is a function of the ratio between two terms: one is the relative intensity differences of a current pixel against its neighbors; the other is the intensity of the current pixel. The orientation component is the gradient orientation of the current pixel. For a given image, the two components are used to construct a concatenated WLD histogram. Experimental results on texture analysis and face detection problems have provided excellent performance. Joint use of LBP and the excitation component of WLD descriptor in dynamic texture segmentation was considered in [9]. This indicates that this component could be useful in replacing the contrast measure of LBP also in other problems.

The local phase quantization (LPQ) descriptor is based on quantizing the Fourier transform phase in local neighborhoods [55]. The phase can be shown to be a blur invariant property under certain commonly fulfilled conditions. In texture analysis, histograms of LPQ labels computed within local regions are used as a texture descriptor similarly to the LBP methodology. The LPQ descriptor has received recently wide interest in blur-invariant face recognition [5]. LPQ is insensitive to image blurring, and it has proven to be a very efficient descriptor in face recognition from blurred as well as sharp images.

Lategahn et al. [34] developed a framework which filters a texture region by a set of filters and subsequently estimates the joint probability density functions by Gaussian mixture models (GMM). Using the oriented difference filters of the LBP method [2], they showed that this method avoids the quantization errors of LBP, obtaining better results than with the basic LBP. Additional performance improvement of the GMM-based density estimator was obtained when the elementary LBP difference filters were replaced by wavelet frame transform filter banks.

References

1. Ahonen, T., Pietikäinen, M.: Soft histograms for local binary patterns. In: Proc. Finnish Signal Processing Symposium, p. 4 (2007)
2. Ahonen, T., Pietikäinen, M.: Image description using joint distribution of filter bank responses. Pattern Recognit. Lett. **30**(4), 368–376 (2009)
3. Ahonen, T., Hadid, A., Pietikäinen, M.: Face recognition with local binary patterns. In: European Conference on Computer Vision. Lecture Notes in Computer Science, vol. 3021, pp. 469–481. Springer, Berlin (2004)
4. Ahonen, T., Hadid, A., Pietikäinen, M.: Face description with local binary patterns: Application to face recognition. IEEE Trans. Pattern Anal. Mach. Intell. **28**(12), 2037–2041 (2006)
5. Ahonen, T., Rahtu, E., Ojansivu, V., Heikkilä, J.: Recognition of blurred faces using local phase quantization. In: Proc. International Conference on Pattern Recognition, pp. 1–4 (2008)
6. Ahonen, T., Matas, J., He, C., Pietikäinen, M.: Rotation invariant image description with local binary pattern histogram Fourier features. In: Scandinavian Conference on Image Analysis. Lecture Notes in Computer Science, vol. 5575, pp. 61–70. Springer, Berlin (2009)
7. Chan, C.H., Kittler, J.V., Messer, K.: Multispectral local binary pattern histogram for component-based color face verification. In: Proc. IEEE Conference on Biometrics: Theory, Applications and Systems, pp. 1–7 (2007)
8. Chan, C.-H., Kittler, J., Messer, K.: Multi-scale local binary pattern histograms for face recognition. In: Proc. International Conference on Biometrics, pp. 809–818 (2007)

9. Chen, J., Zhao, G., Pietikäinen, M.: An improved local descriptor and threshold learning for unsupervised dynamic texture segmentation. In: Proc. ICCV Workshop on Machine Learning for Vision-based Motion Analysis, pp. 460–467 (2009)
10. Chen, J., Shan, S., He, C., Zhao, G., Pietikäinen, M., Chen, X., Gao, W.: WLD: A robust local image descriptor. IEEE Trans. Pattern Anal. Mach. Intell. **32**(9), 1705–1720 (2010)
11. Connah, D., Finlayson, G.D.: Using local binary pattern operators for colour constant image indexing. In: Proc. European Conference on Color in Graphics, Imaging, and Vision, p. 5 (2006)
12. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Proc. IEEE Conference on Computer Vision and Pattern Recognition, vol. 2, pp. 886–893 (2005)
13. Fröba, B., Ernst, A.: Face detection with the modified census transform. In: Proc. International Conference on Face and Gesture Recognition, pp. 91–96 (2004)
14. Fu, X., Wei, W.: Centralized binary patterns embedded with image Euclidean distance for facial expression recognition. In: Proc. International Conference on Natural Computation, vol. 4, pp. 115–119 (2008)
15. Guo, Y., Zhao, G., Pietikäinen, M., Xu, Z.: Descriptor learning based on Fisher separation criterion for texture classification. In: Proc. Asian Conference on Computer Vision, pp. 1491–1500 (2010)
16. Guo, Z.H., Zhang, L., Zhang, D.: A completed modeling of local binary pattern operator for texture classification. IEEE Trans. Image Process. **19**(6), 1657–1663 (2010)
17. Guo, Z.H., Zhang, L., Zhang, D.: Rotation invariant texture classification using LBP variance (LBPV) with global matching. Pattern Recognit. **43**(3), 706–719 (2010)
18. Guo, Z.H., Zhang, L., Zhang, D., Zhang, S.: Rotation invariant texture classification using adaptive LBP with directional statistical features. In: Proc. International Conference on Image Processing, pp. 285–288 (2010)
19. Hafiane, A., Seetharaman, G., Zavidovique, B.: Median binary pattern for texture classification. In: Proc. International Conference on Image Analysis and Recognition, pp. 387–398 (2007)
20. He, C., Ahonen, T., Pietikäinen, M.: A Bayesian local binary pattern texture descriptor. In: Proc. International Conference on Pattern Recognition, pp. 1–4 (2008)
21. He, Y., Sang, N., Gao, C.: Pyramid-based multi-structure local binary pattern for texture classification. In: Proc. Asian Conference on Computer Vision, vol. 3, pp. 1435–1446 (2010)
22. Heikkilä, M., Pietikäinen, M.: A texture-based method for modeling the background and detecting moving objects. IEEE Trans. Pattern Anal. Mach. Intell. **28**(4), 657–662 (2006)
23. Heikkilä, M., Pietikäinen, M., Schmid, C.: Description of interest regions with local binary patterns. Pattern Recognit. **42**(3), 425–436 (2009)
24. Ho An, K., Jin Chung, M.: Cognitive face analysis system for future interactive TV. IEEE Trans. Consum. Electron. **55**(4), 2271–2279 (2009)
25. Huang, D., Wang, Y., Wang, Y.: A robust method for near infrared face recognition based on extended local binary pattern. In: Advances in Visual Computing. Lecture Notes in Computer Science, vol. 4842, pp. 437–446. Springer, Berlin (2007)
26. Huang, X., Li, S.Z., Wang, Y.: Shape localization based on statistical method using extended local binary pattern. In: Proc. International Conference on Image and Graphics, pp. 184–187 (2004)
27. Huang, Y., Wang, Y., Tan, T.: Combining statistics of geometrical and correlative features for 3d face recognition. In: Proc. British Machine Vision Conference, pp. 879–888 (2006)
28. Hussain, S., Triggs, B.: Feature sets and dimensionality reduction for visual object detection. In: Proc. British Machine Vision Conference, pp. 112.1–112.10 (2010)
29. Iakovidis, D.K., Keramidas, E., Maroulis, D.: Fuzzy local binary patterns for ultrasound texture characterization. In: Proc. International Conference on Image Analysis and Recognition, pp. 750–759 (2008)
30. Jin, H., Liu, Q., Lu, H., Tong, X.: Face detection using improved LBP under Bayesian framework. In: Proc. International Conference on Image and Graphics, pp. 306–309 (2004)

31. Kellokumpu, V., Zhao, G., Pietikäinen, M.: Dynamic texture based gait recognition. In: Advances in Biometrics. Lecture Notes in Computer Science, vol. 5558, pp. 1000–1009. Springer, Berlin (2009)
32. Kullback, S.: Information Theory and Statistics. Dover, New York (1968)
33. Lahdenoja, O., Laiho, M., Paasio, A.: Reducing the feature vector length in local binary pattern based face recognition. In: Proc. International Conference on Image Processing, vol. 2, pp. 914–917 (2005)
34. Lategahn, H., Gross, S., Stehle, T., Aach, T.: Texture classification by modeling joint distributions of local patterns with Gaussian mixtures. IEEE Trans. Image Process. **19**, 1548–1557 (2010)
35. Li, B., Meng, M.Q.-H.: Texture analysis for ulcer detection in capsule endoscopy images. Image Vis. Comput. **27**, 1336–1342 (2009)
36. Li, S.Z., Zhao, C., Zhu, X., Lei, Z.: Learning to fuse 3D+2D based face recognition at both feature and decision levels. In: Proc. IEEE International Workshop on Analysis and Modeling of Faces and Gestures, pp. 44–54 (2005)
37. Li, X., Hu, W., Zhang, Z., Wang, H.: Heat kernel based local binary pattern for face representation. IEEE Signal Process. Lett. **17**, 308–311 (2010)
38. Liao, S., Chung, A.C.S.: Face recognition by using enlongated local binary patterns with average maximum distance gradient magnitude. In: Computer Vision—ACCV 2007. Lecture Notes in Computer Science, vol. 4844, pp. 672–679. Springer, Berlin (2007)
39. Liao, S., Law, M., Chung, C.S.: Dominant local binary patterns for texture classification. IEEE Trans. Image Process. **18**, 1107–1118 (2009)
40. Liao, S., Zhao, G., Kellokumpu, V., Pietikäinen, M., Li, S.Z.: Modeling pixel process with scale invariant local patterns for background subtraction in complex scenes. In: Proc. IEEE Conference on Computer Vision and Pattern Recognition, p. 8 (2010)
41. Liao, S., Zhu, X., Lei, Z., Zhang, L., Li, S.Z.: Learning multi-scale block local binary patterns for face recognition. In: Proc. International Conference on Biometrics, pp. 828–837 (2007)
42. Mäenpää, T.: The local binary pattern approach to texture analysis—extensions and applications. PhD thesis, Acta Universitatis Ouluensis C 187, University of Oulu (2003)
43. Mäenpää, T., Pietikäinen, M.: Multi-scale binary patterns for texture analysis. In: Scandinavian Conference on Image Analysis. Lecture Notes in Computer Science, vol. 2749, pp. 885–892. Springer, Berlin (2003)
44. Mäenpää, T., Pietikäinen, M.: Classification with color and texture: Jointly or separately? Pattern Recognit. **37**, 1629–1640 (2004)
45. Mäenpää, T., Pietikäinen, M.: Texture analysis with local binary patterns. In: Chen, C.H., Wang, P.S.P. (eds.) Handbook of Pattern Recognition and Computer Vision, 3rd edn., pp. 197–216. World Scientific, Singapore (2005)
46. Mäenpää, T., Ojala, T., Pietikäinen, M., Soriano, M.: Robust texture classification by subsets of local binary patterns. In: Proc. 15th International Conference on Pattern Recognition, vol. 3, pp. 947–950 (2000)
47. Maturana, D., Soto, A., Mery, D.: Face recognition with decision tree-based local binary patterns. In: Proc. Asian Conference on Computer Vision, 2010
48. Mu, Y.D., Yan, S.C., Liu, Y., Huang, T., Zhou, B.F.: Discriminative local binary patterns for human detection in personal album. In: Proc. IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8 (2008)
49. Nanni, L., Brahnam, S., Lumini, A.: A local approach based on a local binary patterns variant texture descriptor. Expert Syst. Appl. **37**, 7888–7894 (2010)
50. Nanni, L., Lumini, A., Brahnam, S.: Local binary patterns variants as texture descriptors for medical image analysis. Artif. Intell. Med. **49**, 117–125 (2010)
51. Ojala, T., Pietikäinen, M.: Unsupervised texture segmentation using feature distributions. Pattern Recognit. **32**, 477–486 (1999)
52. Ojala, T., Pietikäinen, M., Harwood, D.: A comparative study of texture measures with classification based on feature distributions. Pattern Recognit. **29**(1), 51–59 (1996)

53. Ojala, T., Pietikäinen, M., Mäenpää, T.: Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Trans. Pattern Anal. Mach. Intell. **24**(7), 971–987 (2002)
54. Ojala, T., Valkealahti, K., Oja, E., Pietikäinen, M.: Texture discrimination with multidimensional distributions of signed gray-level differences. Pattern Recognit. **34**(3), 727–739 (2001)
55. Ojansivu, V., Heikkilä, J.: Blur insensitive texture classification using local phase quantization. In: Proc. International Conference on Image and Signal Processing, pp. 236–243 (2008)
56. Petpon, A., Srisuk, S.: Face recognition with local line binary pattern. In: Proc. International Conference on Image and Graphics, pp. 533–539 (2009)
57. Pietikäinen, M., Ojala, T., Xu, Z.: Rotation-invariant texture classification using feature distributions. Pattern Recognit. **33**, 43–52 (2000)
58. Porebski, A., Vandenbroucke, N., Macaire, L.: Haralick feature extraction from LBP images for color texture classification. In: Proc. Workshop on Image Processing Theory, Tools and Applications, pp. 1–8 (2008)
59. Raja, Y., Gong, S.: Sparse multiscale local binary patterns. In: Proc. British Machine Vision Conference, 2006
60. Roy, A., Marcel, S.: Haar local binary pattern feature for fast illumination invariant face detection. In: Proc. British Machine Vision Conference, 2009
61. Shan, C., Gritti, T.: Learning discriminative LBP-histogram bins for facial expression recognition. In: Proc. British Machine Vision Conference, p. 10 (2008)
62. Shan, C., Gong, S., Mcowan, P.: Appearance manifold of facial expression. In: Proc. IEEE ICCV Workshop on Human-Computer Interaction (HCI), pp. 221–230 (2005)
63. Shan, S., Zhang, W., Su, Y., Chen, X., Gao, W.: Ensemble of piecewise FDA based on spatial histograms of local (Gabor) binary patterns for face recognition. In: Proc. International Conference on Pattern Recognition, vol. 4, pp. 606–609 (2006)
64. Smith, R.S., Windeatt, T.: Facial expression detection using filtered local binary pattern features with ECOC classifiers and platt scaling. In: JMLR Workshop on Applications of Pattern Analysis, vol. 11, pp. 111–118 (2010)
65. Sokal, R.R., Rohlf, F.J.: Biometry. Freeman, New York (1969)
66. Sun, N., Zheng, W., Sun, C., Zou, C., Zhao, L.: Gender classification based on boosting local binary pattern. In: Proc. International Symposium on Neural Networks, pp. 194–201 (2006)
67. Swain, M.J., Ballard, D.H.: Color indexing. Int. J. Comput. Vis. **7**(1), 11–32 (1991)
68. Tan, N., Huang, L., Liu, C.: A new probabilistic local binary pattern for face verification. In: Proc. IEEE International Conference on Image Processing, pp. 1237–1240 (2009)
69. Tan, X., Triggs, B.: Enhanced local texture feature sets for face recognition under difficult lighting conditions. In: Analysis and Modeling of Faces and Gestures. Lecture Notes in Computer Science, vol. 4778, pp. 168–182. Springer, Berlin (2007)
70. Tan, X., Triggs, B.: Fusing Gabor and LBP feature sets for kernel-based face recognition. In: Analysis and Modeling of Faces and Gestures. Lecture Notes in Computer Science, vol. 4778, pp. 235–249. Springer, Berlin (2007)
71. Trefny, J., Matas, J.: Extended set of local binary patterns for rapid object detection. In: Proc. Computer Vision Winter Workshop, pp. 1–7 (2010)
72. Turtinen, M., Pietikäinen, M.: Contextual analysis of textured scene images. In: Proc. British Machine Vision Conference, pp. 849–858 (2006)
73. Varma, M., Zisserman, A.: A statistical approach to materials classification using image patch exemplars. IEEE Trans. Pattern Anal. Mach. Intell. **31**, 2032–2047 (2009)
74. Wang, L., He, D.C.: Texture classification using texture spectrum. Pattern Recognit. **23**, 905–910 (1990)
75. Wang, X., Han, T.X., Yan, S.: An HOG-LBP human detector with partial occlusion handling. In: Proc. International Conference on Computer Vision, pp. 32–39 (2009)
76. Wolf, L., Hassner, T., Taigman, Y.: Descriptor based methods in the wild. In: Proc. ECCV Workshop on Faces in Real-Life Images, pp. 1–14 (2008)
77. Yan, S., Shan, S., Chen, X., Gao, W.: Locally assembled binary (LAB) feature with feature-centric cascade for fast and accurate face detection. In: Proc. IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–7 (2008)

78. Yang, H., Wang, Y.: A LBP-based face recognition method with Hamming distance constraint. In: Proc. International Conference on Image and Graphics, pp. 645–649 (2007)
79. Yao, C.H., Chen, S.Y.: Retrieval of translated, rotated and scaled color textures. Pattern Recognit. **36**(4), 913–929 (2003)
80. Yu, L., Liu, H.: Feature selection for high-dimensional data: A fast correlation-based filter solution. In: Proc. 12th Int. Conf. on Machine Learning, pp. 856–863 (2003)
81. Zhang, B., Gao, Y., Zhao, S., Liu, J.: Local derivative pattern versus local binary pattern: Face recognition with high-order local pattern descriptor. IEEE Trans. Image Process. **19**(2), 533–544 (2010)
82. Zhang, G., Huang, X., Li, S., Wang, Y., Wu, X.: Boosting local binary pattern (LBP)-based face recognition. In: Proc. Advances in Biometric Person Authentication, pp. 179–186 (2005)
83. Zhang, L., Chu, R.F., Xiang, S.M., Liao, S.C., Li, S.Z.: Face detection based on multi-block LBP representation. In: Proc. IEEE International Conference on Biometrics, pp. 11–18 (2007)
84. Zhang, L., Zhang, L., Guo, Z., Zhang, D.: Monogenic-LBP: A new approach for rotation invariant texture classification. In: Proc. International Conference on Image Processing, pp. 2677–2680 (2010)
85. Zhang, W., Shan, S., Gao, W., Chen, X., Zhang, H.: Local Gabor binary pattern histogram sequence (LGBPHS): A novel non-statistical model for face representation and recognition. In: Proc. International Conference on Computer Vision, vol. 1, pp. 786–791 (2005)
86. Zhao, D., Lin, Z., Tang, X.: Contextual distance for data perception. In: Proc. International Conference on Computer Vision, pp. 1–8 (2007)
87. Zhao, D., Lin, Z., Tang, X.: Laplacian PCA and its applications. In: Proc. International Conference on Computer Vision, pp. 1–8 (2007)
88. Zhao, G., Pietikäinen, M.: Dynamic texture recognition using local binary patterns with an application to facial expressions. IEEE Trans. Pattern Anal. Mach. Intell. **29**(6), 915–928 (2007)
89. Zhao, G., Ahonen, T., Matas, J., Pietikäinen, M.: Rotation invariant image and video description with local binary pattern features. Under review (2011)
90. Zhao, S., Gao, Y., Zhang, B.: Sobel-LBP. In: Proc. International Conference on Image Processing, pp. 2144–2147 (2008)
91. Zhou, H., Wang, R., Wang, C.: A novel extended local-binary-pattern operator for texture analysis. Inf. Sci. **178**, 4314–4325 (2008)
92. Zhu, C., Bichot, C.-E., Chen, L.: Multi-scale color local binary patterns for visual object classes recognition. In: Proc. International Conference on Pattern Recognition, pp. 3065–3068 (2010)

http://www.springer.com/978-0-85729-747-1

Computer Vision Using Local Binary Patterns
Pietikäinen, M.; Hadid, A.; Zhao, G.; Ahonen, T.
2011, XVI, 212 p., Hardcover
ISBN: 978-0-85729-747-1