AN OPTIMIZATION METHOD FOR TWO-DIMENSIONAL LBP FEATURE VECTORS

Llukman ÇERKEZİ

MS Dissertation

Institute of Graduate Programs
Electrical and Electronics
Engineering Program

May, 2019

AN OPTIMIZATION METHOD FOR TWO-DIMENSIONAL LBP FEATURE VECTORS

Llukman ÇERKEZİ

MASTER OF SCIENCE THESIS

Institute of Graduate Programs
Electrical and Electronics
Engineering Program
Supervisor: Assist. Prof. Dr. Cihan TOPAL

Eskişehir
Eskişehir Technical University
Institute of Graduate Programs
May, 2019

FINAL APPROVAL FOR THESIS

This thesis titled **"An Optimization Method For Two-Dimensional LBP Feature Vectors"** has been prepared and submitted by **Llukman Çerkezi** in partial fullfilment of the requirement in "Eskişehir Technical University Directive on Graduate Education and Examination" for the Degree of Master of Science in **Electrical and Electronics Engineering Department** has been examined and approved on 27/05/2019.

Member	Title, Name and Surname	Signature
Member (Supervisor):	**Assist. Prof. Dr. Cihan TOPAL**	.....
Member	: **Prof. Dr. Ömer Nezih GEREK**	.....
Member	: **Prof. Dr. Hakan ÇEVİKALP**	.....

Director of Graduate School of Sciences

ABSTRACT

AN OPTIMIZATION METHOD FOR
TWO-DIMENSIONAL LBP FEATURE VECTORS

Llukman ÇERKEZİ

Department of Electrical and Electronics Engineering
Eskişehir Technical University, Institute of Graduate Programs, May 2019

Supervisor: Assist. Prof. Dr. Cihan TOPAL

Local binary patterns (LBP) is considered to be one of the most discriminative and computationally efficient descriptor for many computer vision applications. Among numerous variants of LBP, there are also approaches that construct 2-dimensional (2D) histograms to provide a better representation of texture patterns. Those approaches obtain final feature vector by either concatenating marginal histograms of 2D distribution; or flattening the whole distribution in a higher dimensional vector. The resulted feature vector is a more compact one in the former scenario, however, the vector in the latter can provide better accuracy.

In this thesis, we propose a method to make LBP features more discriminative by optimizing projections of joint LBP distribution onto the marginal histograms. In order to find a more efficient representation of the feature vector, we seek for the least redundant marginal histograms of a joint LBP distribution via optimizing several constraints. In this way, we aim to have a more compact feature vector in contrast to the methods which flatten the joint distribution without sacrificing accuracy. Experiments we perform on five popular texture datasets show that the proposed method provides higher recognition rates with the same size feature vectors and comparable results even with lower dimensional vectors.

Keywords: Local Binary Pattern, Texture Recognition, Optimization.

ÖZET

İKİ-BOYUTLU YİÖ ÖZNİTELİK VEKTÖRLERİ İÇİN BİR OPTİMİZASYON YÖNTEMİ

Llukman ÇERKEZİ

Elektrik-Elektronik Mühendisliği Anabilim Dalı
Eskişehir Teknik Üniversitesi, Lisanüstü Eğitim Enstitüsü , Mayıs 2019

Danışman: Dr. Öğr. Üyesi Cihan TOPAL

Yerel İkili Örüntüler (YİÖ) bir çok bilgisayarlı görü uygulamasında en ayırt edici ve hesaplama açısından etkin tanımlayıcılarından biri olarak nitelendirilir. Alanyazında bulunan çok sayıda YİÖ versiyonunun (biçimlerinin) yanı sıra doku örüntülerinin daha iyi bir gösterim sağlaması için 2-boyutlu (2B) histogram yaklaşımları mevcuttur. Bu yaklaşımda final öznitelik vektörü ya 2B dağılımının marjinal histogramlarını birleştirerek ya da 2B dağılımını düzleştirerek tek bir vektör olarak elde edilir. Birinci durumda elde dilen öznitelik vektörü daha kompakt iken ikinci durumda elde edilen öznitelik vektörü daha iyi performans göstermektedir.

Bu tezde birleşik YİÖ dağılımından elde edilen marjinal histogramlarını optimize ederek (eniyileştirerek) daha iyi sınıflandırma sağlayan öznitelikler elde edilmesi için bir eniyileme yöntemi geliştirilmiştir. Öznitelik vektörünün daha etkili bir gösterimini bulmak amacıyla, çeşitli sınırlamalar optimize ederek (eniyileştirerek) birleşik YİÖ dağılımının en az artıklık içeren marjinal histogramları araştırılmıştır. Böylelikle, birleşik dağılımı düzleştirerek tek bir vektöre indirgeyen yöntemlerin aksine doğruluk performansını düşürmeyerek daha kompakt bir öznitelik vektörünün elde edilmesi hedeflenmiştir. Beş farklı doku veri setlerinde yapılan deneylerde önerilen yöntemin aynı boyutta olan öznitelik yaklaşımından daha yüksek tanıma performansı ve hatta daha küçük boyutlu öznitelik vektörüyle kıyaslanabilir sonuçlar elde ettiği görülmüştür.

Anahtar Sözcükler: Yerel İkili Örüntüler, Doku Tanıma, Optimizasyon.

27/05/2019

STATEMENT OF COMPLIANCE WITH ETHICAL PRINCIPLES AND RULES

I hereby truthfully declare that this thesis is an original work prepared by me; that I have behaved in accordance with the scientific ethical principles and rules throughout the stages of preparation, data collection, analysis and presentation of my work; that I have cited the sources of all the data and information that could be obtained within the scope of this study, and included these sources in the references section; and that this study has been scanned for plagiarism with "scientific plagiarism detection program" used by Eskişehir Technical University, and that "it does not have any plagiarism" whatsoever. I also declare that, if a case contrary to my declaration is detected in my work at any time, I hereby express my consent to all the ethical and legal consequences that are involved.

.....

Llukman ÇERKEZİ

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

LBP	Local Binary Pattern
R	Radius of LBP
P	Number of Neighbors in LBP
g_c	Central Pixel
g_p	Neighbouring Pixel
$P_{S,M}(s,m)$	Joint Histogram of Two Different LBP Variants
CLBP	Completed Local Binary Pattern
ELBP	Extended Local Binary Patterns
LTP	Local Ternary Pattern
RILPQ	Rotation Invariant Local Phase Quantization
COV-LPBD	Covariance and Local Binary Pattern Difference
LBC	Local Binary Count
DLBP	Dominant Local Binary Pattern
disCLPB	discriminative Completed Local Binary Pattern
MRELBP	Median Robust Extended Local Binary Pattern
ILBP	Improved Local Binary Pattern
CDLF	Completed Discriminative Local Features
LBPNet	Local Binary Pattern Network
LBC	Local Binary Convolutional
LDP	Local Derivative Pattern
HOG	Histogram of Oriented Gradient
SIFT	Scale Invariant Feature Transform
PCA	Principal Component Analysis
NN	Nearest Neighbour
SVM	Support Vector Machines

1. INTRODUCTION

In this chapter, an introduction on texture recognition will be given in general. Afterwards, the scope of the thesis will be explained.

1.1 Texture Recognition

Texture classification is an active research area in computer vision due to numerous applications e.g. object recognition, medical image analysis, industrial inspection, face analysis and biometrics, remote sensing, etc. A texture is a visual pattern which complies with various statistical properties such as regularity and uniformity of a certain shape. Analogous to other pattern analysis applications, texture classification also consists of two stages, i.e. feature extraction and recognition. However, since a texture comprises of a repeating pattern of a certain structure with different levels of modal variation, there is an amount of redundancy that feature extraction methods should avoid. For this reason, the former stage has substantial importance in texture recognition due to the fact that poor feature representation will not provide good accuracy results regardless the employed classifier. Therefore extracted features should be able to embrace intraclass variations caused by illumination, rotation and scale, as well as it should provide a distinctive representation for sufficient discrimination of individual classes.

Early feature extraction methods consist of various approaches. One of the most popular approach is to utilize statistical analysis of textures for feature extraction purpose [10, 11]. The Bag of Words method with its variants which represents the texture as histogram of discrete vocabulary local features is also an efficient method [12]. Other important texture descriptors include filter banks [13], random features [14], sparse descriptor like Scale Invariant Feature Transform (SIFT) [15] and dense descriptor like Histogram of Oriented Gradients (HOG) [16].

Ojala et al. [17] proposed Local Binary Pattern (LBP) as a texture descriptor which is shown to be very efficient in variety of applications such as face recognition, medical image analysis, texture classification and many others [18]. A LBP feature vector is constructed by pattern encoding and

histogram accumulation steps [19]. In the naïve version of LBP, the center pixel is compared to its neighbor pixels in order to form a bit sequence. Later, that bit sequence is interpreted as a binary number and used to increase the corresponding bin in an histogram. This operation is repeated for all pixels in the image and thus the feature vector is obtained.

Numerous methods are proposed for construction of LBP histograms to improve the accuracy. Among many others, a popular approach is to utilize LBP information to obtain a multi-dimensional (usually 2D) joint histogram [20]. Joint LBP histograms can be built by considering various encoding strategies for different LBP types. Final feature vector is obtained by either concatenating marginal histograms; or flattening the whole joint histogram in a higher dimensional vector by concatenating all rows.

In this thesis, we propose a method to obtain more discriminative LBP features by optimizing projections of 2D joint distribution onto the marginal histograms. For this purpose, we perform several recognition tasks by optimizing several attributes of feature distributions such as entropy, joint entropy, mutual information, correlation and variance. In this way, we seek for the least redundant axes in which the 2D histogram accumulation will be projected in order to have a better feature representation, hence higher accuracy. Experiments we perform on popular texture datasets show that the optimized LBP features outperform the recognition rates obtained by the naive method with the same vector size. In addition, the proposed method achieves comparable accuracy results even with the concatenated lower-dimensional vectors compare to the higher-dimensional vectors obtained by histogram flattening.

1.2 Thesis Organization

The rest of the thesis is organized as follows. Section 2 gives an overview of related work including naïve LBP with its extensions, variants of LBP and applications of LBP. In Section 3 we explain the proposed method in detail. In Section 4 we present our experimental results and discuss our findings. And in Section 5 we provide concluding remarks for the thesis study.

2. RELATED WORK

In this Section, first we will provide detailed information about naïve LBP including some of basic extensions (Section 2.1). In the Section 2.2 we explain in detail a Completed Binary Pattern since it provides important concepts which will be used in this work. Finally, we briefly explain other variants of LBP (Section 2.3) and various applications (Section 2.4) where LBP has been successfully applied.

2.1 Naïve LBP

A LBP code proposed by Ojala et al. [17] of a given pixel in an image is calculated as follows:

$$LBP_{P,R} = \sum_{p=0}^{P-1} s(g_p - g_c)2^p, \; s(x) = \begin{cases} 1, & \text{if } x \geq 0 \\ 0, & \text{otherwise} \end{cases} \tag{2.1}$$

where g_c and g_P are central pixel and corresponding neighbour pixel respectively. Here P is the total number of involved neighbor pixels and R is the radius from central pixel to neighbor pixels as it is shown in Fig. 2.1. The naïve version $LBP_{P,R}$ has two main drawbacks, the dimension of feature vector is exponentially increased depending on neighbour pixel P which will lead to sparse distributions and it is not robust against rotation. For this reason, authors in [17] introduced the next variant of naive LBP code called Uniform

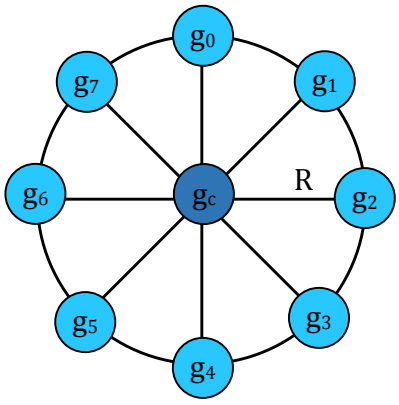


Figure 2.1: *LBP sampling scheme for $P = 8$ and $R = 1$.*

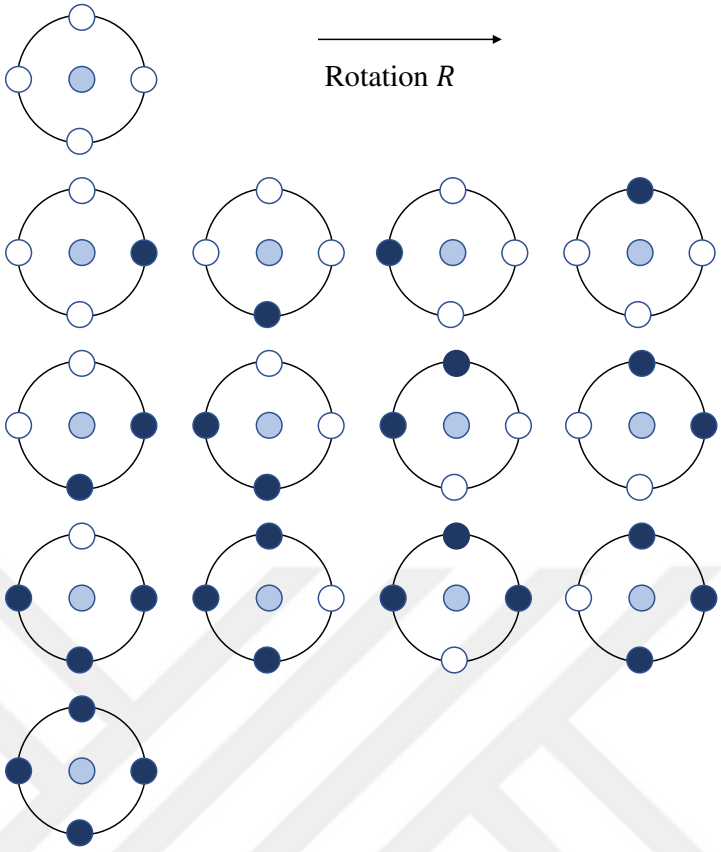


Figure 2.2: *The* 14 *different uniform patterns for sampling rate* $P = 4$.

LBP.

The *uniformity measure* (U) value of LBP patterns is defined as number of bitwise $0-1$ changes, as follows:

$$LBP^{u2}_{P,R} = U(LBP\_S_{P,R}) = |s(g_{P-1} - g_c) - s(g_0 - g_c)| + |\sum_{p=1}^{P-1} s(g_p - g_c) - s(g_{p-1} - g_c)| \tag{2.2}$$

If the value $LBP^{u2}_{P,R}$ is lower or equal then 2, the corresponding pattern is considered uniform, otherwise the pattern will be considered non-uniform. In general for different number of neighborhood P, there are $P \times (P-1) + 2$ different uniform patterns [21] (see Fig. 2.2). There are two main advantages of uniform pattern over the naive LBP code. First, most of the uniform patterns in natural images (textures) are uniform [17]. In our experiments carried out in Outex_TC_00010 [17], KTH-TIPS2b [22], UIUC [12], UMD [23], Brodatz [24] datasets, using $(P, R) = (8, 1)$ uniform patterns account approximately 85% of all patterns.

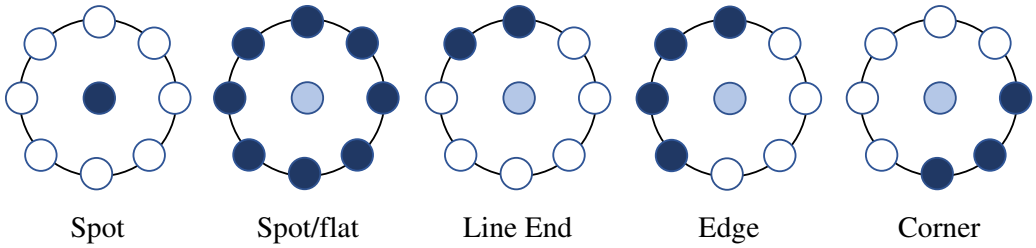


Figure 2.3: *Texture Primitives detected by uniform LBP.*

Second, uniform patterns are more statistically robust than naïve ones, i.e. more robust to noise. On the other hand, feature vector obtained using uniform pattern is more compact since the number of possible of uniform pattern labels is significantly lower [21] (see Table 2.1). Furthermore uniform patterns can be regarded as micro-texton i.e., spots, flat areas, edges, edges end, curves and so on (see Fig. 2.3).

Nevertheless, uniform patterns are not robust to rotation. To achieve the rotation invariance the following condition must be satisfied:

$$LBP^{riu2}_{P,R} = \begin{cases} \sum_{p=0}^{P-1} s(g_p - g_c), & \text{if } U(LBP\_S_{P,R}) \leq 2 \\ P+1, & \text{otherwise} \end{cases} \tag{2.3}$$

Notice that $LBP^{riu2}_{P,R}$ has range of $[0, P+1]$. After the particular LBP is calculated for each pixel in an image, the final feature vector is obtained by histogram of corresponding LBP values (see Fig. 2.4). For more details [17] and [21] provide comprehensive explanation about LBP.

The original version of LBP code has significant limitation since it captures small spatial support area. For $(P, R) = (8, 1)$ naïve LBP operator can capture only local 3×3 neighbourhood, for this reason large scale structures are not considered. On the other hand, adjacent LBP codes are not totally

Table 2.1: *A comparison of featrure vector dimension for naive LBP ($LBP_{P,R}$), uniform LBP ($LBP^{u2}_{P,R}$) and rotation invariant LBP ($LBP^{riu2}_{P,R}$) for different values of P.*

P	$LBP_{P,R}$	$LBP^{u2}_{P,R}$	$LBP^{riu2}_{P,R}$
8	256	59	10
16	65536	243	18
24	16777216	555	26

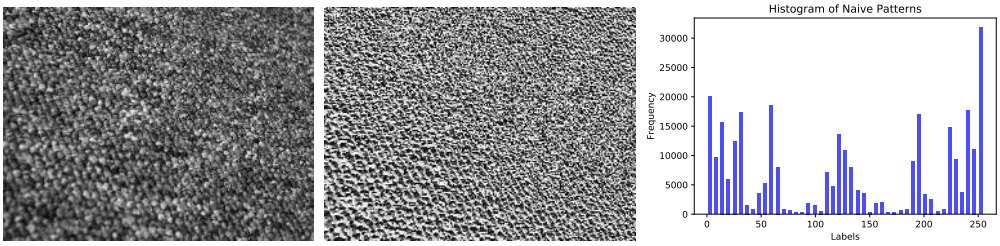


Figure 2.4: *Example of an input texture (left), corresponding LBP image (middle), and LBP histogram (right).*

independent. Figure 2.5 displays two adjacent $LBP_{2,1}$ patterns (the sub figure in the left). The second pixel is the centre pixel of first LBP pattern and also the left neighbour pixel for the second LBP pattern. The third pixel also is the right neighbour pixel for the first LBP Pattern and the centre pixel for the second LBP pattern. Thus, these two pixels will be considered for two different cases. Looking the Eq. 2.1 we observe the second and third pixels take both g_p and g_c values for both cases. Analysing sign function $s(x)$ in Eq. 2.1 we conclude that the second and third pixels' value can not be zero at the same time. The example above shows that adjacent pixels' of LBP code have dependency on each other.

One solution to handle this problem is to use multiscale LBP, in other words, combining N different LBP operators with varying P and R, so each pixel will have N different LBP codes. In this way both micro and macro pattern will be considered. The best way to obtain information is by taking joint distribution of these codes. However the resulted distribution would be in high dimension that results a sparse distribution even for small texture size. For example, joint distribution of $LBP^{u2}_{8,1}$, $LBP^{u2}_{16,2}$ and $LBP^{u2}_{24,3}$ would contain $59 \times 243 \times 555 \approx 7 \times 10^6$ bins, which is impractical. One solution is to take in consideration only marginal histograms. In this case, the statistical independence of the outputs of the different LBP operators can not be

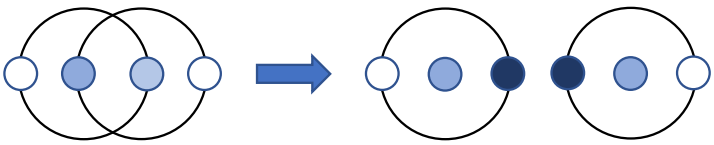


Figure 2.5: *Two adjacent $LBP_{2,1}$ neighbourhoods and an impossible combination of codes. A dark blue disk means the gray level of sample is lower than that of the centre.*

guaranteed. It is shown that using multi-scale analysis often increases the discriminative power of the feature vector [17].

For the aforementioned facts, researchers have come up with different strategies. Instead using only multi-scale LBPs they proposed also to use different variants of LBPs within the same scale. The Completed Local Binary Pattern [20] is the most well-known approach which will be explained in following subsection.

2.2 Completed LBP

Guo et al. [20] created joint distribution of completed LBPs (CLBP), namely three different type of LBP, i.e., LBP_Sign, LBP_Magnitude and LBP_Center. Note that LBP_Sign is the same as naive LBP. CLBP_Magnitude is calculated as follows:

$$LBP\_M = \sum_{p=0}^{P-1} s(|g_p - g_c|, c)2^p \; where \; c = \frac{1}{P}\sum_{p=0}^{P-1} |g_p - g_p| \tag{2.4}$$

we can derive $LBP\_M^{riu2}_{P,R}$ exactly the same way as $LBP^{riu2}_{P,R}$.

Note that the only difference between these two types of LBP is their thresholds. They obtain the final feature vector in two different ways. In the first one, they merge marginal histograms into one vector, and in the second one, they flatten the entire multi-dimensional histogram into one higher-dimensional vector.

Furthermore, they investigate why $LBP\_S_{P,R}$ can extract texture features reasonable well. For this reason, they define local difference vector $[d_0, d_1, ..., d_{P-1}]$ where $d_p = g_p - g_c$. The components g_c and g_p are central pixel and neighbors pixels respectively. The value d_p can be formulated as follows:

$$d_p = s_p * m_p \quad and \quad \begin{cases} s_p = sign(d_p) \\ m_p = \mid d_p \mid \end{cases} \qquad where \quad s_p = \begin{cases} s_p = 1, d_p \geq 0 \\ -1, d_p < 0 \end{cases} \tag{2.5}$$

It can be seen that $LBP\_S_{P,R}$ utilizes only s_p vector, where -1 value is changed to 0. They investigate which component s_p or m_p convey more information to represent the local difference vector d_p. This can be verified by

reconstructing d_p using only one component and checking the reconstruction error. The difference vector d_p can be well modelled by Laplace distribution $Q_L(g) = \frac{e^{\frac{-g}{\lambda}}}{2\lambda}$ depends on the image content. It is obvious that d_p can not be reconstructed using only one component. For this purpose there are used some prior probability distribution of s_p and m_p.

They reconstruct local difference d_p using only magnitude component d_p as follows:

$$d_p^m = m_p \cdot s_p^{'} \tag{2.6}$$

where $s_p^{'}$ is modelled by Bernoulli distribution. Similarly local difference d_p using only sing component s_p as follows:

$$d_p^s = m_p^{'} \cdot s_p \tag{2.7}$$

where m_p can be set as the mean value of the magnitude component m_p. The local difference reconstruction errors for d_p^s and d_p^m are defined as:

$$E_s = E[(d_p - d_p^s)^2] \quad , \quad E_m = E[(d_p - d_p^m)^2] \tag{2.8}$$

They showed both theoretically and experimentally that the reconstruction error of E_m is four times higher than E_s. Thus, it is proven that s_p namely, $LBP\_S_{P,R}$ preserves more information than magnitude component m_p.

2.3 Other Variants of LBP

In the literature there are many variants of LBP proposed by researchers for different applications. This section introduces most well-known variants of LBP.

Extended Local Binary Patterns (ELBP). Liu et al. [25], proposed to use intensity and difference based local features. They use four descriptors central-pixel intensity (CI-LBP), neighbours intensity (NI-LBP), radial difference (RD-LBP) and angular-difference (AD-LBP). Inspired by Guo et al. [20], they combine these descriptors jointly or hybridly. However high dimensional feature vector caused by flattening multi dimensional joint distribution may not be suitable for real time purposes.

Local Ternary Pattern (LTP). Tan et al. [26], propose a new descriptor called LTP a generalization of LBP which is more discriminant and

more robust to noise in uniform regions. Naïve LBP forms its binary pattern by looking the difference between central pixel with neighbours pixels, thus it is sensitive in near-uniform image regions. In order to handle this problem authors proposed 3-level ternary code. If the difference between the center and neighbour pixels is in the range $\pm t$ it is set to 0 value, and ones above the value t assigned to -1 and finally ones below the threshold t assigned to -1. The threshold t is user-specified. Since the generated output may have three different values, they propose a coding scheme that splits each ternary pattern into its positive and negative groups.

Rotation Invariant Local Phase Quantization (RILPQ). Ojansivu at at. [27], proposed a descriptor named Local Phase Quantization (LPQ) for texture classification that is robust to image blur. The descriptor operates in Fourier phase domain computed locally in a window for every pixel in image. Since phase information is used, the descriptor is robust against uniform illumination changes. Rotation invariant version of LPQ named RILPQ is generalized in [28].

Covariance and Local Binary Pattern Difference (COV-LPBD). Hong et al., address the problem of combining LBP(-like) features with other discriminative descriptors, since LBP is an index of discrete patterns [29]. They proposed descriptor named Covariance and LBP Difference (COV-LBPD) which is able to capture intrinsic features in a compact manner.

Local Binary Count (LBC). Zhao et al. proposed a novel descriptor called Local Binary Count (LBC) [30] . On the contrary to LBP which aims to extract the local binary structure; LBC extract local binary difference information and abandon the structural information. They also proposed completed LBC (CLBC) scenario to enhance the performance of texture classification.

Dominant Local Binary Pattern. Liao et al. proposed to combine two sets of features i.e., dominant LBP (DLBP) and features extracted using circularly symmetric Gabor filter response [31]. DLBP captures most dominant patterns while Gabor-based features aim to give global textural information.

Bianconi et al. investigate the effectiveness of Dominant LBP (DLBP) [32]. They conclude that DLBP provides a significant compression rate and retaining information about the patterns' labels improves the discrimination capability of DLBP.

Fathi and Nillchi aimed to utilize more efficient both and non-uniform patterns [33]. Their proposed LBP variant uses a circular majority voting filter and convenient rotation-invariant labeling scheme to obtain regular uniform and non-uniform patterns. Thus proposed LBP variant becomes more discriminative and robust against the noise.

Zhou et al. addressed the problem of uniform patterns in LBP code [34]. Uniform patterns ignore important texture information and are sensitive to noise. Their method combines and classifies "non-uniform" local patterns based on their structure and probability of occurrence and then becomes more robust against noise.

discriminative Completed Local Binary Pattern (disCLBP). Guo et al. presented a new learning framework to obtain discriminative patterns which is formulated into three-layered model [35]. Their framework can be combined with LBP-based approaches. In the first layer, most robust and dominant patterns are learnt while in second layer the most discriminative patterns with respect to each class are estimated. Finally in the last layer, representation capability of features is maximized using patterns obtained in the second layer.

Median Robust Extended Local Binary Pattern (MRELBP). Naive LBP is very sensitive to image noise and suffers to capture microstucture information. Liu et al. [36], proposed MRELBP which compares regional image filter responses (median filter) rather than raw image intensities in order to overcome above problems. Proposed descriptor is able to capture both micro-structure and macro-structure of texture and it is robust against to noise.

Scale Selective Local Binary Pattern. Guo et al. addressed the scale problem of LBP [37]. They build histogram of pre-learned dominant patterns after image is derived by Gaussian filter. Eventually, the maximal frequency among scale space is considered as the scale invariant feature for each pattern.

Improved Local Binary Pattern (ILBP). Lu et al. proposed a new descriptor named ILBP where they discover an important group of primitives such as lines, T-junctions and cross-intersections which forms a non-uniform patterns [38]. They show that these primitives are also robust against monotonic gray-scale variation and rotation and show better performance than LBP.

Pixel Based Local Binary Pattern Descriptor. Cote et al. instead of representing the whole texture image with one LBP histogram, they proposed to increase the classification accuracy trough a novel pixel-based classification mechanism [39]. They assigned label of an image bu aggregating pixels label through a voting process. The proposed method suffers in terms computational complexity since it makes pixel-based classification.

Completed Discriminative Local Features (CDLF). Zhang et al. proposed an adaptive histogram accumulation algorithm (AHA) [19]. AHA assigns different weights for each local region by means of local contrast information. The contribution is high around the edges while it is low at the flat regions. AHA technique could be applied on any encoding strategy including joint distribution approach, however the improvements are not dramatic.

A comprehensive taxonomy about recent developments about LBP can be found in [18] and [40]. There are also some recent studies about deep LBP network inspired by well-known algorithm in deep learning named convolutional neural network (CNN).

Local Binary Pattern Network. Xi et al. [41], proposed a novel architecture named Local Binary Pattern Network (LBPNet). The architecture is inspired by CNN topology whereas instead of constitutional kernels LBP descriptor is replaced. One advantage of model is that avoids model learning since LBP is off-the-shelf descriptor. Extensive experimental results on FERET and LFW datasets show that proposed model achieve comparable results to other unsupervised methods.

Local Binary Convolutional (LBC) Neural Networks. Juefei-Xu et al. [42], motivated by LBP propose Local Binary Convolutional (LBC) network. LBC layer is built up from set of fixed sparse pre-defined binary convolutional filters. The LBP layer need to learn 9× to 169× less parameters which make it more efficient in term of computation cost than standard convolutional layers. They show both theoretically and experimentally that LBC layers are good approximation of standard convolutional layers.

2.4 Applications of LBP

Texture analysis has a variety applications in computer vision. Since LBP has shown efficient performance in texture classification, researches have proposed different methods to use in other application domains. Thus LBP and

Figure 2.6: *An example of facial image divided into* 7×7 *blocks (left). The weights set for* χ^2 *dissimilarity measure (right). The black windows indicate zero impact while the white block indicate high impact [4].*

its variants play an important role in biometric recognition, segmentation, detection and face analysis [21].

Ahonen et al. [4], utilize LBP for face recognition purposes. Since the dataset contain no rotational changes they utilized uniform LBPs. In order to represent better the face, they propose to divide face into several blocks called facial regions and extract for each region local binary patterns. They construct final feature vector by concatenating the local LBP histograms. So both statistics of the facial micro-patterns and their spatial locations are represented. They also find out that some facial regions have more discriminative information than others. So they assign for each region a weighted

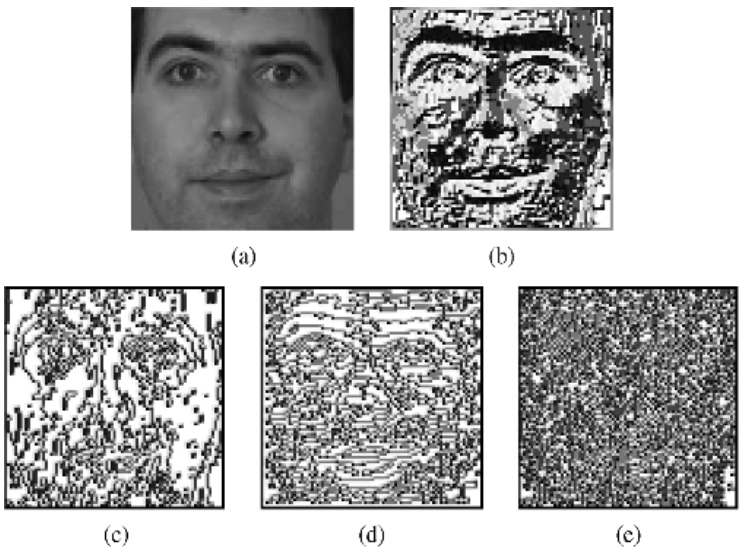


Figure 2.7: *Visualization of LBP and LDP outputs. (a) Input face image. (b) Corresponding LBP Image. (c) The second-order LDP. The third-order and fourth-order LDP in (d) and (e) respectively [5]*

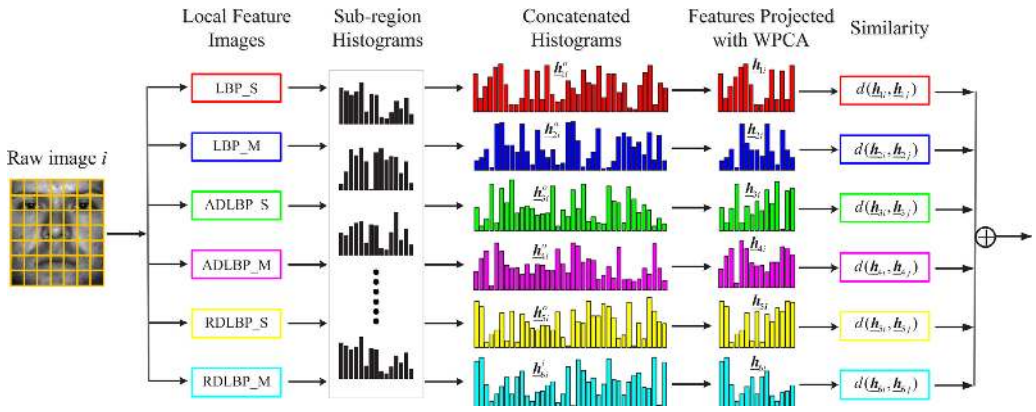


Figure 2.8: *The pipeline of ELBP based face recognition algorithm [6].*

parameter, in order to take this advantage (see Fig. 2.6). Authors in their another study have proposed similar method and discussed possible extensions [43].

Zhang et al. [5], propose a Local Derivative Pattern (LDP) for face recognition purpose. LDP encodes directional pattern features based on local derivative variations (see Fig. 2.7). Proposed descriptor outperforms LBP for both face identification and verification scenarios under various conditions.

Liu et al. [6], use descriptor proposed by Zhuo et al. [34] for face recognition task. They calculate six different LBP-like descriptors from each facial region. Due to the high dimensionality they used whitened PCA to produce more compact and discriminative features (see Fig. 2.8). Extensive experiments carried out in three datasets show that their method outperforms other well known systems. A comprehensive survey about LBP based method for face recognition can be found in [44].

In [7], Shan et al. propose a method for learning discriminative LBP bins using AdaBoost algorithm with application to facial expression recognition (see Fig. 2.9). They validate experimentally that uniform LBP with multiscale variations have explicit impact on performance.

Mu et al. [45], employ LBP as a region descriptor in the task of human detection. They compare LBP descriptor with existing gradient based local feature used in human detection and show that LBP is more discriminative. Nevertheless, existing LBP method does not suit properly in the problem of human detection due to its high complexity and lack of semantic consistency. For this, they propose Semantic-LBP and Fourier-LBP and demonstrate the effectiveness of these two variants over the traditional features for human

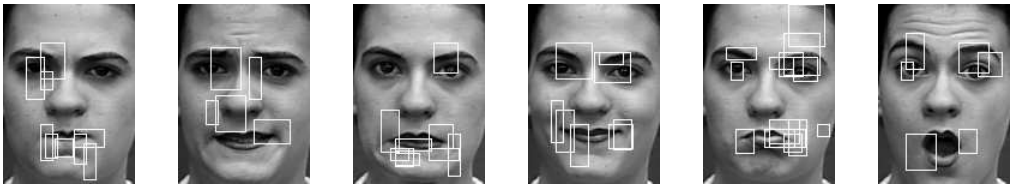


Figure 2.9: *The sub-regions selected by AdaBoost for each facial expression. From left to right: Anger, Disgust, Fear, Joy, Sadness, and Surprise [7].*

detection in INRIA human dataset.

Roy et al. [46], propose Haar Local Binary Patterns which exploits the concepts of Haar feature and LBP for face detection in unfavourable image conditions. Their proposed descriptor is robust against strong illumination conditions, pose and background.

Wang et al. [8], propose similar approach by combining trilinear interpolated Histogram of Oriented Gradients (HOG) with LBP in the scenario of human detection (see Fig. 2.10). The proposed method shows better performance than other state-of-the-art methods in INRIA dataset.

Nanni et al. [47], provide a good comparison of recent variants of LBP-like features in the context of bio-imaging applications. They propose also a novel descriptor named elongated quinary pattern which use elliptical neighbourhood and quinary encoding. Proposed descriptor outperforms naive LBP in three widely-used datasets each comprising different bio-imaging problems.

Yi and Eramian in [9], employ LBP as a sharpness metric for robust segmentation algorithm to separate in and out of focus image regions. Proposed sharpness metric is based on observation that in the blurry regions local image patches have significantly fewer certain local binary pattern compared

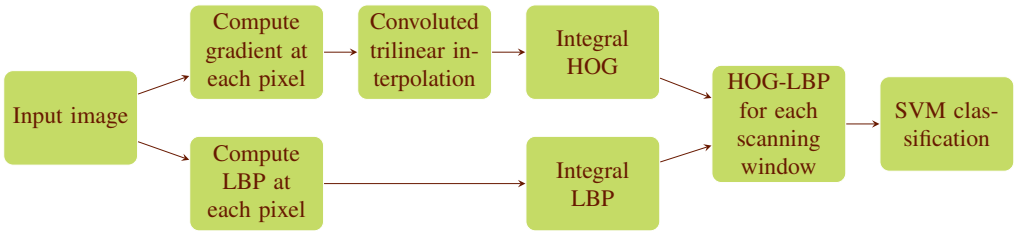


Figure 2.10: *The framework of HOG-LBP detector proposed by [8].*

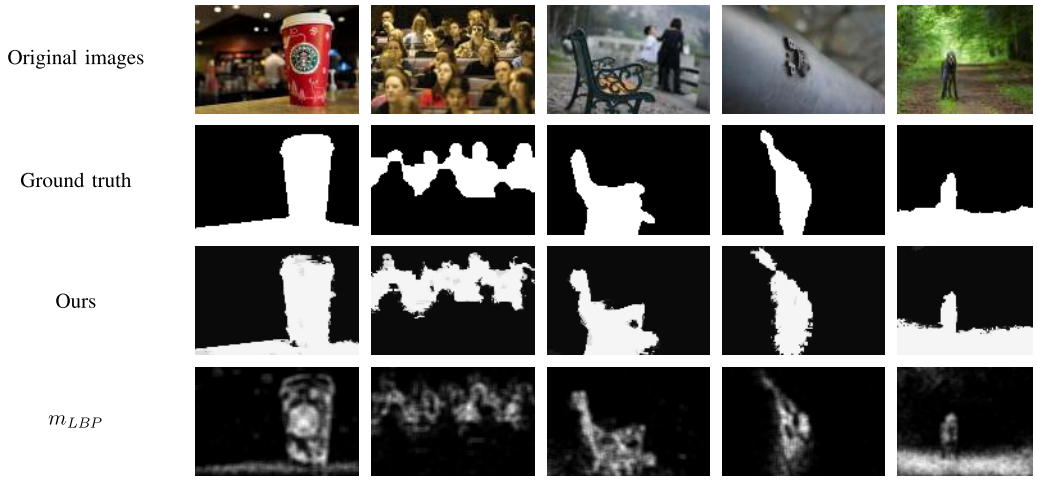


Figure 2.11: *The visual results for defocus blur method proposed [9].*

with those in sharp regions. Sharpness metric achieves comparative results and is not computationally expensive (see Fig. 2.11).

Li et al. [48], exploit LBP for the classification of hyperspectral imagery at high spatial resolution. LBP is used to extract edges, corners, spots along with global Gabor features and original spectral features. Finally these descriptors are fused in one vector for pattern classification process. Extensive experimental results show that proposed method outperforms traditional methods.

Satpathy et al. [49], propose two sets on novel edge-texture descriptor for object recognition i.e. Discriminative Local Binary Pattern and Discriminative Ternary Pattern. These two descriptors are proposed by analysing the weakness of LBP, LTP and RLBP. Proposed descriptors can capture contrast information in images which is necessary for good representation of object counters.

3. Proposed Method

The initial step towards obtaining feature vector is to create a joint 2D LBP histogram denoted as $P_{S,M}(s,m)$ from $LBP\_S^{riu2}_{P,R}$ and $LBP\_M^{riu2}_{P,R}$, respectively. Thus the final feature vector is created either by concatenating marginal histograms P_{LBP_S} and P_{LBP_M} as S_M or by flattening $P_{S,M}(s,m)$ into one vector $S \diagdown M$ (see Table 3.1 for abbreviations).

While the feature vector dimension of S_M is linearly proportional to the number of neighbors in the employed LBP pattern $S \diagdown M$ is quadratically proportional to the number neighbors. In this thesis we aim to find whether there is a more discriminative representation of the 2D LBP histogram which both comprises of a lower dimensional vector and provides an enhanced accuracy. For this purpose, we seek for a better pair of axes to project 2D feature accumulation

3.1 Optimization with Principal Component Analysis

The most intuitive method for this is applying principal component analysis (PCA) to the data and construct the final feature vector as the concatenation of two axes which we obtain via PCA. PCA is a well-established machine learning technique which compute new bases called principal components for an input data and is often utilized for dimension reduction purposes.

On the contrary to many other applications, we do not use PCA for dimension reduction. Instead, here we apply PCA to the 2D LBP distribution to obtain least redundant marginal histograms and represent the new feature by LBP_{PCA}.

For a given image, we compute two different LBP features for each pixel p:

$$p_{LBP} = \begin{bmatrix} LBP\_S^{riu2}_{P,R}(p) & LBP\_M^{riu2}_{P,R}(p) \end{bmatrix}^T \tag{3.1}$$

where p_{LBP} is a 2D vector consisting of 2 LBP values. Then, we construct H matrix where its columns represent p_{LBP} vectors of each pixel in the image:

$$H = \begin{bmatrix} p_{LBP_1} & p_{LBP_2} & \dots & p_{LBP_n} \end{bmatrix} \tag{3.2}$$

where H is a 2 x n matrix and n is the number of pixels in the image.

Once we calculate the 2D LBP vector, we apply PCA to find principal components in order to obtain a more discriminative representation for the data. We apply transformation matrix P to H as the following:

$$PH = Q \tag{3.3}$$

where Q is the transformed 2D LBP histogram. Our aim is to find P such that the covariance matrix of Q denoted by S_Y is diagonal matrix. By diagonalizing covariance matrix of Q, each variable will co-vary as little as possible with other variables. Thus the redundancy would be diminished. S_Y is easily computed as follows

$$\begin{aligned} S_Y &= QQ^T \\ &= (PH)(PH)^T \\ &= P(HH^T)P^T \end{aligned} \tag{3.4}$$

here HH^T is a symmetric matrix ($HH^T = (HH^T)^T$). We know that symmetric matrix can be diagonalized by an orthogonal matrix of its eigenvectors, as follows

$$HH^T = EDE^T \tag{3.5}$$

Table 3.1: *List of Symbols and Notations*

Feature Name	**Feature Type**	**Constraint**	**Symbol**
$LBP\_S\_M^{riu2}_{N,R}$	Marginal	-	S_M
$LBP\_S \setminus M^{riu2}_{N,R}$	Flatten	-	$S \setminus M$
$LBP^{PCA}\_S\_M^{riu2}_{N,R}$	Marginal	Cov matrix	S_M^{PCA}
$LBP^{PCA}\_S \setminus M^{riu2}_{N,R}$	Flatten	Cov matrix	$S \setminus M^{PCA}$
$LBP^{V}\_S\_M^{riu2}_{N,R}$	Marginal	Variance	S_M^{V}
$LBP^{V}\_S \setminus M^{riu2}_{N,R}$	Flatten	Variance	$S \setminus M^{V}$
$LBP^{E}\_S\_M^{riu2}_{N,R}$	Marginal	Entropy	S_M^{E}
$LBP^{E}\_S \setminus M^{riu2}_{N,R}$	Flatten	Entropy	$S \setminus M^{E}$
$LBP^{JE}\_S\_M^{riu2}_{N,R}$	Marginal	Joint Entropy	S_M^{JE}
$LBP^{JE}\_S \setminus M^{riu2}_{N,R}$	Flatten	Joint Entropy	$S \setminus M^{JE}$
$LBP^{MI}\_S\_M^{riu2}_{N,R}$	Marginal	Mutual Inf.	S_M^{MI}
$LBP^{MI}\_S \setminus M^{riu2}_{N,R}$	Flatten	Mutual Inf.	$S \setminus M^{MI}$
$LBP^{C}\_S\_M^{riu2}_{N,R}$	Marginal	Correlation	S_M^{C}
$LBP^{C}\_S \setminus M^{riu2}_{N,R}$	Flatten	Correlation	$S \setminus M^{C}$

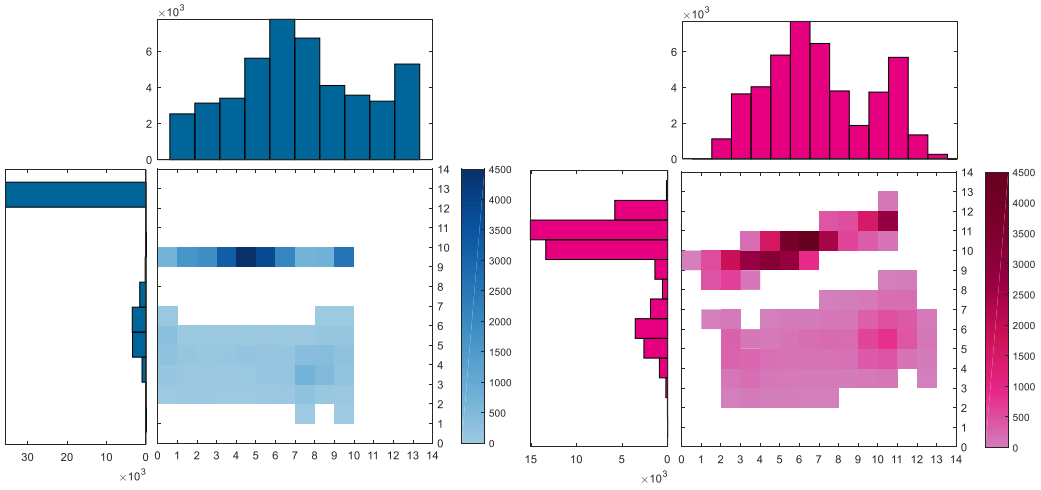


Figure 3.1: *Traditional 2D LBP histogram and optimized 2D LBP histogram with PCA and corresponding marginal histograms.*

where D is a diagonal matrix and E is a matrix of its eigenvectors, i.e. HH^T. If we select P to be a matrix same as E^T and since $P^{-1} = P^T$ holds for orthogonal matrix, then

$$\begin{aligned} S_Y &= P(HH^T)P^T \\ &= P(P^T D P)P^T \\ &= (PP^{-1})D(PP^{-1}) \\ &= D \end{aligned} \tag{3.6}$$

Thus we can conclude that eigenvectors of HH^T are principal components of H. The eigenvector with the highest eigenvalues becomes the most informative one. After obtaining Q, we built our new 2D histogram from it. We obtain PCA based feature vectors S_M^{PCA} and $S\diagdown M^{PCA}$ by concatenating the marginal histograms of our new 2D histogram and by flattening the 2D histogram, respectively. Note that the transformation matrix P, i.e. eigenvector matrix, is actually a 2D rotation matrix. So the corresponding 2D histogram is formed by rotating the original 2D LBP histogram by the rotation matrix R of the rotation angle θ (see Eq. 3.7).

$$R(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \tag{3.7}$$

In this way, we compute principle axes via PCA and project 2D LBP accumulation onto them instead of the original axes in an aim to obtain more

discriminative feature vectors (see Fig. 3.1). Thus we intend to have higher accuracy values without increasing the feature vector size. Although this method intuitively seems reasonable to improve the accuracy, we could not be able to verify this experimentally. When we investigate for a proper explanation, we come up with a reason that PCA assumes the data is linearly spread out in Euclidean space. In our experimental results we observe that 2D LBP distributions does not spread out linearly, hence we could not achieve a higher accuracy.

3.2 Is Naïve 2D LBP Histogram The Most Efficient?

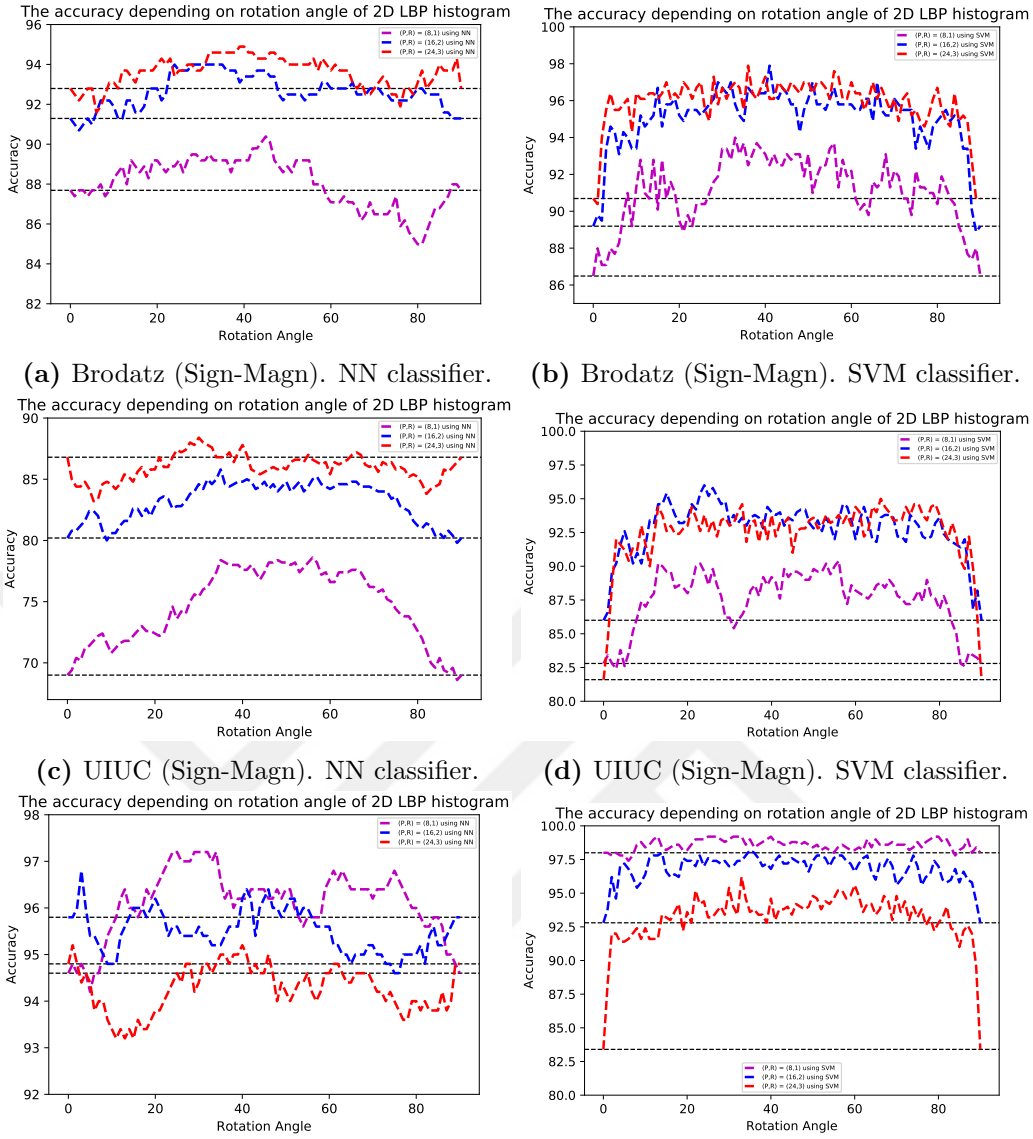
Once we obtain the experimental results and see the disappointing accuracies for S_M^{PCA} and $S \setminus M^{PCA}$, we are confronted the question if there is still room for improvement in the 2D LBP histogram or not? We investigate the answer by a very simple experiment where we take the 2D LBP distribution and transform it by small steps of rotations as in the Eq. 3.8. Here $R(\theta)$ is rotation matrix, P the original LBP pairs and P_{rot} rotated LBP pairs.

$$R(\theta)P_{S,M}(s,m) = P_{rot,S,M}(s,m) \tag{3.8}$$

For each θ rotation angle, we obtain marginal histograms $P_c(\theta)$ of 2D distribution via projection onto x and y axes. Finally, we attain final feature vector as the concatenation of marginal histograms and run an entire texture recognition task for each step. In Fig. 3.2 experimental results are shown for the experiment we performed Brodatz, UMD, and UIUC datasets. It is clearly seen in the figure that different projections can achieve better results up to 2%. For a specific example, UMD dataset using nearest neighbour classifier, results for $P = 8$ and $R = 1$ gives 97.2% whereas it is only around 94.6% for plain rotation at 0°. Therefore, we found that the answer for the above question is yes, there is still potential to have a certain level of improvement in 2D LBP distribution. In the next subsection, we explain the methods that we investigate for exploiting the potential optimizations.

3.3 Proposed Optimization Method

After we experimentally validate that there is potential to optimize feature vectors, we look for an alternative ways to transform the distributions so that



(a) Brodatz (Sign-Magn). NN classifier.

(b) Brodatz (Sign-Magn). SVM classifier.

(c) UIUC (Sign-Magn). NN classifier.

(d) UIUC (Sign-Magn). SVM classifier.

(e) UMD (Sign-Magn). NN classifier.

(f) UMD (Sign-Magn). SVM classifier.

Figure 3.2: *Prospective optimization plot for LBP for three different datasets using nearest neighbour (left) and support vector machines (right) classifiers. The accuracy values indicate the success of recognition task with respect to the corresponding rotation angle of 2D LBP histogram. For this example we use LBP_Sign-LBP_Magnitude LBP pair. LBP_Sign and LBP_Magnitude are defined as in equations 2.1 and 2.4 respectively.*

we achieve better accuracies. Here is a very important detail to emphasize that we rotate all 2D distributions obtained from different textures with the same amount of rotation in advance of projection onto marginal histograms in the experiment shown in Fig. 3.2. However, each texture distribution might have a different amount of rotation and thus even higher classifica-

tion accuracies could be achieved. Therefore, we optimize by minimizing and maximizing several constraints, i.e. variance, correlation, entropy, joint entropy and mutual information to perform simple specific transformations.

In the optimization process, on the contrary to the method that we rotate 2D distributions by the angle obtained via PCA; we rotate 2D LBP distributions $P_{S,M}(s,m)$ by a certain angular step and project it onto marginal histograms. Then we compute aforementioned constraints at each rotation step and seek for a global maximum or minimum along the entire rotation space. In this way, we can obtain feature vectors in different rotations for individual textures. Once we find an extrema, we construct the feature vector by simply concatenating the marginal histograms (see Fig. 3.3 for pipeline of proposed method).

With this optimization scheme, we actually insert an extra step in between constructing the LBP histogram and obtaining the feature vector. In that step, we simply rotate the 2D histogram $P_{S,M}(s,m)$ with a predefined angular step until the concatenated marginal histogram hits a peak on the selected constraint. This additional step of the feature extraction procedure is symmetric, i.e. it is applied in both training and test stages. Thus the algorithm would stop around the same rotation angle for identical textures and obtain similar feature vectors.

In the rest of this section, firstly we explain the statistical concepts employed in this thesis and then how we put in use them for optimization purpose.

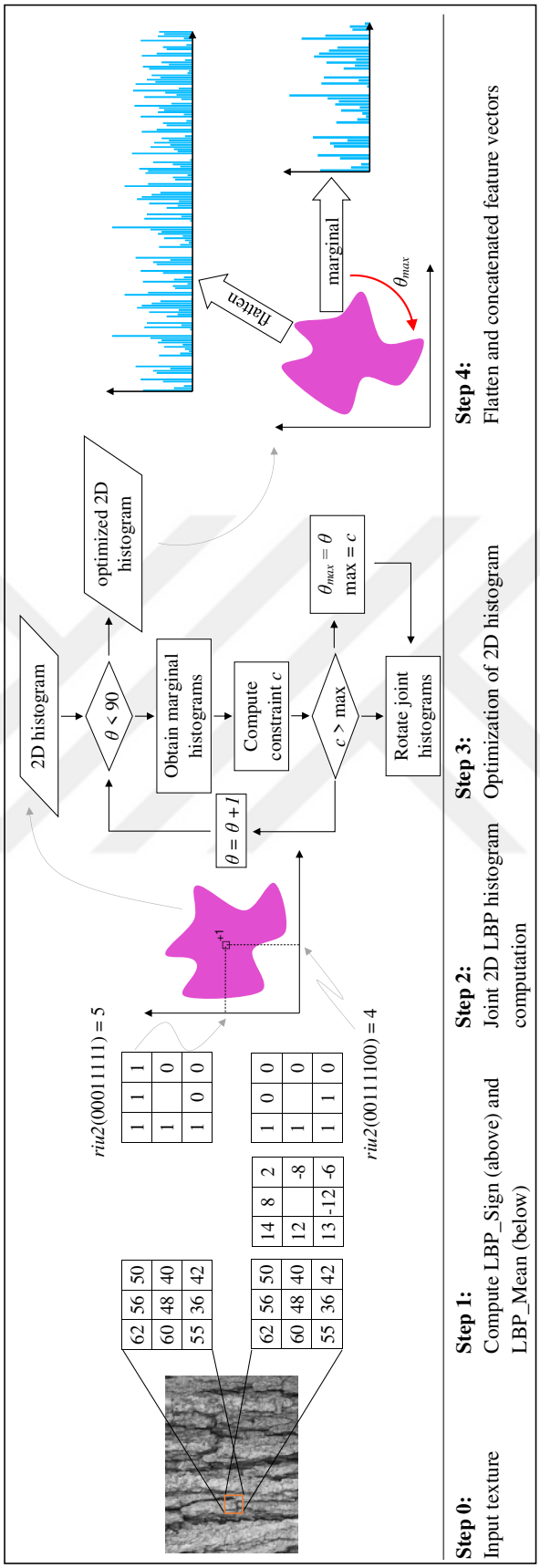


Figure 3.3: *Feature extraction pipeline for proposed method.*

3.4 Statistical Concepts

In order to proceed with optimization constraints we need first to revisit some basic probabilistic and information theory concepts. First we will define a random variable. Random variable is a function which assigns a real value to each possible outcome of the experiment. In this thesis we will work only with discrete random variable whose set of values is finite. Probability mass function (PMF) of random variable X, denoted as p_X, is the probability of the event $X = x$:

$$p(x) = P(\{X = x\}) \tag{3.9}$$

Note that the value of $p(x)$ is always greater than or equal to 0 and $\sum_x p(x) = 1$.

Besides knowing the PMF of X sometimes we want to summarize the random variable X with a single representative number. The most common one is expected value of random variable X defined as:

$$E[X] = \mu_x = \sum_x xp(x) \tag{3.10}$$

$E[X]$ is the weighted average of the all possible values of X. For discrete random variable X expected value is not necessary a value that can be expected to turn up (for more properties on expected value see Table 3.2).

3.4.1 Variance

The second most representative quantity is the variance of random variable X defined as:

$$Var(X) = E[(x - \mu_x)^2] \tag{3.11}$$

where μ_x is mean of discrete random variable X. Variance is the measure of how spread discrete random variable X is around its mean (see Fig.3.4). The expected value and the variance are most associated quantities with random variable X (for more properties on variance see Table 3.2).

3.4.2 Covariance and correlation

The expected value and variance are quantities which provide information only about random variable itself. There are cases where we want to un-

derstand the relation between two random variables X and Y. One of the quantities utilized for this purpose is covariance. The covariance of two random variables X and Y is defined as follows:

$$cov(X, Y) = E[(X - \mu_x)(Y - \mu_y)] \tag{3.12}$$

where μ_x and μ_y are means of X and Y. For $cov(X, Y) = 0$ we say that X and Y are uncorrelated. Positive covariance value indicates that random variable X and Y have same trend, while negative covariance value indicates an opposite trend as shown in Fig. 3.5 (for more properties on covariance see Table. 3.2).

The correlation coefficient ρ of two random variables X and Y is defined as:

$$corr(X, Y) = \rho = \frac{cov(X, Y)}{\sqrt{Var(X)Var(Y)}} \tag{3.13}$$

Correlation may be seen as normalized covariance within range of $[-1, 1]$. High positive or negative correlation indicates high dependence between X and Y while zero correlation implies independence between discrete random variable X and Y. Bertsekas and Tsitsiklis [1] provide more detailed explanation of above concepts.

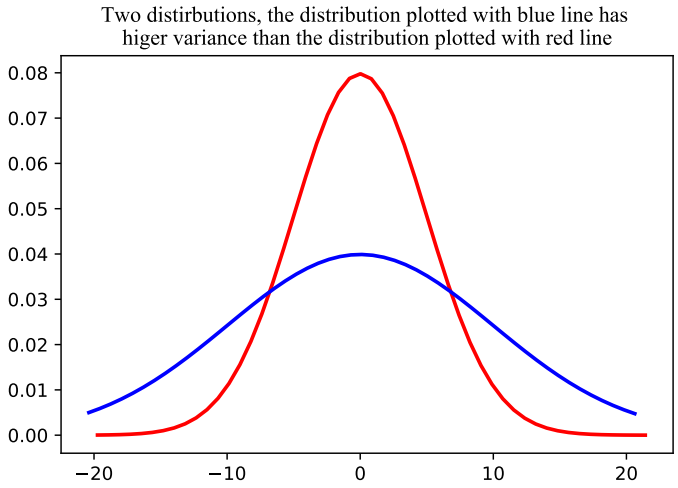


Figure 3.4: *Two different distributions which have different variance values.*

Table 3.2: *Some well known properties of statistical parameters. The proofs of properties for expected value, variance and covariance can be found in [1, 2], while for entropy and mutual information can be found in [3].*

Statical Parameter	**Properties**	**Description**
Probability Mass Function	1. $p(x) \geq 0$ 2. $\sum_x p(x) = 1$	Non-negativity Propertiy Normalization Property
Expected Value	1. $E[aX + b)] = aE[X] + b$ 2. $E[X + Y] = E[X] + E[Y]$	X,Y- random variables a, b - constant
Variance	1. $Var(X) \geq 0$ 2.$Var(X + a) = Var(X)$ 2.$Var(aX) = a^2 Var(X)$	X - random variable a - constant
Covariance	1. $cov(X, X) = Var(X)$ 2.$cov(X, Y) = cov(Y, X)$ 3.$cov(aX, bY) = ab \cdot cov(Y, X)$ 4. $cov(X + a, Y + b) = cov(X, Y))$	X,Y - random variable a,b - constant
Entropy and Joint Entropy	1.$H(X) \geq 0$ 2.$H_a(X) = (\log_a b)H_b(X)$ 3.$H(X, Y) = H(X) + H(X\|Y)$ 4. $H(X\|Y) \neq H(Y\|X)$	X,Y - random variable Changing from one base to another Chain Rule Conditional Entropy
Mutual Information	1.$I(X;Y) = H(X) - H(X\|Y)$ 2. $I(X;Y) = H(Y) - H(Y\|X)$ 3. $I(X;Y) = H(X) + H(Y) - H(X, Y)$ 4. $I(X;Y) = I(Y;X)$ 5. $I(X;Y) \geq 0$ 6. $I(X;X) = H(X) - H(X\|X) = H(X)$	X,Y - random variable Rel. Mutual Information- Entropy Symmetry Property Non-negativity Property

3.4.3 Entropy and joint entropy

One of the most fundamental concept in information theory is entropy. Entropy of a discrete random variable X, denoted as $H(X)$, is defined as following:

$$H(X) = -\sum_{i=1}^{N} p_X(x_i) \log_2 p_X(x_i) \tag{3.14}$$

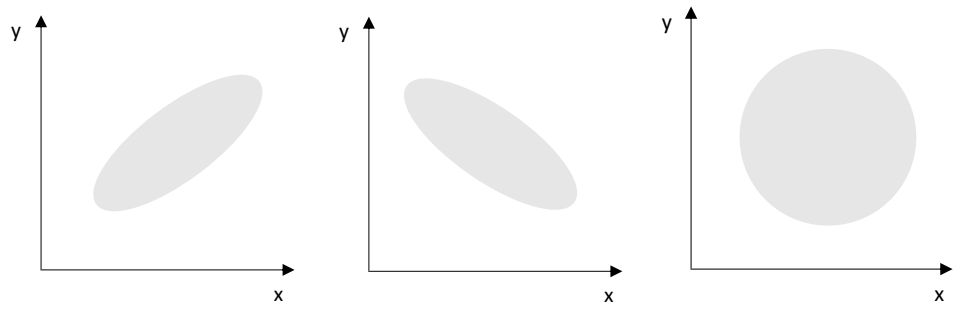


Figure 3.5: *Example of covariance values for three different random variables. Positive (left), negative (middle) and, zero covariance (left).*

where $p_X(x)$ is probability density function of random variable X. The log is to base 2 and entropy is expressed in bits (for change of basis property see Table 3.2). Entropy can be thought as functional of random variable X. Thus its value does not depend on the values taken by random variable X but only on the probability values. $H(x)$ is interpreted as a measure of uncertainty of random variable X. The higher entropy of a discrete random variable X, the less predictable it becomes.

The joint entropy which calculates entropy of pair of random variables X and Y is defined as follows:

$$H(X,Y) = -\sum_{i=1}^{N}\sum_{j=1}^{M} p_{X,Y}(x_i,y_j)\log_2 p_{X,Y}(x_i,y_j) \tag{3.15}$$

where $P(X,Y)$ is joint probability of random variables X and Y.

3.4.4 Mutual information

Another important concept in information theory which provides information on how much one random variable tells about another random variable is Mutual Information (MI). MI between two discrete random variables X and Y is defined as follows:

$$\begin{aligned} I(X,Y) &= H(X) - H(X|Y) \\ &= H(Y) - H(Y|X) \\ &= H(X) + H(Y) - H(X,Y) \end{aligned} \tag{3.16}$$

where $H(X)$ is entropy of random variable X. $H(X|Y)$, conditional entropy, is the uncertainty of discrete random variable X, given the observation of discrete random variable Y. MI, for 2D distributions can also be calculated as follows:

$$I(X,Y) = \sum_{j=1}^{N}\sum_{j=1}^{N} p_{XY}(x_i y_j)\log_2\left(\frac{p_{XY}(x_i y_j)}{p_X(x_i)p_Y(y_j)}\right) \tag{3.17}$$

where $p_X(x)$ and $p_Y(y)$ are marginal distributions over discrete random variable X and Y axes. MI is the reduction of discrete random variable X after observing Y. High MI increases relevance between marginal histograms. From Eq. 3.16 we conclude that mutual information of random variable with itself is equal to entropy of the random variable. This is why entropy

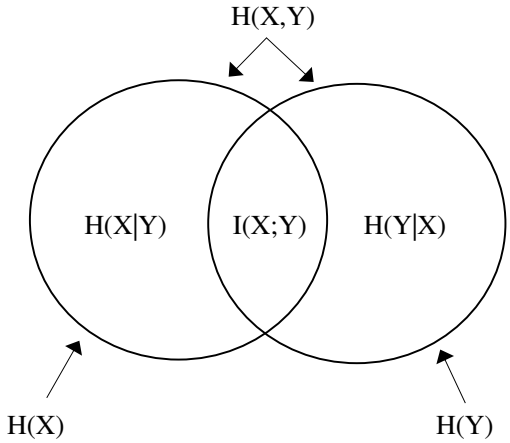


Figure 3.6: *Relation between entropy and mutual information.*

sometimes is referred as self-information [3]. The relation between entropy, joint entropy and mutual is expressed in Venn diagram in Fig. 3.6.

3.5 Optimization Constraints

Here we explain how we employ our statistical constraints in our optimization scenario. As explained in Section 3.3 we rotate 2D LBP distributions $P_{S,M}(s,m)$ by certain angular step $\theta = 1$, project it onto marginal histogram and finally compute statistical constraint. After computing the statistical constraint over entire rotation space we seek for θ which makes maximum or minimum value of statistical constraint. Finally we construct the final feature vector by concatenating the marginal histograms of rotated 2D LBP distribution by θ_{opt}.

- **Maximum Variance Constraint:** look for projections based on following criteria:

$$arg\,max_{\{\theta\,|\,0\leq\theta\leq\frac{\pi}{2}\}}\, var(P_c(\theta)) \tag{3.18}$$

  find θ where we obtain maximum variance of $P_c(\theta)$ where $P_c(\theta)$ is concatenation of marginal histograms at rotation angle θ.

- **Maximum Correlation Constraint:** look for projections based on following criteria:

$$arg\,max_{\{\theta\,|\,0\leq\theta\leq\frac{\pi}{2}\}}\, corr(P_{LBP\_S}(\theta), P_{LBP\_M}(\theta)) \tag{3.19}$$

  finding out θ where we obtain maximum correlation between marginal

histograms $P_{LBP\_S}(\theta)$ and $P_{LBP\_M}(\theta)$.

- **Maximum Entropy Constraint:** We look for projections based on following criteria:

$$arg\,max_{\{\theta|0\leq\theta\leq\frac{\pi}{2}\}}\, H(P_c(\theta)) \tag{3.20}$$

find θ where we obtain maximum entropy of $P_c(\theta)$ where $P_c(\theta)$ is concatenation of marginal histograms at rotation angle θ.

- **Maximum Joint Entropy Constraint:** We look for projections based on following criteria:

$$arg\,max_{\{\theta|0\leq\theta\leq\frac{\pi}{2}\}}\, H(P_{S,M}(s,m),\theta) \tag{3.21}$$

find out θ where we obtain maximum joint entropy of 2-D Histogram.

- **Maximum Mutual Information Constraint:** We look for projections based on following criteria:

$$arg\,max_{\{\theta|0\leq\theta\leq\frac{\pi}{2}\}}\, I(P_{LBP\_S}(\theta), P_{LBP\_M}(\theta)) \tag{3.22}$$

find out θ where we obtain maximum mutual information between marginal histograms P_{LBP_S} and P_{LBP_M} at rotation angle θ.

4. Experimental Results

This section gives quantitative results on the performance of the proposed method. We explain briefly classifiers employed in 4.1. Next, experimental setup and comparative results are explained in sections 4.2 and 4.3 respectively. Finally the discussion about performance analysis of proposed method and execution time analysis are given in sections 4.4 and 4.5.

4.1 Classifiers

In this thesis we use two different classifiers, nearest neighbour and support vector machine.

Nearest neighbour classifier is an example non-parametric models. At the test phase, a test input is assigned to the label of training feature vector which has minimum distance. The most common distance metric to use is Euclidean distance. In this thesis instead of Euclidean distance we use Chi-Square distance metric given in Eq. 4.1. This method is often called as memory based learning.

$$d_{\chi^2}(h,k) = \sum_{i=1}^{N} \frac{(h_i - k_i)^2}{(h_i + k_i)} \tag{4.1}$$

The main drawback with Nearest Neighbour Classifier is that they do not work high dimensional feature vectors [50].

Another well-known classifier is support vector machine (SVM). Since SVMs are more complicated and the aim of thesis is not focusing in details of classifiers no further explanation will be given (for more detail see [51]) We use Dlib machine learning C++ library for implementation of SVM [52].

4.2 Experimental Setup

We perform experiments on five popular texture datasets, i.e. KTH-TIPS2b [22], UIUC [12], UMD [23], Brodatz [24], and Outex_TC_00010 [17]. More detail on attributes of each dataset is provided in Table 4.1. In datasets where test/train scenario is not predefined, we apply 10-fold cross validation.

Table 4.1: *List and properties of the employed datasets in the experiments. The table presents the properties of each dataset including number of classes, image resolutions, and variety of samples in view point, scale and illumination changes.*

Dataset	# Class	#Samples \Class	Total Samples	Sample Size	Test/Train Split?	View-point	Scale	Illumi-nation
UIUC	25	40	1000	320 × 240	No	Yes	Yes	No
UMD	25	40	1000	320 × 240	No	Yes	Yes	No
KTH-TIPS2b	11	432	4.752	200 × 200	Yes	No	Yes	Yes
Brodatz	111	9	999	215 × 215	No	No	Yes	No
Outex_TC10	24	180	4.320	128 × 128	Yes	No	No	No

Outex_TC_00010 dataset contains 24 different texture categories. Each category includes 20 samples for each ten rotation angles (0°, 5°, 10°, 15°, 30°, 45°, 60°, 75°, and 90°). Texture samples with 0° rotation angle are utilized for train phase and other for test phase.

Texture samples in UMD and UIUC datasets share common properties like significant scale and view point changes, arbitrary rotations, and uncontrolled illumination conditions. Additionally, textures in UMD dataset has resolution four times of textures in UIUC dataset. KTH-TIPS2b dataset is propsoed in extension of CURet dataset. Textures consist of four physical samples, three different viewing angles, four illumination condition and nine different scales.

Note that we need to do zero padding to 2D histograms in advance to rotation to prevent them exceeding the range (see Table 4.2). Thus rotated 2D distributions will be larger than original ones. The implementation of algorithm is carried out in C++ Language, Intel i5 3.5 GHz CPU, 16GB RAM.

Table 4.2: *2D histogram dimensions for original and rotated versions*

Parameters	Histograms	
(P, R)	Original 2D Histogram	Rotated 2D Histogram
(8, 1)	10 × 10	16 × 16
(16, 2)	18 × 18	26 × 26
(24, 3)	26 × 26	40 × 40

4.3 Methods in Comparison

The naive methods build LBP feature vectors by either concatenating marginal histogram S_M or flattening the histogram in one vector $S \diagdown M$. We optimize naive 2D LBP feature vectors with respect to the proposed optimization methods and take both flattened and concatenated forms of them into consideration. Although it cannot help to improve the accuracy, we also provide results based on axes obtained from PCA denoted by S_M^{PCA}. In order to keep notations concise, we will use abbreviations in Table 3.1 in the rest of the paper.

In the experiments, we examine the performances of all the methods in three main comparisons. In the first experiment, we compare methods which obtain the eventual feature histogram by concatenating two marginal histograms and present the results in Table 4.4. In the second experiment, we assess the flattening approach for the same 2D LBP methods and present the results in Table 4.5. In Table 4.6 we combine best results from Table 4.4 and Table 4.5 in order to observe the overall results. For each dataset and parameter, considering both classifiers, the highest score is shadowed, and those scores which are within 1% of the highest are boldfaced. We evaluate the methods with respect to the number of bold and highlighted scores across all datasets for all parameters. In addition, we provided number of bold results for each classifier and sort the entire list with respect to the total number of bolds in order to demonstrate the eventual performance of each method.

4.4 Performance Analysis

Table 4.4 and Table 4.5 provide the experimental results carried out in aforementioned datasets for proposed methods and corresponding naive approach, S_M and $S \diagdown M$, respectively.

In Table 4.4 we see that the optimized feature vectors with mutual information S_M^{MI} and variance S_M^{V} takes the first and the second places, respectively, both with SVM classifier. The naive approaches without optimization follow them in the next two rows. S_M^{PCA} gives the lowest scores for all metrics and in most cases performs even worse than S_M for all datasets. Other methods, i.e. S_M^{E}, S_M^{JE}, S_M^{C}, S_M^{V} pro-

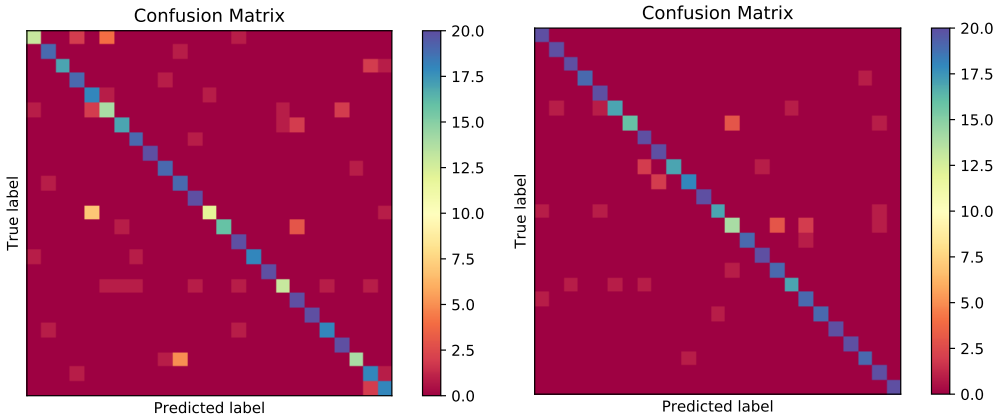


Figure 4.1: *Confusion matrices for experimental results obtained in UIUC dataset with naïve (left) and optimized (right) feature vectors of* $(P, R) = (24, 3)$.

vided a mediocre performance in the experiments. In terms of accuracy rates, S_M^{MI} shows higher results for both classifiers in UIUC, UMD and KTHTIPS2b datasets. For Brodatz dataset, S_M^{MI} is within 1% range for all parameters in both classifiers while S_M could provided the best result. S_M^{MI} performed relatively worse in Outex TC10 dataset compare to its performance in other dataset. Briefly, S_M^{MI} could provide the best or within 1% accuracy results when combined with SVM classifier.

In Table 4.5 we present experimental results that we obtain via flattening 2D LBP distributions. According to the experiments, accuracy results of $S \diagdown M$ and $S \diagdown M^{MI}$ were very close. In the recognition tasks that we run on five datasets and three different parameters for each; $S \diagdown M$ and $S \diagdown M^{MI}$ could perform within the 1% range to the best results for 9 and 8 times, respectively. Similar to its performance in marginal feature vector generation approach, S_M^{PCA} could not provide a promising recognition accuracy via flattening. To summarize, $S \diagdown M$ and $S \diagdown M^{MI}$ provided comparable results where $S \diagdown M$ performed slightly better for the experiments that we construct a higher dimensional feature vector compare to concatenating marginal histograms.

We compare the results obtained from both feature vector construction methods in Table 4.6 which is composed with the best results of Table 4.4 and Table 4.5. In addition, since dimensions of the feature vectors in this table vary, we also provide them for all parameter settings to ease the comparison. In overall results, S_M^{MI} and S_M performed the same, i.e. with three #Best and nine #Bold scores both with SVM classifier. However,

S_M^{MI} takes the first place due to the fact that it achieved the same result with a significantly lower dimensional feature vector. When we investigate the breakdown of the algorithms into datasets, we see that S_M^{MI} performs better in UIUC, UMD and KTHTIPS2b datasets which have more challenging samples due to strong rotation and illumination changes.

In Fig. 4.1 and Fig. 4.2 we present confusion matrices of naïve and optimized feature vectors of S_M^{MI} for UIUC and UMD datasets, respectively. Note that the confusion matrices obtained by the optimization method using mutual information constraint look more tidy with a more stable diagonal values compare to the naïve ones. In Table 4.7 several samples of textures are shown with corresponding naïve and optimized 2D histograms, respectively.

4.5 Execution Time Analysis

Here we aim to give some information regarding the analysis of execution time of the proposed method. For this purpose, we use textures from Brodatz [24] dataset with dimension of 215×215.

In Table 4.3 there are shown average execution times in millisecond for proposed methods including all immediate steps. As shown in table LBP_Sign have lower time cost that LBP_Magn. This is from the fact that LBP_Magn needs one more step to calculate threshold than LBP_Sign which use central pixel as threshold value. As expected, the accumulation of histogram has the lowest time cost.

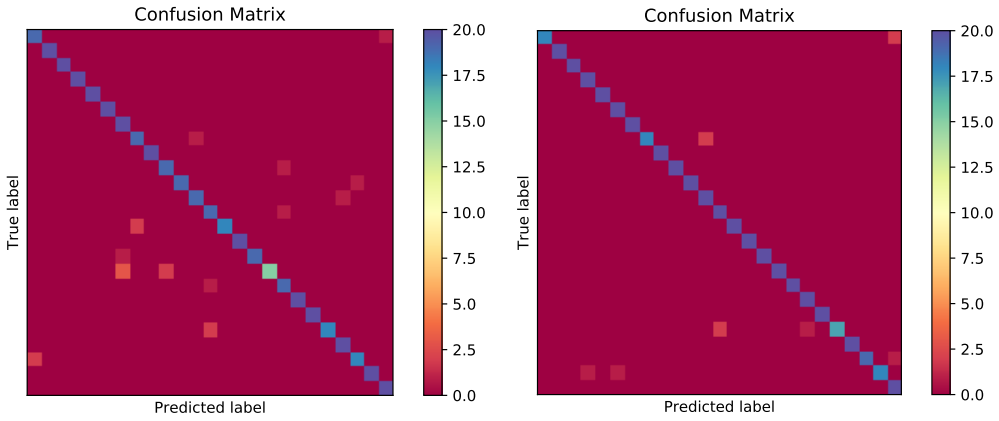


Figure 4.2: *Confusion matrices for experimental results obtained in UMD dataset with naïve (left) and optimized (right) feature vectors of* $(P, R) = (16, 2)$.

Table 4.3: *Average execution times of the proposed method including all intermediate steps. Time is given in milliseconds.*

Method	P	R	LBP Sign	LBP Magn	Histogram	Optimization	Total
S_M	8	1	35.38 ms	39.08 ms	0.18 ms	–	74.64 ms
S_M	16	2	67.57 ms	74.34 ms	0.18 ms	–	141.67 ms
S_M	24	3	98.44 ms	107.98 ms	0.21 ms	–	206.63 ms
S_M^{MI}	8	1	35.15 ms	39.04 ms	0.17 ms	0.99 ms	75.45 ms
S_M^{MI}	16	2	67.39 ms	73.94 ms	0.21 ms	2.07 ms	143.61 ms
S_M^{MI}	24	3	99.46 ms	108.73 ms	0.18 ms	3.32 ms	211.69 ms
S_M^{V}	8	1	36.60 ms	39.72 ms	0.18 ms	0.58 ms	77.08 ms
S_M^{V}	16	2	67.22 ms	74.32 ms	0.17 ms	1.31 ms	143.02 ms
S_M^{V}	24	3	98.03 ms	106.62 ms	0.25 ms	2.42 ms	207.32 ms
S_M^{E}	8	1	35.48 ms	39.43 ms	0.19 ms	0.71 ms	75.81 ms
S_M^{E}	16	2	67.75 ms	74.25 ms	0.18 ms	1.56 ms	143.74 ms
S_M^{E}	24	3	98.10 ms	106.72 ms	0.18 ms	2.70 ms	207.70 ms
S_M^{JE}	8	1	35.42 ms	39.08 ms	0.18 ms	0.96 ms	75.64 ms
S_M^{JE}	16	2	67.45 ms	73.96 ms	0.18 ms	1.99 ms	143.58 ms
S_M^{JE}	24	3	100.19 ms	108.64 ms	0.18 ms	3.29 ms	212.30 ms
S_M^{C}	8	1	35.34 ms	38.80 ms	0.17 ms	0.59 ms	74.90 ms
S_M^{C}	16	2	67.49 ms	73.96 ms	0.18 ms	1.32 ms	142.95 ms
S_M^{C}	24	3	98.37 ms	107.63 ms	0.20 ms	2.43 ms	208.63 ms

On the other hand S_M^V and S_M^C tend to have the lowest execution time. This is expected since the calculation of variance and correlation require less operations than other statistical constraints. Mutual information S_M^{MI} constraint despite its efficient performance tends to have slowest computation speed among other constraints. This is due to fact that mutual information requires to compute the entropy of marginal histograms separately and compute the joint entropy. Nevertheless, taking in consideration the computation time of other steps, mutual information constitutes approximately 1.5% of total computation time. Generally, we conclude that proposed optimization step has not a significant effect in total computation time.

Table 4.4: *Experimental results for marginal approaches. For each feature and classifier pair (each row), the best score is shadowed, and those scores which are within the 1% of the highest are boldfaced. We evaluate our methods according to the number of the bold scores and sort the table in descending order. In addition, we also underline the best result for each dataset to indicate the highest performance regardless the feature parameters.*

Method	Classifier	UIUC			UMD			KTH TIPS 2B			Brodatz			Outex_TC10			#Best	#Bold
		(8,1)	(16,2)	(24,3)	(8,1)	(16,2)	(24,3)	(8,1)	(16,2)	(24,3)	(8,1)	(16,2)	(24,3)	(8,1)	(16,2)	(24,3)		
S_M^{MI}	SVM	**89.72**	**95.20**	**93.40**	**98.02**	**98.30**	**96.70**	**70.64**	**71.25**	**66.95**	**83.33**	**87.70**	**88.40**	**91.69**	**97.44**	**98.59**	8	15
S_M^{V}	SVM	**89.54**	86.40	**92.30**	**97.92**	95.60	**96.04**	**69.94**	69.46	**66.78**	82.19	78.03	83.27	**92.31**	96.14	**98.04**	1	8
S_M	NN	70.60	81.90	84.72	95.22	95.56	95.56	61.27	63.10	62.15	82.83	**87.82**	**88.64**	**91.48**	**96.45**	**98.25**	2	5
S_M	SVM	86.10	92.88	90.74	**98.08**	97.04	93.60	66.99	68.32	65.35	**83.57**	73.82	71.08	89.81	**97.31**	**98.61**	3	4
S_M^{JE}	SVM	85.67	92.20	91.24	96.18	**97.30**	**96.44**	68.57	69.12	**67.29**	75.61	83.25	80.81	86.61	96.43	97.31	1	3
S_M^{E}	SVM	88.58	92.64	87.14	**97.68**	96.86	**96.42**	**70.23**	69.29	65.46	80.07	80.67	81.12	89.42	95.20	96.61	0	3
S_M^{MI}	NN	77.02	86.46	85.90	96.12	96.02	95.38	62.79	64.72	65.67	**84.21**	**87.71**	**87.71**	90.20	96.25	97.03	0	3
S_M^{C}	SVM	81.72	92.40	90.86	97.02	96.46	93.46	67.98	68.38	65.29	73.61	83.67	86.48	89.71	94.45	**98.30**	0	1
S_M^{C}	NN	71.81	79.88	84.14	94.32	94.18	95.48	61.46	62.85	62.11	77.68	85.34	**88.37**	88.93	93.61	96.67	0	1
S_M^{PCA}	SVM	46.04	63.52	56.36	69.30	72.66	72.98	54.10	61.95	55.40	50.63	58.04	60.01	76.40	75.93	84.42	0	0
S_M^{V}	NN	77.01	81.62	84.82	95.86	94.30	94.58	62.66	63.01	65.06	83.10	81.17	86.59	90.41	94.92	96.87	0	0
S_M^{E}	NN	76.35	83.82	81.84	95.32	95.26	93.46	61.73	64.05	64.28	81.50	83.67	83.94	89.42	93.22	95.72	0	0
S_M^{JE}	NN	71.01	80.88	83.72	93.58	94.30	94.74	61.78	62.09	62.81	77.38	84.48	83.70	85.57	95.52	96.14	0	0
S_M^{PCA}	NN	42.74	60.88	61.34	74.00	79.94	79.36	50.27	57.00	56.86	53.27	66.24	67.35	74.73	75.15	87.21	0	0

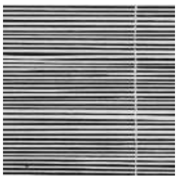
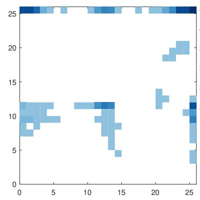
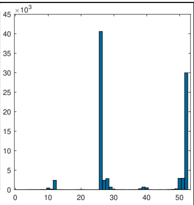
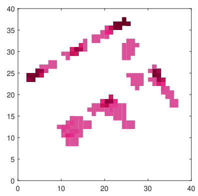
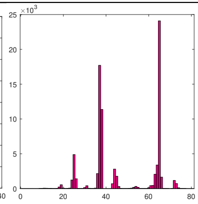
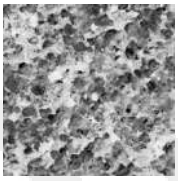
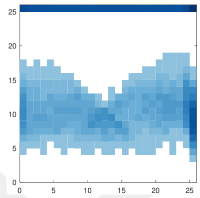
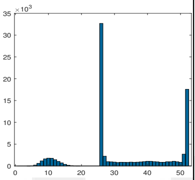
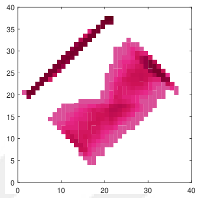
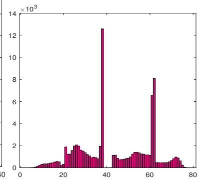
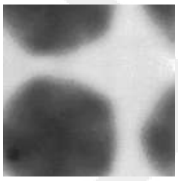
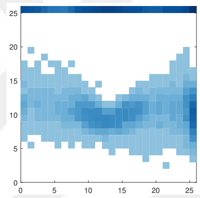
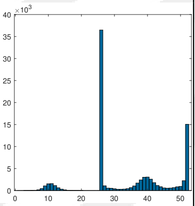
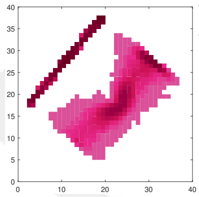
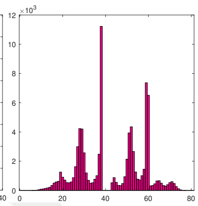
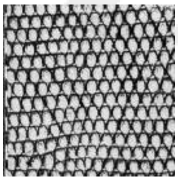
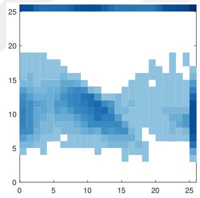
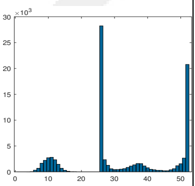
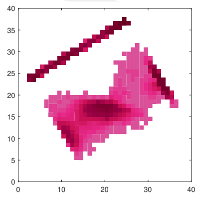
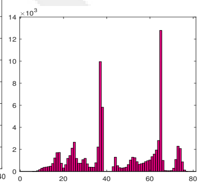
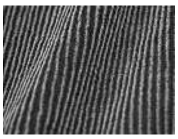
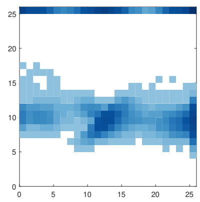
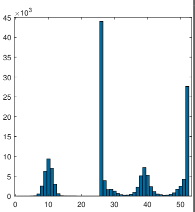
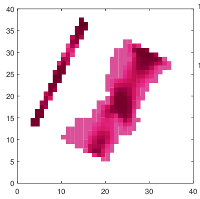
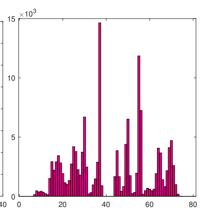
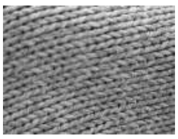
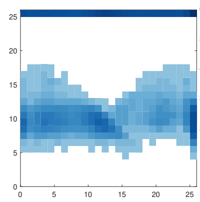
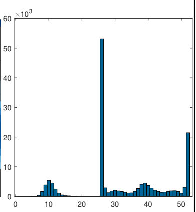
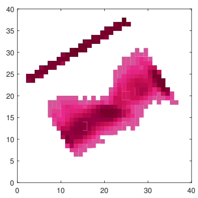
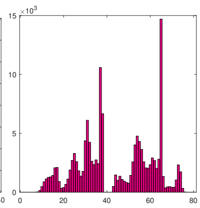
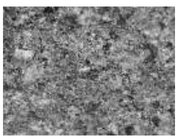
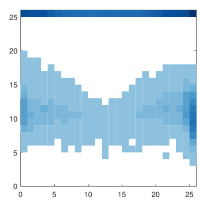
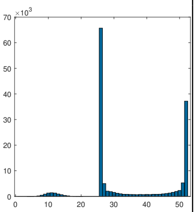
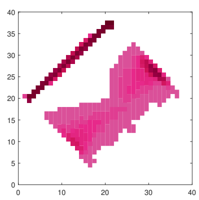
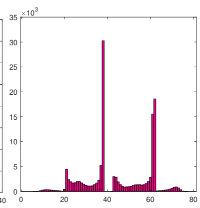
Table 4.5: *Experimental Results for flattening approaches. The meanings of bold, underlined and shaded indicators are the same with the Table 4.4.*

Method	Classifier	UIUC			UMD			KTH TIPS 2B			Brodatz			Outex_TC10			#Best	#Bold
		(8,1)	(16,2)	(24,3)	(8,1)	(16,2)	(24,3)	(8,1)	(16,2)	(24,3)	(8,1)	(16,2)	(24,3)	(8,1)	(16,2)	(24,3)		
$S \setminus M$	NN	82.50	90.14	91.76	**97.06**	**97.46**	**96.56**	65.02	65.17	63.90	**88.19**	**92.95**	**93.00**	**92.77**	**98.04**	**98.58**	4	9
$S \setminus M^{MI}$	SVM	**90.46**	**95.48**	**93.52**	96.88	**98.30**	**95.58**	**71.63**	**70.26**	66.14	85.21	90.07	89.42	91.87	96.79	**98.17**	3	8
$S \setminus M$	SVM	**90.92**	**95.06**	**92.88**	**97.94**	**97.76**	94.34	**72.55**	**70.81**	63.08	75.06	79.36	86.93	91.38	**97.57**	**98.64**	3	8
$S \setminus M^{MI}$	NN	80.66	89.80	90.56	95.46	97.14	**95.58**	64.01	64.83	64.09	86.60	**91.98**	90.93	**93.30**	**97.52**	**98.15**	1	5
$S \setminus M^{C}$	SVM	82.86	**95.10**	**92.88**	94.80	**98.02**	94.20	68.81	**71.25**	63.29	75.16	87.14	90.91	90.59	94.84	**98.22**	1	5
$S \setminus M^{V}$	SVM	**90.76**	86.46	**92.58**	**97.10**	95.52	**96.42**	70.91	68.79	65.69	83.87	79.21	84.83	91.87	96.57	**97.99**	0	5
$S \setminus M^{JE}$	SVM	88.62	93.48	**92.70**	96.38	**97.80**	**96.62**	70.13	70.22	**67.29**	79.74	86.51	85.16	87.23	96.45	97.57	2	4
$S \setminus M^{V}$	NN	80.76	84.40	89.58	95.28	95.46	**95.56**	62.89	64.03	63.55	84.00	85.31	88.81	**92.68**	96.53	**97.73**	0	3
$S \setminus M^{E}$	SVM	89.04	92.80	87.34	**96.80**	**97.30**	94.00	70.28	69.73	65.00	81.20	82.80	82.08	90.52	95.00	96.40	0	2
$S \setminus M^{JE}$	NN	77.98	88.54	90.04	95.14	**96.68**	**95.82**	63.84	63.90	64.01	82.74	90.13	88.72	87.16	96.71	97.18	0	2
$S \setminus M^{C}$	NN	73.58	88.82	91.16	92.58	96.42	**96.44**	62.52	64.43	63.84	75.52	89.94	**92.43**	89.92	94.92	97.60	0	2
$S \setminus M^{PCA}$	SVM	45.64	57.44	53.20	75.74	74.00	62.58	55.59	59.59	54.31	52.16	58.58	55.92	75.88	79.14	82.89	0	0
$S \setminus M^{E}$	NN	79.96	87.38	86.18	94.90	96.14	94.48	63.13	63.36	63.72	82.34	87.43	86.74	90.57	94.76	96.87	0	0
$S \setminus M^{PCA}$	NN	40.66	49.92	53.30	70.18	75.58	73.88	48.61	54.69	51.05	49.03	60.27	61.62	71.17	77.91	82.13	0	0

Table 4.6: *Composition of the best results of Table 4.4 and Table 4.5. The meanings of bold, underlined and shaded indicators are the same with the Table 4.4.*

Method	Classifier	UIUC			UMD			KTH TIPS 2B			Brodatz			Outex_TC10			Feature Dimension			#Best	#Bold
		(8,1)	(16,2)	(24,3)	(8,1)	(16,2)	(24,3)	(8,1)	(16,2)	(24,3)	(8,1)	(16,2)	(24,3)	(8,1)	(16,2)	(24,3)	(8,1)	(16,2)	(24,3)		
S_M^{MI}	SVM	89.72	**95.20**	**93.40**	**98.02**	**98.30**	**96.70**	70.64	**71.25**	**66.95**	83.33	87.70	88.40	91.69	**97.44**	**98.59**	32	56	80	3	9
$S \setminus M$	SVM	**90.92**	**95.06**	**92.88**	**97.94**	**97.76**	94.34	**72.55**	**70.81**	63.08	75.06	79.36	86.93	91.38	**97.57**	**98.64**	100	324	676	3	9
$S \setminus M$	NN	82.50	90.14	91.76	97.06	**97.46**	**96.56**	65.02	65.17	63.90	**88.19**	**92.95**	**93.00**	**92.77**	**98.04**	**98.58**	100	324	676	4	8
$S \setminus M^{MI}$	SVM	**90.46**	**95.48**	**93.52**	96.88	**98.30**	95.58	**71.63**	**70.26**	66.14	85.21	90.07	89.42	91.87	96.79	**98.17**	256	784	1600	3	7
$S \setminus M^{C}$	SVM	82.86	**95.10**	**92.88**	94.80	**98.02**	94.20	68.81	**71.25**	63.29	75.16	87.14	90.91	90.59	94.84	**98.22**	256	784	1600	1	5
S_M^{V}	SVM	89.54	86.40	92.30	**97.92**	95.60	**96.04**	69.94	69.46	**66.78**	82.19	78.03	83.27	**92.31**	96.14	**98.04**	32	56	80	0	5
$S \setminus M^{JE}$	SVM	88.62	93.48	**92.70**	96.38	**97.80**	**96.62**	70.13	70.22	**67.29**	79.74	86.51	85.16	87.23	96.45	97.57	256	784	1600	1	4
$S \setminus M^{MI}$	NN	80.66	89.80	90.56	95.46	97.14	95.58	64.01	64.83	64.09	86.60	**91.98**	90.93	**93.30**	**97.52**	**98.15**	256	784	1600	1	4
S_M^{JE}	SVM	85.67	92.20	91.24	96.18	**97.30**	**96.44**	68.57	69.12	**67.29**	75.61	83.25	80.81	86.61	96.43	97.31	32	56	80	1	3
S_M	SVM	86.10	92.88	90.74	**98.08**	97.04	93.60	66.99	68.32	65.35	83.57	73.82	71.08	89.81	**97.31**	**98.61**	20	36	52	1	3
S_M	NN	70.60	81.90	84.72	95.22	95.56	95.56	61.27	63.10	62.15	82.83	87.82	88.64	91.48	96.45	**98.25**	20	36	52	0	1

Table 4.7: *Several textures samples with corresponding naïve and optimized 2D histograms, respectively.*

Texture	Naïve 2D Histogram	Naïve Feature Vector	Optimized 2D Histogram	Optimized Feature Vector
				
				
				
				
				
				
				

5. CONCLUSION

In this thesis we propose an improvement method for 2D LBP approaches to extract more discriminative feature vectors. Our method suggests modifying 2D LBP distribution before constructing the feature vectors via either concatenation of marginal histograms or flattening the whole distribution. Prior to the feature vector extraction, we seek for new projection axes by applying a certain transformation and utilize several constraints i.e. variance, correlation, entropy, joint entropy, and mutual information as stopping criteria. We perform a comprehensive set of experiments including five well known texture datasets in the literature and two classification methods.

According to the results, the proposed method outperforms naïve marginal approach in almost all experiments and provides the best results for mutual information as the optimization constraint. In addition, it provides comparable results in flattening approach where the dimension of resulted feature vectors are quadratically proportional to the size of 2D LBP histogram. In comparison of all marginal and flattening approaches, we see that the optimized marginal features provide promising accuracy values even with lower dimensional feature representations than higher dimensional vectors obtained via flattening. In all experiments, we show that the proposed optimization algorithm boosts the recognition accuracies regardless the number of neighbor and radius parameters of the employed LBP feature.

REFERENCES

[1] D. P. Bertsekas and J. N. Tsitsiklis, *Introduction to Probability*. Athena Scientific, 2008.

[2] R. W. Hamming, *The Art of Probability: For Scientists and Engineers*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1991.

[3] T. M. Cover and J. A. Thomas, *Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing)*. New York, NY, USA: Wiley-Interscience, 2006.

[4] T. Ahonen, A. Hadid, and M. Pietikänen, "Face recognition with local binary patterns," in *European Conference on Computer Vision (ECCV)*, pp. 469–481, 2004.

[5] B. Zhang, Y. Gao, S. Zhao, and J. Liu, "Local derivative pattern versus local binary pattern: Face recognition with high-order local pattern descriptor," *IEEE Transactions in Image Processing*, vol. 19, no. 2, pp. 533 – 544, 2010.

[6] L. Liu, P. Fieguth, G. Zhao, M. Pietikänen, and D. Hu, "Extended local binary patterns for face recognition," *Information Science*, vol. 358-359, pp. 56–72, 2016.

[7] C. Shan and T. Gritti, "Learning discriminative lbp-histogram bins for facial expression recognition," in *British Machine Vision Conference (BMVC)*, 2008.

[8] X. Wang, T. X. Han, and S. Yan, "An hog-lbp human detector with partial occlusion handling," in *International Conference on Computer Vision (ICCV)*, pp. 32–39, 2009.

[9] X. Yi and M. Eramian, "Lbp-based segmentation of defocus blur," *IEEE Transactions in Image Processing*, vol. 25, no. 4, pp. 1626 – 1638, 2016.

[10] T. Leung and J. Malik, "Representing and recognizing the visual appearance of materials using three-dimensional textons," *International*

Journal of Computer Vision, vol. 43, pp. 29–44, 2008.

[11] M. Varma and A. Zisserman, "A statistical approach to material classification using image patch exemplars," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 21, no. 11, pp. 2032 – 2047, 2009.

[12] S. Lazebnik, C. Schmid, and J. Ponce, "A sparse texture representation using local affine regions," *IEEE Transactions on Patterns Analysis and Machine Intelligence*, vol. 27, no. 8, pp. 1265 – 1278, 2005.

[13] B. Manjunath and W. Ma, "Texture features for browsing and retrieval of image data," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 8, pp. 837 – 842, 1996.

[14] L. Liu and P. Fieguth, "Texture classification from random features," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 83, pp. 574 – 586, 2012.

[15] D. Lowe, "Distinctive image features from scale invariant points," *International Journal on Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[16] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 886–893, 2005.

[17] O. Timo, P. Matti, and M. Topi, "Multiresulution gray-scale and rotation invariant texture classification with local binary patterns," *IEEE Transactions on Patterns Analysis and Machine Intelligence*, vol. 24, no. 7, pp. 971 – 987, 2002.

[18] L. Liu, P. Fieguth, Y. Guo, Z. Wang, and M. Pietikänen, "Local binary features for texture classification: Taxonomy and experimental study," *Pattern Recognition*, vol. 62, pp. 135–160, 2016.

[19] Z. Zhang, S. Liu, X. Mei, B. Xiao, and L. Zheng, "Learning completed discriminative local features for texture classification," *Pattern Recognition*, vol. 67, pp. 263–275, 2017.

[20] Z. Guo, L. Zhang, and D. Zhang, "A completed modeling of local binary pattern operator for texture classification," *IEEE Transactions in Image Processing*, vol. 19, no. 6, pp. 1657 – 1663, 2010.

[21] M. Pietikänen, A. Hadid, G. Zhao, and T. Ahonen, *Computer Vision Using Local Binary Patterns*. Springer, 2011.

[22] P. Mallikarjuna, M. Fritz, A. Targhi, E. Hayman, B. Caputo, and J. Eklundh, "The kth-tips and kth-tips2 databases," 2006.

[23] L. Liu, P. Feiguth, X. Wang, M. Pietikänen, and D. Hu, "A new texture descriptor using multifractal analysis in multi-orientation wavelet pyramid," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 161 – 168, 2010.

[24] P. Brodatz, "Textures: A photographic album for artists and designers," 1966.

[25] L. Liu, L. Zhao, Y. Long, G. Kuang, and P. Fieguth, "Extended local binary patterns for texture classification," *Image and Vision Computing*, vol. 30, no. 2, pp. 86–99, 2012.

[26] X. Tan and B. Triggs, "Enhanced local texture feature sets for face recognition under difficult lighting conditions," *IEEE Transactions in Image Processing*, vol. 19, no. 6, pp. 1635 – 1650, 2010.

[27] V. Ojansivu and J. Heikkilä, "Blur insensitive texture classification using local phase quantization," in *International Conference on Image and Signal Processing*, 2008.

[28] V. Ojansivu, E. Rahtu, and J. Heikkila, "Rotation invariant local phase quantization for blur insensitive texture analysis," in *International Conference on Pattern Recognition (ICPR)*, 2008.

[29] X. Hong, G. Zhao, M. Pietikänen, and X. Chen, "Combining lbp difference and feature correlation for texture description," *IEEE Transactions in Image Processing*, vol. 23, no. 6, pp. 2557 – 2568, 2014.

[30] Y. Zhao, D. Huang, and W. Jia, "Completed local binary count for rotation invariant texture classification," *IEEE Transactions in Image Processing*, vol. 21, no. 10, pp. 4492 – 4497, 2012.

[31] S. Liao, M. Law, and A. Chung, "Dominant local binary patterns for texture classification," *IEEE Transactions in Image Processing*, vol. 18, no. 5, pp. 1107 – 1118, 2009.

[32] F. Bianconi, E. Gonzalez, and A. Fernandez, "Dominant local binary patterns for texture classification: Labelled or unlabelled ?," *Pattern Recognition Letters*, vol. 65, pp. 8–14, 2015.

[33] A. Fathi and A. R. N. Nillchi, "Noise tolerant local binary operator for efficient texture analysis," *Pattern Recognition Letters*, vol. 33, pp. 1093–1100, 2012.

[34] H. Zhou, R. WANG, and C. Wang, "A novel extended local-binary-pattern operator for texture recognition," *Information Sciences*, vol. 178, pp. 4314–4325, 2008.

[35] Y. Guo, G. Zhao, and M. Pietikänen, "Discriminative feature for texture description," *Pattern Recognition*, vol. 45, pp. 3834–3843, 2012.

[36] L. Liu, S. Lao, P. W. Fieguth, Y. Guo, X. Wang, and M. Pietikänen, "Median robust extended local binary pattern for texture classification," *IEEE Transactions in Image Processing*, vol. 25, no. 3, pp. 1368 – 1381, 2016.

[37] Z. Guo, X. Wang, J. Zhuo, and J. You, "Robust texture image representation by scale selective local binary patterns," *IEEE Transactions in Image Processing*, vol. 25, no. 2, pp. 687 – 699, 2016.

[38] F. Lu and J. Huang, "An improved local binary pattern operator for texture classification," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2016.

[39] M. Cote and A. Albu, "Robust texture classification by aggregating pixel-based lbp statisticss," *IEEE Signal Processing Letters*, vol. 22, no. 11, pp. 2102 – 2106, 2015.

[40] L. Liu, P. Feiguth, X. Wang, M. Pietikänen, and D. Hu, "Evaluation of lbp and deep texture descriptors with a new robustness benchmark," in *European Conference on Computer Vision (ECCV)*, 2016.

[41] M. Xi, L. Chen, D. Polajnar, and W. Tong, "Local binary pattern network : A deep learning approach for face recognition," in *International Conference on Image Processing (ICIP)*, 2016.

[42] F. Juefei-Xu, V. N. Boddeti, and M. Savvides, "Local binary convolu-

tional neural networks," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[43] T. Ahonen, A. Hadid, and M. Pietikänen, "Face description with local binary patterns: Application to face recognition," *IEEE Transactions on Patterns Analysis and Machine Intelligence*, vol. 28, no. 12, pp. 2037–2041, 2006.

[44] D. Huang, C. Shan, M. Ardabilian, Y. Wang, and L. Chen, "Local binary patterns and its applications to facial image analysis: A survey," *IEEE Transactions on Systems, Man and Cybernetics - Part C (Applications and Reviews)*, vol. 41, no. 26, pp. 765 – 781, 2011.

[45] Y. Mu, S. Yan, Y. Lu, T. Huang, and B. Zhou, "Discriminative local binary patterns for human detection in personal album," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1–8, 2008.

[46] A. Roy and S. Marcel, "Haar local binary pattern feature for fast illumination invariant face detection," in *British Machine Vision Conference (BMVC)*, 2009.

[47] L. Nanni, A. Lumini, and S. Brahnam, "Local binary patterns variants as texture descriptors for medical image analysis," *Artificial Intelligence in Medicine*, vol. 49, pp. 117–125, 2010.

[48] W. Li, C. Chen, H. Su, and Q. Du, "Local binary patterns and extreme learning machine for hyperspectral imagery classification," *IEEE Transactions in Geoscience and Remote Sensing*, vol. 53, no. 7, pp. 3681 – 3693, 2015.

[49] A. Satpathy, X. Jiang, and H. Lung, "Lbp based edge-texture features for object recognition," *IEEE Transactions in Image Processing*, vol. 23, no. 5, pp. 1953 – 1964, 2014.

[50] K. P. Murphy, *Machine learning : a probabilistic perspective*. MIT Press, 2013.

[51] E. Alpaydin, *Introduction to Machine Learning*. The MIT Press, 2nd ed., 2010.

[52] D. E. King, "Dlib-ml: A machine learning toolkit," *Journal of Machine*

Learning Research, vol. 10, pp. 1755–1758, 2009.

CURRICULUM VITAE

Name Surname : Llukman Cerkezi
Foreign Languages: : Turkish, English, Bosnian
Place of Birth and Year : Kosovo / 1993
E-mail : llukmancerkezi@gmail.com

Education ve Professional Background:

- 2016- Present, Eskisehir Technical University, Electrical and Electronics Engineering, Eskisehir
- 2012-2016, Sakarya University, Electrical and Electronics Engineering, Sakarya
- 2011-2012, Gazi University, Turkish Language Learning, Research and Application Center, Ankara
- 2008-2011, Eqrem Çabej, Mathematics and Natural Sciences High School, Vushtrri

Publications:

- Ll. Cerkezi, C. Topal. Gender recognition with uniform local binary patterns. Signal Processing and Communications Applications Conference, 1-4, 2018.
- Ll. Cerkezi, G. Cetinel. RDWT and SVD based secure digital image watermarking using ACM. Signal Processing and Communications Applications Conference, 149-152,2016
- G. Cetinel, Ll. Cerkezi. Robust Chaotic Digital Image Watermarking Scheme based on RDWT and SVD. Int. Journal of Image, Graphics and Signal Processing, 8, 2016
- G. Cetinel, Ll. Cerkezi.Chaotic digital image watermarking scheme based on DWT and SVD. Int. Conference on Electrical and Electronics Engineering, 251-255, 2015