A Robust and Compact Descriptor Based on Center-Symmetric LBP

Jinwei Xiao Gangshan Wu
State Key Laboratory for Novel Software Technology, Nanjing University, China
E-mail: xjw87610@gmial.com gswu@nju.edu.cn

Abstract

Center-symmetric local binary pattern (CS-LBP) is a novel texture feature which utilizes texture to describe the local regions. It combines the good property of local binary pattern (LBP) and SIFT. It has been extended to a region descriptor and achieved promising performance in many applications. However, it is sensitive to noise and less efficient due to its high dimensional descriptor vector. Due to these, we propose a novel descriptor based on CS-LBP operator denoted as PCA-CS-LBP. Our proposed descriptor achieves better noise robustness using the difference of pixels instead of the rough comparing of pixels. Besides, PCA is employed and applied to generate a more compact representation. Comparisons between our descriptor and standard CS-LBP descriptor are given on a standard image matching dataset. Experimental results show that our descriptor is outperforms the standard CS-LBP descriptor in most cases.

1 Introduction

More recently, the local binary pattern (LBP) has received considerable attentions in texture analysis ,face recognition ,gender classification and image retrieval for its efficiency, simple theory and computation [1, 2, 3, 4, 5, 6, 7, 8, 9]. The basic version of LBP operator only considers the eight neighbors of a pixel[10], and later, it has been extended and yield many modifier versions[11].

In order to use LBP for region descriptor, Heikkilä introduced CS-LBP operator in[12].The proposed CS-LBP descriptor combines the good property of SIFT and LBP which makes it effective for region descriptor. In[12],Heikkailä made a comparison between SIFT and the CS-LBP descriptor on image matching and his experimental result shows the performance of CS-LBP is almost equally promising as the popular SIFT descriptor while CS-LBP is on average 2 to 3 times faster. Due to its effective and efficiency, CS-LBP has been widely used in many applications[13, 14, 15, 16, 17, 18, 19]. In[13],the CS-LBP descriptor was applied in image retrieval task. Zheng in[17] extracted CS-LBP descriptor on blocks of a detection window for pedestrian detection application. Later, they extended the standard CS-LBP into pyramid CS-LBP for pedestrian detection. The CS-LBP descriptor is also applied to video related applications. In[16],Xue.G employed CS-LBP descriptor and applied it on dynamic background substraction application. Douze.M in[19] employed the CS-LBP descriptor on the video copy detection task and he used the CS-LBP descriptor to filter the false returned videos. However, there are some shortcomings of CS-LBP descriptor. The discriminative ability of CS-LBP descriptor is limited because CS-LBP only compares the pair pixel values which ignores the exact difference of the pair pixels. Two patches may have similar or even same CS-LBP descriptor even though they may differ a lot in vision. Besides, the CS-LBP descriptor is sensitive to noise especially noise on flat images. In order to deal with this problems of CS-LBP, we proposed a novel descriptor, denoted as PCA-CS-LBP, based on CS-LBP. Our proposed descriptor is more discriminative and robust to noise than the standard CS-LBP descriptor. In addition, it is more compact than the standard CS-LBP.

The rest of the paper is organized as follows. In section 2 we will give a brief review of LBP operator, CS-LBP operator and the standard CS-LBP region descriptor proposed in[15]. In the next section, we will give a detail description of our descriptor and its construction. Section 4 will provides details of our experimental methodology.

2 Preliminary

2.1 The LBP operator

Local binary pattern (LBP) is a texture descriptor which codifies local primitives (such as curved edges, spots, flat areas, etc) into a feature histogram. LBP and its extensions outperform existing texture descriptors both with respect to performance and to computation efficiency.

The basic version of LBP feature of a pixel is assigned by thresholding the 3×3-neighborhood of each pixel with

978-0-7695-4541-7/11 $26.00 © 2011 IEEE
DOI 10.1109/ICIG.2011.32

the center pixel's value. Let g_c be the center pixel graylevel and g_i $(i = 0, 1, ..., 7)$ be the graylevel of each surrounding pixel. If g_i is smaller than g_c, the binary result of the pixel is set 0, otherwise to 1. All the results are combined to an 8-bit binary value. The decimal value of the binary is the LBP feature. See Figure.1 for an illustration of computing the basic LBP feature.

In order to be able to deal with textures at different

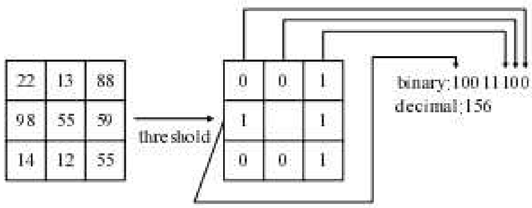


Figure 1. the basic LBP operator.

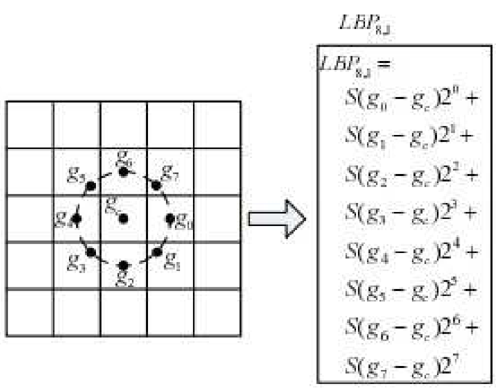


Figure 2. The LBP feature of a pixel's circular neighborhoods with R=1,P=8.

scales, the original LBP has been extended to arbitrary circular neighborhoods[11]by defining the neighborhoods as a set of sampling points evenly spaced on a circle centered at a pixel to be labeled. It allows any radius and number of sampling points. If a sample point does not fall at integer coordinates, the pixel value is bilinearly interpolated. Let (P, R) denote the circular neighborhood, where P is the number of sampling points on the circle and R is the radius of the circle. The LBP feature $LBP_{(P,R)}$ can be computed as Eq(1):

$$LBP_{(P,R)} = \sum_{i=0}^{P-1} S(g_c - g_i)2^i, S(x) = \begin{cases} 1, & \text{if } x > 0 \\ 0, & \text{otherwise} \end{cases} \quad (1)$$

Here g_c is the center pixel's graylevel and $g_i (i = 0, 1, ..., P - 1)$ is the graylevel of each sampling pixel on the circle. See Figure.2 for an illustration of computing the LBP feature of a pixel's circular neighborhoods with R=1 and P=8.

Further extensions to the original LBP are called uni-

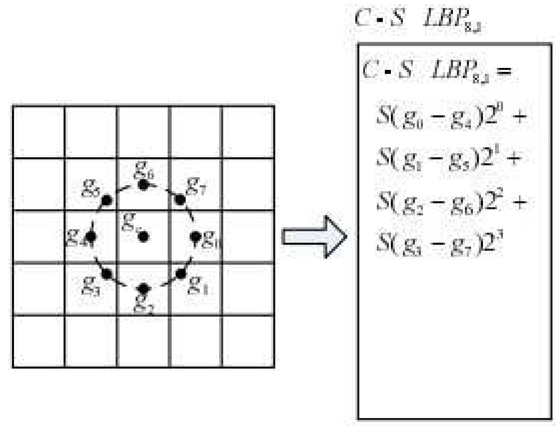


Figure 3. The CS-LBP features for a neighborhood of 8 pixels.

form patterns[11]. A LBP pattern is called uniform if the binary pattern contains at most bitwise transitions from 0 to 1 or vice versa when the bit pattern is considered circular. For example, the bit pattern 11000011, 11110000 and 00111100 arise from different rotations of the same local pattern and they all correspond to the normalized sequence 00001111. Uniform pattern reduces the LBP pattern number from 256 to 58 and is successfully applied to face detection in[11].

2.2 The center-symmetric LBP operator

The center-symmetric LBP(CS-LBP) is another modified version of LBP[12]. It is proposed to solve the problem LBP which has a long histogram. As shown in Figure.3, instead of comparing each neighbor pixel with the center pixel, the center-symmetric pairs of pixels will be compared. The CS-LBP feature computation can be described as Eq(2):

$$CS - LBP_{(P,R,T)} = \sum_{i=1}^{P/2-1} S(g_i - g_{(i+P/2)})2^i$$
$$S(x) = \begin{cases} 1, & \text{if } x > T \\ 0, & \text{otherwise} \end{cases} \quad (2)$$

Where g_i and $g_{(i+P/2)}$ correspond to the graylevel of center-symmetric pairs of pixels of N equally spaced pixels on a circle of radius R. It should be noticed that CS-LBP is closely related to gradient operator, because like some gradient operators it considers graylevel differences between pairs of opposite pixels in a neighborhood. In this way, the CS-LBP feature takes advantages of both the properties of the LBP and the gradient based feature.

2.3 The CS-LBP descriptor

Heillikä in[15] used CS-LBP feature to describe the region around an interest point and their experiments show that the performance is almost equally promising as the popular SIFT descriptor and it has been shown that the CS-LBP descriptor is on average 2 to 3 times faster than SIFT which is because the CS-LBP feature needs only simple arithmetic operations while the SIFT requires time consuming inverse tangent computation when computing the gradient orientation.

The CS-LBP descriptor is a 3D histogram of CS-LBP locations and values. Location is quantized into a 4×4 location grid and CS-LBP operator is applied on each cell of the grid, resulting in a 256-dimensional descriptor. In[15], the CS-LBP descriptor accepts a region detected by region detectors.First, the input region is divided into 4×4 cells with local grid. Then for each cell a CS-LBP histogram is built, resulting in a 3D histogram of CS-LBP locations and values. In order to avoid boundary effects in which the descriptor abruptly changes as a feature shifts from one cell to another, bilinear interpolation over x and y dimensions is used to share the weight of the feature between 4 nearest cells. The share for a cell is determined by the bilinear interpolation weights. Last,the final descriptor is obtained by concatenating the CS-LBP histograms computed for the cells. To reduce the effects of large descriptor elements the descriptor is first normalized to unit length. Then the influence of large descriptor elements is reduced by thresholding the descriptor entries that each one is no larger than 0.2 and renormalizing to unit length.

3 The PCA-CS-LBP descriptor

In the previous section, we briefly describe the CS-LBP operator and CS-LBP region descriptor. The CS-LBP descriptor only compares the center-symmetric pairs of pixels while ignores the exact difference of the pairs of pixels. As a result, two pixels may have same CS-LBP value while the two neighbor regions around two pixels differ a lot in visual content. Besides, the CS-LBP value of a pixel can be easily affected by noise. In order to deal with these problems, we modifier the standard CS-LBP operator and use the difference value of the center-symmetric pairs of pixels to describe a pixel. The modified CS-LBP denoted as D-CS-LBP can be described as Eq(3):

$$\begin{aligned} &D-CS-LBP_{(P,R)} = \left[S_0, S_1, ..., S_{\frac{P}{2}-1}\right] \\ &S_i = g_i - g_{i+\frac{P}{2}} (i = 0, 1, ..., \tfrac{P}{2} - 1) \end{aligned} \tag{3}$$

Here the meaning of P and R is same with CS-LBP. As we can see from Eq(3), the D-CS-LBP of a pixel is a vector containing the difference of the center-symmetric pairs of pixels and the length of D-CS-LBP is N/2. For P=8 and R=1, the length of D-CS-LBP vector will be 8/2=4. Then the D-CS-LBP descriptor of a patch can be obtained by concatenating the D-CS-LBP vector of each pixel for the patch. The D-CS-LBP descriptor reduces the effect of noise while maintains the good property of CS-LBP that CS-LBP is robust to monotonic intensity change. But this D-CS-LBP descriptor is too long and low compact and is not convenient to use in practical applications. So we apply Principal Component Analysis to obtain a compact descriptor denoted as PCA-CS-LBP.

Principal Component Analysis[20] is a standard technique for dimensionality reduction and has been applied to a broad class of computer vision problems, including object recognition[21] and face recognition[22]. Though, PCA suffers from a number of shortcomings[23], such as its implicit assumption of Gaussian distributions and its restriction to orthogonal linear combinations, it remains popular due to its simplicity. The idea of applying PCA to image is not novel. Ke.Y in[24]applied PCA to the SIFT descriptor and proposed PCA-SIFT descriptor. In this paper, we apply PCA in a similar way with Ke.Y[24].

Our proposed descriptor accepts the same input as the standard CS-LBP descriptor: regions that extracted by region detector. Our PCA-CS-LBP algorithm can be summarized as the following steps:(1) pre-computed an eigenspace to express the D-CS-LBP of local patches; (2) given a patch, compute its local image D-CS-LBP values; (3) project the CS-LBP image vector using the eigenspace to derive a compact feature vector. This feature vector is significantly smaller than the standard CS-LBP feature vector, and can be used with the same matching algorithms.

3.1 Offline computation of patch eigenspace

PCA enables us to linearly project high-dimensional samples onto a low-dimensional feature space. For our application (encoded by the patch eigenspace) can be pre-computed once and stored.

As discussed above, the input vector is obtained by computing D-CS-LBP descriptor for a input patch. We then normalize this vector to unit magnitude to minimize the impact of variations in illumination.

In this paper, we use Hessian-Affine region detector for detecting interest points. The detected regions are first affine normalized to a size 41×41 before computing the descriptors. This size is as per the standards of the detector literature.

To build our eigenspace, we adopt a similar approach that stated in PCA-SIFT[24]. We run Hessian-Affine detector[25][26]algorithm on a diverse collection of images and collected 10,000 patches. Each patch is processed as described above to create a $4\times39\times39$=6084 element vector, and PCA is applied to the covariance matrix for PCA-CS-LBP. The images used in building the eigenspace are discarded and not used in any of the matching experiments.

3.2 Feature representation

To find the feature vector for a given image patch, we simply create its 6084-element normalized D-CS-LBP descriptor vector and project it into our feature space using the stored eigenspace. We empirically determined good values for the dimensionality of the feature space, n, in this paper, we use n=36 which is similar to PCA-SIFT[24]. The standard CS-LBP descriptor representation employs 256-element vectors; using PCA-CS-LBP results in significantly space benefits.

4 Evaluation

The proposed descriptors are evaluated on the standard dataset using the standard matching protocol provided by[27]. The underlying performance measure for this protocol is the recall versus false positive ratio. The performance of the designed method was compared with native CS-LBP descriptor. The CS-LBP and used is our own implementation of the algorithm based on the LBP programs provided by[28]. The CS-LBP implementing is with the following parameters Radius=2, Number of nearest neighbors=8,and Threshold=0.01. These parameters were also reported to be performing well in[12]. Our implementation of PCA-CS-LBP is based on the CS-LBP operator with following parameters Radius=1, Number of nearest neighbors=8. The Threshold used in CS-LBP descriptor is not needed in PCA-CS-LBP algorithm. The region detector employed for both the CS-LBP descriptor and our proposed PCA-CS-LBP is Hessian-Affine detector provided in[23].

The standard dataset contains different image sets with different geometric and photometric transformations. It covers six different types of changes for having both pairs of structured and textured scenes. The transformations provided are: viewpoint change, scale change, image rotation, image blur, illumination change and JPEG compression. For each category there are a set of six images with established ground truth homographes. For a given detector and descriptor pair, the performance of the descriptor is measured using the following steps:

1. Accurate number of correspondence is measured by projecting the regions detected on one image on to other and if the overlap error is below a threshold, then the patches are said to corresponding. The overlap threshold is set to 0.5 in this paper.

2. The ground truth number of correspondences also depends on the matching strategy used. Here, we test the descriptor using Nearest Neighbor method. In Nearest Neighbor method, two points are said to be corresponding if the distance between their descriptors is the minimum and is below a threshold. This implies there is one to one matching.

3. Finally, for performance evaluation, the threshold parameter is varied to obtain a plot of recall versus (1-precision).

$$\text{recall} = \frac{\text{No.of correct matches}}{\text{No.of correspondences}} \tag{4}$$

$$\text{1-precision} = \frac{\text{No.of false matches}}{\text{Total.no of matches}} \tag{5}$$

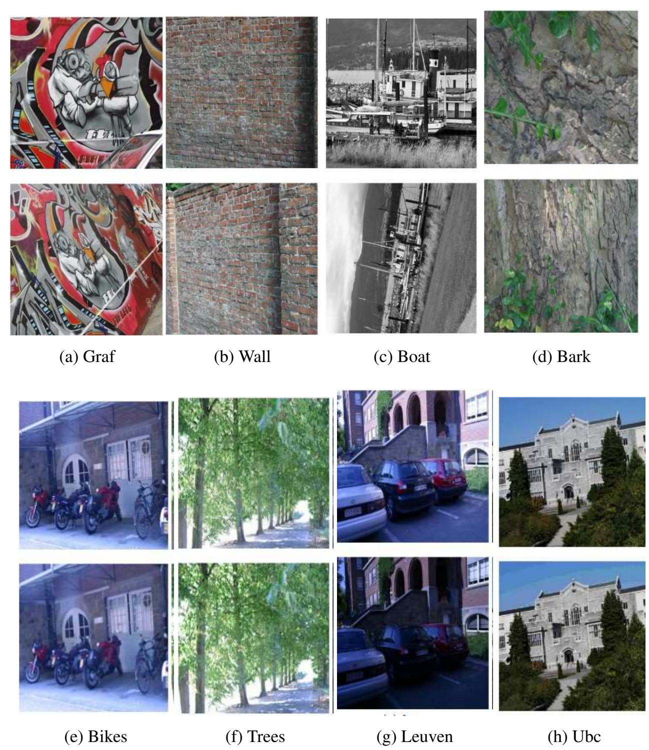
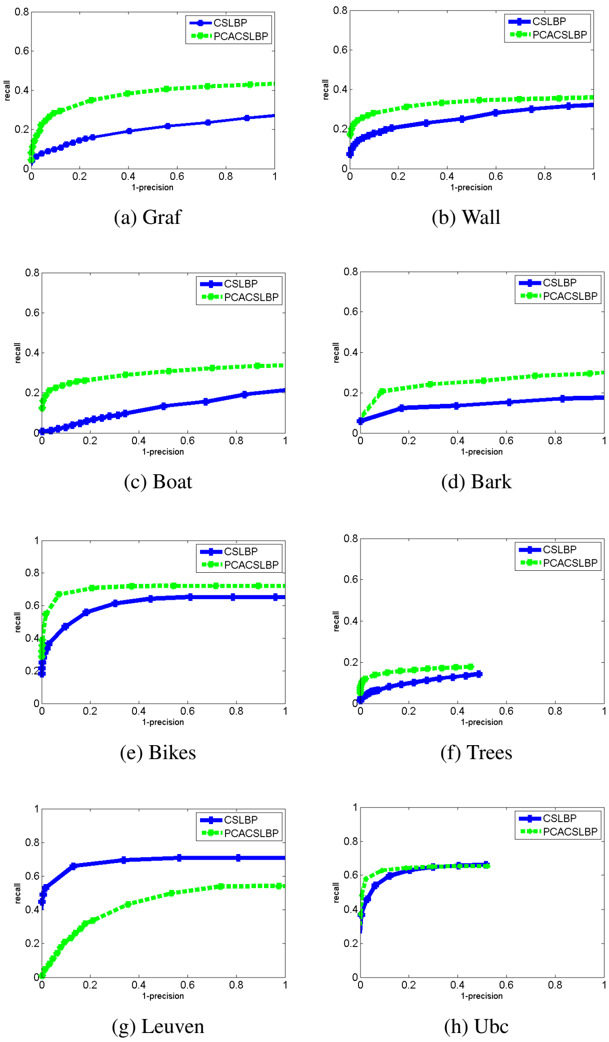


Figure 4. Image pairs used for evaluation



(a) Graf (b) Wall (c) Boat (d) Bark (e) Bikes (f) Trees (g) Leuven (h) Ubc

Figure 5. Performance of PCACSLBP and CS-LBP descriptors on Hessian Affine Regions. The green line is for PCA-CS-LBP while blue one for CS-LBP.

We have evaluated the descriptors based on nearest neighbor based matching. The performance results for matching images pairs shown in Figure.4 are shown in the Figure.5.

From Figure.5, we can see that our descriptor outperforms the standard CS-LBP descriptor at most cases. The only case that the CS-LBP descriptor is better than our descriptor is the pair of Leuven images. The Leuven images suffer the light changes. Under the light change conditions, it results in a shift of the gray value of a pixel. The new pixel value may beyond gray level and it will be set as 0 or 255(for gray level of 256). In this situation, the difference of two pixels after light changing transform is different from the original. Well, as for the CS-LBP descriptor, the magnitude relation of two pixels can remain the same even when the new value of a pixel is beyond gray level. For the CS-LBP descriptor, the magnitude relation of two pixels may change when the new value of this two pixels are both beyond gray level after light changing transform. But for slight light change, our descriptor still performs well. Actually, situations of images suffering great light changes are not common. So our descriptor can achieve good performance at most situations.

5 Conclusion

In this paper, we propose a novel descriptor based on CS-LBP operator. Our descriptor achieves better performance than the standard CS-LBP at most cases. Experimental results also show the improvement of our descriptor. In the future, we will apply our descriptor to actual applications. The work to advanced our descriptor is also under our consideration.

6 Acknowledgement

This work is Supported by the National Science Foundation of China under Grant Nos. 61021062

References

[1] G. Zhao and M. Pietikäinen. Dynamic texture recognition using local binary patterns with an application for facial expressions. *IEEE Transactions on Pattern Analysis and Machine Interlligence*, 29(6):915–928, 2007.

[2] M. Heikkila and M. Pietikäinen. A texture-based method for modeling the background and detecting moving objects. *IEEE Transactions on Pattern Analysis and Machine Interlligence*, 28(4):657–662, 2006.

[3] S. Liao and A. Chung. Texture classification by using advanced local binary patterns and spatial distribution of dominant patterns. *IEEE International Conference on Acoustics, Speech and Signal Processing*, pages I–1221, 2007.

[4] B. Froba and A. Ernst. Face detection with the modified census transform. *International Conference on Automatic Face and Gesture Recognition*, pages 91–96, 2004.

[5] H. Jin, Q. Liu, H. Lu, and X. Tong. Face detection using improved lbp under bayesian framework. *Proceedings of the Third International Conference on Image and Graphics*, pages 306–309, 2004.

[6] G. Zhao and M. Pietikäinen. Local binary pattern descriptors for dynamic texture recognition. *International Conference on Pattern Recognition*, 2:211–214, 2006.

[7] N. Sun, W. Zheng, C. Sun, C. Zou, and L. Zhao. Gender classification based on boosting local binary pattern. *Advances in Neural Networks-ISNN*, pages 194–201, 2006.

[8] T. Ahonen, A. Hadid, and M. Pietikäinen. Face description with local binary patterns: Application to face recognition. *IEEE Transactions on Pattern Analysis and Machine Interlligence*, 28(12):2037–2041, 2006.

[9] V. Takala, T. Ahonen, and M. Pietikäinen. Block-based methods for image retrieval using local binary patterns. *Image Analysis*, pages 882–891, 2005.

[10] T. Ojala, M. Pietikäinen, and D. Harwood. A comparative study of texture measures with classification based on featured distributions. *Pattern recognition*, 29(1):51–59, 1996.

[11] T. Ojala, M. Pietikäinen, and T. Maenpaa. Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. *IEEE Transactions on Pattern Analysis and Machine Interlligence*, 24(7):971–987, 2002.

[12] M. Heikkila, M. Pietikäinen, and C. Schmid. Description of interest regions with local binary patterns. *Pattern Recognition*, 42(3):425–436, 2009.

[13] S. Junding, Z. Shisong, and W. Xiaosheng. Image retrieval based on an improved cs-lbp descriptor. *IEEE International Conference on Information Management and Engineering (ICIME)*, pages 115–117, 2010.

[14] H. Lu and Z. Zheng. Two novel real-time local visual features for omnidirectional vision. *Pattern Recognition*, 43(12):3938–3949, 2010.

[15] R. Mattivi and L. Shao. Spatio-temporal dynamic texture descriptors for human motion recognition. *Intelligent Video Event Analysis and Understanding*, pages 69–91, 2010.

[16] G. Xue, J. Sun, and L. Song. Dynamic background subtraction based on spatial extended center-symmetric local binary pattern. *IEEE International Conference on Multimedia and Expo (ICME)*, pages 1050–1054, 2010.

[17] Y. Zheng, C. Shen, and X. Huang. Pedestrian detection using center-symmetric local binary patterns. *International Conference on Image Processing*, 1(4):1, 2010.

[18] Y. Zheng, C. Shen, R. Hartley, and X. Huang. Pyramid center-symmetric local binary/trinary patterns for effective pedestrian detection. *Asia Conference on Computer Vision(ACCV)*, pages 281–292, 2011.

[19] M. Douze, H. Jégou, and C. Schmid. An image-based approach to video copy detection with spatio-temporal post-filtering. *IEEE Transactions on Multimedia*, 12(4):257–266, 2010.

[20] I. Jolliffe. *Principal component analysis*. Wiley Online Library, 2002.

[21] H. Murase and S. Nayar. Detection of 3d objects in cluttered scenes using hierarchical eigenspace. *Pattern Recognition Letters*, 18(4):375–384, 1997.

[22] M. Turk and A. Pentland. Face recognition using eigenfaces. *Computer Vision and Pattern Recognition*, pages 586–591, 1991.

[23] J. Karhunen and J. Joutsensalo. Generalizations of principal component analysis, optimization problems, and neural networks. *Neural Networks*, 8(4):549–562, 1995.

[24] Y. Ke and R. Sukthankar. Pca-sift: A more distinctive representation for local image descriptors. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2004.

[25] K. Mikolajczyk, T. Tuytelaars, C. Schmid, A. Zisserman, J. Matas, F. Schaffalitzky, T. Kadir, and L. Gool. A comparison of affine region detectors. *International journal of computer vision*, 65(1):43–72, 2005.

[26] K. Mikolajczyk and C. Schmid. An affine invariant interest point detector. *Europe Conference on Computer Vision*, pages 128–142, 2002.

[27] `Http://www.robots.ox.ac.uk/~vgg/research/affine`.

[28] `Http://www.ee.oulu.fi/mvg/page/lbp_bibliography`.

[29] S. Liao, X. Zhu, Z. Lei, L. Zhang, and S. Li. Learning multi-scale block local binary patterns for face recognition. *Advances in Biometrics*, pages 828–837, 2007.