Speed Up Learning based Descriptor for Face Verification

Hai Wang, Bongnam Kang, Jongmin Yoon and Daijin Kim
Department of Computer Science and Engineering, Pohang University of Science and Technology, Pohang, Kyungbuk, 790-784, Republic of Korea

Keywords: Face Recognition, Feature Extraction, LE Descriptor, LBP.

Abstract: Many state of the art face recognition algorithms use local feature descriptors known as Local Binary Pattern (LBP). Many extensions of LBP exist, but the performance is still limited. Recently Learning Based Descriptor was introduced for face verification, it showed high discrimination power, but compared with LBP, it's expensive to compute. In this paper, we propose a novel coding approach for Learning Based Descriptor (LE) descriptor which can keep the most discriminative LBP like feature as well as significantly shorten the feature extraction time. Since the proposed method speed up the LE descriptor's feature extraction time, we call it Speeded Up Learning Descriptor or SULE for short. Tests on LFW standard benchmark show the superiority of SULE with respect of several state of the art feature descriptors regularly used in face verification applications.

1 INTRODUCTION

Face Recognition is a very traditional research topic in computer vision community. In the past twenty years, a large number of approaches that tackled the face recognition were proposed, such as the subspace analysis of PCA(Turk and Pentland, 1991), ICA(Bartlett, 2002) and their extensions. While these face recognition approaches commonly assume that face images are well aligned and have a similar pose, in many practical applications it is impossible to meet these conditions. To this end, face recognition algorithms based on textures of small pathes of face images, known as local appearance descriptors, have shown excellent performance on standard face recognition datasets. Such as Gabor feature(Zheng, 2007), HOG(Alps and Montbonnot, 2005), SIFT(Lowe, 2004), SURF(Bay, 2008) and histogram of Local Binary Pattern(LBP)(Ahonen and Pietikinen, 2006). A comparison of various local descriptor based face recognition algorithms may be found in (Mikolajczyk and Schmid, 2005) and (Cao and Yin, 2010).

Among all the current popular local descriptors in the literature, histogram of ULBP has become popular for face recognition task due to their simplicity, computational efficiency, and robustness to change in illumination. The success of LBP has inspired several variations. These include local ternary pattern(Tan and Triggs, 2010), high order derived LBP(Ahonen and Pietikainen, 2007), multi scale LBP(Liao and Lei, 2007), patch based LBP(Wolf and Taigman, 2008), LBP on Gabor magnitude images(Zhang and Zhang, 2005) and Decision Tree based LBP(Solar and Correa, 2009), to cite a few. However, these methods have some drawbacks, either too complicated to compute such as LBP on Gabor magnitude images, or show better performance on certain database but has poor performance on some other databases. Among all of these LBP variations, decision tree based LBP corporate the LBP with the supervised learning, however, it has several disadvantages such as the feature size is too large for practical usage and the performance greatly depends on the supervised learning procedure. Inspired by Decision Tree based LBP, (Huang and Miller, 2007) proposed a learning based descriptor with unsupervised learning. Compared with other local features, it has showed a better performance on several challenging databases. But until now, no literature focus on the computation complexity of LE descriptor.

In this paper, our main contribution is to propose a coding method that explicitly learns discriminative descriptor from the training data the same as that in original Learning based descriptor (LE). Our learning method is based on K-d tree approach. As a testing scenario, we find that our approach is much faster and efficient than the original LE descriptor while can maintain almost the same recognition performance in the same experiment setting.

Wang H., Kang B., Yoon J. and Kim D..
Speed Up Learning based Descriptor for Face Verification.
DOI: 10.5220/0004278007620767
In *Proceedings of the International Conference on Computer Vision Theory and Applications* (VISAPP-2013), pages 762-767
ISBN: 978-989-8565-47-1

The rest of this paper is organized as follows. Section 2 introduces some basic related work in the field of local descriptor, Section 3 presents the LE descriptor. Section 4 discusses the disadvantage of the LE descriptor and presents the main details of our approach. In Section 5, extensive experiments are conducted and the experimental results are given. Finally, Section 6 concludes our work and figures out some possible future work.

2 RELATED WORK

Several local appearance descriptors have been used in face recognition task. Among all of them, LBP and its extensions are the most widely used. Here, we constrain our discussion within the LBP based extension descriptors. (Tan and Triggs, 2010) proposed local ternary pattern(LTP), instead of directly comparing the neighbour pixel's value with the center value, LTP compares the neighbour pixel's value with the center value with a certain threshold. By doing so, it can remove some noise in the image, but the improvement is very limited. (Ahonen and Pietikainen, 2007) proposed High order derived LBP, which calculates the LBP value based on the previous LBP coded image, in literature, it showed that the third derivative LBP has the best performance, but its performance is not stable, i.e., in some databases, it has good performance, in some databases, the performance improvement is not so significant. (Liao and Lei, 2007) proposed Multi Scale LBP(MS-LBP), in this approach, they compare the neighbour blocks' mean gray value instead of compare the pixel level's gray value, it has been demonstrated better performance when used for classification problem, as for face recognition, the performance is worse than original LBP. (Wolf and Taigman, 2008) proposed patch based LBP, specifically, three patch LBP and four patch LBP, compared with LBP, it's efficient to compute but the performance improvement is not so significant, in some cases, even a little worse. (Marr and Hildreth, 2005) proposed soft LBP, instead of assign each value a certain binary value, they assign each pixel with a range of values, and each value with some probability, due to the comparison is not direct between pixels, so this method is not robust to illumination, and also, due to for each pixel, it has different probability assigned to it with certain values, it's computation complicated. (Zhang and Zhang, 2005) and (Xie and Gao, 2008) proposed LBP on Gabor magnitude images, it showed better performance and has been widely used, but to calculate Gabor image, we need to convolve the image with Gabor filters, which is expensive, so this method is not suitable for the face recognition which requires high speed. (Solar and Correa, 2009) proposed Decision Tree based LBP, by combining the LBP with the supervised learning, it has showed good performance, but the performance is not stable, also, the feature size is extremely large to put it into practical usage.

3 LEARNING BASED DESCRIPTOR

Learning based descriptor is a novel feature extraction method for face recognition. Compared with Local Binary Pattern, it corporates the feature extraction with the unsupervised learning method, and it's more robust to pose and facial expression variation. The specific steps of LE descriptor can be briefly summarized as follows:

- DoG Filter

  To remove the noise and illumination difference in the image, each face image is feed to the DoG filter(Hartigan and Wong, 1979) first. The continued face image processing are based on the filtered image.

- Sampling and Normalization

  At each pixel, sample its neighboring pixels in the ring based pattern to form a low level feature vector. We sample $r\times 8$ pixels at even intervals on the ring of radius r. In each sampling pattern, it has three different parameters, i.e., ring number, ring radius, sampling number of each ring. After sampling, we normalize the sampled feature vector into unit length with L1 norm.

- Learning based Encoding

  An encoding method is applied to encode the normalized feature vector into discrete codes. Unlike many handcrafted encoders, in LE approach, the encoder is specifically trained for the face in an unsupervised manner from a set of training face images. In the paper, they recommend three unsupervised learning methods: K-means(McNames, 2001), PCA tree(Dasgupta and Freund, 2007), and random projection tree(Bentley, 1975). After the encoding, the input image is turned into a coded image.

- Histogram Representation

  After the image has been encoded into coded image, following the method described in Ahone et als work, the encoded image can be divided into a grid of patches. A histogram of the LE codes is computed in each patch and the patch histogram is

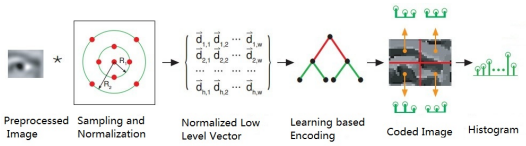

Figure 1: Pipeline of LE descriptor.

concatenated to form the descriptor of the whole face image.

The overall procedure of the LE descriptor pipeline is shown as following Figure 1:

In the paper, although the performance difference of different coding methods is very small as they stated within 1% difference, random projection tree encoding method shows the best performance among the three coding methods, in the following section, we use the random projection tree as the LE descriptor's default encoding method.

4 SPEED UP LEARNING BASED DESCRIPTOR

4.1 Discussion on Learning based Descriptor

From previous description of the LE descriptor, we can find that the key lies in the learning based coder, one of the most important characteristics of the learning approach is that they should partition the normalized feature vector space into the same size, so that in the discrete feature space, each discrete bin can be hit by the same frequency. But in all the three coding methods, it's expensive to calculate the corresponding bin number. For example, for one pixel, if we use 2 rings, and assume the vector size for each pixel is 25, and suppose the bin size is 64, then if we use K means to encode the pattern, to assign a pixel discrete value, we need to find the nearest cluster center, this operation needs 64×25 multiplications and a certain number of additions to calculate the Euclidean distance. If we use random projection tree or PCA tree, then we need 6×25 multiplications and certain number of additions. Compared with LBP, it's too expensive. In addition, the multiplication is performed between two fractions, if we consider its implementation on the embedding system, due to most of the embedding system doesn't have the Floating Point Unit (FPU), these floating point multiplications will take certain time and constrain the practical usage of LE descriptor.

From this point, it's necessary for us to find some alternatives which can code the normalized unit feature without using too many multiplications while can keep the high discrimination power of the LE descriptor. By analysis the procedure of LE descriptor, we can find the key is to partition the normalized unit feature space into the different same size partitions, not the partition method itself is critical to the performance. In addition, we notice that the performance difference caused by different partition methods is not significant. From this point, we can use some other much simpler method such as K-d tree to partition the feature space, and the details can be found in the following section.

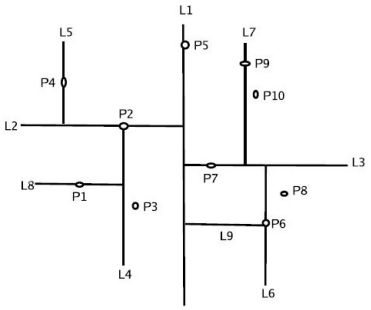

Figure 2: Partition the space.

4.2 Kd Tree based Learning Descriptor

K-d tree is a d dimension binary tree, it can be used to fast find the nearest neighbor in the d dimension vector space, or to partition the vector space, it's a kind of extension of the 1 dimension binary tree. Given some data points, we can build a corresponding K-d tree as the manner of building a one dimension binary tree, the only difference is that in K-d tree, each time when we pick one dimension data as the pivot, and then compare all data points with the pivot using the corresponding picked one dimension value. By such a manner, we can divide the space into smaller spaces recursively. When used for nearest neighbor search, we first need to build K-d tree using the gallery data points, then given some query points, according to the tree we build, we can find the nearest neighbor in O(N*log*N). When used for partition the space, we can first decide the tree level n, then the space can be partitioned to 2^{n-1} sub spaces, and each space has the same number of data points. An illustration diagram of K-d tree is shown in Figure 2.

Figure 2 gives the space partition result. Notice in this case, the K-d tree is balanced, in our K-d tree based learning based descriptor, we use the balanced K-d tree.

If we use K-d tree to partition the spaces, we can partition the current space to two sub spaces each time, by recursively partition the spaces, finally we

can partition the space into several sub spaces. Random projection tree and PCA tree also can partition the space into several subspaces, but the partition approach is different, in random projection or PCA tree, for one time partition, we need to multiply the feature vector with the coefficients, if we recursively partition the space, we need multiply it with the coefficients consecutively. In K-d tree, for each time partition, we just need compare the corresponding coordinate with the pivot, if we recursively partition the space, we just need compare it with the corresponding coordinate consecutively.

5 EXPERIMENT

To demonstrate our proposed approach, in this section, we compare the performance of our proposed SULE with the original LE descriptor in terms of face verification task.

To evaluate our approach with the original LE descriptor fairly, all the experiment settings for LE descriptor and our SULE descriptor are exactly the same, the only difference is the coding method. In our experiment, we use the face image from LFW(Huang and Miller, 2007) for unsupervised learning. LFW database is a very challenging data set with 13,233 face images of 5,749 persons collected from the web. It has two views for its own data. View 1 partitions the 5,749 persons into training and testing set for development purposes such as model selection and parameter tuning. View 2 divides them into ten disjoint folds and specifies 300 positive pairs and 300 negative pairs within each fold: positive pairs are instances of the same person while negative pairs are those of different persons. We follow the LFW configuration that allows us to use the face data on View 1 to train the SULE and LE data. Also, we normalize the LFW database by two eye positions which can be detected by our eye detector. As suggested from the LE paper, more images used for training cannot guarantee a better performance, here we choose 100 images with size 250×250 to train, we believe our training sample is enough. After training, we then use the View 2 face data to evaluate our SULE and LE descriptor.

To demonstrate our approach, we don't use any other advanced face verification or face recognition techniques, i.e., we just crop the face image to 150×80, then extract features for the cropped face image and then directly use Chi-Square matching to calculate the distance of two feature vectors.

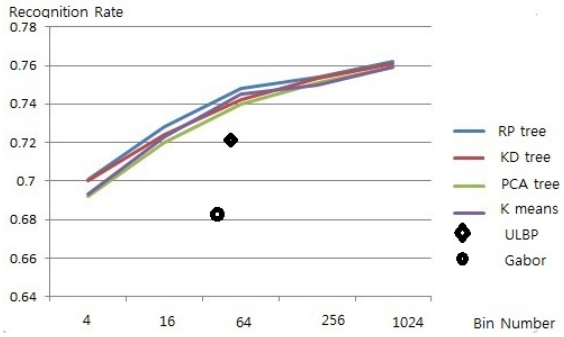

Figure 3: Performance comparison vs. Learning method.

Table 1: Performance Evaluation.

Approach	SULE(%)	LE(%)	ULBP(%)
Performance	74.23	74.81	70.36

5.1 Experimental Results

First, we give the performance comparison result with different bin numbers in each histogram.

From Figure 3, we can see that with different bin numbers (partition the spaces into different number subspaces), the performance difference between our SULE descriptor and the LE descriptor is very small, which coincide our early assumption that that the coding method is not critical for the final performance. Also, we can find that with large bin number, we can get a better performance. In the continuous experiment, we use 64 bins. Here, we give the specific performance using 64 bins in Table I.

Second, in the LE paper, they suggest after feature extraction, we can use PCA to lower the dimension of the feature, combined with the cosine similarity, it can improve the performance. Here, we also follow the rules, the performance is shown in Figure 4. Due to large code number cause large feature size, and this make PCA intractable, here, we just show the performance of using 64 code number using Random Projection tree and Kd tree.

From Figure 4, we can find that in this experiment settings, our SULE descriptor can achieve almost the same performance compared with LE descriptor.

In addition, we give the ROC curve using different descriptors in Figure 5.

We also give the computation complexity for calculating the SULE and LE descriptor in Table II. The result is under the 64 code number settings, notice that this doesn't include the preprocessing time in case of LE descriptor and SULE descriptor due to the preprocessing time just takes up a small part of total processing time in these two descriptors.

In table II, C means comparison operation, M

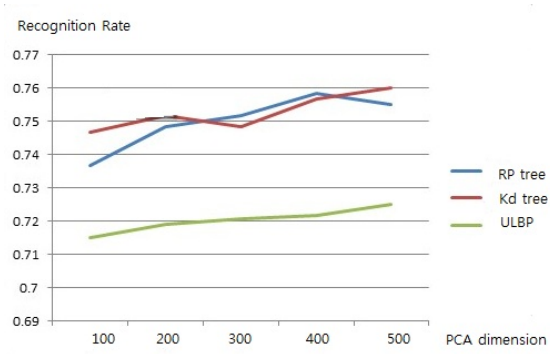

Figure 4: Principal Component Dimension.

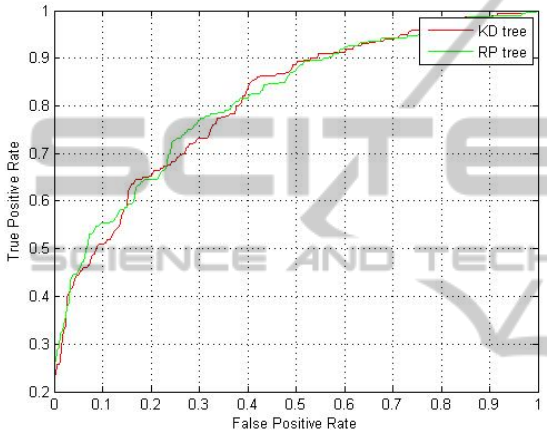

Figure 5: ROC curve of descriptors with different coding methods.

Table 2: Computation Time.

Approach	SULE(C)	LE(M)	ULBP(C)
Operation	6	25*64	8

means the multiplication operation. From Table II, we can find that our SULE is much simpler than LE and more efficient to compute, especially when we implement the algorithm on embedding system. If we don't consider the preprocessing time, compared with ULBP, our approach is still has a slightly advantage in terms of feature extraction time, consider the preprocessing step, the overall computation time of SULE is a little larger than ULBP. But to the best of our knowledge, this is a good trade off between the computation time and recognition performance among several LBP based extensions.

5.2 Results Discussion

From the experimental results from face verification, we can see that our proposed Speed Up Learning Descriptor can keep the high discrimination feature of face images as that for Learning Descriptor, meanwhile, our SULE is much simple and efficient to compute. In the task of face verification, our SULE can achieve comparable performance with LE descriptor in all the different experiment settings, all these results demonstrate that our SULE is a good alternative of LE descriptor. In addition, we firmly believe that the simplification of our SULE can inspire some other extensions of LE descriptor. The first one is that our method is efficient to calculate, if we don't care the feature size, we can extract features using large bin number without significantly increasing the computation time, as we know, large bin number always guarantees a better performance. The second point is that our method doesn't need multiplication, this makes our method is especially suitable for face recognition application on embedding system, and the performance of our SULE is also quite competitive compared with other face recognition method on embedding system. In addition, high speed face recognition on embedding system still remains a difficult problem, we think that our approach can be good solution to this problem. The next point is that, instead of training our encoder with the holistic face images, we can train our encoder with different face patches, each patch has a encoder which can capture the structure details of each face patch better, and we believe that patch based SULE can further improve the verification accuracy. The final point is that we think our SULE can extend to some other task such as human/car classification, eye detection and object detection.

6 CONCLUSIONS

In this paper, we proposed a novel method that uses training pattern to create discriminative LE descriptor by using K-d tree. The algorithm obtains encouraging results on standard database while significantly reduce the computation complexity. In particular, with respect to a face recognizer based on the widely used LBP, our approach also presents a considerable increasing in recognition accuracy, demonstrating the advantages of using K-d tree to train the feature pattern.

As future work, our current implementation does not use different face patches to train different encoders, also we don't consider weights to different face patches. Incorporating weights has been shown to be an effective strategy in various similar works, such as (Solar and Correa, 2009). Finally, although our SULE and LE use DoG filter to make our approach robust to illumination, but compared with ULBP, the DoG filter's effect is still not enough, to

make SULE and LE descriptor more robust to illumination, we need explore some other simple preprocessing technique to solve the illumination problem.

ACKNOWLEDGEMENTS

This research was supported by the Implementation of Technologies for Identification, Behavior, and Location of Human based on Sensor Network Fusion Program through the Ministry of Knowledge Economy (Grant Number: 10041629) and also this research was supported by the Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education, Science and Technology(2012-0008835).

REFERENCES

Ahonen, T. and Pietikainen, M. (2007). Soft histograms for local binary patterns. In *Finnish Signal Processing Symposium*. SciTePress.

Ahonen, T. and Pietikinen, M. (2006). Face description with local binary patterns: Application to face recognition. *IEEE Transaction on Pattern Analysis And Machine Intelligence*, pages 2037–2041.

Alps, I. and Montbonnot (2005). Histograms of oriented gradients for human detection. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. SciTePress.

Bartlett, M. (2002). Face recognition by independent component analysis. *IEEE Trans. on Neural Networks*, pages 1450–1464.

Bay, H. (2008). Surf: Speeded up robust features. *Computer Vision and Image Understanding*, pages 346–359.

Bentley, J. (1975). Multidimensional binary search trees used for associative searching. *Communications of the ACM*, pages 509–517.

Cao, Z. and Yin, Q. (2010). Face recognition with learning-based descriptor. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition(CVPR)*. SciTePress.

Dasgupta, S. and Freund, Y. (2007). Random projection trees and low-dimensional manifolds. *UCSD Technical Report CS2007-0890*, pages 1615–1630.

Hartigan, J. and Wong, M. (1979). Algorithm as 136: A k-means clustering algorithm. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, pages 100–108.

Huang, G. and Miller, E. (2007). Labeled faces in the wild: A database for studying face recognition in unconstrained environments.

Liao, S. and Lei, Z. (2007). Learning multi-scale block local binary patterns for face recognition. *Advances in Biometrics*, pages 828–837.

Lowe, D. (2004). Distinctive image features from scale-invariant keypoints. *International Journal Computer Vision*, pages 91–110.

Marr, D. and Hildreth, E. (2005). Theory of edge detection. *Proceedings of the Royal Society of London*, pages 1615–1630.

McNames, J. (2001). A fast nearest neighbor algorithm based on a principal axis search tree. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 964–976.

Mikolajczyk, K. and Schmid, C. (2005). Performance evaluation of local descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1615–1630.

Solar, R. and Correa, M. (2009). Recognition of faces in unconstrained environments: A comparative study. *EURASIP Journal on Advances in Signal Processing*, pages 1–20.

Tan, X. and Triggs, B. (2010). Enhanced local texture feature sets for face recognition under difficult lighting conditions. *IEEE Transactions on Image Processing*, pages 1635–1650.

Turk, M. and Pentland, A. (1991). Eigenfaces for recognition. *Journal of Cognitive Neurosicence*, pages 71–86.

Wolf, L. and Taigman, Y. (2008). Descriptor based methods in the wild. In *Real-Life Images Workshop at ECCV*. SciTePress.

Xie, S. and Gao, W. (2008). V-lgbp: Volume based local gabor binary patterns for face representation and recognition. In *IEEE Conference on Pattern Recognition(ICPR)*. SciTePress.

Zhang, W. and Zhang, H. (2005). Local gabor binary pattern histogram sequence (lgbphs): A novel non-statistical model for face representation and recognition. In *IEEE Conference on Computer Vision(ICCV)*. SciTePress.

Zheng, Z. (2007). Gabor feature-based face recognition using supervised locality preserving projection. *Signal Processing*, pages 2473–2483.