IDIAP RESEARCH REPORT

HAAR LOCAL BINARY PATTERN FEATURE FOR FAST ILLUMINATION INVARIANT FACE DETECTION

Anindya Roy Sébastien Marcel

Idiap-RR-28-2009

SEPTEMBER 2009

Centre du Parc, Rue Marconi 19, P.O. Box 592, CH - 1920 Martigny
T +41 27 721 77 11 F +41 27 721 77 12 info@idiap.ch www.idiap.ch

Haar Local Binary Pattern Feature for Fast Illumination Invariant Face Detection

Anindya Roy
Anindya.Roy@idiap.ch

Sébastien Marcel
Sebastien.Marcel@idiap.ch

École Polytechnique Fédérale de Lausanne
Lausanne, Switzerland
Idiap Research Institute
Martigny, Switzerland

Abstract

Face detection is the first step in many visual processing systems like face recognition, emotion recognition and lip reading. In this paper, we propose a novel feature called Haar Local Binary Pattern (HLBP) feature for fast and reliable face detection, particularly in adverse imaging conditions. This binary feature compares bin values of Local Binary Pattern histograms calculated over two adjacent image subregions. These subregions are similar to those in the Haar masks, hence the name of the feature. They capture the region-specific variations of local texture patterns and are boosted using AdaBoost in a framework similar to that proposed by Viola and Jones. Preliminary results obtained on several standard databases show that it competes well with other face detection systems, especially in adverse illumination conditions.

1 Introduction

The main challenge for a face detection system is to successfully detect faces in an arbitrary image, irrespective of variations in illumination conditions, background, pose, scale, expression and the identity of the person. Numerous approaches have been proposed to counter these issues. Most of these approaches can be organized in three categories: feature-based approaches [5], appearance-based approaches [23] and boosting-based approaches [20]. The third approach, which involves the boosting of simple local features called Haar features in a cascade architecture, was introduced in 2001 by Viola and Jones [20]. It has become very popular since then because it shows very good results both in terms of accuracy and speed (with the use of Integral Image concept), and is quite suitable for real-time applications. Since the initial work of Viola and Jones, most of the research in face detection has focused on the improvement of their cascade architecture. Related works can be classified in mainly two possible directions: alternative boosting algorithms [10], [18] or alternative architecture designs [9], [17].

However, most of these boosting-based methods which are derived from the Haar feature set have a common limitation. This is the *vulnerability of the Haar feature set to variations in illumination conditions*, for example, where there is a strong side illumination either from

left or right, or the dynamic range of the image intensity varies from region to region over the face (ref. Sec. 2.3, Fig.6). Thus, there is a need to improve the robustness of the system to take into account these illumination variations, but retaining the richness of the feature set, and the advantages of efficient feature selection by boosting and fast evaluation of the features using the Integral Image concept.

The Local Binary Pattern (LBP) introduced by Ojala et al. [12] is one such operator which is robust to monotonic illumination variations (ref. Fig. 1). Thus, various face detection systems have been proposed using LBP or its variants, such as Improved Local Binary Patterns (ILBP) [7], Multi-Block Local Binary Patterns [25], the Modified Census Transform (MCT) [4], [14] and the Locally Assembled Binary (LAB) features [22].

In this paper, we propose a new type of feature called the Haar Local Binary Pattern (HLBP) feature which combines the advantages of both Haar and LBP. This feature compares the LBP label counts in two adjacent image subregions, i.e. it indicates whether the number of times a particular LBP label occurs in one region is greater or lesser than the number of times it occurs in another region, offset by a certain threshold. These two subregions are represented by a set of masks similar to Haar masks [20]. Thus, our features are able to capture the region-specific variation of local texture patterns. This makes our features more robust to illumination variations, which may be quite complex and concentrated over certain subregions of the image only (strong side illumination), compared to Haar and LBP individually. Since each LBP label count is actually a particular bin value of the spatial histogram [24], our features are also robust to slight variations in location and pose.

To our knowledge, this is the first time individual LBP label counts have been combined with Haar features for face detection. Since each HLBP feature is linked with exactly one LBP label, there is no need to consider the entire LBP histogram in training and test, as in [4]. Thus our system is more efficient in terms of storage requirements as well as speed (ref. Sec.3.2). This makes it more suitable for use on mobile devices for instance. We use a variation of the Integral Histogram [21] to calculate our features, which further increases the speed.

We tested our proposed approach using several standard databases against two standard face detection systems. The first is the baseline system based on Haar features [20]. The second is the system based on MCT [4] which is one of the best performing systems representing the state of the art today. [1]

The rest of the paper is organized as follows: we first introduce the proposed HLBP features in Sec. 2. We report the experiments and discuss the results in Sec. 3. Finally, conclusions are given in Sec. 4.

2 The Proposed Framework : Face Detection using HLBP features

In the current work, we unite the two popular concepts of Boosted Haar features [20] and Local Binary Patterns [12], so as to use the advantages of both in the task of face detection.

[1] A public demonstration of the MCT-based face detection system can be found at *http://www.idiap.ch/onlinefacedetector* .

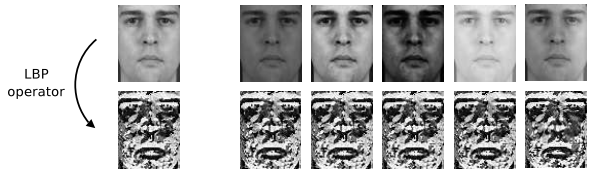

Figure 1: LBP robustness to monotonic gray-scale transformations. On the top row, the original image (left) as well as several images (right) obtained by varying the brightness, contrast and illumination. The bottom row shows the corresponding LBP images which are almost identical. Please see Fig. 6 for more complex illumination changes considered in our experiments.

2.1 General Boosting Framework

The central concept of our framework (as in the Viola and Jones' face detector) is to use boosting, that linearly combines simple weak classifiers $f_j(I)$ to build a strong ensemble, $F(I)$ as follows :

$$F(I) = \sum_{j=1}^{n} \alpha_j f_j(I). \tag{1}$$

The selection of weak classifiers $f_j(I)$ as well as the estimation of the weights α_j are learned by the boosting procedure. An input image I is detected as a face if $F(I)$ is higher than a certain threshold Θ which is also given by the boosting procedure [20] and is rejected otherwise. Each weak classifier f_j is associated with a weak feature, called the Haar feature in Viola and Jones' system. Here, instead of the Haar feature, we use a different set of weak features which we call Haar Local Binary Pattern (HLBP) features.

2.2 The proposed HLBP features

We assume that our input is an $N \times M$ 8-bit gray-level image, which can be represented as an $N \times M$ matrix I, each of whose elements satisfy, $0 \leq I(x,y) \leq 2^8$. In the first stage, we calculate the LBP image I_{LBP} [12] from the original input image I. The LBP operator can be applied at different scales. However, after extensive preliminary testing, we have found the $LBP_{4,1}$ operator as the optimal LBP operator in our case. At a given pixel position (x_c, y_c), the $LBP_{4,1}$ operator is defined as an ordered set of binary comparisons of pixel intensities between the center pixel (x_c, y_c) and its four surrounding pixels, $\{(x_i, y_i)\}_{i=0}^{3}$ (ref. Fig. 2). The decimal form of the resulting 4-bit word is called the LBP code or LBP label of the center pixel and can be expressed as,

$$I_{LBP}(x_c, y_c) = \sum_{n=0}^{3} s(I(x_n, y_n) - I(x_c, y_c))2^n. \tag{2}$$

where $I(x_c, y_c)$ is the gray-level value of the center pixel (x_c, y_c) and $\{I(x_n, y_n)\}_{n=0}^{3}$ are the gray-level values of the 4 surrounding pixels. The function $s(x)$ is defined as,

$$s(x) = \begin{cases} 1 & \text{if } x \geq 0, \\ 0 & \text{if } x < 0. \end{cases} \tag{3}$$

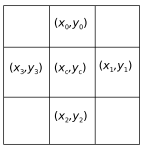

Figure 2: The $LBP_{4,1}$ label for a particular pixel (x_c, y_c) is calculated by comparing its intensity with each one of its four neighbors (vertical and horizontal only), $\{x_i, y_i\}_{i=0}^{3}$, and forming a 4-bit word. Unlike the $LBP_{8,1}$ case, the 4 diagonal neighbors are not considered.

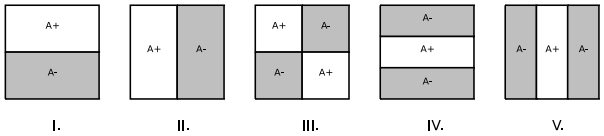

Figure 3: The five types of masks used for the calculation of both Haar and HLBP features, I. Bihorizontal, II. Bivertical, III. Diagonal, IV. Trihorizontal, V. Trivertical.

In the second stage, we calculate the Integral Histogram set $\{I_k^H\}_{k=1}^{N_{labels}}$ [21] of the LBP image I_{LBP}. Here, N_{labels} indicates the number of LBP labels depending on the LBP operator used, and here it has a value of 16 (2^4). Thus the Integral Histogram set consists of $N_{label} = 16$ Integral Histograms. The individual pixels $I_k^H(x,y)$ of the k-th Integral Histogram I_k^H is calculated as the number of pixels above and to the left of the pixel (x,y) in the LBP image I_{LBP} which have a label k, as follows,

$$I_k^H(x,y) = \sum_{u \le x, v \le y} \delta_k(u,v) \tag{4}$$

where $\delta_k(u,v) = 1$ if the label of the pixel at location (u,v) in the LBP image I_{LBP} is k, and is zero otherwise. Using the following pair of references, for all $k \in \{1, N_{label}\}$:

$$i_k^H(x,y) = \quad i_k^H(x,y-1) + \delta_k(x,y) \tag{5}$$
$$I_k^H(x,y) = \quad I_k^H(x-1,y) + i_k^H(x,y) \tag{6}$$

where $i_k^H(x,0) = 0$ for any x and k, the Integral Histogram set can be calculated by one pass over the LBP image. In the third and final stage, the Integral Histogram set will enable us to calculate the proposed HLBP features directly in an efficient and fast way as with Integral Image for the original Haar features. A particular HLBP feature is defined by the following parameters : mask type T (one out of five, ref. Fig. 3), LBP label k (one out of sixteen for $LBP_{4,1}$), position (x,y) of the mask inside the image plane, size (w,h) of the mask, a threshold θ and a direction p (either +1 or -1). It can be observed that a HLBP feature has exactly the same definition as a Haar feature except the addition of the parameter k. To calculate the value of a particular feature $f_{T,k,x,y,w,h,\theta,p}(I)$, its corresponding mask of size (w,h) is placed on the LBP image I_{LBP} at the location (x,y). Like in Viola and Jones' system, each mask type divides the mask region into two areas (ref. Fig. 3), a positive (A_+) and a

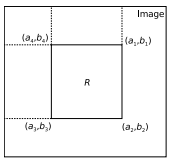

Figure 4: Calculation of the sum of LBP label counts within region R using Integral Histogram (ref. Eqn. 10).

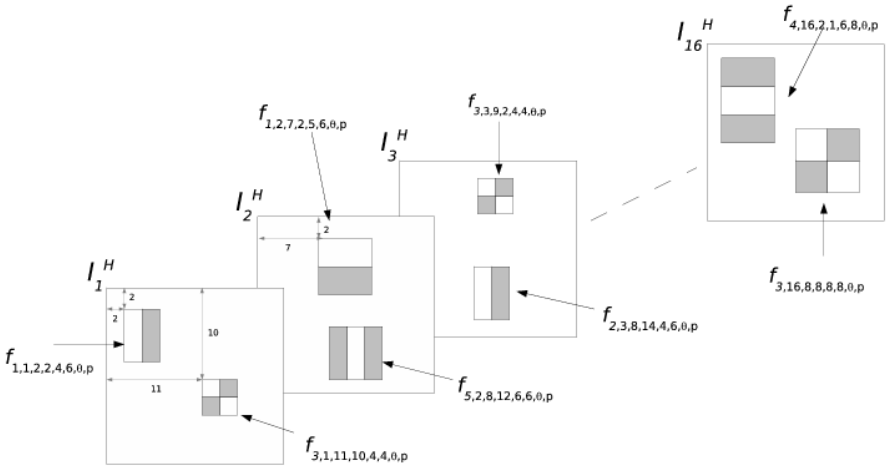

Figure 5: The HLBP features $f_{T,k,x,y,w,h,\theta,p}$ are calculated by placing the corresponding mask at the specified location (x,y) inside the Integral Histogram I_k^H and with the specified size (w,h). Examples of eight different masks corresponding to eight different features have been shown in the figure.

negative (A_-) region. If we define,

$$S_{A+} = \sum_{(u,v)\in A_+} \delta_k(u,v) \tag{7}$$
$$S_{A-} = \sum_{(u,v)\in A_-} \delta_k(u,v) \tag{8}$$

with $\delta_k(u,v)$ as defined[2], then the HLBP feature value is given simply by,

$$f_{T,k,x,y,w,h,\theta,p}(I) = \begin{cases} 1 & \text{if } p.(S_{A_+} - S_{A_-}) > p.\theta, \\ -1 & \text{if } p.(S_{A_+} - S_{A_-}) \leq p.\theta \end{cases} \tag{9}$$

Thus, the HLBP feature is a binary feature, as the normal Haar feature. In other words, the HLBP feature indicates whether region A_+ (region A_-) has θ pixels more with the LBP label k compared to region A_- (region A_+), given $p = 1$ ($p = -1$), i.e. the spatial count differences of the LBP label k (ref. Sec. 1). However, to calculate S_{A+} and S_{A-} we do not need to use the above equations 7 and 8. They can each be calculated directly by only a few

[2]For the Viola and Jones' system, $\delta_k(u,v)$ is replaced by $I(u,v)$, the pixel intensity at location (u,v).

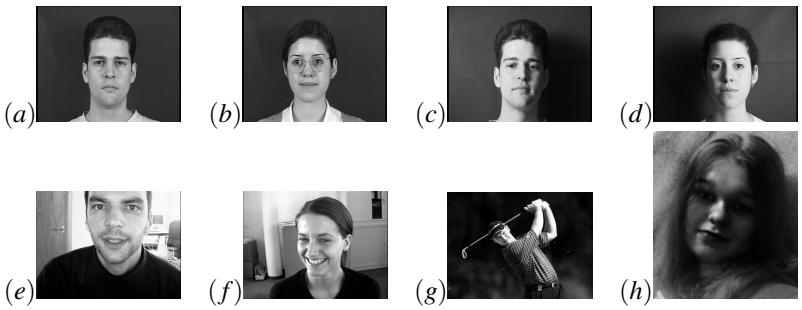

Figure 6: Example images from the databases used in our experiments: (a)-(b) XM2VTS Normal set, (c)-(d) XM2VTS Darkened set, (e)-(f) BioID database, (g)-(h) Fleuret database.

references to the corresponding Integral Histogram I_k^H as in usual Haar features, as follows. Let us denote by $(a_1,b_1),(a_2,b_2),(a_3,b_3),(a_4,b_4)$ the four corners of a generic rectangular region R, like A_+ or A_- (ref. Fig. 3). Then the sum S_R (as in Eqns.7 and 8) can be calculated directly as (ref. Fig. 4),

$$S_R = I_k^H(a_2,b_2) - I_k^H(a_3,b_3) - I_k^H(a_1,b_1) + I_k^H(a_4,b_4) \tag{10}$$

Thus finally, each such HLBP feature can also be calculated with just a few references to the pertinent Integral Histogram I_k^H, allowing our algorithm for real time implementation just as with normal Haar features.

2.3 Advantage of HLBP features over Haar features

The HLBP features involve counting the number of pixels in a region having a certain LBP label k, instead of summing over pixel intensities as with Haar features. Now, due to adverse illumination conditions, the pixel intensities in an image I may change. However, the LBP label of a pixel is much more robust to illumination changes as shown in Fig. 1. Thus, the number of pixels within a region having a particular LBP label will also remain more or less constant with varying illumination. More precisely, if we observe footnote[2], the term $I(u,v)$, the pixel intensity at location (u,v), changes with varying illumination. Hence the final Haar feature value will also change. In contrast, if we observe the defining Eqns. 7 and 8, in Sec. 2.2 for the calculation of HLBP features, we see that $I(u,v)$ has been replaced by $\delta_k(u,v)$, which is 1 if the LBP label of pixel (u,v) is k, the feature parameter, and 0 otherwise. According to definition of LBP, since LBP code is robust to illumination changes, $\delta_k(u,v)$ is also robust to illumination changes. Thus the final HLBP feature value, as defined in Eqn. 9, remains robust too. This observation has motivated us to combine the LBP concept with the Haar feature framework to obtain the advantages of both.

3 Experiments

We implemented a face detection system using our proposed HLBP features, and compared its performance against two other reference face detection systems.

Database	Number of images	Illumination conditions	Other challenging aspects
XM2VTS Normal set [11]	2360	Uniform illumination	-
XM2VTS Darkened set [11]	1180	Strong side-illumination	-
BioID [6]	1521	Non-uniform illumination	Images were obtained in real world conditions featuring a large variety of illumination, background and face size.
Fleuret [3]	580	Non-uniform illumination	Images from real life situations were collected from the web, showing large variations in illumination, background and face size and slight variations in pose.

Table 1: Description of the databases used in our experiments

3.1 Reference systems and databases used

The first reference system is the one by Viola and Jones [20] using normal Haar features. It provides the baseline for Haar feature-based systems. The second reference system is the one by Froba et al. [4][14] using Modified Census Transform (MCT). It is one of the LBP variants representing the current state of the art. To calculate the MCT, Froba et al. compare each pixel in a 3×3 grid against the average of the intensity values within that grid, instead of the center pixel as in LBP (ref. Sec.2.2). This leads to a 9-bit code and a $511(2^9 - 1)$-bin Lookup table (LUT), each entry of which stores the log-likelihood ratio of a particular code. This LUT has to be stored for each feature. The face detector is implemented as a cascade of classifier stages, where each stage calculates the sum of LUT bins corresponding to the MCT-codes at particular locations in the test image.

We implemented our system and both the reference systems as cascades of 5 stages. Each stage had a strong classifier boosted from the set of weak classifiers (ref. Sec.2.1). The stages had 5, 10, 20, 50 and 200 weak classifiers respectively. Thus, the number of features is the same for all the 3 systems.

For training, we used two internally created databases consisting of face and non-face images extracted from BANCA(Spanish Corpus) [1], Essex, Feret [13], ORL [15], Stirling and Yale [2] databases. For testing, we used 1) the standard XM2VTS database [11], [8], taking into account two cases, the Normal set with normal lighting conditions and the Darkened set with adverse or side illumination, 3) the BioID database [6] and 4) an additional database from Fleuret et al. [3]. Examples from each database used for testing are shown in Fig.6. A brief description of each database is given in table 1.

3.2 Results and discussions

The face detection performance of the three systems are given in Fig.7 in terms of ROC curves on each of the four databases. We discuss these results and various other aspects of the system below.

Performance From Fig.7, we observe that our system (HLBP) performs reasonably well on all the four databases. However, its performance is noteworthy especially for the three cases with adverse imaging conditions, i.e, XM2VTS Darkened set, BioID database and the Fleuret database (please refer to Table 1 for more details). For the XM2VTS Darkened set, it outperforms Haar by a wide margin. Although MCT is able to achieve an initial higher True Positive Rate (TPR), HLBP is able to outperform MCT as soon as the number of false positives are allowed to reach 50. From this point onwards, MCT is not able to improve its TPR further, while HLBP is able to improve it by a significant amount. For the BioID

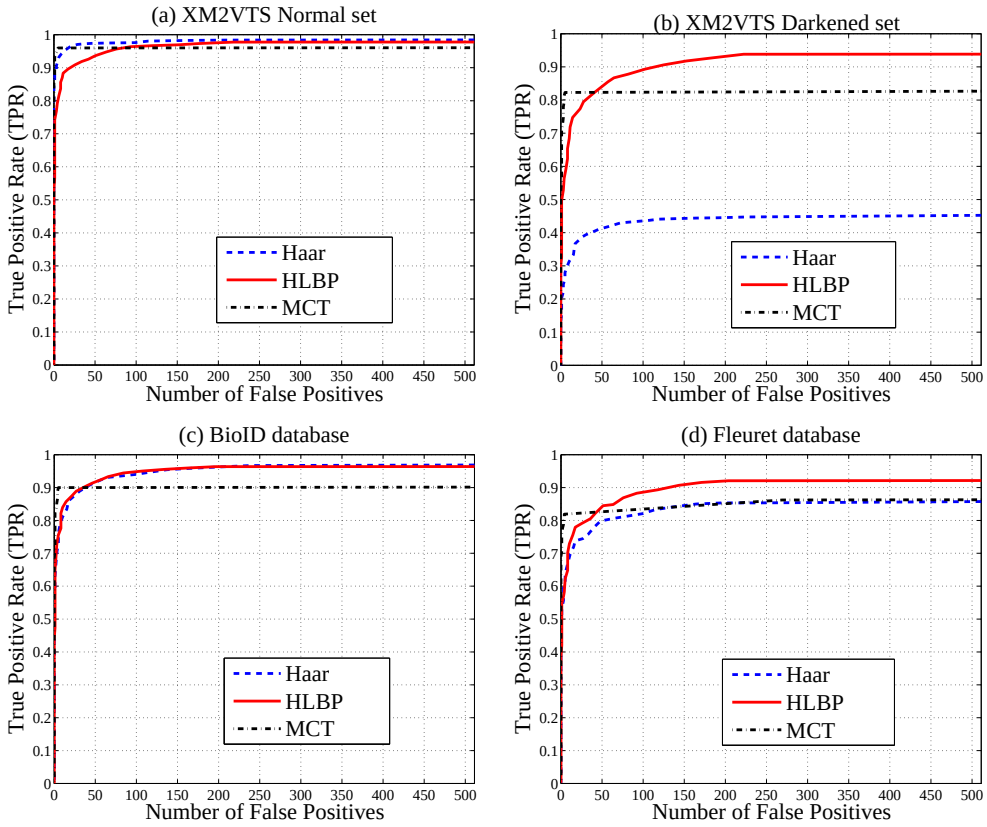

Figure 7: Comparison of face detection performance on different datasets by the three systems using Haar, HLBP and MCT features: (a) XM2VTS Normal set, (b) XM2VTS Darkened set, (c) BioID database and (d) Fleuret database.

database, HLBP performs as well as Haar and soon outperforms MCT after an initial higher TPR by MCT. MCT is not able to handle the variation in face size and pose as well as HLBP and keeps rejecting some of the faces. For the Fleuret database also, HLBP outperforms Haar by a wide margin, and outperforms MCT also, after an initial higher performance by the latter. It is true that HLBP is not able to outperform the two systems for the XM2VTS Controlled set, however this is not so significant since most real world situations would correspond to the other three cases.

Storage requirements and number of parameters In Table 2, we enlist all the parameters required to define a Haar, HLBP and MCT feature respectively. We observe that the number of parameters required is within 10 for Haar and HLBP, while it is 513 for MCT. The major difference for MCT comes from the 511-bin LUT (ref. Sec.3.1) which is not required for Haar and HLBP. Thus a single MCT feature is much more complex to represent than a Haar or HLBP feature. We also give an estimate of the minimum number of bits required to store these parameters based on their ranges and types. For Haar and HLBP, it is around $26+2\times N_f$ bits, where N_f is the number of bits required to store a floating point number. For MCT, it is $10+511\times N_f$. With $N_f = 32$ bits or 4 bytes, the value used in our system, Haar

Parameter Type	Number of para-meters	Range/Type of each parameter	Minimum number of bits per parameter	Total number of bits required	Haar	HLBP	MCT
Location	2 (x,y)	1-19	5	10	✓	✓	✓
Size	2 (w,h)	6-19	4	8	✓	✓	-
Mask Type, T	1	1-5	3	3	✓	✓	-
Direction, p	1	$\{-1,1\}$	1	1	✓	✓	-
LBP Label, k	1	1-16	4	4	-	✓	-
Feature weight, α	1	`float`	N_f	N_f	✓	✓	-
Threshold, θ	1	`float`	N_f	N_f	✓	✓	-
Lookup Table (LUT)	511	`float`	N_f	$511 \times N_f$	-	-	✓
			Total number of parameters per feature		8	9	513
			Total number of bits per feature		$22+$ $2 \times N_f$	$26+$ $2 \times N_f$	$10+$ $511 \times N_f$

Table 2: Comparison of storage requirements (in bits) and the number of free parameters per feature of the 3 systems, Haar, HLBP and MCT [4]. Each row lists a parameter and a checkmark (✓) in a particular column indicates that this parameter is required for the definition of the corresponding feature. Please refer to Sec.2.1, 2.2 (Eqn.9) and Sec.3.1 for more details about each parameter. Here N_f denotes the number of bits required to store one floating point number. It is compiler-dependent. In our setup it is 32 bits or 4 bytes, a typical value.

requires 86 bits, HLBP 90 bits and MCT requires 16362 bits. Thus, MCT has a much higher storage complexity than HLBP and Haar in terms of bits per feature and also in terms of total number of bits to represent the model, since exactly the same number of features were used for all the three systems (ref. Sec. 3.1). Thus HLBP is able to achieve comparable results with MCT using a model as simple as Haar but much simpler than MCT. This justifies the use of HLBP in low memory applications involving embedded devices and mobile phones rather than MCT. Further, a model with higher number of parameters (MCT) entails a higher classification risk at test time due to overfitting on the training set [19].

Training and test time At first glance, the total number of possible features should be 16 times more for HLBP than for Haar since every Haar feature can be associated with one out of 16 possible LBP labels to give one HLBP feature. However, since HLBP is derived from histograms or counts of the $LBP_{4,1}$ labels and not the pixel intensity themselves, we do not use all possible windows at all locations and scales, but only use windows which have a minimum size of 6 pixels. This is because smaller sized windows would not be useful in filling up the histogram. This reduced the number of features to around 100,000 which compares favorably with the Haar feature set which number around 64,000, for a window size of 19×19. This leads to comparable training times for the two algorithms. In fact, HLBP is able to reject about 81.2% of the non-faces in the first stage compared to 75.5% for Haar, leading to a further reduction in its training time. For MCT, a 511-bin LUT needs to be calculated for each individual feature (ref. Sec.3.1) which is avoided by our system, thus making it faster. For testing, we use exactly the same setup (number of stages and number of classifiers at each stage) for the three systems, the only difference from Haar being the calculation of the $LBP_{4,1}$ image as a preprocessing in HLBP. However, the calculation of the $LBP_{4,1}$ image can be done in one pass over the image using only two relational operations per pixel. Also, this operation is only needed once per scale. Hence, the relative increase in computation time is negligible. MCT also requires a similar preprocessing step as for HLBP (ref. Sec.3.1).

Originality of proposed method Certain other systems also involve either Local Binary Patterns and / or boosted Haar-like features, similar to Viola and Jones. However, they are different from our proposed system. The Multi-Block Local Binary Pattern [25] and Locally Assembled Binary Feature [22] extend the idea of LBP by comparing sums of intensities over image patches to calculate the LBP label itself. The object detection framework by Zhang et al. [24] uses the concept of spatial histograms of Local Binary Patterns. Their features measure the similarity between model and test histograms using histogram intersection [16]. However, none of these methods compare counts of individual LBP labels in two regions as we do. Our method tries to capture the region-specific variation of certain local texture patterns, which is not done in [25],[22] and [24]. Wang et al. [21] have used Fisher Linear Discriminant on Histogram features for Face Detection. However, there is no use of LBP concept which is the major contribution of our work. Furthermore, the inclusion of Fisher Linear Discriminant increases the computational complexity at test time.

4 Conclusion

In this paper, we have introduced a new type of feature called the HLBP feature which combines the concepts of Haar feature introduced by Viola and Jones, with Local Binary Patterns, harnessing the advantages of both for the problem of face detection. Our features are able to model the region-specific variations of local texture and are relatively robust to wide variations in illumination, pose and background, and also slight variations in pose. Experiments have shown that our system performs significantly better in such adverse imaging conditions than normal Haar features and performs reasonably better than MCT features with much less storage and computation requirements.

5 Acknowledgements

The authors would like to thank the Swiss National Science Foundation (projects 200020-122062 and 51NF40-111401) and the FP7 European MOBIO project (IST-214324) for their financial support.

References

[1] E. Bailly-Bailliere, S. Bengio, F. Bimbot, M. Hamouz, J. Kittler, J. Mariethoz, J. Matas, K. Messer, V. Popovici, F. Poree, B. Ruiz, and J.P. Thiran. The BANCA database and evaluation protocol. In *Proc. of the 4th Intl. Conf. on Audio- and Video-Based Biometric Person Authentication (AVBPA)*, pages 625–638, 2003.

[2] P. Belhumeur, J. Hespanha, and D. Kriegman. Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 19(7):711–720, 1997.

[3] F. Fleuret. Fast Binary Feature Selection with Conditional Mutual Information. *Journal of Machine Learning Research*, 5:1531–1555, 2004.

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

[5] B. Heisele, T. Serre, M. Pontil, and T. Poggio. Component-based Face Detection. In *Computer Vision and Pattern Recognition*, pages 657–662, 2001.

[6] O. Jesorsky, K. Kirchberg, and R. Frischholz. Robust Face Detection using the Hausdorff Distance. In *Proc. of the 3rd Intl. Conf. on Audio- and Video-Based Biometric Person Authentication (AVBPA)*, pages 90–95, 2001.

[7] H. Jin, Q. Liu, H. Lu, and X. Tong. Face detection using improved LBP under bayesian framework. In *Proc. of the Third International Conference on Image and Graphics (ICIG)*, pages 306–309, 2004.

[8] J. Luettin and G. Maitre. Evaluation protocol for the extended M2VTS database (XM2VTSDB). Idiap Communication 98-05, Idiap, 2000.

[9] H.T. Luo. Optimization design of cascaded classifiers. In *Computer Vision and Pattern Recognition*, pages 480–485, 2005.

[10] S.W. Lyu. Infomax Boosting. In *Computer Vision and Pattern Recognition*, pages 533–538, 2005.

[11] K. Messer, J. Matas, J. Kittler, J. Lüttin, and G. Maitre. XM2VTSDB: The Extended M2VTS Database. In *2nd Intl. Conf. Audio- and Video-based Biometric Person Authentication, AVBPA'99*, pages 72–77, 1999. URL http://www.ee.surrey.ac.uk/Research/VSSP/xm2vtsdb/docs/messer-avbpa99.pdf.

[12] T. Ojala, M. Pietikainen, and D. Harwood. A comparative study of texture measures with classification based on feature distributions. *Pattern Recognition*, 29:51–59, 1996.

[13] P. Phillips, H. Moon, S. Rizvi, and P. Rauss. The Feret evaluation methodology for face recognition algorithms. *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, 22(10):1090–1104, 2000.

[14] Yann Rodriguez. *Face Detection and Verification using Local Binary Patterns*. PhD Thesis, École Polytechnique Fédérale de Lausanne, 2006.

[15] F. Samaria and S. Young. HMM-based Architecture for Face Identification. *Image and Vision Computing*, 12(8):537–543, October 1994.

[16] B. Schiele. *Object Recognition using Multidimensional Receptive Field Histograms*. PhD thesis, I.N.P.Grenoble, 1997.

[17] J. Sochman and J. Matas. WaldBoost-Learning for time constrained sequential detection. In *Computer Vision and Pattern Recognition*, pages 150–156, 2005.

[18] J. Sun, J.M. Rehg, and A.F. Bobick. Automatic cascade training with perturbation bias. In *Computer Vision and Pattern Recognition*, pages 276–283, 2004.

[19] V.N. Vapnik. *Statistical Learning Theory*. Wiley-Interscience, 1989.

[20] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 511–518, 2001.

[21] H. Wang, P. Li, and T. Zhang. Histogram Features-Based Fisher Linear Discriminant for Face Detection. In *Asian Conference on Computer Vision*, pages 521–530, 2006.

[22] S. Yan, S. Shan, X. Chen, and W. Gao. Locally Assembled Binary (LAB) Feature with Feature-centric Cascade for Fast and Accurate Face Detection. In *Computer Vision and Pattern Recognition*, 2008.

[23] M.-H. Yang, D. Roth, and N. Ahuja. A SNoW-based face detector. In *Advances in Neural Information Processing Systems*, pages 855–861, 2000.

[24] H. Zhang, W. Gao, X. Chen, and D. Zhao. Learning Informative Features for Spatial Histogram-Based Object Detection. In *Proc. of International Joint Conference on Neural Networks, Montreal, Canada*, pages 1806–1811, 2005.

[25] L. Zhang, R. Chu, S. Xiang, S. Liao, and S.Z. Li. Face detection based on Multi-Block LBP representation. In *2nd Intl. Conf. on Biometrics (ICB)*, pages 11–18, 2007.