An Improvement of Energy-Transfer Features Using DCT for Face Detection

Radovan Fusek, Eduard Sojka, Karel Mozdřeň, and Milan Šurkala

Technical University of Ostrava, FEECS, Department of Computer Science,
17. listopadu 15, 708 33 Ostrava-Poruba, Czech Republic
{radovan.fusek,eduard.sojka,karel.mozdren,milan.surkala}@vsb.cz

Abstract. The basic idea behind the energy-transfer features (ETF) is that the appearance of objects can be successfully described using the function of energy distribution in the image. This function has to be reduced into a reasonable number of values. These values are then considered as the vector that is used as an input for the SVM classifier. The process of reducing can be simply solved by sampling; the input image is divided into the regular cells and inside each cell, the mean of the values is calculated. In this paper, we propose an improvement of this process; the Discrete Cosine Transform (DCT) coefficients are calculated inside the cells (instead of the mean values) to construct the feature vector. In addition, the DCT coefficients are reduced using the Principal Component Analysis (PCA) to create the feature vector with a relatively small dimensionally. The results show that using this approach, the objects can be efficiently encoded with the relatively small set of numbers with promising results that outperform the results of state-of-the-art detectors.

1 Introduction

In the area of feature based detectors, many methods have proved to be very effective using the sliding window technique. The basic idea behind the sliding window technique is that the window scans the image in different scales. The image inside each window is examined; the features that are capable to describe the objects of interest are calculated inside each window (image). The features that are obtained are combined into the final feature vector. This vector is then used as an input for the trainable classifier (e.g. neural network, support vector machine, random forest).

The main contribution of this paper is an improvement of the sliding window detector that is based on energy-transfer features (ETF). The face detector based on these features was presented in [6]. The basic idea of these features is that the appearance (shape) of objects can be described using the energy distribution. Inside the image, the transfer of energy is solved by making use of the physical laws. The image can be imagined as a rectangular plate with the thermal conductivity properties; the image gradient can be considered as a thermal insulator (the places with the high gradient indicate the low conductivity and

A. Elmoataz et al. (Eds.): ICISP 2014, LNCS 8509, pp. 511–519, 2014.

vice versa). To simulate the temperature transfer, the temperature sources are placed in the form of a regular grid into the image. Inside the image, the temperature is transfered from these sources during a certain chosen time. After this time, the distribution of temperature is investigated; the image is divided into the rectangular non-overlapping cells and inside each cell the mean temperature is calculated. In this work, we propose an improvement of this process; instead of the mean temperature inside each cell, we use the Discrete Cosine Transform (DCT) coefficients to encode the function of temperature distribution. After DCT, the DC coefficients represent the average temperatures of the regions and the AC coefficients represent temperature changes across the regions. It is obvious that the information obtained after DCT are more descriptive and can be used to effectively encode the function of temperature distribution. Finally, the PCA (Principal Component Analysis) is used to create the feature vector with the relatively small dimensionality.

The rest of the paper is organized as follows. In Section 2, the state-of-the art features are mentioned. In Section 3, the propose process of feature extraction is described and the results are shown in Section 4.

2 Related Works

The image features that can be used in object detectors can be divided into two areas; sparse (keypoint detectors) and dense. The sparse type features are based on the keypoint detectors; the features are calculated inside the image areas that are located in the neighborhoods of the keypoint. The dense type features are calculated over the whole image that is usually divided into the regular (overlap or non-overlap) regions and within these regions the features are calculated. One of the most popular descriptor based on the keypoints was proposed by David Lowe [10]. The method is called Scale Invariant Feature Transform (SIFT). The method consists of a Difference of Gaussian (DoG) keypoint detector; the histogram gradient orientations inside the regions around the keypoints are calculated. The Speeded up Robust Feature (SURF) descriptor by Bay et al.[2] is also one of the widely used keypoint descriptors. The authors used the Haar-wavelet responses with the fast calculation via the integral image thanks to which SURF is faster than SIFT. The very fast method called Binary Robust Independent Elementary Features (BRIEF) was proposed by Calonder et al.[4]. In BRIEF, the binary string that contains the results of intensity difference of pixels are used and the descriptor similarity is evaluated using the Hamming distance. In [13], the authors proposed the another binary descriptor with the rotation and noise invariant properties called Oriented Fast and Rotated BRIEF (ORB). In a similar way, Leutenegger et al.[8] proposed Binary Robust Invariant Scalable Keypoints (BRISK) with the rotation and scale invariant properties consists of the FAST-based detector. Finally, the Fast Retina Keypoint (FREAK) descriptor that also uses the binary strings was proposed in [1].

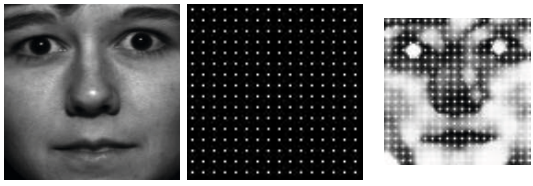
In the area of dense features (without the key-point detector), three types of features are considered as state-of-the-art; Histogram of Oriented Gradients

(HOG), Haar-like features, and Local Binary Patterns (LBP). The HOG descriptors [5] are inspired by SIFT and the descriptors can be regarded as the dense version of SIFT. Viola and Jones [15] proposed the detection framework that used the Haar-like features combined with the integral image and AdaBoost algorithm. LBP were proposed by Ojala et al.[11] for texture analysis, hoverer, many variations of LBP were proposed for solving the problem of face detection and recognition.

3 Proposed Method

The main principle behind the energy-transfer features (ETF) is based on the fact that the shape of the objects can be described using the function of energy distribution. The advantages of this function can be characterized as follows. Suppose that the appearance of the objects is defined using the function of edge information only (e.g gradient sizes, directions). In the cases that the edges are very thin, the samples can miss the important information about the edges, however, the samples can capture the information about the areas of objects rather than the edge direction or size in this particular case. Moreover, the edges can be corrupted (e.g. due to the noise), which causes that the invalid edge information can be obtained. On the other hand, in the energy-transfer features, the information about the object areas is important and it is used for object description.

Suppose that the temperature source is placed inside the object area. Since the gradient of the image represents the thermal insulator, this area will be filled with the certain temperature distribution after the temperature transfer process. The values of temperature will be approximately constant inside this area; these values can be investigated and used for description of the object. Since the real-life objects (Fig. 1(a)) consist of the areas with different properties (e.g. sizes, shapes), the temperature sources are placed in the form of a regular grid (Fig. 1(b)). The visualization of temperature distribution is shown in Fig. 1(c).



(a) (b) (c)

Fig. 1. The real-life image (a). The regular grid of sources (b). The visualization of distribution of temperature from these sources (c). The value of temperature is depicted by the level of brightness.

In ETF, the thermal field inside the input image is solved by making use of the following equation [12]

$$\frac{\partial I(x,y,t)}{\partial t} = \mathrm{div}(c\nabla I), \tag{1}$$

where $I(x,y,t)$ represents the temperature at a position (x,y) at a time t, div is a divergence operator, ∇I is the temperature gradient and c stands for thermal conductivity. For the source points and arbitrary time $t \in [0,\infty)$, we set $I(x_s,y_s,t) = 1$, where (x_s,y_s) are the coordinates of the source points (i.e. we hold the temperature constant during the whole process of transfer, which is in contrast with the usual diffusion approaches). In all remaining points, we take into account the initial condition $I(x,y,0) = 0$. The equation is solved iteratively. The conductivity in Eq. 1 is determined by

$$c = g(\|E\|), \tag{2}$$

where E is an edge estimate. We define the edge estimate E as the gradient of original image $E = \nabla B$, where B is the brightness function. The function $g(\cdot)$ has the form of [12]

$$g(\|\nabla B\|) = \frac{1}{1 + \left(\frac{\|\nabla B\|}{K}\right)^2}, \tag{3}$$

where K is a constant representing the sensitivity to the edges [12].

Once the temperature field over the input image is obtained (at a chosen time t), the temperature values should be sampled. For this purpose, we divide the input image inside the sliding window into the blocks. The blocks are divided into the cells (Fig. 2).

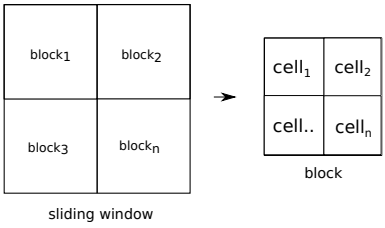


Fig. 2. The blocks and cells inside the input image (sliding window)

We experimented with the different sizes of the blocks and cells. We observed that the best results were obtained using 16×16 blocks and 8×8 cells; inside the cells the DCT coefficients are computed and composed to the final feature vector. In the case of cells with 8×8 pixels (similarly in JPEG compression), each cell

consists of 1 DC coefficient and 15 AC coefficients after DCT. The coefficients that are located in the upper left corner contain the most of information (low frequencies). On the other hand, the bottom right coefficients represent higher frequencies that can be discarded. Therefore, instead of the encoding the whole set of the coefficients, we encode the upper left coefficients only.

To encode the upper left coefficients, we create three patterns of these coefficients (Fig. 3) for our experiments (similarly in [14]). In these patterns, the three AC regions are created. These regions represent the different frequencies and different information can be encoded using different patterns. To reduce the quantity of the coefficients, the mean of coefficients is calculated inside these regions. It means that each 8×8 cell is represented by four values; 1 DC coefficient + 3 averages of AC coefficients. The final feature vector is composed from these values.

We observed that the DC coefficient that represents the average energy of cells is the most important coefficient. Therefore, the pattern Fig. 3(a) is constructed to encode the coefficients that are in the close proximity to DC coefficient. In the pattern Fig. 3(b), the coefficients are grouped into horizontal, vertical, and diagonal form; this pattern is construct to capture the different direction information. The pattern Fig. 3(c) is constructed as another option of the pattern Fig. 3(b). In this pattern, the differences between the sizes of the direction areas are reduced; the regions have approximately the same size. We experimented with the many variations of the patterns, however, these three patterns achieved best results.

Using this approach to encode the coefficients, the dimensionality of the feature vector of each block is 16; one 16×16 block that contains four 8×8 cells is encoded by 16 values: 4 DC coefficient + 12 averages of AC coefficients.

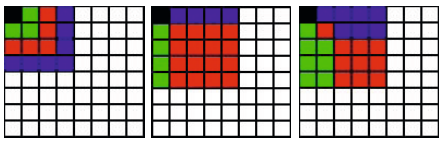


Fig. 3. The three different options of AC patterns. The areas are depicted by three different colors in that the averages of coefficients are calculated.

In our work, this dimensionality is even further reduced using PCA. Finally, the support vector machine classifier with the radial basis function kernel is trained over the proposed descriptors (in the next step) to create the final classifier.

4 Experiments

For the training phase, the positive set consists of 2300 faces and 4300 non-faces. The positive set contains the face images from the BIOID database combined with the Extended Yale Face Database B [7]. The images were manually cropped on the area of faces only. The negative set consists of 3000 images that were obtained from the MIT-CBCL database combined with the 1300 hard negative examples. For the detector based on ETF, the training images were resized to the size of 80×80 pixels (the size of sliding window was also set to this size) and the sliding window scanned 10 different resolutions of input image; the thermal fields were computed for each resolution. We experimented with the parameters of ETF and DCT, and we suggest the following configurations: $ETF_{DCT(a)}$, $ETF_{DCT(b)}$, $ETF_{DCT(c)}$. The configurations were designed with the size of temperature sources: 1 pixel; the distance between the sources: 5 pixels, the number of iterations (time) for the transfer of temperature: 100, the size of block: 16×16 pixels; the size of cell: 8×8 pixels (four cells inside each block); the horizontal step size of blocks: 8 pixels (blocks are overlapped). From Fig. 3(a), the DCT pattern is used in the $ETF_{DCT(a)}$ configuration; from Fig. 3(b) in $ETF_{DCT(b)}$ and from Fig. 3(c) in $ETF_{DCT(c)}$. All configurations consist of 1296 descriptors for one position of sliding window.

For comparison, we use the detectors that are based on the HOG features, LBP (Local Binary Patterns) features [9] and Haar features (Viola-Jones detection framework). For the HOG features, we used the identical size of the samples (80×80 pixels). We used the classical parameters of HOG descriptors; the size of block: 16×16 pixels; the size of cell: 8×8 pixels; the horizontal step size: 8 pixels; the number of bins: 9. This configuration consists of 2916 HOG descriptors for one position of sliding window; this configuration is denoted as HOG. The SVM classifier is trained over the HOG descriptors similarly in the ETF based detector. The detector based on the Viola-Jones detection framework is denoted as $Haar$, the detector based on the LBP features is denoted as LBP; we create the cascade classifiers and the training images were resized to the 19×19 pixels for these detectors. We used the identical training set (2300 positive and 4300 negative samples) and testing set for all detectors. To test the detectors, we collected 350 images from the Faces in the Wild dataset [3]. In Table 1, the detection results are shown.

The detector based on the Haar features achieved a good number of true positives (sensitivity 88.17%), nevertheless, this detector needed to increase the number of training samples. This is also the problem of LBP based detector; the number of false positives of these detectors is rather large. The detector based on HOG achieved better overall results (F1 84.67%) than LBP and HAAR based detectors, however, the HOG based detector detected faces in the wrong places (precision 73.17%). Moreover, this detector using 2× more descriptors than ETF based detector.

Due to the fact that only the detector based on the $ETF_{DCT(b)}$ configuration (with the dimensionality of feature vector 1296) achieved a very low number of false positives (precision 98.90%), we used PCA (Principal Component Analysis)

Table 1. The detection performance

	Precision	Sensitivity	F1 score
$ETF_{DCT(a)}$	97.26%	79.24%	87.33%
$ETF_{DCT(b)}$	**98.90**%	80.13%	88.53%
$ETF_{DCT(c)}$	91.84%	87.95%	89.85%
$ETF\_PCA_{DCT(b)}$	94.24%	91.29%	**92.74**%
HOG	73.17%	97.99%	84.67%
$Haar$	79.80%	88.17%	83.78%
LBP	66.27%	74.55%	70.17%

to reduce the number of descriptors; we used the 200 principal components (corresponding to the largest eigenvalues) for the detector based on this feature subset. This step had positive effect on the training phase of classifier; using reduced feature vector, the classifier was able to detect more faces (sensitivity 91.29%). The detector using PCA is denoted as $ETF\_PCA_{DCT(b)}$. This detector achieved best detection results in the test (F1 92.74%, detection results are shown in Fig. 4).

Clearly, the benefit of the DCT coefficients combined with ETF is visible; the important image information is encoded using DCT, moreover, this information can be reduced without negative impact on the detection result. The first reduction is performed when the mean of coefficients is calculated inside each region, the second reduction is performed using PCA. Finally, with this approach, faces can be efficiently encoded with the relatively small set of numbers with promising results that outperform the results of state-of-the-art detectors.

It is important to note that the complexity measurement of the proposed method can be divided into two parts; the temperature transfer process and the process of composing the feature vector. We have developed the GPU (CUDA) and CPU (SSE/AVX) versions for solving the temperature transfer process; the time of GPU version is 40 milliseconds, time of CPU version is 150 milliseconds for 640×480 images and 150 iterations. The calculation of DCT and compose the feature vector take approximately 1 milliseconds for one position of sliding window (80×80 pixels). The recognition time depends on the chosen classifier.

5 Conclusion

We proposed the improvement of encoding the temperature (energy) distribution that is useful for object description. The improvement is based on the fact that the important image information can be described using DCT coefficients. We use this premise and we propose the way to encode the distribution of temperature using DCT with very promising detection results. We also have shown that the DCT coefficients can be further compressed by PCA to achieve the feature vector of reasonable dimensionality.

Fig. 4. The detection results of $ETF\_PCA_{DCT(b)}$ configuration. The results are without the postprocessing (the detection results are not merged).

Acknowledgments. This work was supported by the SGS in VSB Technical University of Ostrava, Czech Republic, under the grant No. SP2014/170.

References

1. Alahi, A., Ortiz, R., Vandergheynst, P.: FREAK: Fast Retina Keypoint. In: IEEE Conference on Computer Vision and Pattern Recognition. IEEE, New York (2012), CVPR 2012 Open Source Award Winner
2. Bay, H., Tuytelaars, T., Van Gool, L.: Surf: Speeded up robust features. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) ECCV 2006, Part I. LNCS, vol. 3951, pp. 404–417. Springer, Heidelberg (2006)
3. Berg, T.L., Berg, A.C., Edwards, J., Forsyth, D.: Who's in the picture. In: Saul, L.K., Weiss, Y., Bottou, L. (eds.) Advances in Neural Information Processing Systems, vol. 17, pp. 137–144. MIT Press, Cambridge (2005)
4. Calonder, M., Lepetit, V., Strecha, C., Fua, P.: Brief: Binary robust independent elementary features. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part IV. LNCS, vol. 6314, pp. 778–792. Springer, Heidelberg (2010)
5. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005, vol. 1, pp. 886–893 (June 2005)

6. Fusek, R., Sojka, E., Mozdren, K., Surkala, M.: Energy-transfer features and their application in the task of face detection. In: 2013 10th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS), pp. 147–152 (2013)
7. Lee, K., Ho, J., Kriegman, D.: Acquiring linear subspaces for face recognition under variable lighting. IEEE Trans. Pattern Anal. Mach. Intelligence 27(5), 684–698 (2005)
8. Leutenegger, S., Chli, M., Siegwart, R.: Brisk: Binary robust invariant scalable keypoints. In: 2011 IEEE International Conference on Computer Vision (ICCV), pp. 2548–2555 (2011)
9. Liao, S., Zhu, X., Lei, Z., Zhang, L., Li, S.Z.: Learning multi-scale block local binary patterns for face recognition. In: Lee, S.-W., Li, S.Z. (eds.) ICB 2007. LNCS, vol. 4642, pp. 828–837. Springer, Heidelberg (2007)
10. Lowe, D.: Object recognition from local scale-invariant features. In: The Proceedings of the Seventh IEEE International Conference on Computer Vision, 1999, vol. 2, pp. 1150–1157 (1999)
11. Ojala, T., Pietikäinen, M., Harwood, D.: A comparative study of texture measures with classification based on featured distributions. Pattern Recognition 29(1), 51–59 (1996), http://dx.doi.org/10.1016/0031-3203(95)00067-4
12. Perona, P., Malik, J.: Scale-space and edge detection using anisotropic diffusion. IEEE Trans. Pattern Anal. Mach. Intell. 12, 629–639 (1990), http://dx.doi.org/10.1109/34.56205
13. Rublee, E., Rabaud, V., Konolige, K., Bradski, G.: Orb: An efficient alternative to sift or surf. In: 2011 IEEE International Conference on Computer Vision (ICCV), pp. 2564–2571 (2011)
14. Tsai, T., Huang, Y.P., Chiang, T.W.: Image retrieval based on dominant texture features. In: 2006 IEEE International Symposium on Industrial Electronics, vol. 1, pp. 441–446 (July 2006)
15. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2001, vol. 1, pp. I-511– I-518 (2001)