Extracting Spatiotemporal Interest Points using Global Information

Shu-Fai Wong and Roberto Cipolla
Department of Engineering University of Cambridge
Cambridge CB2 1PZ, UK
{sfw26, cipolla}@eng.cam.ac.uk

Abstract

Local spatiotemporal features or interest points provide compact but descriptive representations for efficient video analysis and motion recognition. Current local feature extraction approaches involve either local filtering or entropy computation which ignore global information (e.g. large blobs of moving pixels) in video inputs. This paper presents a novel extraction method which utilises global information from each video input so that moving parts such as a moving hand can be identified and are used to select relevant interest points for a condensed representation. The proposed method involves obtaining a small set of subspace images, which can synthesise frames in the video input from their corresponding coefficient vectors, and then detecting interest points from the subspaces and the coefficient vectors. Experimental results indicate that the proposed method can yield a sparser set of interest points for motion recognition than existing methods.

1. Introduction

Methods based on local features or interest points have shown a promising result in motion recognition [14, 1, 6, 12, 11]. This part-based approach (see Figure 1 for an illustration) uses only several 'interesting' parts of a whole spatiotemporal (ST) volume for analysis and thus avoids problems such as non-stationary backgrounds. In addition, unlike the holistic approach which analyses a whole ST volume (e.g. [3, 15]), this approach is more tolerant to large geometric variations between intra-class samples and is more flexible. Recent works such as [1] also showed that the part-based approach yields higher recognition accuracy than the holistic approach.

Although this approach provides a good performance in motion recognition, there is only a limited number of reported works on spatiotemporal interest point (or cuboid) detection. The most straight-forward way to detect cuboids is to extend 2D interest point detection algorithms (e.g. those in [13]) to 3D video analysis. In [7], Laptev extended

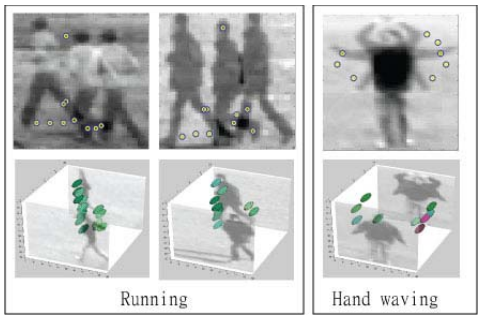


Figure 1. The top row shows superposition images of two walking sequences and a hand waving sequence. Spatiotemporal (ST) interest points (detected by the method proposed in [1]) are also displayed. The bottom row shows the 3D visualisation of those ST points with colour indicating the similarity of their appearances.

2D Harris corner detector to a 3D Harris detector which detects regions having high intensity variations in both spatial and temporal dimensions (i.e. spatiotemporal corners). However, it usually gives only a small amount of detections which is insufficient for most part based classifiers.

Dollár et al. [1] improved the 3D Harris detector by relaxing its constraints so that a larger amount of detections can be obtained. Their detector applies Gabor filtering (for detecting intensity variations) on the temporal domain only and selects regions which give high responses. By setting appropriate spatial and temporal scales, the detector can give a large amount of interest points correspond to regions undergo complex motions.

An entropy approach [12, 4] has also been proposed to obtain a large amount of detections. In these works, entropy is first computed for each small ST volume or cuboid, and cuboids with large entropy values are reported as salient points. Their scales can be determined through maximising the entropy values.

The above detection methods use solely local information such as the distribution of pixel intensities within a small region. The implication is that a certain number of detections may be generated by video noises instead of the

motion being analysed. An interest point selection scheme is required if there is a need to have a compact representation of the original ST volume.

Discriminative interest points can be selected using boosting [6, 16]. Given a large number of training samples, boosting can select the most discriminative and representative interest points from a set of randomly generated cuboids. However, boosting usually requires a huge amount of training samples, making it less applicable to small motion datasets.

In this paper, we present a novel method which uses global information for detecting and selecting interest points from a single video. It extracts structural information (e.g. the location of moving parts) from the video input and detects cuboids in regions having a larger probability of containing the relevant motion.

This paper has 3 main contributions: we propose a novel method which utilises global information for detecting interest points in videos. In addition, we evaluated the performance of several state-of-the-art detectors in motion recognition tasks. Finally, we show that our method gives the most compact representation for the motion recognition tasks.

2. Approach

2.1. Latent structure

2.1.1 Dynamic Textures

Inspired by the work on extraction of dynamic textures [2] where a video sample can be broken down into a latent representation and a dynamic generation model, we can use a similar approach for extracting global information from a video input. In [2], given a sequence of images $y(t)_{t=1..\tau}$, the dynamic texture associated with the sequence can be formulated as:

$$x(t+1) = Ax(t) + Bv(t), \quad (1)$$

$$y(t) = \phi(x(t)) + w(t). \quad (2)$$

In the equations, $x(t)$ is a latent representation of an input image $y(t)$ and ϕ defines the transformation from the latent representation to the image representation. The dynamic expressed in the images is represented by A which describes the transition of the corresponding latent representations. To achieve a compact representation of a video input, ϕ can be approximated as a linear transformation which converts x in a lower dimension space to y in a higher dimension space. Thus, the original sequence $y(t)_{t=1..\tau}$ can now be compressed into a set of subspaces ϕ and a set of coefficients $x(t)_{t=1..\tau}$. If the motion involved is periodic, the original sequence can be compressed further as an initial state coefficient $x(0)$, a set of subspaces ϕ, and transition matrices A and B.

Through the above dynamic texture analysis, a set of compact representations, which can synthesise a denoised version of the original video, is obtained. Even if the motion involved is not periodic, it is still possible to represent the sequence by a set of subspaces ϕ and a set of coefficients $x(t)_{t=1..\tau}$. In this paper, we will focus on this set of representations which give the global information for interest point detection.

In [2], the latent representation is obtained by singular value decomposition (SVD) where the data matrix Y with each image represented as a column vector is decomposed to $U\Sigma V^T$. The set of subspaces C is then obtained as U and the set of coefficients X is then obtained as ΣV^T. This representation, however, does not have much intuitive meaning in identifying interest points.

2.1.2 Non-negative matrix factorisation

Instead of using SVD, we propose to use non-negative matrix factorisation (NNMF) [8] to obtain the subspace representation. Similar to SVD, the objective function involves minimising the difference between the original data matrix and the approximated matrix: $\|Y - CX\|^2$, where Y is the data matrix, C is the subspaces and X is the coefficient matrix. NNMF adds a non-negativity constraint to this objective function such that all entries in C and X are positive. The decomposition can be done using an iterative algorithm involving multiplicative update rules [9] in each iteration, itr:

$$C_{ia}^{itr+1} = C_{ia}^{itr} \frac{(Y(X^{itr})^T)_{ia}}{(C^{itr}X^{itr}(X^{itr})^T)_{ia}}, \forall i, a, \quad (3)$$

$$X_{bj}^{itr+1} = X_{bj}^{itr} \frac{((C^{itr+1})^T Y)_{bj}}{((C^{itr+1})^T C^{itr+1} X^{itr})_{bj}}, \forall b, j. \quad (4)$$

According to [8], there are two interesting observations in NNMF decomposition: (1) each subspace image contains a sparse set of positive pixels, and (2) this decomposition gives a large fraction of vanishing coefficients. Based on these observations, an intuitive notion of this decomposition is that each original image $y(t)$ can be decomposed to a set of small parts which correspond to the positive entries in the subspace matrices. An example in [8] shows that given a set of face images, the positive entries in the subspaces usually correspond to facial components such as eyes, noses and mouths.

Instead of obtaining facial components, we exploit NNMF to extract a sparse set of 'motion components' from the original image sequence in this work. Given a sequence of images organised in a column matrix $I = [I_1..I_\tau]$, a feature matrix is obtained from $Y = I - \overline{I}$ where $\overline{I}$ is the mean matrix. The feature matrix is then decomposed

into Y^+ with positive entries only and Y^- with negative entries only. Each matrix is decomposed using NNMF: $Y^+ = C^+X^+$ and $-Y^- = C^-X^-$. The main idea is simplified and summarised in Figure 2.

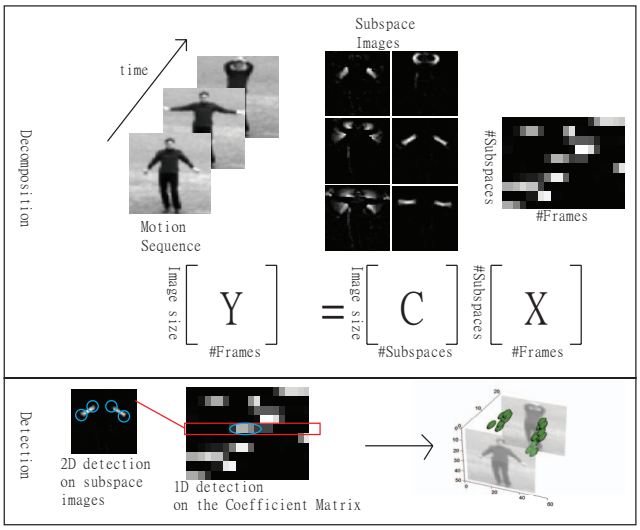


Figure 2. An overview of our proposed method which decomposes a data matrix into a subspace matrix and a coefficient matrix for interest point detection.

As shown in Figure 2, the subspaces or the 'motion components' obtained usually correspond to the moving parts such as moving hands. In addition, the coefficient matrices are sparse (see also Figure 3). These imply (1) each input image Y_t (i.e. each column in the feature matrix Y) can be considered as a combination of these 'motion components' which are shared by the whole image sequence and (2) motion occurs when some of these 'motion components' appear or disappear (i.e. the transition from zero to positive values or from positive to zero along rows in the coefficient matrix).

These implications can be exploited in interest point detection based on our intuitive understanding that interest regions usually appear near the tip regions of these 'motion components' (e.g. finger, shoulder and some joint regions) and the appearance or disappearance of these tip regions may be distinctive features for motion recognition. Therefore, instead of analysing the raw image sequence, it is possible to detect and select interest regions from the 'motion components' and their associated coefficients. In other words, the original 3D interest point detection problem is broken down into a problem on 2D interest point detection in the subspace images and a problem on 1D interest point detection along each coefficient vector as shown in Figure 2.

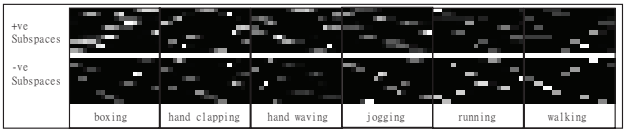


Figure 3. The coefficient matrices obtained from different motion classes in KTH motion dataset.

2.2. Spatial Interest Points

In order to obtain a set of 2D interest points and their spatial scales, Difference of Gaussian (DoG) detector [10] is exploited. We chose DoG because it is relatively faster than other commonly used spatial interest point detectors which cannot improve the final recognition performance significantly.

2.3. Temporal Interest Points

Difference of Gaussian (DoG) detector was modified to detect interest points and report their scales in each coefficient vector. It is also possible to locate 1D interest points by multi-scale non-maximum suppression and maxima detection for better efficiency.

2.4. Ranking of Interest Points

Eventually, the spatial interest points from each subspace image and the temporal interest points from the corresponding coefficient vector can give a set of spatiotemporal interest points which indicate the appearance or disappearance of the 'motion components'. However, not all reported interest points are relevant. This is because the feature matrix Y is given by subtracting the mean image from the input image but not subtracting a background image from it and the non-zero values in the feature image Y_t do not imply non-stationary or motion areas. To remove irrelevant interest points, each reported interest point is evaluated to see whether it contains motion. In this work, a saliency measure is chosen for ranking interest points so that points with homogeneous textures are removed. As in [5], the saliency measure is defined as:

$$R = -\sum_{l \in L} p(l) \log p(l), \tag{5}$$

where l is one of a set of L possible intensity values (e.g. 255 if the input is a greyscale image) and $p(l)$ gives the probability of observing the intensity value l within the reported interest region. A finite number of the most salient points are selected eventually.

3. Implementation details

Various state-of-the-art spatiotemporal interest point detectors were compared with ours in the experiments. A brief

review of these detectors and the setting used will be presented in the following sub-sections.

3.1. Laptev Detector

Laptev [7] extends the Harris corner detector to analyse videos by modifying the Harris corner function to:

$$R = \det(\mu) - k\mathrm{trace}^3(\mu), \quad (6)$$

$$\mu = g(\cdot;\sigma^2,\tau^2) * \begin{pmatrix} L_x^2 & L_xL_y & L_xL_t \\ L_xL_y & L_y^2 & L_yL_t \\ L_xL_t & L_yL_t & L_t^2 \end{pmatrix}, \quad (7)$$

where $g(\cdot;\sigma^2,\tau^2)$ is a 3D Gaussian smoothing kernel with a spatial scale σ and a temporal scale τ and $L_{x,y,z}$ are the gradient functions along x, y, z directions. The points with large R are detected. In the experiments, σ and τ were set to 2 and 4 respectively and k was set to 0.005.

3.2. Dollár et al. Detector

Dollár et al. [1] detector applies Gabor filtering on temporal domain and select regions which give high responses, R:

$$R = (I * g * h_{ev})^2 + (I * g * h_{od})^2, \quad (8)$$

$$h_{ev}(t;\tau,\omega) = -cos(2\pi t\omega)e^{-t^2/\tau^2}, \quad (9)$$

$$h_{od}(t;\tau,\omega) = -sin(2\pi t\omega)e^{-t^2/\tau^2}, \quad (10)$$

where g is a 2D Gaussian smoothing kernel $g(x,y;\sigma)$, and h_{ev} and h_{od} are the quadrature pair of a 1D Gabor filter. There are 2 main parameters in the detector, namely σ and τ which correspond to spatial and temporal scales of the detector, and we can set $\sigma = 2$, $\tau = 4$, and $\omega = 4/\tau$ as recommended by Dollár et al. [1].

3.3. Saliency Detector

Saliency detectors [5, 12, 4] use the same response function in both spatial and spatiotemporal analysis. The response given by a region under investigation is shown in Equation 5. In the experiment, the default size of the regions was $\sigma = 2$ and $\tau = 4$.

3.4. NNMF Detector

The details of our proposed detector, NNMF detector, can be found in Section 2. Although this method can select spatial and temporal scale automatically, we fix it for a fair comparison between different detectors. The scales used are $\sigma = 2$ and $\tau = 4$ (i.e. the same as the other detectors). The number of subspaces used was 10.

3.5. Feature Encoding

The interest regions returned (i.e. cuboids) are encoded into feature vectors for further analysis. The size of the cuboid to be encoded is around three times the scales along each dimension. According to our experiments and suggestion from Dollár et al. [1], spatiotemporal gradient feature was used to obtain the best recognition result.

4. Experiments

We conducted experiments in three different domains, namely human activity, facial expression and hand gesture. To obtain a fair judgement of the performance of our detector, we compare our result to those obtained using different detectors mentioned in the previous section. The performance of a detector is evaluated by measuring the accuracy of motion classification using the interest points detected by it. The algorithms for motion recognition and the datasets used will be described in the following subsections.

4.1. Algorithms for motion recognition

Quantised feature vector: Through interest point detection, each video sample gives a set of interest points which are encoded as spatiotemporal gradient features. Interest points from a certain amount of samples are used to form an appearance codebook using k-mean clustering on the gradient features. Similarly, a location codebook is formed from clustering of the spatiotemporal locations of the interest points. The interest point features are then vector quantised into codewords according to the codebooks and each video sample is eventually represented as a histogram of codewords. This histogram represents both appearance and location information of interest points and samples having a similar set of interest points and a similar distribution of them in spatiotemporal space will give similar histograms.

Probabilistic latent semantic analysis (pLSA): It is a popular unsupervised method for learning object categories from interest point features and it was implemented based on Niebles et al. work [11]. Histogram features of training or testing samples are concatenated to form a co-occurrence matrix which is an input of the pLSA algorithm. The number of topics used was the same as the number of motion classes in the datasets used. The number of iteration in the EM algorithm was set to 100 and the algorithm was implemented in Matlab running in a P4 1GB memory computer.

Support Vector Machines (SVM): The algorithm was also implemented in Matlab. The input is a normalised histogram (i.e. a column vector in the co-occurrence matrix used in the pLSA algorithm). The SVM kernel used is a radial kernel. This implementation can be considered as a simplified version of Schuldt et al. work [14].

Nearest Neighbour classifier (NNC): As in Dollár et al. work [1], a NNC classifier based on χ^2 distance was im-

plemented in Matlab. The input is a normalised histogram which is the same as the one used in the above methods.

4.2. Datasets

4.2.1 Human activity dataset: KTH dataset

The human activity dataset used in the experiments were collected by Schuldt et al. [14]. There are totally six human activities including boxing, hand clapping, hand waving, jogging, running and walking (see Figure 4). There are 25 subjects engaged in the above activities in totally 4 different scenarios including indoor, outdoor, changes in clothing and variations in scale. Each video sample contains one subject engaged in a single activity in a certain condition and there are totally 593 samples in the dataset.

The raw video samples were processed such that each sample contains one iteration of the action and the processed samples have an average size of ($x : 80 \times y : 80 \times t : 15$). The processed videos from 5 subjects were used to generate the codebooks, and these videos were not used in training and test stages. We performed leave-one-out cross-validation to evaluate our algorithm. In other words, videos from 19 subjects were used in training and the remaining one was used in the test. The result is reported as the average of 20 runs.

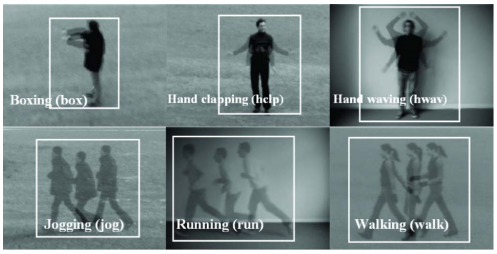


Figure 4. Superposition images of action classes in KTH dataset.

4.2.2 Facial expression dataset

The facial expression dataset used were collected by Dollár et al. [1]. There are 2 subjects in the dataset and each of them expresses six different emotions (anger, disgust, fear, joy, sadness and surprise) for 8 times under 2 different lighting conditions (see Figure 5). Each video sample contains one facial expression and the average size of the samples is ($x : 152 \times y : 194 \times t : 100$). There are totally 192 samples in the dataset. The codebook was generated from videos captured from one lighting condition. We performed leave-one-out cross-validation to evaluate our algorithm so that samples from one subject in a certain lighting set-up were used in the test and the remaining were used in training.

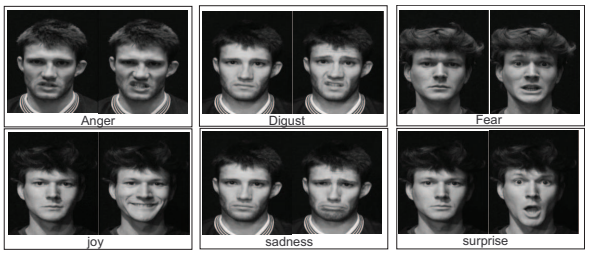


Figure 5. Sample images from the facial expression dataset used.

4.2.3 Hand gesture dataset

The hand gesture dataset used were obtained by us. There are nine different gestures (see Figure 6) signed by two subjects under 5 lighting set-ups and each subject was asked to sign for 10 times in each set-up so that the dataset contains 900 samples. Each video sample contains one gesture and the average size of the samples is ($x : 320 \times y : 240 \times t : 100$). The codebook was generated from samples captured in the normal lighting condition and we performed leave-one-out cross-validation for evaluation where samples captured from any three of the remaining lighting conditions were used in training.

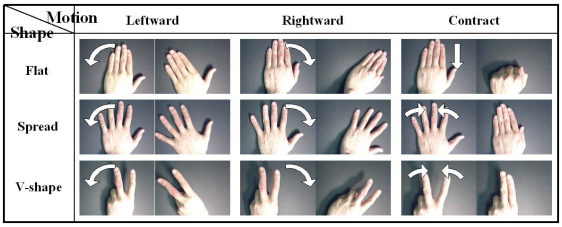


Figure 6. Nine gesture classes in the gesture dataset used.

5. Results

5.1. Human Activity

We ran this experiment on KTH dataset for 20 times using leave-one-out cross-validation. In average, the training set contained around 450 samples in each run, and each video sample gave 30 interest points. The histogram features had 200 entries. Table 1 summarises the recognition results obtained from various methods and Figure 7(a) shows the confusion matrix obtained by our method.

5.2. Facial Expression

We ran this experiment on Dollár et al. dataset for 3 times. In each test, the training set contained around 96 samples and each sample gave 30 interest points. The histogram features had 100 entries. Table 1 summarises the results obtained from various methods and Figure 7(b) shows

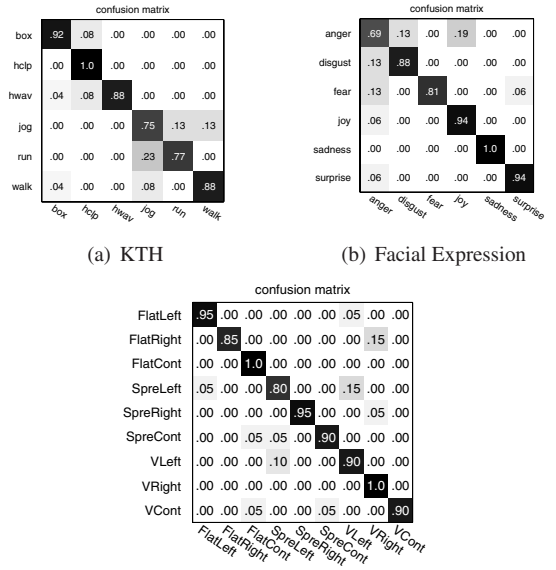
the confusion matrix obtained by our method.

5.3. Gesture

We ran this experiment on our gesture dataset for 4 times. In each test, the training set contained around 540 samples and each sample gave 30 interest points. The histogram features had 200 entries. Table 1 summarises the results obtained from various methods and Figure 7(c) shows the confusion matrix obtained by our method.

	Laptev	Dollár et al.	Saliency	NNMF
KTH:				
- pLSA	20.57	64.08	61.97	73.24
- SVM	29.79	85.92	66.90	86.62
- NNC	26.95	75.67	64.79	80.06
Facial Expression:				
- pLSA	23.96	60.33	40.63	68.50
- SVM	36.46	80.56	61.46	87.50
- NNC	31.25	75.00	44.79	78.13
Gesture:				
- pLSA	15.24	64.44	60.56	78.33
- SVM	29.27	88.89	71.11	91.67
- NNC	25.61	79.44	64.44	87.22

Table 1. Comparison of recognition results (i.e. the accuracy in percentage) obtained from various methods.



(a) KTH (b) Facial Expression

(c) Gesture

Figure 7. Confusion matrices generated by applying our detector and SVM on the three datasets used.

The experimental results indicate that our NNMF detector and Dollár et al. detector lead to a better recognition accuracy than Laptev detector and saliency detector while our detector's performance is slightly better than Dollár et al.'s. Figure 8 to 11 show the detection results obtained from these detectors. It can be observed that Laptev detector gives insufficient interest points while saliency detector reports a lot of points which is not discriminative enough. Therefore, their performances are the worst. The figures also show that our NNMF detector usually reports interest points near the moving limbs and facial components. It seems that these points are more discriminative than those given by Dollár et al. detector and lead to a slightly better recognition accuracy. This will be verified in the next experiment.

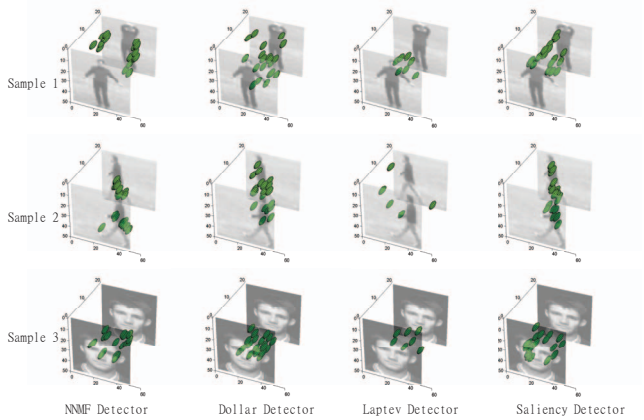


Figure 11. Each row shows the 3D visualisation of interest points obtained by applying various detectors on the samples illustrated in Figure 8, 9, and 10.

5.4. Number of interest points

In this experiment, the maximum number of interest points used for recognition was reduced in each run. By doing so, we can roughly know about the minimum amount of interest points required for achieving certain recognition accuracy, say 70%. The recognition algorithm used was Nearest Neighbour Classifier (NNC) which is the simplest classifiers among the others used in the previous experiment. Table 2 summarises the results obtained using our detector and Figure 12 shows the confusion matrix obtained by applying our detector and NNC which used only 5 interest points for recognition.

	Number of interest points used				
Dataset	25	20	15	10	5
KTH	80.99	79.58	80.28	76.06	71.13
Facial Expression	77.08	80.21	76.04	77.08	65.63
Gesture	84.44	80.00	77.78	70.56	65.56

Table 2. The effect of reducing the number of interest points in recognition tasks.

The experimental results show that relatively high recognition accuracy (i.e. 80%) could be maintained if more than 15 NNMF interest points were used and the minimum amount of NNMF interest points required for achieving an

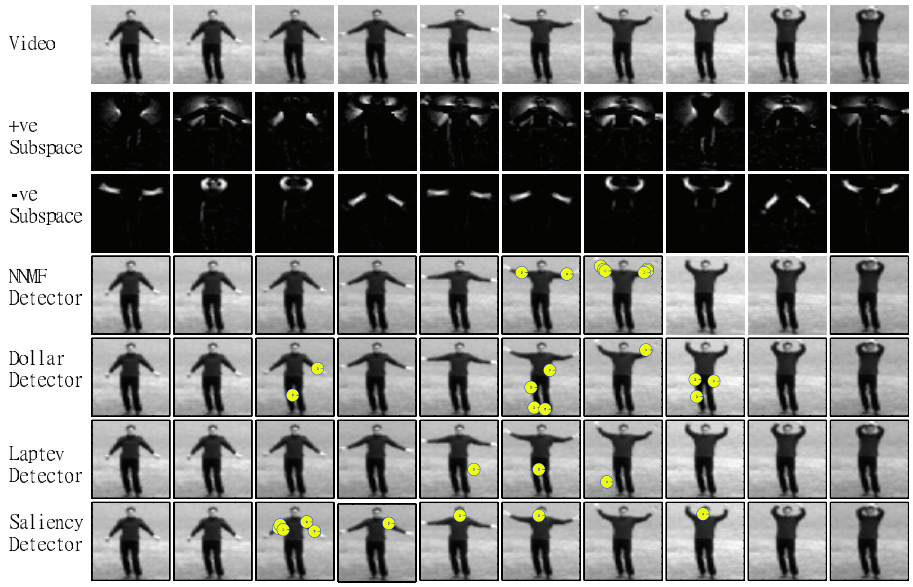


Figure 8. Interest point detection results on sample 1 ('hand-waving' class in KTH dataset). The first row shows the image sequence in the sample. The second and third rows show the subspace images obtained by NNMF. The last four rows show the interest points detected by various methods compared in the experiment.

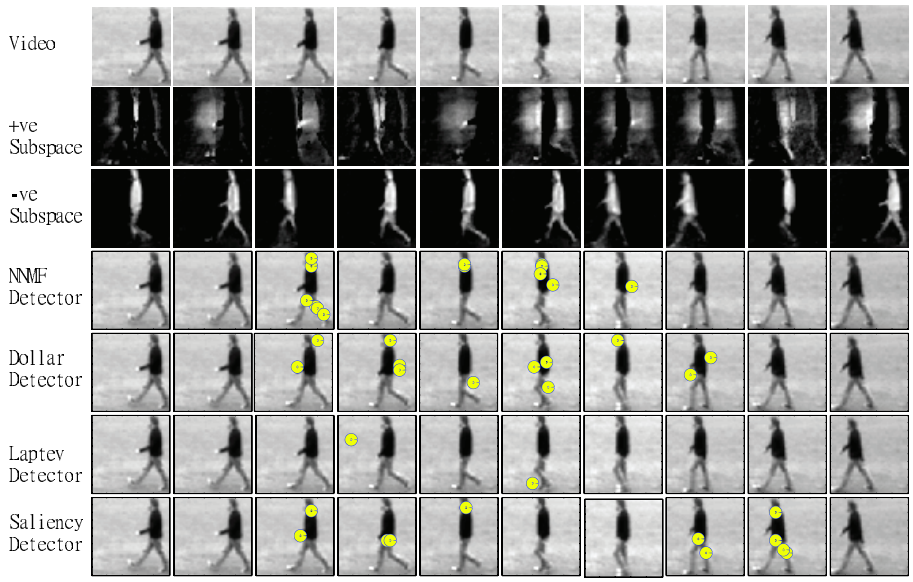


Figure 9. Interest point detection results on sample 2 ('walking' class in KTH dataset). The first row shows the image sequence in the sample. The second and third rows show the subspace images obtained by NNMF. The last four rows show the interest points detected by various methods compared in the experiment.

accuracy rate of 70% is 10. Even if the number was reduced further to 5, the recognition accuracy is not too bad (still over 65%) and the confusion matrices show that several motion classes can still be recognised accurately (over 80%). Other detectors, on the other hand, lead to recognition accuracy below 50% if only 5 interest points were used. These observations show that our NNMF detector can give a sparse set of discriminative points.

6. Conclusion

This paper introduces a novel method for detecting spatiotemporal interest points. Unlike recent works which use local information only, our detector utilises global information (i.e. the organisation of pixels in a whole

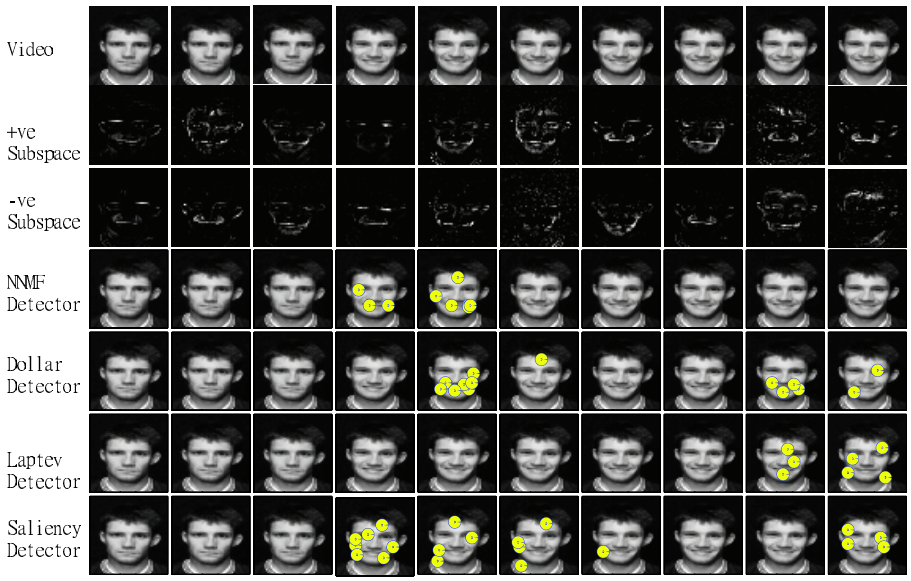
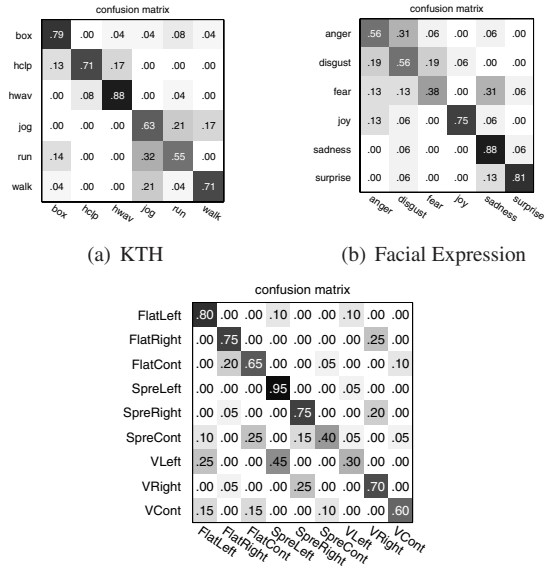


Figure 10. Interest point detection results on sample 3 ('joy' class in Facial Expression dataset). The first row shows the image sequence in the sample. The second and third rows show the subspace images obtained by NNMF. The last four rows show the interest points detected by various methods compared in the experiment.



(a) KTH (b) Facial Expression

(c) Gesture

Figure 12. Confusion matrices generated by applying our detector and Nearest Neighbour classifier which compares 5 interest points.

video) to detect and select a sparse set of interest points. Experimental results indicate that our detector can select a compact and discriminative set of interest points for motion recognition tasks.

Acknowledgements. SW is funded by the Croucher Foundation.

References

[1] P. Dollár, V. Rabaud, G. Cottrell, and S. Belongie. Behavior recognition via sparse spatio-temporal features. In *ICCV Workshop: VS-PETS*, 2005.

[2] G. Doretto, A. Chiuso, Y. N. Wu, and S. Soatto. Dynamic textures. *IJCV*, 51(2):91–109, 2003.

[3] A. A. Efros, A. C. Berg, G. Mori, and J. Malik. Recognizing action at a distance. In *ICCV*, pages 726–733, 2003.

[4] H. Hung and S. Gong. Quantifying temporal saliency. In *BMVC*, pages 742–749, 2004.

[5] T. Kadir, A. Zisserman, and M. Brady. An affine invariant salient region detector. In *ECCV*, 2004.

[6] Y. Ke, R. Sukthankar, and M. Hebert. Efficient visual event detection using volumetric features. In *ICCV*, 2005.

[7] I. Laptev. On space-time interest points. *IJCV*, 64(2-3):107–123, 2005.

[8] D. D. Lee and H. S. Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, October 1999.

[9] D. D. Lee and S. H. Seung. Algorithms for non-negative matrix factorization. In *NIPS*, volume 13, pages 556–562, 2000.

[10] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004.

[11] J. C. Niebles, H. Wang, and F.-F. Li. Unsupervised learning of human action categories using spatial-temporal words. In *BMVC*, 2006.

[12] A. Oikonomopoulos, I. Patras, and M. Pantic. Spatiotemporal salient points for visual recognition of human actions. *IEEE Trans. Systems, Man, and Cybernetics, Part B*, 36(3):710–719, June 2006.

[13] C. Schmid, R. Mohr, and C. Bauckhage. Evaluation of interest point detectors. *IJCV*, 37(2):151–172, 2000.

[14] C. Schuldt, I. Laptev, and B. Caputo. Recognizing human actions: A local svm approach. In *ICPR*, pages 32–36, 2004.

[15] E. Shechtman and M. Irani. Space-time behavior based correlation. In *CVPR*, pages 405–412, 2005.

[16] P. Smith, N. da Vitoria Lobo, and M. Shah. Temporalboost for event recognition. In *ICCV*, pages 733–740, 2005.