University of Wollongong

Research Online

Faculty of Informatics - Papers (Archive)

Faculty of Engineering and Information
Sciences

1-1-2009

Kernel PCA of HOG features for posture detection

Peng Cheng

University of Wollongong, pc175@uow.edu.au

Wanqing Li

University of Wollongong, wanqing@uow.edu.au

Philip Ogunbona

University of Wollongong, philipo@uow.edu.au

Follow this and additional works at: <https://ro.uow.edu.au/infopapers>

Part of the [Physical Sciences and Mathematics Commons](#)

Recommended Citation

Cheng, Peng; Li, Wanqing; and Ogunbona, Philip: Kernel PCA of HOG features for posture detection 2009, 415-420.

<https://ro.uow.edu.au/infopapers/2126>

Research Online is the open access institutional repository for the University of Wollongong. For further information contact the UOW Library: research-pubs@uow.edu.au

---

Kernel PCA of HOG features for posture detection

Abstract

Motivated by the non-linear manifold learning ability of the Kernel Principal Component Analysis (KPCA), we propose in this paper a method for detecting human postures from single images by employing KPCA to learn the manifold span of a set of HOG features that can effectively represent the postures. The main contribution of this paper is to apply the KPCA as a non-linear learning and open-set classification tool, which implicitly learns a smooth manifold from noisy data that scatter over the feature space. For a new instance of HOG feature, its distance to the manifold that is measured by its reconstruction error when mapping into the kernel space serves as a criterion for detection. And by combining with a newly developed KPCA approximation technique, the detector can achieve almost real-time speed with neglectable loss of performance. Experimental results have shown that the proposed method can achieve promising detection rate with relatively small size of positive training dataset.

Keywords

kernel, pca, hog, features, detection, posture

Disciplines

Physical Sciences and Mathematics

Publication Details

Cheng, P., Li, W. & Ogunbona, P. (2009). Kernel PCA of HOG features for posture detection. International Conference: Image and Vision Computing New Zealand (pp. 415-420). Wellington, New Zealand: IEEE.

Kernel PCA of HOG features for posture detection

Peng Cheng ^{#1}, Wanqing Li ^{#2}, Philip Ogunbona ^{#3}

School of Computer Science and Software Engineering

University of Wollongong

Wollongong, Australia

Email: ¹pc175@uow.edu ²wanqing@uow.edu ³philipo@uow.edu

Abstract—Motivated by the non-linear manifold learning ability of the Kernel Principal Component Analysis (KPCA), we propose in this paper a method for detecting human postures from single images by employing KPCA to learn the manifold span of a set of HOG features that can effectively represent the postures. The main contribution of this paper is to apply the KPCA as a non-linear learning and open-set classification tool, which implicitly learns a smooth manifold from noisy data that scatter over the feature space. For a new instance of HOG feature, its distance to the manifold that is measured by its reconstruction error when mapping into the kernel space serves as a criterion for detection. And by combining with a newly developed KPCA approximation technique, the detector can achieve almost real-time speed with neglectable loss of performance. Experimental results have shown that the proposed method can achieve promising detection rate with relatively small size of positive training dataset.

I. INTRODUCTION

Vision-based posture detection refers to the problem of locating specific human postures in image or video. Solution to the problem not only benefits for human motion analysis [1], but also for semantic image/video retrieval. In this paper, we are particular interested in detecting postures from images under realistic assumptions: (a) training samples are only available for the postures to be detected, and (b) there are no negative samples (e.g. samples for uninterested postures and non-human scene).

Intuitively, the problem may be addressed in two steps in which humans are detected first and then their postures are classified. Under our assumptions, the two step solution becomes infeasible because there are no sufficient training samples for reliable human detection [2], [3]. In addition, the postures are need to be classified into an open-set of categories representing the postures to be detected and background (including the uninterested postures) for which we don't have any training samples.

A posture represents a subset of the configurations of the human body. One of the key issues in posture detection is the effective representation of the postures. Study on the problem of estimating 3D human poses from images or image sequences has provided us with much knowledge with this respect. The essence of the pose estimation is to seek an effective representation of the pose appearance in the images and the 3D joint configuration of the body that is most likely to generate the appearance. Effective representation of the appearance pattern for a given 3D configuration would reduce the ambiguity of estimation. In [4], [5], the appearance

pattern is approximated by a cluster representing as a Gaussian Mixture Model (GMM) and, in [6], [7], [8], the pattern is represented by a non-linear smooth manifold. The the intrinsic relations between the posture configuration and the appearance is achieved through a regressor. According to the results obtained by [6], [7], [8], manifold representation usually has a better performance than cluster representation. This can be explained by the fact that the body articulation that causes the major in-class (within-posture) variations is concentrated on a few joint angles and these variations can be effectively enclosed in a lower-dimensional manifold. In contrast, the piecewise GMM cluster representation will become too sparse and hard to be accurately estimated when the number of examples is too few or the dimensionality of the feature space is too high (i.e, curse of dimensionality). The ability of manifold representation in dealing with cluttered background has been further demonstrated by the work [9] on estimating upper-body poses by combining the regressor with histograms of orientation (HOG) [10] feature and non-negative matrix factorization (NMF) based background subtraction scheme.

In this paper, we further develop the idea of [9] and devise a new method for posture detection from a single image. We also use HOG as our image feature, but the NMF-based background suppression and regression-based manifold representation are replaced with kernel PCA [11]. For each posture to be detected we learn its in-class variations through embedding its training examples in one manifold. For a test image, the Bhattacharyya distance between its HOG feature and its projection onto each manifold are used to decide whether the test image contains a specified posture or not. The experimental results have shown that the proposed approach can successfully identify specific human postures from cluttering background with average 94% detection rate at 0.05 false positive rate, by using small number of training examples for each posture and no negative examples at all.

A major problem in applying KPCA is that the computational cost is proportional to the number of training samples that define the kernel models and is often not affordable in practice when the number is large. This problem can be alleviated by introducing a newly developed KPCA approximation technique [12], which aims to accelerate the KPCA mapping process by discarding a number of redundant training samples, of which contribution to the KPCA mapping are trivial. This technique is designed to balance the tradeoff between computational efficiency and accuracy of the distance

measuring, and allows the proposed detection approach to be performed with significant smaller computation cost without noticable loss of performance, resulting an almost real-time detector. Performance and efficiency are compared between the detectors using the proposed approximated KPCA model and the original KPCA model, the results have shown that the detector can be applied twice as faster as before after the approximation algorithm is introduced, without significant loss of detection rate.

The rest of this paper is organized as follows. Section II presents the proposed method, including a brief description of kernel PCA (KPCA) for manifold learning and the proposed detection algorithm based on the Bhattacharyya distance. Section III propose a method that can accelerate the speed of detection with minimum loss of performance. Experimental results are given in Section IV. Section V concludes the paper.

II. PROPOSED METHOD

The proposed posture detection method adopts the HOG as the features and consists of two phases: training and detection. In the training phase, the manifold is learned through KPCA for each posture to be detected from its training samples. In the detection phase, HOG is extracted from the test image and project to the manifold of every learned postures. The reconstruction error between the original HOG and the HOG preimage of the test image in the KPCA space is used to determine whether the test image contains a human in a specific posture. Details of the proposed method is presented below.

A. Extraction of HOG

We construct the histogram of gradient (HOG) feature [10] by dividing the image into multiple overlapping blocks of the same size and quantizing the gradient direction of all pixels into 9 directions. For each block, its histogram is formed such that the occurrence for each direction represents the total gradient magnitude of the gradient along the direction and the histogram reflects the weighted gradient distribution inside each block. The HOG of each block is then normalized so the sum of elements of each block HOG is 1, and the HOG of the entire image is formed by concatenating the block HOGs. If the gradient magnitudes of all pixels in one block are zero, the corresponding block HOG will be normalized to uniformly distributed (each of its elements are set to $1/9$). In our approach, each block is an 8×8 image patch that has 50% overlapping with its neighbors, in total 31×15 blocks are used to cover a 128×64 image window. The HOG for the image window is a $31 \times 15 \times 9 - d$ vector, and the sum of all elements in one HOG is 31×15 . Finally, all components of the HOG are square-rooted, this is to allow Bhattacharyya distance between 2 HOGs being evaluated faster, we will explain the reason of doing so in later context.

According to the report [13], HOG outperforms other local descriptors in locality and invariance characteristics. The HOG has also been proved effective in many previous works [14] [15] [9].

B. Manifold representation using Kernel PCA

Kernel PCA (KPCA) is a non-linear manifold learning technique that can be regarded as a kernel expansion of the conventional linear PCA [16]. It features a kernel mapping $\phi(x) : x \rightarrow k(x, \cdot)$ on dataset $X = x_1, x_2, \dots, x_n$ followed by a linear principal component analysis (PCA) on the mapped dataset $\Phi = \phi(x_1), \phi(x_2), \dots, \phi(x_n)$, where $k(\cdot, \cdot)$ is a predefined kernel function. If the kernel function $k(\cdot, \cdot)$ is positive-semidefinite (we will only use the Gaussian radial basis function (RBF) kernel: $k(x, y) = e^{-\frac{\|x-y\|^2}{2\sigma^2}}$), the mapped dataset Φ is always in a infinite-dimensional space (a.k.a. feature space) where all its elements are independent. Therefore this design allows non-linear pattern of the original dataset to be effectively straightened in the infinite-dimensional feature space, thus makes a linear PCA able to find the non-linear principal manifold span of the original dataset as it extract the linear principal components of the mapped dataset in the feature space. Here we will not elaborate the mechanism of KPCA, but only briefly describe its algorithm:

Given an set of HOGs $X = \{x_1 \dots x_n\}$ extracted from the n training samples for a specific posture, where x_i is the HOG for the i 'th sample. A KPCA model that represents its manifold span can be train in 2 steps[11]:

- 1) Define the kernel $k(x, y)$ and calculate the kernel matrix

$$K = [K_{ij}], K_{ij} = k(x_i, x_j)$$

and centered kernel matrix

$$\hat{K} = HKH$$

where $H = I_n - 1_n$, I_n is an $n \times n$ identity matrix and 1_n denotes a $n \times n$ matrix in which each element takes the value $1/n$.

- 2) Compute d largest eigenvalues $\lambda_1 \dots \lambda_d$ and the corresponding eigenvectors

$$A = [a_1, \dots, a_d]$$

of $\hat{K}$. where $a_i, i = 1, 2, \dots, d$ are all n dimensional column vectors and scaled such that $|a_i| = \frac{1}{\lambda_i}$.

Matrix A and all training samples are stored as the parameters of the KPCA model, they implicitly define the shape of the manifold span of the HOGs. For a new test image, the projection of its HOG z onto the first d principal components in the feature space can be given by:

$$y(z) = w(z) - b \quad (1)$$

$$w(z) = (H_A)^T (k(x_1, z), \dots, k(x_n, z))^T \quad (2)$$

$$= (H_A)^T k(X, z)$$

$$b = (H_A)^T K 1_{1,n} \quad (3)$$

where $H_A = HA$ and both $w(z)$ and b are d -dimensional vectors representing the projection of $\phi(z)$ onto the principal components and the mean of $\phi(x)$ over the n samples, and $1_{1,n}$ is the column vector of 1_n

In the learning stage, KPCA will be applied to the training samples of each posture which results in a set of KPCA models, each model consisting of the training samples and an eigenmatrix A . These learned models will implicitly represent the manifold representation of postures. Comparing to other manifold learning technique KPCA has some unique advantages. First, the smoothness of the manifold is strictly lower-bounded by the smoothness of the kernel function $k(\cdot, \cdot)$, which can be easily tuned to have the best generalization ability and avoid overfitting problem. Second, the KPCA is almost immune to noises in the training dataset, and noises in the test datum can be effectively removed by projecting it onto the manifold, which is equivalent to projecting it onto the linear subspace of the principal components in the feature space. This ability can be used for background suppression and occlusion compensation as demonstrated in handwriting recognition [16]. Last, this non-linear learning method is easy to perform and only involves linear operations and eigendecomposition.

C. Detection

The detection of trained postures is formulated as a problem of open-set classification which classifies a test image window into one of the learned postures or as negative (background/unknown posture). Let z be the test image and $p(c_i|z)$ be the probability of posture c_i given z . z is considered containing posture c_k , if

$$\begin{cases} k = \arg \max_i p(c_i|z) \\ p(c_k|z) > th \end{cases} \quad (4)$$

where th is a threshold, and $p(c_i|z)$ is the decision function of the classifier. Using Bayesian rule it becomes:

$$p(c_i|z) \propto \frac{p(z|c_i)}{p(z)} \quad (5)$$

where $p(z|c_i)$ is the conditional probability of z given posture c_i and $p(z)$ is the prior probability of z . As each c_i is represented by a manifold in the HOG space, the closest HOG to z that lies in the manifold span of c_i can be readily expressed as the KPCA reconstruction result of z [16] [17], which is the reverse mapping from $\phi_P(z)$ in the feature subspace to the original HOG space:

$$r(z) = \phi^{-1}(\phi_P(z)) \quad (6)$$

where $\phi_P(z)$ is the projection of $\phi(z)$ onto the first d principal components. The obtained $r(z)$ can be regarded as an HOG of reference that illustrates 'what z should look like' when z is assumed to be in the class of c_i , meanwhile, since the HOG is a histogram-based feature, we can assume that the conditional probability $\ln(p(z|c_i))$ is proportional to the exponential of Bhattacharyya distance between the histogram of z and $r(z)$

$$p(z|c_i) = e^{c_1 D_B(z, r(z))} \quad (7)$$

where D_B is the Bhattacharyya distance and c_1 is a constant. The KPCA reconstruction is a recursive and slow optimization

process, however, referring back to the fact that each component of z is actually the square root of the original HOG component, $D_B(z, r(z))$ becomes:

$$\begin{aligned} D_B(z, r(z)) &= \langle z, r(z) \rangle \\ &= \frac{1}{2}(|z|^2 + |r(z)|^2 - |z - r(z)|^2) \end{aligned} \quad (8)$$

Also since both $|z|$ and $|r(z)|$ are constant, and $|z - r(z)|^2$ can be calculated from the kernel function (in our case only the Gaussian RBF kernel is used), the distance becomes:

$$D_B(z, r(z)) = c_2 + \frac{1}{2} \ln(2\sigma^2 \langle \phi(z), \phi_P(z) \rangle) \quad (9)$$

where $c_2 = 15 \times 31$ and σ is the width of the kernel $k(\cdot, \cdot)$. This distance function is very similar to the decision function of the novelty detector proposed by Hoffmann [18]. The only difference is that our distance function measures the reconstruction error in the HOG space but Hoffmann's function measures it in the feature space. Furthermore, since $\phi_P(x)$ is in the principal component subspace, $\langle \phi(z), \phi_P(z) \rangle$ can be replaced by $\langle (y_n(z) + b_n), (y(z) + b_n) \rangle$, where $y(z)$ is obtained by Eq.(13), and $y_n(z)$ and b_n are the mapping result $y(z)$ and centering offset b respectively when all n eigenvectors of $\hat{K}$ are used as principal components.

The $p(z|c_i)$ can be a good decision function in many open-set classification problems, but in our case, $p(z|c_i)$ alone cannot provide enough discriminative power, this is because each block of the HOG is a 9-bin histogram vector, and the intrinsic distribution of the binned gradient direction can be regarded as uniform categorical distribution in most cases, so the prior distribution of each block HOG is actually a multinomial distribution, and will peak when its components are equal to the mean vector. Therefore, $p(z)$ in Eq.(13) will not be a constant and must be taken into account: as the multinomial distribution is hard to calculate, again we approximate $p(z)$ by the exponential of the Bhattacharyya distance between the HOG of the image and the mean HOG vector:

$$p(z) = e^{c_1 D_B(z, (\frac{1}{3})_{15 \times 31 \times 9})} \quad (10)$$

$$= e^{c_1 (c_2 - \frac{1}{2} |z - (\frac{1}{3})_{15 \times 31 \times 9}|^2)} \quad (11)$$

Where $(\frac{1}{3})_{15 \times 31 \times 9}$ denotes the mean HOG vector with each element being $1/3$. Combining Eq.(12), Eq.(7), Eq.(9) and Eq.(10) together we will have:

$$p(c_i|z) \propto e^{\frac{c_1}{2} [\ln(2\sigma^2 \langle \phi(z), P P^T \phi(z) \rangle) + |z - (\frac{1}{3})_{15 \times 31 \times 9}|^2]} \quad (12)$$

Substitute $p(c_i|z)$ with Eq.(12) the decision function Eq.(4) will become:

$$\begin{cases} k = \arg \max_i \ln(\langle \phi(z), \phi_P(z) \rangle) + |z - (\frac{1}{3})_{15 \times 31 \times 9}|^2 \\ \ln(\langle \phi(z), \phi_P(z) \rangle) + |z - (\frac{1}{3})_{15 \times 31 \times 9}|^2 > \frac{2}{c_1} \ln(th) \end{cases} \quad (13)$$

Parameters c_1 governs the tradeoff between false positive and false negative and can be tuned to attain the best detection result.

III. GREEDY KPCA APPROXIMATION

Obtaining the mapping function $y(z)$ and $y_n(z)$ for open-set classification involves comparing z to each of the training samples $X = x_1, x_2, \dots, x_n$. When n is very large, the proposed classifier often becomes intolerably slow for real-time detection. The classifier is even slower than an RBF kernel SVM, in which only a subset of training samples are required to be kept as support vectors. To overcome this problem, we introduce a greedy KPCA approximation algorithm [12] that replace the original KPCA model, given by X and A , by a reduced KPCA model, given by a reduced-size training dataset $\tilde{X}$ and smaller matrix $\tilde{A}$, such that the mapping of a new observation, z in the reduced KPCA model is sufficiently close to the mapping of the observation in the original KPCA model. If the size of $\tilde{X}$ is m and the corresponding matrix A is $n \times m$, then the speed gain of using the new model will be m/n .

The approximation works by incrementally select a subset from the training samples X , of which feature vectors Φ_X constitute a linear subspace of the KPCA mapping space that can covers the majority of the variation of $y(z)$, so $y(z)$ can be approximated by its projection onto the subspace with minimum error. This can be effectively achieved by performing Gram-Schmidt orthogonalization on Φ_X in each iteration, and maximize the contribution of the chosen orthogonalized column vector of Φ_X to the remaining approximation error. In this paper we will only give the implementation of this algorithm, and suggest readers to refer to the original paper [12] for further details.

- 1) Given a training dataset X in the HOG space and the Gaussian RBF kernel function $k(.,.)$, calculate the KPCA model and obtain the mapping of the training dataset $W_X = w(x_1), w(x_2), \dots, w(x_n)$. Initialize $V = D_n^{\frac{1}{2}} W_X$.
- 2) Calculate the inner-product matrix $N = V^T D_n^{\frac{1}{2}}$, find one of its rows with the largest norm.
- 3) Add the i^{th} column vector of V to the reduced set $\tilde{X} = \tilde{X} \cup v_i$. v_i can be removed from V .
- 4) Update V by orthonormalize its column vectors V with v_i .
- 5) Repeat step 3 to 5 until m examples are chosen or the mapping error is lower than a threshold.
- 6) Obtain $\tilde{H}_A$ by $\tilde{H}_A = (D_n^{\frac{1}{2}} W_{\tilde{X}})^+$ and $\tilde{b}$ by $\tilde{b} = (\tilde{H}_A)^T K(\tilde{X}, X) 1_{1,n}$.

Therefore the KPCA parameter H_A can be replaced by $\tilde{H}_A$ and b can be replaced by $\tilde{b}$, comparing to the training of the original KPCA model, the complexity of this training algorithm only increases for $O(m(2n^2 + 2nd))$, which is almost neglectable comparing to the large computational cost in eigendecomposition of K , however, the speed of the mapping function $y(z)$ will be significantly increase, and for our

Fig. 1. ROC curves of the proposed detector and Dalal's detector on human detection task with different kernel width w

detection framework, the mapping speed is far more important than the training speed.

IV. EXPERIMENTAL RESULTS

In our first experiment we tested our approach for human detection task. (Though this approach was not initially designed for this task) We use the INRIA database for training and testing, 489 images and their mirror image from the INRIA positive training dataset are used for training, their background are manually removed. the removal of background is crucial to the performance of the classifier, since there is no negative example, if the background is included in the positive examples the learning approach will be unable to automatically 'weight down' the background regions as in SVM, and HOG components from these regions will be learned as part of the posture manifold, and will eventually affect the decision making of the detector. Such effect can only be avoided if the examples densely cover every possible HOG pattern of background, but this is impractical. In another hand, if the background regions are removed, the corresponding HOG components will become uniformly distributed, which coincide with our assumption that the binned HOG of one block in the background region should follows the uniform categorical distribution. According to our experiment, the detection result will not make sense if the background of the examples are not removed.

In test stage, INRIA positive test dataset consisting of 589 human images is used as positive test samples and 9060 images randomly cropped from INRIA negative test dataset (mainly consisting of landscape images) are used as negative test samples. (Background in test images are not removed) The result is compared to Dalal's work [10] that uses linear SVM for classification. The ROC curves of the linear SVM-based detector and the proposed detector are shown in figure 1.

This performance is much worse than the SVM-based detector if under the condition that absolutely no negative sample is used. Since in human detection or pedestrian detection, the variations of HOGs of human bodies are usually not continuous or not smooth, in which case the manifold

Fig. 2. ROC curves of the proposed detector in human detection task with different kernel width w when negative examples are introduced, the result is compared to Dalal's detector.

Fig. 3. Typical images for the 12 postures

representation is not particularly suitable. And even if it can be enclosed in a lower-dimensional manifold, its dimensionality will be too high and cannot be easily estimated from a relatively small training set. Hence it is more important to find discriminative features rather than representative features, and the SVM is more. Introducing a new negative class ('Non-posture' class) denoted by an extra KPCA model learned from the negative training dataset will lead to a close-set classifier, of which performance is almost identical to the detector using linear SVM (the ROC curves of both classifiers are shown in figure 2), however such improvement will not bring any superiority over the SVM-based detector in either performance or efficiency, meanwhile one of the most attractive features of the proposed detector—its ability to function without negative examples—will be lost. As stated before, this experiment is not intend to show the performance of our method as it is not designed for the human detection task.

In the following experiments, the proposed approach is tested in multi-posture case, training and test samples are extracted from Weizmann action dataset [19]. The dataset contains 93 low resolution video (188×144 , 25 fps) sequences for 10 actions. Nine subjects played each action once. Over 2000 images were manually cropped and divided into 12 postures to form the posture corpus. Figure 3 shows typical images for the 12 postures. The corpus was randomly divided into training and testing set at a ratio of 7:3, and the same negative test dataset used in previous experiment are used for another time. A number of experiments were conducted to evaluate the performance of the proposed method.

The second experiment aims to find the detection rate of the KPCA-based detector, each test image is classified into 13

(a)

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	NG
P1	22	0	0	0	0	0	0	0	0	0	0	0	0
P2	1	66	0	0	0	0	2	0	0	0	0	0	1
P3	0	0	19	0	0	0	0	0	0	0	0	0	5
P4	0	0	0	23	0	0	0	0	0	0	0	0	6
P5	0	0	0	0	41	0	0	0	0	0	0	0	0
P6	0	0	0	0	0	22	0	0	0	0	0	0	3
P7	0	0	3	0	0	0	66	0	0	0	0	0	1
P8	0	0	0	0	0	0	0	16	0	0	0	0	12
P9	0	0	0	0	0	0	0	0	27	0	3	0	2
P10	0	0	7	0	0	0	0	3	0	40	0	0	3
P11	0	0	0	0	0	0	0	0	2	0	47	0	9
P12	0	0	0	0	0	0	0	0	0	0	0	26	10
NG	38	38	55	1	1	16	24	0	0	0	0	0	8887

(b)

Fig. 4. (a) ROC curve of the proposed detector on Weizmann action database with different kernel width w (b) The confusion matrix of the proposed detector. P1-P12 represents the 12 postures and NG represents the negative samples

classes: 12 for postures and 1 for background. The experiment was carried out for several time, each time with a different RBF kernel width σ , The ROC curves of the detector and corresponding confusion matrix at the best detection/false alarm ratio are shown in Figure 4 (a) and (b) respectively. This result start to show some potential of the proposed detector, it is able to achieve 94% detection rate at 0.05 false positive rate, though the false positive rate is still a bit higher than expected, we believe it can be further improved by introducing more accurate statistical modeling of the background.

The third experiment aims to test the accuracy of the proposed KPCA approximation algorithm and how it will affect the performance of the detector. The experiment was carried out for several time, each time with a different number of selected examples m , The ROC curves of the detector using reduced model is compared to that of the detector using original KPCA model. These ROC curves are shown in Figure 5.

In the last experiment, we test the detector that was used in previous 2 experiments on a few images extracted from MPEG-7 ETRI video sequences and INRIA human dataset. The detector was conducted from each image using sliding window. Each window image is classified into either one of the trained postures or non-posture. Figure 6 shows several results where the bounding boxes indicate detected postures related to walking, run and standing. In the last image, the

Fig. 5. ROC curve of the proposed detector on Weizmann action database with different size of the reduced set m in KPCA approximation

Fig. 6. Some of the detection results on the images from ETRI and INRIA datasets

standing posture was not detected. This is because the training sample for the standing posture were all from side viewpoints.

V. CONCLUSION

In this paper, we propose a new method for detecting postures from single images using KPCA manifold learning. The performance is quite promising considering only about 150 samples per posture was used without involving any negative samples. Our future work will be to further improve the method for higher detection rate, and to explore the potential of the method to deal with multiple viewpoints and occlusion.

REFERENCES

- [1] W. Li, Z. Zhang, and Z. Liu, "Expandable data-driven graphical modeling of human actions based on salient postures," *Circuits and Systems for Video Technology, IEEE Transactions on*, vol. 18, no. 11, pp. 1499–1510, 2008.
- [2] B. Wu and R. Nevatia, "Detection and tracking of multiple, partially occluded humans by bayesian combination of edgelet based part detectors," *International Journal of Computer Vision*, vol. 75, no. 2, pp. 247–266, 2007.
- [3] Z. Lin, L. Davis, D. Doermann, and D. DeMenthon, "Hierarchical part-template matching for human detection and segmentation," in *Proc. International Conference on Computer Vision*, 2007, pp. 1–8.
- [4] K. Grauman, G. Shakhnarovich, and T. Darrell, "Inferring 3d structure with a statistical image-based shape model," in *Proc. International Conference on Computer Vision*, 2003, pp. 641–647.

- [5] A. Agarwal and B. Triggs, "Monocular human motion capture with a mixture of regressors," in *IEEE Workshop on Vision for Human-Computer Interaction*, 2005, p. 72.
- [6] M. Brand, "Shadow puppetry," in *Proc. International Conference on Computer Vision*, vol. 2, 1999, p. 1237.
- [7] A. Agarwal and B. Triggs, "3d human pose from silhouettes by relevance vector regression," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, 2004, pp. II–882–II–888.
- [8] A. Elgammal and C. Lee, "Inferring 3d body pose from silhouettes using activity manifold learning," in *Proc. IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2, 2004, pp. II–681–II–688.
- [9] A. Agarwal and B. Triggs, "A local basis representation for estimating human pose from cluttered images," in *Proc. Asian Conference on Computer Vision*, vol. 3851/2006, 2006, pp. 50–59.
- [10] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. Computer Vision and Pattern Recognition*, vol. 1, 2005, pp. 886–893.
- [11] B. Scholkopf, S. Mika, A. Smola, G. Ratsch, and K. Muller, "Kernel pca pattern reconstruction via approximate pre-images," in *Proceedings of the 8th International Conference on Artificial Neural Networks, Perspectives in Neural Computing*, 1998, pp. 147–152.
- [12] P. Cheng, "Greedy approximation of kernel pca by minimizing the mapping error," 2009, (Accepted).
- [13] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1615–1630, 2005.
- [14] N. Dalal, B. Triggs, and C. Schmid, "Human detection using oriented histograms of flow and appearance," in *Proc. European Conference on Computer Vision*, vol. 3952, 2006, pp. 428–441.
- [15] Y. Wu and T. Yu, "A field model for human detection and tracking," *Pattern Analysis and Machine Intelligence*, vol. 28, no. 5, pp. 753–764, 2006.
- [16] S. Mika, B. Scholkopf, A. Smola, K. Muller, M. Scholz, and G. Ratsch, "Kernel pca and de-noising in feature spaces," *Advances in Neural Information Processing Systems*, vol. 11, pp. 536–542, 1999.
- [17] J. Kwok and I. Tsang, "The pre-image problem in kernel methods," *IEEE transactions on neural networks*, vol. 15, no. 6, pp. 1517–1525, 2004.
- [18] H. Hoffmann, "Kernel PCA for novelty detection," *Pattern Recognition*, vol. 40, no. 3, pp. 863–874, 2007.
- [19] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri, "Actions as space-time shapes," in *The Tenth IEEE International Conference on Computer Vision*, 2005, pp. 1395–1402.