Noname manuscript No.
(will be inserted by the editor)

Generic Human Action Recognition from a Single Example

Hae Jong Seo · **Peyman Milanfar**

Received: date / Accepted: date

Abstract We present a novel human action recognition method based on space-time locally adaptive regression kernels and the matrix cosine similarity measure. The proposed method operates using a *single* example (e.g., short video clip) of an action of interest to find similar matches. It does not require prior knowledge (learning) about actions being sought; and does not require foreground/background segmentation, or any motion estimation or tracking. Our method is based on the computation of the so-called local steering kernels as space-time descriptors from a query video, which measure the likeness of a voxel to its surroundings. Salient features are extracted from said descriptors and compared against analogous features from the target video. This comparison is done using a matrix generalization of the cosine similarity measure. The algorithm yields a scalar resemblance volume with each voxel here, indicating the likelihood of similarity between the query video and all cubes in the target video. By employing nonparametric significance tests and non-maxima suppression, we detect the presence and location of actions similar to the given query video. High performance is demonstrated on the challenging set of action data (Shechtman and Irani 2007b) indicating successful detection of actions in the presence of fast motion, different contexts and even when multiple complex actions occur simultaneously within the field of view of the camera. Further experiments on the Weizmann dataset (Gorelick et al. 2007) and the KTH dataset (Schuldt et al. 2004) for action categorization task demonstrate that the proposed method achieves improvement over other (state-of-the-art) algorithms.

Keywords Action Recognition · Space-time local regression kernels · Matrix Cosine Similarity · Significance testing

University of California at Santa Cruz
1156 High Street, Santa Cruz, CA, USA
Tel.: +1-831-4594141
Fax: +1-440-3322312
E-mail: rokaf@soe.ucsc.edu

1 Introduction

A huge number of videos (BBC[1],Youtube[2]) are available online today and the number is rapidly growing. Human actions constitute one of the most important parts in movies, TV shows, and consumer-generated videos. Analysis of human actions in videos is considered a very important component in computer vision systems because of such applications as human-computer interaction, content-based video retrieval, visual surveillance, analysis of sports events and more. The term “action” refers to a simple motion pattern as performed by a single subject, and in general lasts only for a short period of time, namely just a few seconds. Action is often distinguished from activity in the sense that action is an individual atomic unit of activity. In particular, human action refers to physical body motion. Recognizing human actions from video is a very challenging problem due to the fact that physical body motion can look very different depending on the context: 1) similar actions with different clothes, or in different illumination and background can result in a large appearance variation; 2) the same action performed by two different people may look dissimilar in many ways.

Over the last two decades, many studies have attempted to tackle this problem broadly by means of time-series non-parametric approaches, parametric approaches, and volumetric approaches. For instance, 2-D templates (Bobick and J.W.Davis 2008), 3-D object models (Gorelick et al. 2007), and manifold learning methods (Elgammal and Lee. 2004) are categorized into time-series non-parametric approaches while Hidden Markov Models (Starner et al. 1998), linear dynamical systems (Mazzaro et al. 2005), and non-linear dynamical systems (Pavlovic et al. 2000) are called parametric approaches. Volumetric approaches (part-based approach: [Niebles and Fei-Fei. 2007; Niebles et al. 2008], subvolume matching: [Shechtman and Irani. 2007b; Ke et al. 2005], and tensor-based approach: [Kim et al. 2007]) tend to outperform the other two approaches. We refer the interested reader to Turaga et al. 2008, and references therein for a good summary.

As examples of the volumetric approach, Niebles and Fei-Fei 2007 and Niebles et al. 2008 considered videos as spatiotemporal bag-of-words by extracting space-time interest points and clustering the features, and then used a probabilistic Latent Semantic Analysis (pLSA) model to localize and categorize human actions. However, the performance of these methods can degrade due to 1) the lack of enough training samples; 2) misdetections and occlusions of the interest point since they ignore global space-time information. Shechtman and Irani 2007b employed a three dimensional correlation scheme for only action detection. They focused on subvolume matching in order to find similar motion in the two space-time cubes, which can be computationally heavy. Ke et al. 2005 presented an approach which uses boosting on 3-D Haar-type features inspired by Haar-like features in 2-D object detection (Viola and Jones 2004.) While these features are very efficient to compute, many examples are required to train an action detector in order to achieve good performance. Recently, Kim et al. 2007 generalized canonical correlation analysis to tensors and showed very good accuracy on KTH dataset, but their method requires a manual alignment process for camera motion compensation. Ning et al. 2008 proposed a system to search for human actions using a coarse-to-fine approach with a five-layer hierarchical space-time model. These volumetric methods do not require background subtraction, motion estimation, or complex models of body configuration and kinematics. They tolerate variations in appearance, scale, rotation, and movement to some extent. Methods such as those in Shechtman and Irani. 2007b; Ning

[1] http://www.bbcmotiongallery.com/Customer/index.aspx

[2] http://www.Youtube.com

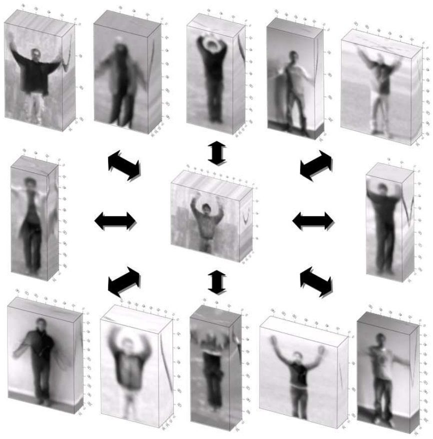


Fig. 1 (a) A hand-waving action and possibly similar actions

et al. 2008; and Yeo et al. 2008 which aim at recognizing actions based solely on only one query (what we shall call training-free) are very useful but challenging for video retrieval from the web (e.g., viewdle[3],videosurf[4]). In these methods, a single query video is provided by users and every gallery video in the database is compared with the given query, posing a video-to-video matching problem.

1.1 Problem Specification

We present a novel approach to the problem of human action recognition as a video-to-video matching problem. Here, recognition is generally divided into two parts: category classification and detection/ localization. The goal of action classification is to classify a given action query into one of several pre-specified categories (for instance, 6 categories from KTH action dataset (Schuldt et al. 2004): boxing, hand clapping, hand waving, jogging, running, and walking), while action detection is meant to separate an action of interest from the background in a target video (for instance, spatiotemporal localization of a ballet turn (1 second) from a long ballet video sequence (25 seconds)). This paper tackles both action detection and category classification problems simultaneously by searching for an action of interest within other “target” videos with only a *single* “query” video. In order to avoid the disadvantages of learning-based methods which require a large number of training examples, we focus on a sophisticated feature representation with an efficient and reliable similarity matching scheme which at the same time, allows us to avoid the difficult problem of explicit motion estimation.

In general, the target video may contain actions similar to the query, but these will typically appear in completely different context (See Fig. 1.) Examples of such differences can range from rather simple optical or geometric differences (such as different clothes, lighting, action speed and scale changes); to more complex inherent structural differences such as for instance a hand-drawn action video clip (e.g., animation) rather than a real human action.

[3] http://www.viewdle.com

[4] http://www.videosurf.com

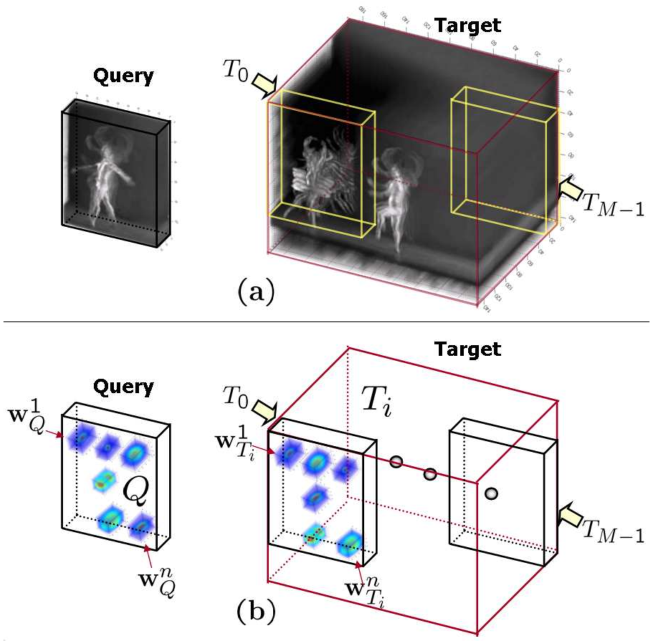


Fig. 2 Action detection problem (a) Given a query video Q, we wish to detect/localize actions of interest in a target video T. T is divided into a set of overlapping cubes (b) space-time local steering kernels (3-D LSKs) capture the geometric structure of underlying data.

1.2 Overview of the Proposed Approach

In this paper, our contributions to the action recognition task are mainly two-fold. First, we propose a novel feature representation that is derived from space-time local regression kernels which capture the underlying structure of the data exceedingly well, even in the presence of significant distortions. The most salient characteristics are then computed by performing dimensionality reduction, namely Principal Component Analysis (PCA) on a collection of the space-time local regression kernels. Second, we propose to use a training-free nonparametric detection scheme, an earlier version of which for 2-D object detection was proposed by Seo and Milanfar[5]. Furthermore, we extended the detection scheme to action category classification by automatically cropping a short action clip from larger videos.

The key idea behind local regression kernels is to robustly obtain local data structures by analyzing the radiometric (pixel value) differences based on estimated gradients, and use this structure information to determine the shape and size of a canonical kernel (descriptor). The motivation to use these local regression kernels is the earlier successful work on adaptive kernel regression for image denoising, interpolation (Takeda et al. 2007) and deblurring (Takeda et al. 2008b). Takeda et al.[6] extended the kernel regression framework to super-resolution by introducing space-time local *steering* kernels (3-D LSK) which capture the essential local behavior of a spatiotemporal neighborhood. The 3-D LSK is fundamentally based on the comparison of neighboring voxels in both space and time, which implicitly contains information about the local motion of the voxels across time, thus requiring no explicit motion estimation.

[5] Downloadable from `http://www.soe.ucsc.edu/~rokaf/paper/TrainingFreeGenericObjectDetection_FinalRevision_Mar10.pdf`.

[6] Downloadable from `http://www.ee.ucsc.edu/~milanfar/publications/journal/SpatiotemporalKernelRegression.pdf`.

Recently, Seo and Milanfar[5] proposed to use local steering kernels as descriptors for generic 2-D object detection and demonstrated a high detection accuracy in challenging sets of real-world objects. Action recognition addressed in this paper is considered to be more challenging than static (2-D) object recognition due to problems such as variations in individual motion and camera motion. However, motion provides an additional discriminative power and 3-D LSKs can implicitly capture local motion information exceedingly well.

Inspired by these earlier works, we propose to use 3-D LSKs for the problems of detection/localization of actions of interest between a query video and a target video. Denoting the target video (T), and the query video (Q), we compute a dense set of 3-D LSKs from each. These densely computed descriptors are highly informative, but taken together tend to be over-complete (redundant). Therefore, we derive features by applying dimensionality reduction (namely PCA) to these resulting arrays, in order to retain only the salient characteristics of the 3-D LSKs.

Generally, T is bigger than the query video Q. Hence, we divide the target video T into a set of overlapping cubes indexed by i, which are the same size as Q (See Fig. 2(a)). The feature collections from Q and T_i form feature volumes $\mathbf{F}_Q$ and $\mathbf{F}_{T_i}$. We compare the feature volumes $\mathbf{F}_{T_i}$ and $\mathbf{F}_Q$ from the i^{th} cube of T and Q to look for matches. Inspired in part by many studies (Fu et al. 2008; Fu and Huang 2008; Liu 2007, 2008; Lin et al. 2005; Ma et al. 2007) which took advantage of cosine similarity over the conventional Euclidean distance, we employ "Matrix Cosine Similarity" as a similarity measure which generalizes the (vector) cosine similarity between two vectors (Schneider and Borlund. 2007; Ahlgren et al. 2003; Rodgers and Nicewander 1988). The optimality properties of this approach were introduced in Seo and Milanfar[5] using a naive Bayes framework, which leads to the use of the Matrix Cosine Similarity (MCS). In order to deal with the case where the target video may not include any actions of interest or when there are multiple occurrences of action of interest in the target video, we also adopt the idea of a significance test and non-maxima suppression (Devernay 1995.)

Very recently, Shechtman and Irani 2007a introduced a space-time local self-similarity descriptor for action detection and showed performance improvement over their previous approach (Shechtman and Irani 2007b). It is worth mentioning that this (independently derived) local space-time self-similarity descriptor is a special case of 3-D LSK and is also related to a number of other local data adaptive metrics such as Optimal Space-Time Adaptation (OSTA) (Boulanger et al. 2005) and Non-Local Means (NLM) (Buades et al. 2008) which have been used very successfully for video restoration in the image processing community.

Fig. 3 shows an overview of our proposed framework for action detection and category classification. Before we begin a more detailed description, we highlight some aspects of the proposed framework.

- We propose a novel feature representation derived from densely computed 3-D LSKs. Since the calculation of 3-D LSKs is stable in the presence of uncertainty in the data (Takeda et al. 2007), our approach is robust even in the presence of noise. In addition, normalized 3-D LSKs provide a certain invariance to illumination changes (see Fig. 5.)
- As opposed to Shechtman and Irani 2007b who filtered out "non-informative" descriptors in order to reduce the time complexity, we automatically obtain the most salient feature volumes by applying Principal Components Analysis (PCA) to a collection of 3-D LSKs as similarly done in the approach of Ali and Shah 2008 where kinematic features were derived from optical flow by applying PCA. The proposed method is feasible

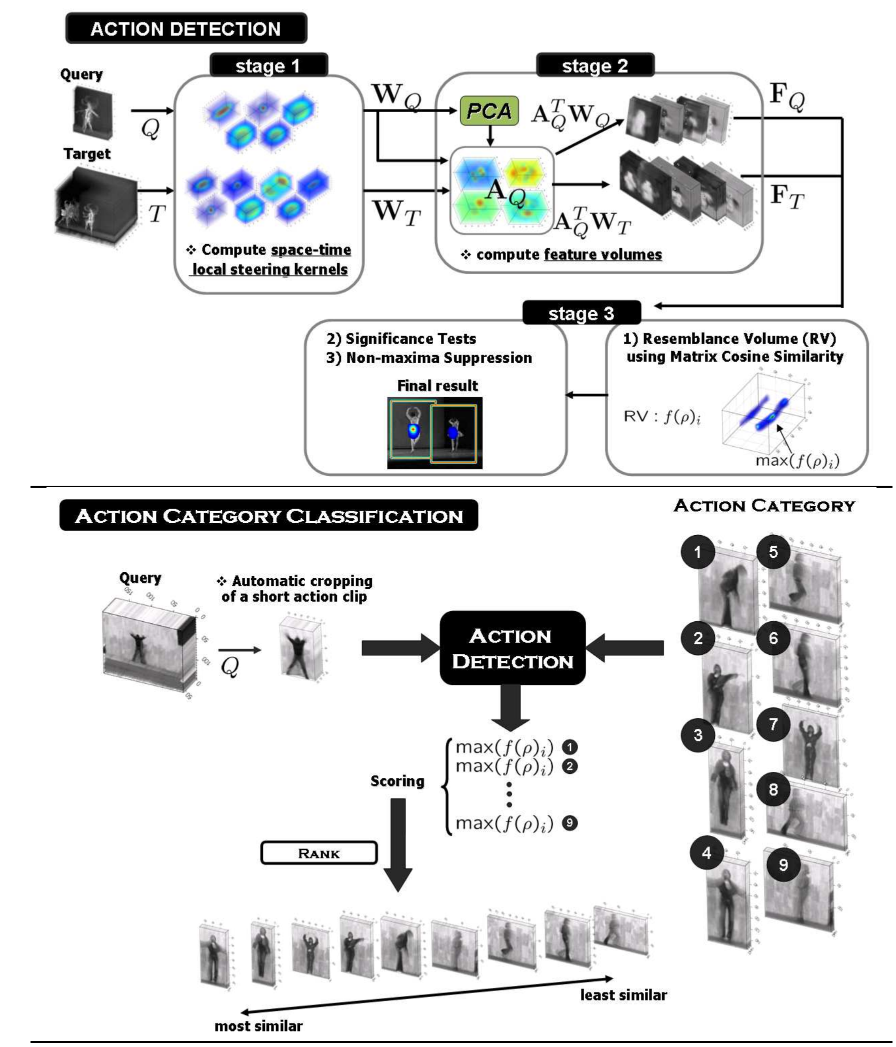


Fig. 3 System overview. Top: Action detection framework (There are broadly three stages.) Bottom: Action Category Classification

in practice because the dimension of features after PCA is significantly reduced (e.g., from say $3 \times 3 \times 7 = 64$, to 3 or 4), even though the descriptors are densely computed.

- We apply a nonparametric generic object detection framework[5] to the action detection problem and extend it to action category classification by developing a method which automatically crops a short action clip.
- The proposed method is tolerant to small deformations (i.e., ±20% scale change, ±15 degree rotation change) of the query and can detect multiple actions that occur simultaneously in the field of view of the camera using multiple queries.
- From a practical standpoint, it is important to note that the proposed framework operates using a single example of an action of interest to find similar matches; does not require

any prior knowledge (learning) about actions being sought; and does not require any pre-processing step or segmentation of the target video.

This paper is organized as follows. In the next section, we specify the algorithmic aspects of our action detection framework, using a novel feature representation which results from the "space-time local *steering* kernel" (3-D LSK) followed by PCA and a reliable similarity measure (the "Matrix Cosine Similarity"). In Section 3, we extend the proposed detection framework to action category classification. In Section 4, we demonstrate the performance of the system with comprehensive experimental results, and we conclude this paper in Section 5.

2 Action Detection From a Single Query

As outlined in the previous section, our approach to detect actions consists broadly of three stages. Below, we describe each of these steps in detail. In order to make the concepts more clear, we first briefly describe the local steering kernels in 2-D. For extensive detail on this subject, we refer the reader to Takeda et al. 2007.

2.1 Local Steering Kernel as a descriptor

2.1.1 Local Steering Kernel in 2-D (LSK)

The key idea behind LSK is to robustly obtain the local structure of images by analyzing the radiometric (pixel value) differences based on estimated gradients, and use this structure information to determine the shape and size of a canonical kernel. The local steering kernel is defined as follows:

$$K(\mathbf{x}_l-\mathbf{x};\mathbf{C}_l)=\frac{\sqrt{\det(\mathbf{C}_l)}}{h^2}\exp\left\{\frac{(\mathbf{x}_l-\mathbf{x})^T\mathbf{C}_l(\mathbf{x}_l-\mathbf{x})}{-2h^2}\right\},\quad l\in[1,\cdots,P], \tag{1}$$

where $\mathbf{x}_l=[x_1,x_2]_l^T$ is the space-time coordinates, h is a global smoothing parameter, P is the total number of samples in a local analysis window around a sample position at $\mathbf{x}_l$, and the matrix $\mathbf{C}_l\in\mathbb{R}^{(2\times2)}$ is a covariance matrix estimated from a collection of first derivatives along spatial axes. The covariance matrix $\mathbf{C}_l$ modifies the shape and size of the local kernel in a way which robustly encodes the local geometric structures.

As apparent from Fig. 4(a), the shape of the LSK's is not simply a Gaussian, despite the simple definition above. It is important to note that this is because for each pixel $\mathbf{x}_l$ in the vicinity of $\mathbf{x}$, a different steering matrix $\mathbf{C}_l$ is used, therefore leading to a far more complex and rich set of possible shapes for the resulting LSKs. The same idea is valid in 3-D as well, as described below.

2.1.2 Space-Time Local Steering Kernel (3-D LSK)

Now, we introduce the time axis to the data model so that $\mathbf{x}_l=[x_1,x_2,t]_l^T$: x_1 and x_2 are the spatial coordinates, t is the temporal coordinate. In this setup, the covariance matrix $\mathbf{C}_l$ can be naively estimated as $\mathbf{J}_l^T\mathbf{J}_l$ with

$$\mathbf{J}_l=\begin{bmatrix} z_{x_1}(\mathbf{x}_1), & z_{x_2}(\mathbf{x}_1), & z_t(\mathbf{x}_1) \\ \vdots & \vdots & \vdots \\ z_{x_1}(\mathbf{x}_P), & z_{x_2}(\mathbf{x}_P), & z_t(\mathbf{x}_P) \end{bmatrix}$$

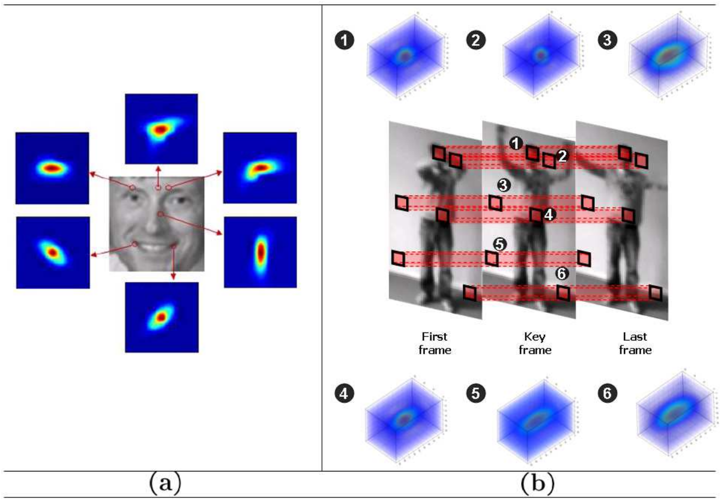


Fig. 4 (a) Examples of 2-D LSK in various regions. (b) Examples of space-time local steering kernel (3-D LSK) in various regions. Note that key frame means the frame where the center of 3-D LSK is located.

where $z_{x_1}(\cdot), z_{x_2}(\cdot)$, and $z_t(\cdot)$ are the first derivatives along x_1-, x_2-, and $t-$ axes, and P is the total number of samples in a *space-time* local analysis window (or cube) around a sample position at $\mathbf{x}$. For the sake of robustness, we compute a more stable estimate of $\mathbf{C}_l$ by invoking the singular value decomposition (SVD) of $\mathbf{J}_l$ with regularization as:

$$\widehat{\mathbf{C}}_l = \gamma_l \sum_{q=1}^{3} a_q^2 \mathbf{v}_q \mathbf{v}_q^T \in \mathbb{R}^{(3\times 3)}, \tag{2}$$

with

$$a_1 = \frac{s_1 + \lambda'}{\sqrt{s_2 s_3} + \lambda'}, \quad a_2 = \frac{s_2 + \lambda'}{\sqrt{s_1 s_3} + \lambda'}, \quad a_3 = \frac{s_3 + \lambda'}{\sqrt{s_1 s_2} + \lambda'}, \quad \gamma_l = \left(\frac{s_1 s_2 s_3 + \lambda''}{P}\right)^{\alpha} \tag{3}$$

where λ' and λ'' are regularization parameters that dampen the noise effect and restrict γ_l and the denominators of a_q's from being zero. The singular values (s_1, s_2, and s_3) and the singular vectors ($\mathbf{v}_1, \mathbf{v}_2$, and $\mathbf{v}_3$) are given by the compact SVD of $\mathbf{J}_l$:

$$\mathbf{J}_l = \mathbf{U}_l \mathbf{S}_l \mathbf{V}_l^T = \mathbf{U}_l \mathrm{diag}[s_1, s_2, s_3][\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3]^T, \tag{4}$$

Then, the covariance matrix $\widehat{\mathbf{C}}_l$ modifies the shape and size of the local kernel in a way which robustly encodes the space-time local geometric structures present in the video (See Fig. 4 (b) for an example.) Similarily to 2D case, 3-D LSKs are formed as follow:

$$K(\mathbf{x}_l - \mathbf{x}; \mathbf{C}_l) = \frac{\sqrt{\det(\mathbf{C}_l)}}{h^2} \exp\left\{-\frac{(\mathbf{x}_l - \mathbf{x})^T \mathbf{C}_l (\mathbf{x}_l - \mathbf{x})}{2h^2}\right\}, \quad \mathbf{C}_l \in \mathbb{R}^{(3\times 3)}. \tag{5}$$

In the 3-D case, orientation information captured in 3-D LSK contains the motion information implicitly. It is worth noting that a significant strength of using this implicit framework (as opposed to the direct use of estimated motion vectors) is the flexibility it provides in terms of smoothly and adaptively changing the parameters defined by the singular values

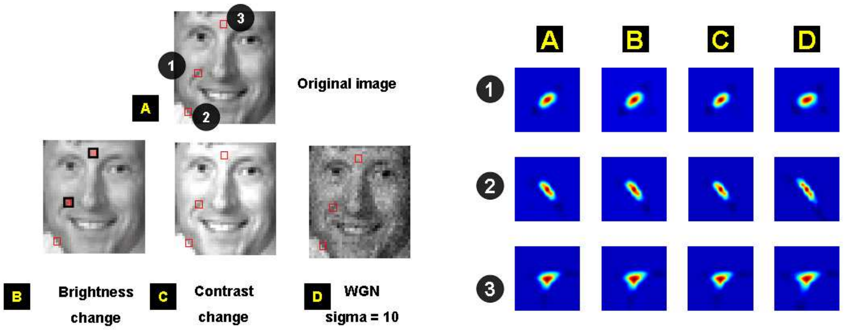


Fig. 5 Invariance and robustness of 2-D LSK weights $W(\mathbf{x}_l - \mathbf{x}; 2)$ in various challenging conditions. Note that WGN means White Gaussian Noise.

in Equation 3. This flexibility allows the accommodation of even complex motions, so long as their magnitudes are not excessively large. For a more in depth analysis of local steering kernels, we refer the interested reader to Takeda et al. 2007, 2008a.

In what follows, at a position $\mathbf{x}$, we will essentially be using (a normalized version of) the function $K(\mathbf{x}_l - \mathbf{x}; \mathbf{C}_l)$ as a function of $\mathbf{x}_l$ and $\mathbf{C}_l$ to represent a video's inherent local space-time geometry. To be more specific, the 3-D LSK function $K^j(\mathbf{x}_l - \mathbf{x}; \mathbf{C}_l)$ is densely calculated and normalized as follows

$$W_Q^j(\mathbf{x}_l - \mathbf{x}) = \frac{K_Q^j(\mathbf{x}_l - \mathbf{x}; \mathbf{C}_l)}{\sum_{l=1}^{P} K_Q^j(\mathbf{x}_l - \mathbf{x}; \mathbf{C}_l)}, \quad j \in [1, \cdots, n], \quad l \in [1, \cdots, P],$$

$$W_T^j(\mathbf{x}_l - \mathbf{x}) = \frac{K_T^j(\mathbf{x}_l - \mathbf{x}; \mathbf{C}_l)}{\sum_{l=1}^{P} K_T^j(\mathbf{x}_l - \mathbf{x}; \mathbf{C}_l)}, \quad j \in [1, \cdots, n_T], \quad l \in [1, \cdots, P], \tag{6}$$

where n and n_T are the number of 3-D LSKs in the query video Q and the target video T respectively [7]. Next, we describe some key properties of the above.

2.2 Feature representation

Seo and Milanfar[5] have shown that the normalized LSKs in 2-D follow a power-law (i.e., a long-tail) distribution. That is to say, the features are scattered out in a high dimensional feature space, and thus there basically exists no dense cluster in the descriptor space. The same principle applies to 3-D LSK case. In order to illustrate and verify that the normalized 3-D LSKs also satisfy this property and follow a power-law distribution, we computed an empirical bin density (100 bins) of the normalized 3-D LSKs (using a total of $50,000$ 3-D LSKs) computed from 90 videos from Weizmann action dataset (Gorelick et al. 2007) using the K-means clustering method (See Fig. 6.) The utility of this observation becomes clear in the next paragraphs.

[7] Note that videos here are gray scale. The case of color is worth treating independently and is discussed in the manuscript `http://www.soe.ucsc.edu/~rokaf/paper/TrainingFreeGenericObjectDetection_FinalRevision_Mar10.pdf`.

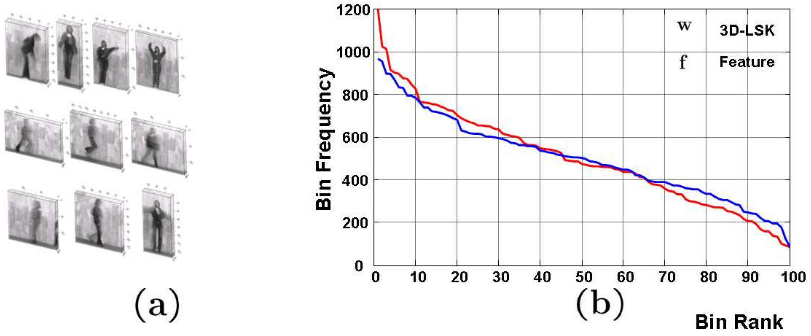


Fig. 6 (a) Some example video sequences (Weizman dataset) where 3-D LSKs were computed. (b) Plots of the bin density of 3-D LSKs and their corresponding low-dimensional features.

In the previous section, we computed a dense set of 3-D LSKs from Q and T. These densely computed descriptors are highly informative, but taken together tend to be over-complete (redundant). Therefore, we derive features by applying dimensionality reduction (namely PCA) to these resulting arrays, in order to retain only the salient characteristics of the 3-D LSKs. As also observed in Boiman et al. 2008, an ensemble of local features with even little discriminative power can together offer significant discriminative power. However, both quantization and informative feature selection on a long-tail distribution can lead to a precipitous drop in performance. Therefore, instead of any quantization and informative feature selection, we focus on reducing the dimension of 3-D LSKs using PCA to enhance the discriminative power and reduce computational complexity. [8]

This idea results in a new feature representation with a moderate dimension which inherits the desirable discriminative attributes of 3-D LSK. The distribution of the resulting features sitting on the low dimensional manifold also tends to follow a power-law distribution as shown in Fig. 6 (b) and this attribute of the features allows us to the use "Matrix Cosine Similarity" measure which will be illustrated in Section 2.3. Seo and Milanfar[5] have illustrated that a naive Bayes decision rule based on these features for object detection leads to the use of "Matrix Cosine Similarity".

In order to organize $W_Q(\mathbf{x}_l - \mathbf{x})$ and $W_T(\mathbf{x}_l - \mathbf{x})$, which are densely computed from Q and T, let $\mathbf{W}_Q$,$\mathbf{W}_T$ be matrices whose columns are vectors $\mathbf{w}_Q$,$\mathbf{w}_T$, which are column-stacked (rasterized) versions of $W_Q(\mathbf{x}_l - \mathbf{x})$,$W_T(\mathbf{x}_l - \mathbf{x})$ respectively:

$$\mathbf{W}_Q = [\mathbf{w}_Q^1, \cdots, \mathbf{w}_Q^n] \in \mathbb{R}^{P\times n}, \qquad \mathbf{W}_T = [\mathbf{w}_T^1, \cdots, \mathbf{w}_T^{n_T}] \in \mathbb{R}^{P\times n_T}. \tag{7}$$

As described in Fig. 3, the next step is to apply PCA to $\mathbf{W}_Q$ for dimensionality reduction and to retain only its salient characteristics. Applying PCA to $\mathbf{W}_Q$ we can retain the first (largest) d principal components[9] which form the columns of a matrix $\mathbf{A}_Q \in \mathbb{R}^{P\times d}$. Next, the lower dimensional features are computed by projecting $\mathbf{W}_Q$ and $\mathbf{W}_T$ onto $\mathbf{A}_Q$:

$$\mathbf{F}_Q = [\mathbf{f}_Q^1, \cdots, \mathbf{f}_Q^n] = \mathbf{A}_Q^T \mathbf{W}_Q \in \mathbb{R}^{d\times n}, \quad \mathbf{F}_T = [\mathbf{f}_T^1, \cdots, \mathbf{f}_T^{n_T}] = \mathbf{A}_Q^T \mathbf{W}_T \in \mathbb{R}^{d\times n_T}. \tag{8}$$

[8] Ali and Shah 2008 also pointed out that interest point descriptor-based action recognition methods have a limitation in that useful pieces of global information may be lost.

[9] Typically, d is selected to be a small integer such as 3 or 4 so that 80 to 90% of the "information" in the LSKs would be retained. (i.e., $\frac{\sum_{i=1}^{d}\lambda_i}{\sum_{i=1}^{P}\lambda_i} \geq 0.8$ (to 0.9) where λ_i are the eigenvalues.)

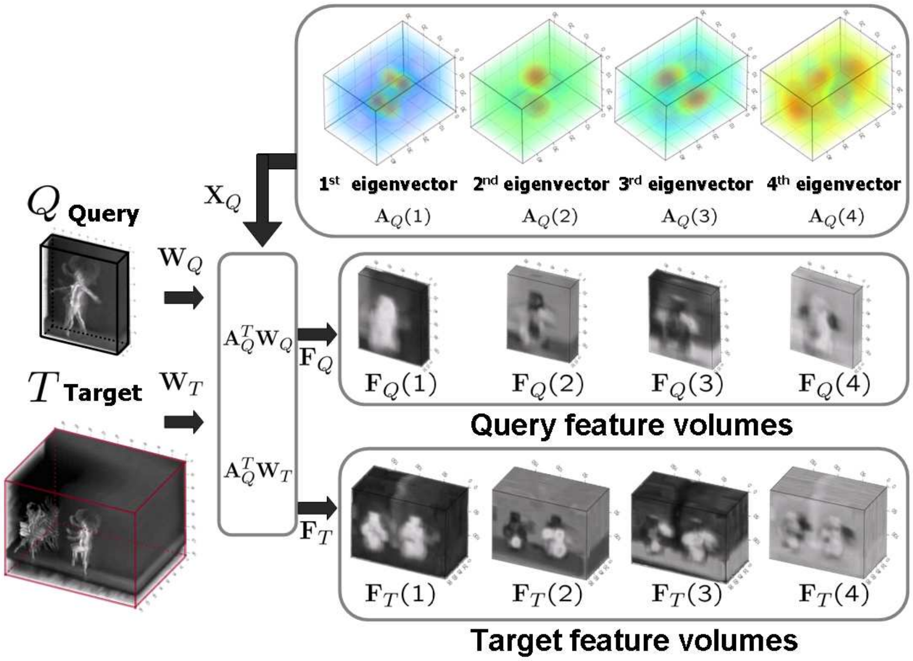


Fig. 7 Ballet action : $\mathbf{A}_Q$ is learned from a collection of 3-D LSKs $\mathbf{W}_Q$, and Feature row vectors of $\mathbf{F}_Q$ and $\mathbf{F}_T$ are computed from query Q and target video T respectively. Eigenvectors and feature vectors were transformed to volume and up-scaled for illustration purposes.

Fig. 7 illustrates the principal components in $\mathbf{A}_Q$ and shows what the features $\mathbf{F}_Q, \mathbf{F}_T$ look like for a ballet video sequence. In Fig. 8, we can see that the principal components from different actions in the KTH dataset (Schuldt et al. 2004) look distinct from one another. We can infer that the derived feature volumes should have a good discriminative power.

It is worth noting that a similar approach was also taken by Ke and Sukthankar 2004 where PCA was applied to interest point descriptors such as SIFT, leading to enhanced performance. Very recently, Ali and Shah 2008 proposed a set of kinematic features that extract different aspects of motion dynamics present in the optical flow. They obtained bags of kinematic modes for action recognition by applying PCA to a set of kinematic features. We differentiate our proposed method from Ali and Shah 2008 in the sense that 1) motion information is implicitly contained in 3-D LSK while Ali and Shah 2008 explicitly compute optical flow; 2) Background subtraction was used as a pre-processing while our method is fully automatic, 3) Ali and Shah 2008 employed multiple instance learning while the proposed method does not involve any training phase.

2.3 Detecting Similar Actions using the Matrix Cosine Measure

2.3.1 Matrix Cosine Similarity

The next step in the proposed framework is a decision rule based on the measurement of a “distance” between the computed feature volumes $\mathbf{F}_Q, \mathbf{F}_{T_i}$. We were motivated by earlier works such as Ma et al. 2007; Fu et al. 2008; and Fu and Huang 2008, that have shown the

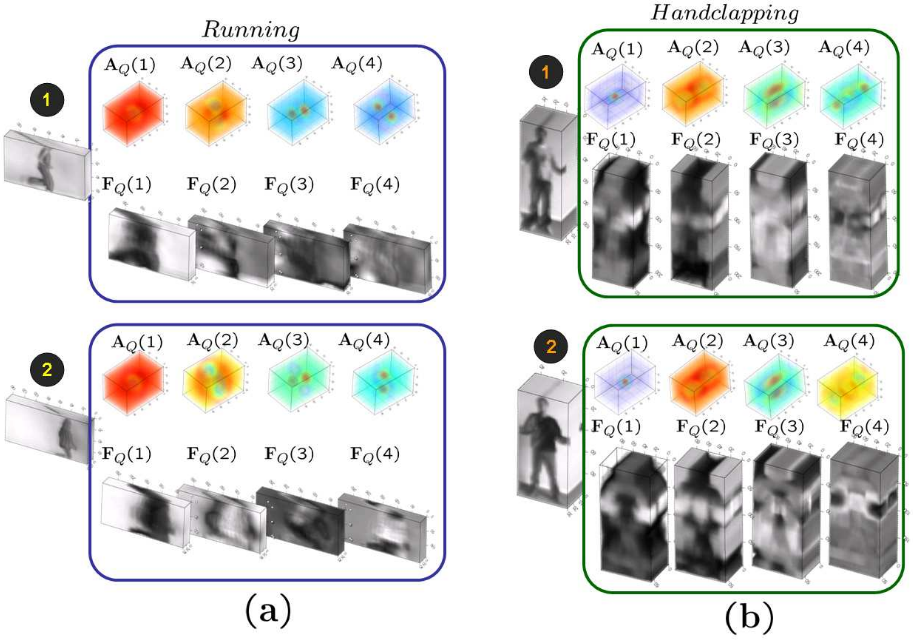

Running
1
2
$\mathbf{A}_Q(1)$ $\mathbf{A}_Q(2)$ $\mathbf{A}_Q(3)$ $\mathbf{A}_Q(4)$
$\mathbf{F}_Q(1)$ $\mathbf{F}_Q(2)$ $\mathbf{F}_Q(3)$ $\mathbf{F}_Q(4)$
(a)
Handclapping
(b)

Fig. 8 Comparison between Eigenvectors $\mathbf{A}_Q$'s and Feature row vectors of $\mathbf{F}_Q$'s from 2 action categories (running VS. hand clapping) from the KTH dataset (Schuldt et al. 2004). Eigenvectors and feature vectors were transformed to volume and up-scaled for illustration purposes. While $\mathbf{A}_Q$'s and $\mathbf{F}_Q$'s of running actions by two different people are similar, they are totally different from those of hand clapping actions.

effectiveness of correlation-based similarity. Recently, Seo and Milanfar[5] have proposed a nonparametric detection framework based on "Matrix Cosine Similarity" and have achieved excellent performance in 2-D object detection. Here, we propose to use this framework for action detection. For a more in depth analysis on the detection framework, we refer the interested reader to Seo and Milanfar[5].

The "Matrix Cosine Similarity (MCS)" between two feature matrices $\mathbf{F}_Q, \mathbf{F}_{T_i}$ which consist of a set of vectors can be defined as the "Frobenius inner product" between two normalized matrices as follows:

$$\rho(\mathbf{F}_Q,\mathbf{F}_{T_i}) =< \overline{\mathbf{F}}_Q,\overline{\mathbf{F}}_{T_i} >_F = \text{trace}\Big(\frac{\mathbf{F}_Q^T\mathbf{F}_{T_i}}{\|\mathbf{F}_Q\|_F\|\mathbf{F}_{T_i}\|_F}\Big) \in [-1,1], \tag{9}$$

where, $\overline{\mathbf{F}}_Q = \frac{\mathbf{F}_Q}{\|\mathbf{F}_Q\|_F} = [\frac{\mathbf{f}_Q^1}{\|\mathbf{F}_Q\|_F},\cdots,\frac{\mathbf{f}_Q^n}{\|\mathbf{F}_Q\|_F}]$ and $\overline{\mathbf{F}}_{T_i} = \frac{\mathbf{F}_{T_i}}{\|\mathbf{F}_{T_i}\|_F} = [\frac{\mathbf{f}_{T_i}^1}{\|\mathbf{F}_{T_i}\|_F},\cdots,\frac{\mathbf{f}_{T_i}^n}{\|\mathbf{F}_{T_i}\|_F}]$.

Equation 9 can be rewritten as a weighted sum of the standard cosine similarities $\rho(\mathbf{f}_Q,\mathbf{f}_{T_i}) = \frac{\mathbf{f}_Q^T\mathbf{f}_{T_i}}{\|\mathbf{f}_Q\|\|\mathbf{f}_{T_i}\|}$ (Ma et al. 2007; Fu et al. 2008; Fu and Huang 2008) between each pair of corresponding feature vectors (i.e., columns) in $\mathbf{F}_Q, \mathbf{F}_{T_i}$ as follows:

$$\rho(\mathbf{F}_Q,\mathbf{F}_{T_i}) = \sum_{\ell=1}^{n}\frac{{\mathbf{f}_Q^{\ell}}^T\mathbf{f}_{T_i}^{\ell}}{\|\mathbf{F}_Q\|_F\|\mathbf{F}_{T_i}\|_F} = \sum_{\ell=1}^{n}\rho(\mathbf{f}_Q^{\ell},\mathbf{f}_{T_i}^{\ell})\frac{\|\mathbf{f}_Q^{\ell}\|\|\mathbf{f}_{T_i}^{\ell}\|}{\|\mathbf{F}_Q\|_F\|\mathbf{F}_{T_i}\|_F}. \tag{10}$$

The weights are represented as the product of $\frac{\|\mathbf{f}_Q^{\ell}\|}{\|\mathbf{F}_Q\|_F}$ and $\frac{\|\mathbf{f}_{T_i}^{\ell}\|}{\|\mathbf{F}_{T_i}\|_F}$ which indicate the relative importance of each feature in the feature sets $\mathbf{F}_Q, \mathbf{F}_{T_i}$. We see here an advantage of the MCS

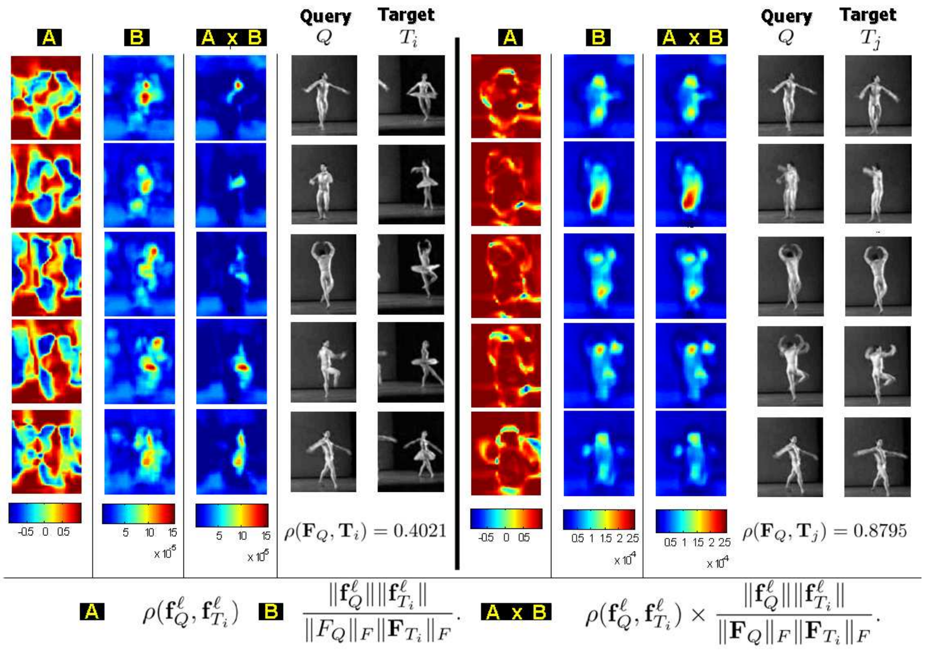


Fig. 9 Examples of cosine similarities and their corresponding weights throughout the features $\mathbf{f}_Q^\ell,\mathbf{f}_{T_i}^\ell$

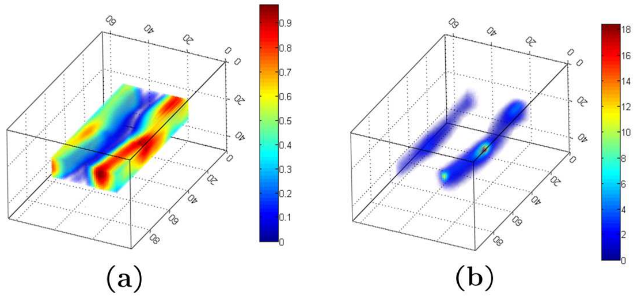
in that it takes account of the strength and angle similarity of vectors at the same time. Hence, this measure not only generalizes the cosine similarity naturally, but also overcomes the disadvantages of the conventional Euclidean distance which is sensitive to outliers. Fig. 9 shows examples of the computation of the MCS, which indicate that it provides a reliable measure of similarity.

We compute $\rho(\mathbf{F}_Q,\mathbf{F}_{T_i})$ over M (a possibly large number of) target cubes and this can be efficiently implemented by column-stacking the matrices $\mathbf{F}_Q,\mathbf{F}_{T_i}$ and simply computing the (vector) cosine similarity between two long column vectors as follows:

$$\begin{aligned}\rho_i \equiv \rho(\mathbf{F}_Q,\mathbf{F}_{T_i}) &= \sum_{\ell=1}^{n} \frac{{\mathbf{f}_Q^\ell}^T \mathbf{f}_{T_i}^\ell}{\|\mathbf{F}_Q\|_F\|\mathbf{F}_{T_i}\|_F} \\ &= \sum_{\ell=1}^{n}\sum_{j=1}^{d} \frac{f_Q^{(\ell,j)} f_{T_i}^{(\ell,j)}}{\sqrt{\sum_{\ell=1}^n \sum_{j=1}^d |f_Q^{(\ell,j)}|^2}\sqrt{\sum_{\ell=1}^n \sum_{j=1}^d |f_{T_i}^{(\ell,j)}|^2}}, \\ &= \rho(\text{colstack}(\mathbf{F}_Q),\text{colstack}(\mathbf{F}_{T_i})) \in [-1,1], \qquad (11)\end{aligned}$$

where $f_Q^{(\ell,j)}, f_{T_i}^{(\ell,j)}$ are elements in the ℓ^{th} vector $\mathbf{f}_Q^\ell$ and $\mathbf{f}_{T_i}^\ell$ respectively, and colstack$(\cdot)$ means an operator which column-stacks (rasterizes) a matrix.

It is worth noting that Shechtman and Irani 2007b proposed 3-D volume correlation score (global consistency measure between query and target cube) by computing a weighted average of local consistency measures. The difficulty with that method is that local consistency values should be explicitly computed from each corresponding subvolume of the query and target video. Furthermore, the weights to calculate a global consistency measure



(a) (b)

Fig. 10 (a) Resemblance volume (RV) which consists of $|\rho_i|$ (b) Resemblance volume (RV) which consists of $f(\rho_i)$. Note that Q and T are the same examples shown in Fig 2 where a female and a male ballet dancer are taking turning actions.

are based on a sigmoid function, which is somewhat ad-hoc. Here, we claim that our MCS measure is better motivated, and more general than their global consistency measure for action detection.

The next step is to generate a so-called “resemblance volume” (RV), which will be a volume of voxels, each indicating the likelihood of similarity between the Q and T. As for the final test statistic comprising the values in the resemblance volume, we use the *proportion* of shared variance (ρ_i^2) to that of the “residual” variance $(1-\rho_i^2)$. More specifically, RV is computed as follows:

$$\text{RV} : f(\rho_i) = \frac{\rho_i^2}{1-\rho_i^2}. \tag{12}$$

In Fig. 10, examples of resemblance volume (RV) based on $|\rho_i|$ and $f(\rho_i)$ are presented. Red color represents higher resemblance. As is apparent from these typical results, qualitatively, the resemblance volume generated from $f(\rho_i)$ provides better contrast and dynamic range in the result $(f(\rho_i) \in [0,\infty])$. More importantly, from a quantitative point of view, we note that $f(\rho_i)$ is essentially the Lawley-Hotelling Trace statistic (Tatsuoka 1988 ; Calinski et al. 2006), which is used as an efficient test statistic for detecting correlation between two data sets. Furthermore, it is worth noting that historically, this statistic has been suggested in the pattern recognition literature as an effective means of measuring the separability of two data clusters (e.g. Duda et al. 2000.)

2.3.2 Non-Parametric Significance Test

If the task is to find the most similar cube (T_i) to the query (Q) in the target video, one can choose the cube which results in the largest value in the RV (i.e., max $f(\rho_i)$) among all the cubes, no matter how large or small the value is in the range of $[0,\infty]$. This, however, is unwise because there may not be *any* action of interest present in the target video. We are therefore interested in two types of significance tests. The first is an overall test to decide whether there is any sufficiently similar action present in the target video at all. If the answer is yes, we would then want to know how many actions of interest are present in the target video and where they are. Therefore, we need two thresholds: an overall threshold τ_o and

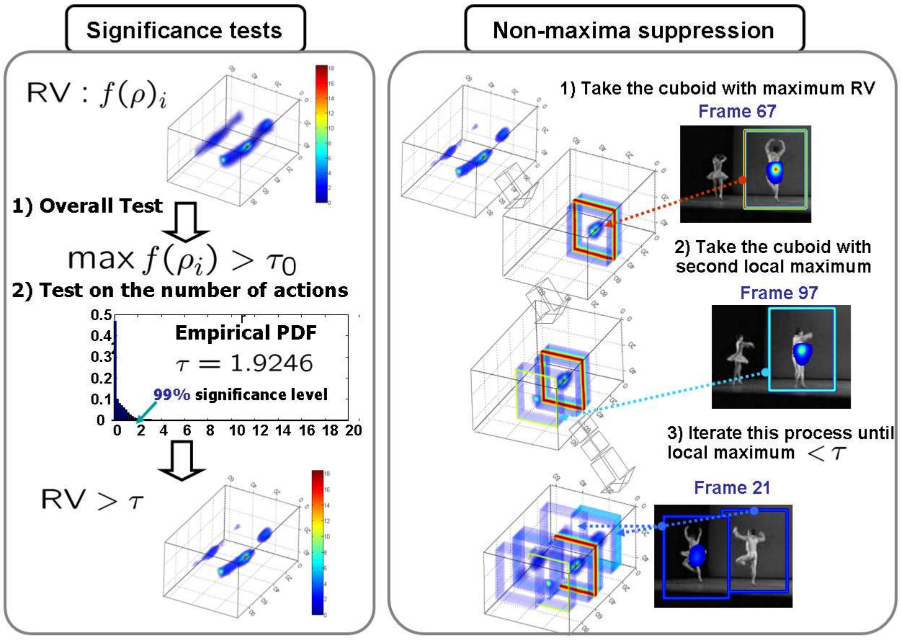


Fig. 11 Note that query and target are same as those in Fig. 2. Left: two significance tests, Right: Non-maxima suppression (Devernay 1995)

a threshold τ to detect the (possibly) multiple occurrences of similar actions in the target video.

In a typical scenario, we set the overall threshold τ_o to be, for instance, 0.96 which is about 50% of variance in common[10] (i.e., $\rho^2 = 0.49$). In other words, if the maximal $f(\rho_i)$ is just above 0.96, we decide that there exists at least one action of interest. The next step is to choose τ based on the properties of $f(\rho_i)$. When it comes to choosing the τ, there is need to be more careful. If we have a basic knowledge of the underlying distribution of $f(\rho_i)$, then we can make predictions about how this particular statistic will behave, and thus it is relatively easy to choose a threshold which will indicate whether the pair of features from the two videos are sufficiently similar. But, in practice, we do not have a very good way to model the distribution of $f(\rho_i)$. Therefore, instead of assuming a type of underlying distribution, we employ the idea of nonparametric testing. Namely, we compute an empirical probability density function (PDF) from M samples $f(\rho_i)$ and set τ so as to achieve, for instance, a 99 percent significance level in deciding whether the given values are in the extreme (right) tails of the distribution. This approach is based on the assumption that in the target video, most cubes do not contain the action of interest (in other words, action of interest is a relatively rare event), and therefore, the few matches will result in values which are in the tails of the distribution of $f(\rho_i)$.

[10] This in effect represents an unbiased choice reflecting our lack of prior knowledge about whether any similar actions are present at all.

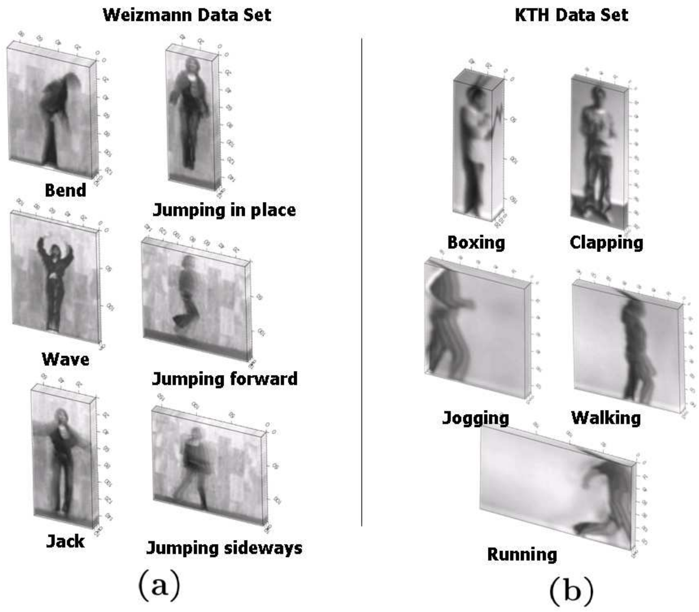


Fig. 12 Some examples (a) the Weizmann action dataset (Gorelick et al. 2007) (b) the KTH action dataset (Schuldt et al. 2004)

2.3.3 Non-maxima Suppression

After the two significance tests with τ_o, τ are performed, we employ the idea of non-maxima suppression (Devernay 1995) for the final detection. We take the volume region with the highest $f(\rho_i)$ value and eliminate the possibility that any other action is detected within some radius[11] of the center of that volume again. This enables us to avoid multiple false detections of nearby actions already detected. Then we iterate this process until the local maximum value falls below the threshold τ. Fig. 11 shows a graphical illustration of significance tests and non-maxima suppression.

3 Action Category Classification

As opposed to action detection, action category classification (Turaga et al. 2008, and references therein) aims to classify a given action query into one of several pre-specified categories as shown in Fig. 3. Examples of human actions from the Weizmann action dataset (Gorelick et al. 2007) and the KTH action dataset (Schuldt et al. 2004) are shown in Fig. 12. In earlier discussion on action detection, we assumed that in general the query video is smaller than target video. Now we relax this assumption, and thus we need a preprocessing step which selects a valid human action from the query video. This idea would not only allow us to extend the proposed detection framework to action category classification, but also improve both detection and classification accuracy by removing unnecessary background from the query video. Once the query video is cropped as a short action clip, the cropped query is searched against each labeled video in the database, and the value of the

[11] The size of this “exclusion” region will depend on the application at hand and the characteristics of the query video.

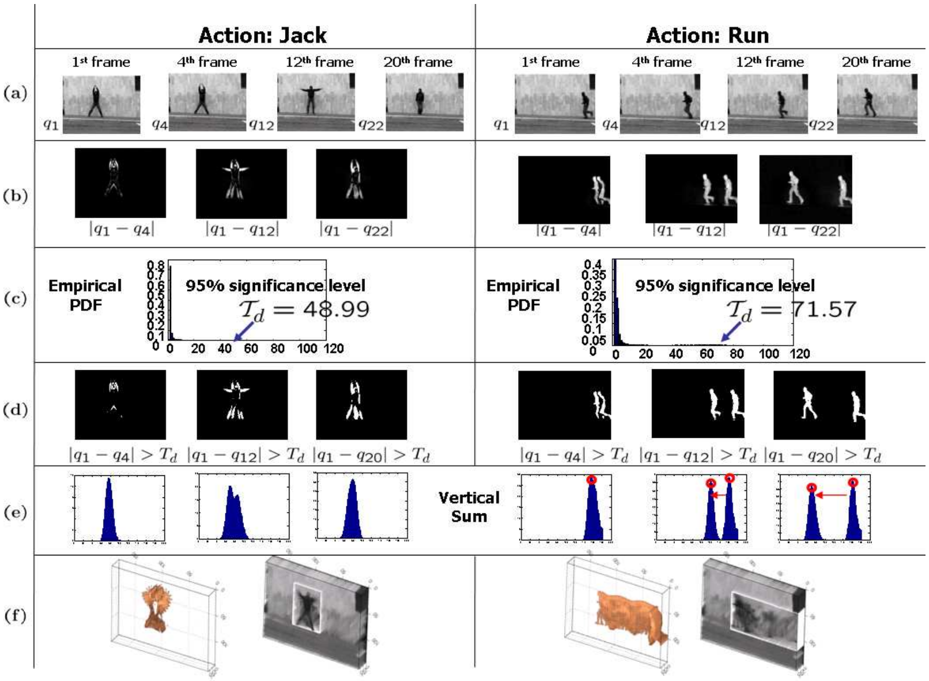


Fig. 13 Automatic action cropping (a) Query frames (b) absolute difference images (c) empirical PDF of absolute difference images (d) thresholded absolute difference images (e) vertical sum corresponding to thresholded absolute difference images (f) action shape by absolute difference images and query with a bounding cube containing action.

resemblance volume (RV) is viewed as the likelihood of similarity between the query and each labeled video. Then we classify a given query video as one of the predefined action categories using a nearest neighbor (NN) classifier.

3.1 Automatic Action Cropping

In this section, we introduce a procedure which automatically segments from the original query video a small cube with 1 second (25 frames) length [12] that contains human action. We further decide whether the action has a direction or not (for instance, in case of running, walking, and jogging actions, is it moving to the left or right?). First, in order to reduce the effect of noise, the query video is spatially blurred with a Gaussian kernel of size [5 × 5] with $\sigma = 1.5$. Then we assign the first frame as a key frame, and compute absolute difference images by subtracting each frame from the key frame. Next, we utilize the idea of non-parametric significance testing again. With a collection of absolute difference images, we compute an empirical PDF from all the difference values and set a threshold τ_d so as to achieve, a 95% significance level in deciding whether the given difference values are in the extreme (right) tails of the empirical PDF of the absolute difference values. The approach is based on the assumption that in the query video, human action is a rare event and thus

[12] Ning et al. 2008 pointed out that 1 second length video clip typically confines human action. We also found that from our experiments, 25 frames is long enough for the query.

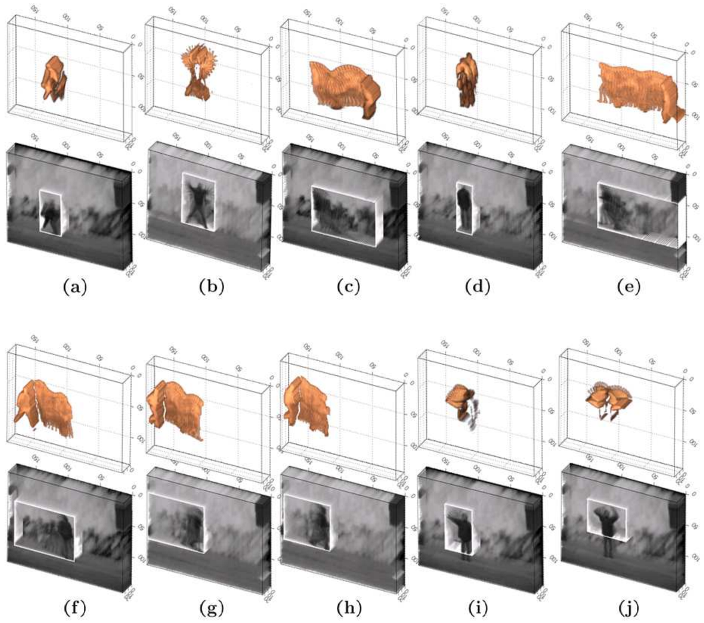


Fig. 14 Examples of action shape by thresholded absolute difference images and query videos with a bounding cube containing action on the Weizmann action dataset (a) bend (b) jump in place (c) wave with one hand (d) wave with two hands (e) run (f) jump (g) side (h) walk (i) skip (j) jack.

results in values which are in the tails of the distribution. After thresholding each absolute difference image with τ_d, we calculate a vertical sum to check whether there is a direction associated with the action. In order to also reduce noise effect from the vertical sum, we apply a Gaussian blurring to the vertical sum with a kernel of size 5 with $\sigma = 3$. If there are two maxima throughout all frames and one is deviating from the other, we classify this query video as an action with a dominant direction. Otherwise, the query video is considered to have no direction. What we do next is to crop only valid human action region by fitting a 3-D rectangular box to a collection of thresholded absolute difference images. Figure 13 shows the entire procedure of automatic action cropping on two action categories (Jack and Run).

4 Experimental Results

In this section, we demonstrate the performance of the proposed method with comprehensive experiments on three datasets; namely, the general action dataset (Shechtman and Irani 2007b), the Weizmann action dataset (Gorelick et al. 2007), and the KTH action dataset (Schuldt et al. 2004). While the general action dataset was used to evaluate detection performance of the proposed method, the Weizmann action dataset and the KTH action dataset were exploited for action categorization. Comparison is made with other state-of-the-art methods that have reported their results on these datasets.

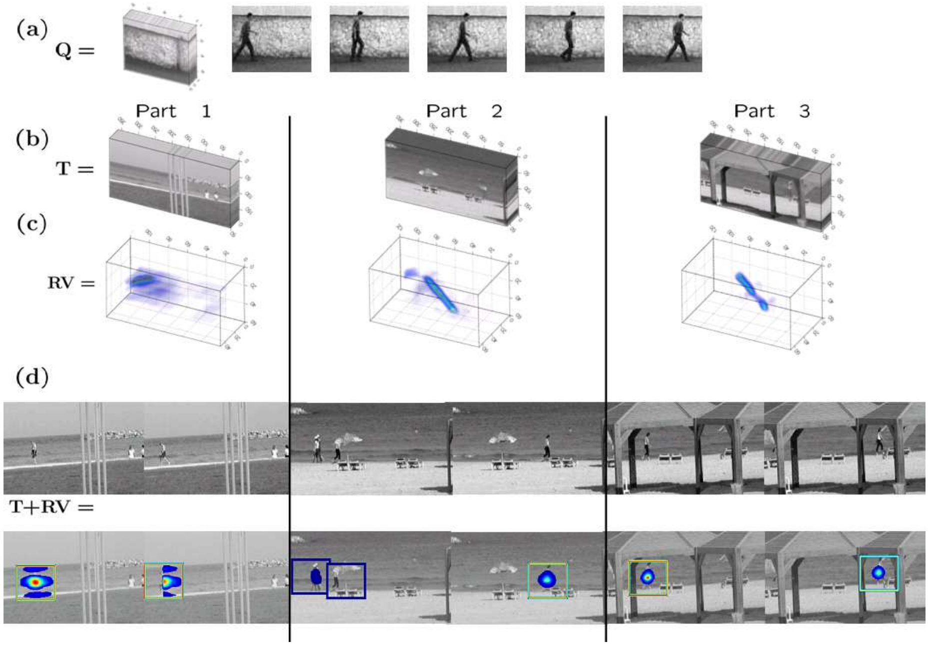


Fig. 15 Results searching for walking person on the beach (a) query video (a short walk clip) (b) target video (c) Resemblance volumes (RV) (d) a few frames from T (e) frames with resemblance volume on top of it.

4.1 Action Detection

In this section, we show several experimental results on searching with a short query video against a (typically longer and larger) target video. Our method detects the presence and location of actions similar to the given query and provides a series of bounding cubes with resemblance volume embedded around detected actions. Note that no background/foreground segmentation is required in the proposed method. This method can also handle modest amount of variations in rotation (up to ±15 degree), and spatial and temporal scale change (up to ±20%). Once given Q and T (typically Q of 60 × 70 pixels and T of 180 × 360 pixels), we blur and downsample both Q and T by a factor of 3 in order to reduce the time-complexity. We then compute 3-D LSK of size 3×3 (space) $\times 7$ (time) as descriptors so that every space-time location in Q and T yields a 63-dimensional local descriptor $\mathbf{W}_Q$ and $\mathbf{W}_T$ respectively. The reason why we choose a lager time axis size than space axis of the cube is that we focus on detecting similar actions regardless of different appearances, thus we give a higher priority to temporal evolution information than spatial appearance. The smoothing parameter h for computing 3-D LSKs was set to 2.1. We end up with $\mathbf{F}_Q, \mathbf{F}_T$ by reducing dimensionality from 63 to $d = 4$ and then, we obtain RV by computing the MCS measure between $\mathbf{F}_Q, \mathbf{F}_T$. The threshold τ for each test example was determined by the 99 percent confidence level. We applied our method to 3 different examples : *i.e.* detecting 1) walking people, 2) ballet turn actions, and 3) multiple actions in one video. Shechtman and Irani 2007b have tested their method on these videos using the same query and Shechtman and Irani 2007a and Ning et al. 2008 also tested their methods on some of these videos. Visually, we achieved similar (or even better) performance as compared to the methods in Shechtman and Irani 2007a; Shechtman and Irani 2007b; Ning et al. 2008 as shown in Fig 15, 16, and

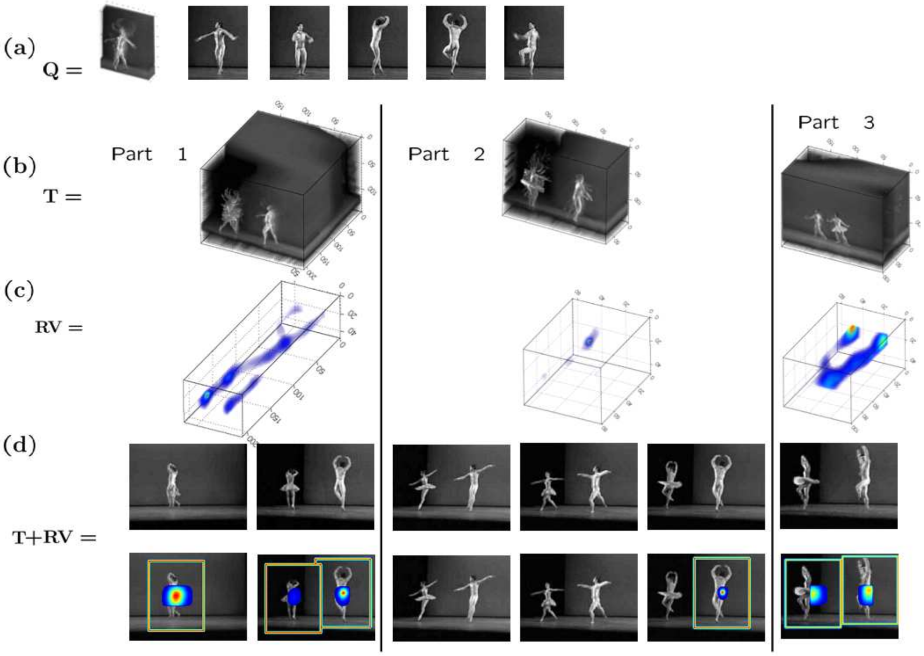


Fig. 16 Results searching for ballet turn on the ballet video (a) query video (a short ballet turn clip) (b) target video (c) resemblance volumes (RV) (d) a few frames from T (e) video frames with resemblance volume on top of it.

17. It is worth noting here that these training-free action detection methods did not provide either threshold values or describe how they selected threshold values in reporting detection performance. On the other hand, the threshold values are automatically chosen in our algorithm with respect to the confidence level as explained earlier.

Fig. 15 shows the results of searching for instances of walking people in a target beach video (460 frames of 180 × 360 pixels). The query video contains a very short walking action moving to the right (14 frames of 60 × 70 pixels) and has a background context which is not the beach scene. In order to detect walking actions in either directions, we used two queries (Q and its mirror-reflected version) and generated two RVs. By voting the higher score among values from two RVs at every space-time location, we arrived at one RV which includes correct locations of walking people in the correct direction. Fig. 15 (a) shows a few sampled frames from Q. In order to provide better illustration of T, we divided T into 3 non-overlapping sections. Fig. 15 (b) and (c) represent each part of T and its corresponding RV respectively. Red color represents higher resemblance while blue color denotes lower resemblance values. Fig. 15 (d) and (e) show a few frames from T and RVs superimposed on T respectively.

Fig. 16 shows the results of detecting ballet turning action in a target ballet video (284 frames of 144 × 192 pixels). The query video contains a single turn of a male dancer (13 frames of 90 × 110 pixels). Fig. 16 (a) shows a few sampled frames from Q. Next, Fig. 16 (b) and (c) represent each part of T and its corresponding RV respectively. Fig. 16 (d) and (e) show a few frames from T and RVs superimposed on T respectively. Most of the turns of the two dancers (a male and a female) were detected even though this video contains

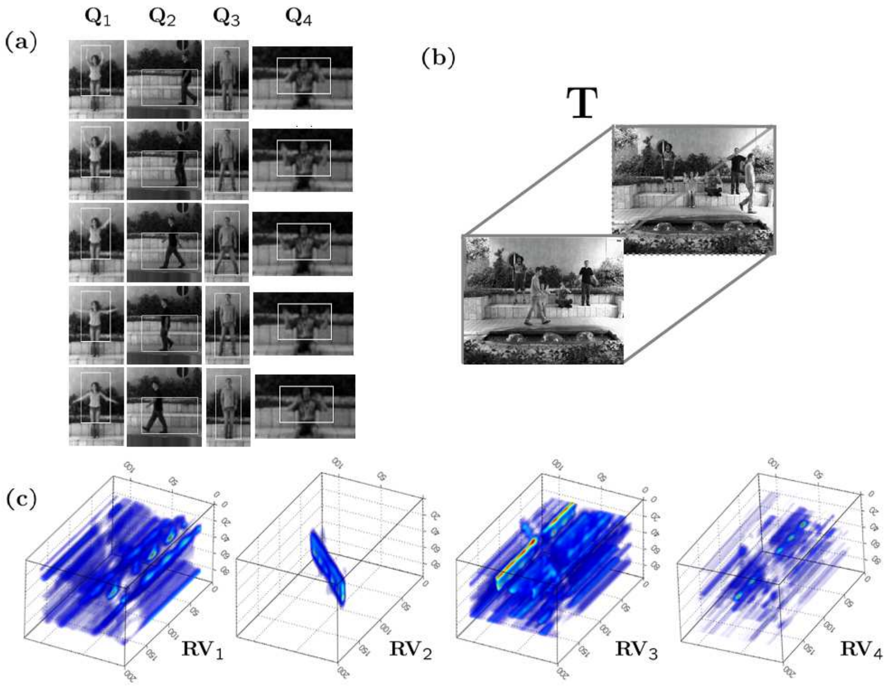


Fig. 17 Results searching for multiple actions (a) four different short video queries. Note that white boxes represent automatic cropping results as explained in Sec 3.1. (b) target video T (c) resemblance volumes (RV)s with respect to each query.

very fast moving parts and contains large variability in spatial scale and appearance (the female dancer wearing a skirt) as compared to the given query Q. We observed that one of the female dancer's turning actions was missed because of large spatial scale variation as compared to the given Q. However, we can easily deal with this problem by either adjusting the significance level or using a multi-scale approach as done in Seo and Milanfar[5]. The detection result of the proposed method on this video visually outperforms those in Shechtman and Irani 2007b; Ning et al. 2008 which had a number of miss detections and false alarms, and compares favorably to that in Shechtman and Irani 2007a in terms of visual detection accuracy.

Fig. 17 shows the results of detecting 4 different actions ("walk", "wave", "clap", and "jump") which occur simultaneously in a target video (120 frames of 288 × 360 pixels). Four query videos were matched against the target video independently. Fig. 17 (a) and (b) show a few sampled frames from Q and T respectively. White boxes in Fig. 17 (a) represent automatic cropping results as explained in Sec 3.1. The resulting RVs are shown in Fig. 17 (c). Most of the actions were detected although one of two "clap" actions on the target video was missed.

In all the above examples, we used the same parameters. It is evident, based on all the results above, that the proposed training-free action detection based on 3-D LSK works well and is robust to modest variations in spatiotemporal scale.

4.2 Action Category Classification

Our baseline algorithm is designed for detecting actions in videos, but this method can also be extended to action classification as explained in Section 3. We conducted an extensive set of experiments to evaluate the action classification performance of the proposed method on the Weizmann action dataset (Gorelick et al. 2007) and the KTH action dataset (Schuldt et al. 2004).

4.2.1 Weizmann Action Data Set

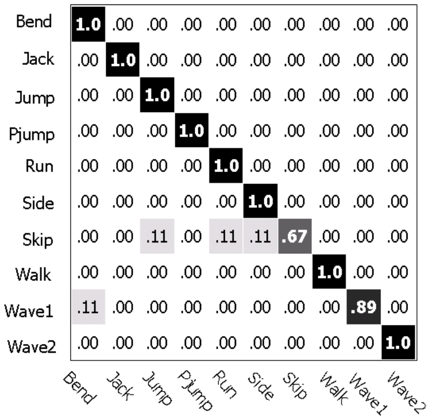
The Weizmann action dataset contains 10 actions (bend, jack, jump forward, jump in place, jump sideways, skip, run, walk, wave with two hands, and wave with one hand) performed by 9 different subjects (See Fig. 12 (a).) This dataset contains videos with static cameras and simple background, but it provides a good testing environment to evaluate the performance of the algorithm when the number of categories are large compared to the KTH dataset (a total of 6 categories). The testing was performed in a "leave-one-out" setting, *i.e.*, for each run the videos of 8 subjects are labeled and the videos of the remaining subject are used for testing (query). We applied the automatic action cropping method introduced in Section 3.1 to the testing video. Then the resulting short action clip is matched against the remaining labeled videos using the proposed method. We classify each testing video as one of the 10 action types by 3-NN (nearest neighbor) as similarly done in Ning et al. 2008. The results are reported as the average of nine runs. We were able to achieve a recognition rate of 96% for all ten actions. The recognition rate comparison is provided in Table 1 as well. The proposed method which is training-free performs favorably against state-of-the-art methods (Niebles et al. 2008; Junejo et al. 2008; Liu et al. 2008; Jhuang et al. 2007; Ali and Shah 2008; Batra et al. 2008) which largely depend on training[13].

Table 1 Comparison of average recognition rate on the Weizmann dataset (Gorelick et al. 2007)

Our Approach (1-NN)	Juenjo *et al.* (Junejo et al. 2008)	Liu *et al.* (Liu et al. 2008)
90%	95.33%	90%
Our Approach (2-NN)	Niebles *et al.* (Niebles et al. 2008)	Ali *et al.* (Ali and Shah 2008)
90%	90%	95.75%
Our Approach (3-NN)	Jhuang *et al.* (Jhuang et al. 2007)	Batra *et al.* (Batra et al. 2008)
96%	98.8%	92%

We further provide the results using 1-NN and 2-NN for comparison. We observe that these results also compare favorably to several state-of-the-art methods even though our method involves no training phase, and requires no background/foreground segmentation. It is worth noting that the method in Jhuang et al. 2007 is not designed for action localization, but only for action classification. Fig. 18 shows the confusion matrix for our method. Note that our method is mostly confused by similar action classes, such as "skip" with "jump","run", and "side".

[13] It is worth noting that different groups employed different experimental methodologies. There are broadly two main evaluation methods: 1) leave-one-out and 2) split-data-equally. The split-data-equally means that the a collection of video sequences are divided into two equal sets randomly: one for training examples and the other for testing (query). Since our method does not involve any training, we adopted the leave-one-out in this paper.



	Bend	Jack	Jump	Pjump	Run	Side	Skip	Walk	Wave1	Wave2
Bend	1.0	.00	.00	.00	.00	.00	.00	.00	.00	.00
Jack	.00	1.0	.00	.00	.00	.00	.00	.00	.00	.00
Jump	.00	.00	1.0	.00	.00	.00	.00	.00	.00	.00
Pjump	.00	.00	.00	1.0	.00	.00	.00	.00	.00	.00
Run	.00	.00	.00	.00	1.0	.00	.00	.00	.00	.00
Side	.00	.00	.00	.00	.00	1.0	.00	.00	.00	.00
Skip	.00	.00	.11	.00	.11	.11	.67	.00	.00	.00
Walk	.00	.00	.00	.00	.00	.00	.00	1.0	.00	.00
Wave1	.11	.00	.00	.00	.00	.00	.00	.00	.89	.00
Wave2	.00	.00	.00	.00	.00	.00	.00	.00	.00	1.0

Fig. 18 Average confusion matrix for the Weizmann action dataset. (Here, 3-NN was used as similarly done in Ning et al. 2008.)

4.2.2 KTH Action Data Set

In order to further verify the performance of our algorithm, we also conducted experiments on the KTH dataset. The KTH action dataset contains six types of human actions (boxing, hand waving, hand clapping, walking, jogging, and running), performed repeatedly by 25 subjects in 4 different scenarios: outdoors (c_1), outdoors with camera zoom (c_2), outdoors with different clothes (c_3), and indoors (c_4). Some samples are shown in Fig. 12 (b). This dataset seems more challenging than the Weizmann dataset because there are large variations in human body shape, view angles, scales, and appearance. The "leave-one-out" cross validation is again used to measure the performance. More specifically, for each run the videos of 24 subjects are designated as labeled video sets and the videos of the remaining subject is used for testing. Fig. 19 shows the confusion matrices from our method for each scenario. Fig. 20 shows the average confusion matrix across all scenarios. We were able to achieve a recognition rate of 95.66% on these six actions. The recognition rate comparison with competing methods is provided in Table 2 as well. It is worth noting that our method outperforms all the other state-of-the-art methods and is fully automatic like the method in Ning et al. 2008 while the method in Kim et al. 2007 manually aligned the actions in space-time. Table 3 further shows that our scenario-wise recognition rates are consistently higher than those reported in Ning et al. 2008, and Jhuang et al. 2007.

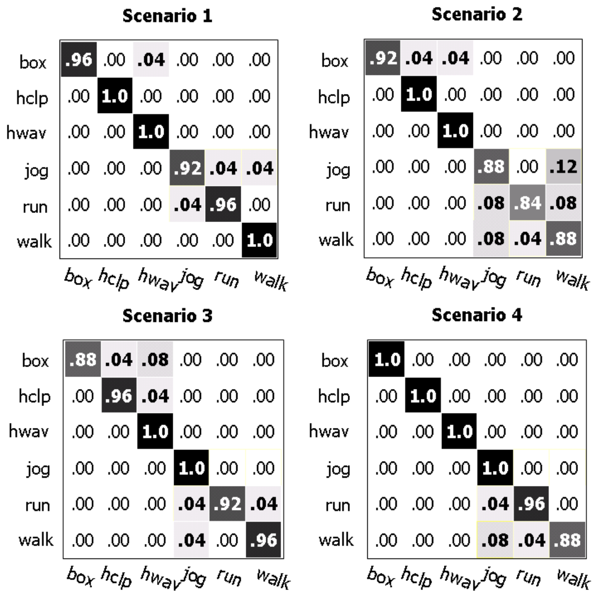
Table 2 Detailed comparison of recognition rate on the KTH dataset. *Avg* is the average across 4 scenarios.

Methods	c_1	c_2	c_3	c_4	*Avg*
Our Approach	**97.33**%	**92.67**%	**95.3**%	**97.32**%	**95.66**%
Ning *et al.* (Ning et al. 2008) (3-NN)	95.56 %	82.41 %	90.66 %	94.72%	92.09%
Jhuang *et al.* (Jhuang et al. 2007)	96.0 %	89.1 %	89.8 %	94.8%	91.7%

Our system is designed with recognition accuracy as a high priority. A typical run of the action detection system takes a little over 1 minute on a target video T (50 frames of 144×192 pixels, Intel Pentium CPU 2.66 Ghz machine) using a query Q (13 frames of 90×110). Most of the run-time is taken up by the computation of MCS (about 9 seconds,

Table 3 Comparison of average recognition rate on the KTH dataset

Our Approach (1-NN)	Kim *et al.* (Kim et al. 2007)	Ning *et al.* (Ning et al. 2008)
89%	95.33%	92.31% (3-NN)
Our Approach (2-NN)	Ali *et al.* (Ali and Shah 2008)	Niebles *et al.* (Niebles et al. 2008)
93%	87.7%	81.5%
Our Approach (3-NN)	Dollar *et al.* (Dollar et al. 2005)	Wong *et al.* (Wong et al. 2007)
95.66%	81.17%	84%


Scenario 1

	box	hclp	hwav	jog	run	walk
box	.96	.00	.04	.00	.00	.00
hclp	.00	1.0	.00	.00	.00	.00
hwav	.00	.00	1.0	.00	.00	.00
jog	.00	.00	.00	.92	.04	.04
run	.00	.00	.00	.04	.96	.00
walk	.00	.00	.00	.00	.00	1.0

Scenario 2

	box	hclp	hwav	jog	run	walk
box	.92	.04	.04	.00	.00	.00
hclp	.00	1.0	.00	.00	.00	.00
hwav	.00	.00	1.0	.00	.00	.00
jog	.00	.00	.00	.88	.00	.12
run	.00	.00	.00	.08	.84	.08
walk	.00	.00	.00	.08	.04	.88

Scenario 3

	box	hclp	hwav	jog	run	walk
box	.88	.04	.08	.00	.00	.00
hclp	.00	.96	.04	.00	.00	.00
hwav	.00	.00	1.0	.00	.00	.00
jog	.00	.00	.00	1.0	.00	.00
run	.00	.00	.00	.04	.92	.04
walk	.00	.00	.00	.04	.00	.96

Scenario 4

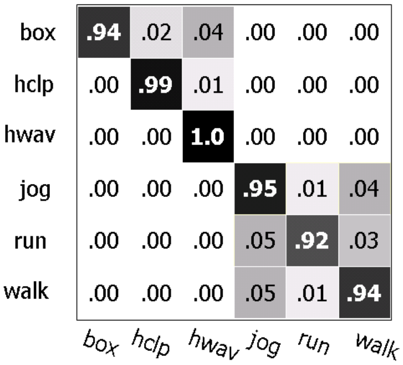
	box	hclp	hwav	jog	run	walk
box	1.0	.00	.00	.00	.00	.00
hclp	.00	1.0	.00	.00	.00	.00
hwav	.00	.00	1.0	.00	.00	.00
jog	.00	.00	.00	1.0	.00	.00
run	.00	.00	.00	.04	.96	.00
walk	.00	.00	.00	.08	.04	.88

Fig. 19 Tables of confusion matrix for the KTH action dataset in each scenario (Here, 3-NN was used as similarly done in Ning et al. 2008.)

and 16.5 seconds for the computation of 3-D LSKs from Q and T respectively, which needs to be computed only once.) There are many factors that affect the precise timing of the calculations, such as query size, complexity of the video, and 3-D LSK size. Our system runs in Matlab but could be easily implemented using multi-threads or parallel programming as well as General Purpose GPU for which we expect a significant gain in speed. Even though our method is stable in the presence of moderate amount of camera motion, our system can benefit from camera stabilization methods as done in Medioni et al. 2001 and Veit et al. 2004 in case of large camera movements.

5 Conclusion and Future Directions

In this paper, we have proposed a novel action recognition algorithm by employing *space-time local steering kernels* (3-D LSKs) which robustly capture underlying space-time data structure; and by using a training-free nonparametric detection scheme based on "Matrix Cosine Similarity" (MCS) measure. The proposed method can automatically detect in the



	box	hclp	hwav	jog	run	walk
box	.94	.02	.04	.00	.00	.00
hclp	.00	.99	.01	.00	.00	.00
hwav	.00	.00	1.0	.00	.00	.00
jog	.00	.00	.00	.95	.01	.04
run	.00	.00	.00	.05	.92	.03
walk	.00	.00	.00	.05	.01	.94

Fig. 20 Average confusion matrix for the KTH action dataset across all scenarios (Here, 3-NN was used as similarly done in Ning et al. 2008.)

target video the presence, the number, as well as location of actions similar to the given query video. In order to increase the detection accuracy and further deal with action classification, we developed a simple but effective automatic action cropping method. Challenging sets of real-world human action experiments demonstrated that the proposed approach achieves a high recognition accuracy and improves upon other state-of-the-art methods. Unlike most of the state-of-the-art methods that involve training phases, background/foreground segmentation, and manual aligning of actions, the proposed method operates using a *single* example of an action of interest to find similar matches; does not require any prior knowledge (learning) about actions being sought; and does not require any segmentation or pre-processing step of the target video. In order to improve time-efficiency of the proposed method, a coarse-to-fine approach can be applied or a background subtraction based on space-time saliency detection (Mahadevan and Vasconcelos 2008; Marat et al. 2009) can be utilized. Since local regression kernels in 2-D and in 3-D were originally designed for image (video) restoration, the proposed framework should solve the joint problem of simultaneous super-resolution and recognition when there might be a low-resolution query while the database contains only high-resolution images (videos). By computing local regression kernels from images (video) at once, we may be able to not only detect objects (actions) of interest, but denoise, deblur, and super-resolve images (videos) at the same time. These aspects of the work are the subject of ongoing research.

Acknowledgements This work was supported in part by AFOSR Grant FA 9550-07-01-0365.

References

Ahlgren P, Jarneving B, Rousseau R (2003) Requirements for a cocitation similarity measure, with special reference to peasron's correlation coefficient. Journal of the American Society for Information Science and Technology 54(6):550–560

Ali S, Shah M (2008) Human action recognition in videos using kinematic features and multiple instance learning. Accepted for publication in IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)

Batra D, Chen T, Sukthankar R (2008) Space-time shapelets for action recogntion. IEEE Workshop on Motion and video Computing (WMVC) pp 1–6

Bobick A, JWDavis (2008) The recognition of human movement using temporal templates. IEEE Transactions on Pattern Analysis and Machine Intelligence 23:1257–1265

Boiman O, Shechtman E, Irani M (2008) In defense of nearest-neighbor based image classification. IEEE Conference on Computer Vision and Pattern Recognition (CVPR) pp 1–8

Boulanger J, Kervrann C, Bouthemy P (2005) Space-time adaptation for patch-based image sequence restoration. IEEE Transactions on Pattern Analysis and Machine Intelligence 29:1096–1102

Buades A, Coll B, Morel JM (2008) Nonlocal image and movie denoising. International Journal of Computer Vision 76(2):123–139

Calinski T, Krzysko M, Wolynski W (2006) A comparison of some tests for determining the number of nonzero canonical correlations. Communication in Statistics, Simulation and Computation 35:727–749

Devernay F (1995) A non-maxima suppression method for edge detection with sub-pixel accuracy. Technical report, INRIA (RR-2724)

Dollar P, Rabaud V, Cottrell G, Belongie S (2005) Behavior recognition via sparse spatio-temporal features. In proceeding of Visual Surveillance and Performance Evaluation of Tracking and Surveillance (VS-PETS) pp 65–72

Duda R, Hart P, Stark D (2000) Pattern Classification, 2nd Edition. John Wiley and Sons Inc, New York

Elgammal A, Lee C (2004) Inferring 3d body pose from silhouettes using activity manifold learning. IEEE Conference on Computer Vision and Pattern Recognition (CVPR) pp 681–688

Fu Y, Huang TS (2008) Image classification using correlation tensor analysis. IEEE Transactions on Image Processing 17(2):226–234

Fu Y, Yan S, Huang TS (2008) Correlation metric for generalized feature extraction. IEEE Transactions on Pattern Analysis and Machine Intelligence 30(12):2229–2235

Gorelick L, Blank M, Shechtman E, Irani M, Basri R (2007) Actions as space-time shapes. IEEE Transactions on Pattern Analysis and Machine Intelligence 29:2247–2253

Jhuang H, TSerre, LWolf, TPoggio (2007) A biologically inspired system for action recogntion. IEEE International Conference on Computer Vision(ICCV) pp 1–8

Junejo I, Dexter E, Laptev I, Prez P (2008) Cross-view action recognition from temporal self-similarities. In Proc European Conference Computer Vision (ECCV'08) 2:293–306

Ke Y, Sukthankar R (2004) PCA-SIFT: A more distinctive representation for local image descriptors. IEEE Conference on Computer Vision and Pattern Recognition (CVPR) pp 506–513

Ke Y, Sukthankar R, Hebert M (2005) Efficient visual event detection using volumetric features. IEEE Conference on Computer Vision and Pattern Recognition (CVPR) pp 166–173

Kim T, Wong S, Cipolla R (2007) Tensor canonical correlation analysis for action classification. IEEE Conference on Computer Vision and Pattern Recognition (CVPR) pp 1–3

Lin D, Yan S, Tang X (2005) Comparative study: Face recognition on unspecific persons using linear subspace methods. IEEE International Conference on Image Processing 3:764–767

Liu C (2007) The bayes decision rule induced similarity measures. IEEE Transactions on Pattern Analysis and Machine Intelligence 29(6):1086–1090

Liu C (2008) Clarification of assumptions in the relationship between the bayes decision rule and the whitened cosine similarity measure. IEEE Transactions on Pattern Analysis and Machine Intelligence 30(6):1116–1117

Liu J, Ali S, Shah M (2008) Recognizing human actions using multiple features. IEEE Conference on Computer Vision and Pattern Recognition (CVPR) pp 1–8

Ma Y, Lao S, Takikawa E, Kawade M (2007) Discriminant analysis in correlation similarity measure space. International Conference on Machine Learning 227:577–584

Mahadevan V, Vasconcelos N (2008) Background subtraction in highly dynamic scenes. IEEE Conference on Computer Vision and Pattern Recognition (CVPR) pp 1–6

Marat S, Phuoc TH, Granjon L, Guyader N, Pellerin D, G-Dogue A (2009) Modelling spatio-temporal saliency to predict gaze direction for short videos. Published online: International Journal of Computer Vision (IJCV)

Mazzaro M, Sznaier M, Camps O (2005) A model validation approach to gait classification. IEEE Transactions on Pattern Analysis and Machine Intelligence 27:1820–1825

Medioni G, Cohen I, Bremond F, Hongeng S, Nevatia R (2001) Event detection and analysis from video streams. IEEE Transactions on Pattern Analysis and Machine Intelligence 23:873–890

Niebles J, Fei-Fei L (2007) A hierarchical models of shape and appearance for human action classification. IEEE Conference on Computer Vision and Pattern Recognition (CVPR) pp 1–8

Niebles J, Wang H, Fei-Fei L (2008) Unsupervised learning of human action categores usig spatial-temporal words. International Journal of Computer Vision 79:299–318

Ning H, Han T, Walther D, Liu M, Huang T (2008) Hierarchical space-time model enabling efficient search for human actions. IEEE Transactions on Circuits and Systems for Video Technology, in press

Pavlovic V, Rehg J, MacCormick J (2000) Learning switching linear models of human motion. In Advances in Neural Information Processing Systems pp 981–987

Rodgers J, Nicewander W (1988) Thirteen ways to look at the correlation coefficient. The American Statistician 42(1):59–66

Schneider JW, Borlund P (2007) Matrix comparison, part 1: Motivation and important issues for measuring the resemblance between proximity measures or ordination results. Journal of the American Society for Information Science and Technology 58(11):1586–1595

Schuldt C, Laptev I, Caputo B (2004) Recognizing human actions: A local svm approach. IEEE Conference on Pattern Recognition (ICPR) 3:32–36

Shechtman E, Irani M (2007a) Matching local self-similarities across images and videos. In Proc of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) pp 1–8

Shechtman E, Irani M (2007b) Space-time behavior-based correlation -or- how to tell if two underlying motion fields are similar without computing them? IEEE Transactions on Pattern Analysis and Machine Intelligence 29:2045–2056

Starner T, Weaver J, Pentland A (1998) Real-time american sign language recognition using desk and wearable computer based video. IEEE Transactions on Pattern Analysis and Machine Intelligence 20:1371–1375

Takeda H, Farsiu S, Milanfar P (2007) Kernel regression for image processing and reconstruction. IEEE Transactions on Image Processing 16(2):349–366

Takeda H, van Beek P, Milanfar P (2008a) Spatio-temporal video interpolation and denoising using motion-assisted steering kernel (MASK) regression. IEEE International Conference on Image Processing (ICIP) pp 637–640

Takeda H, Farsiu S, Milanfar P (2008b) Deblurring using regularized locally-adaptive kernel regression. IEEE Transactions on Image Processing 17:550–563

Tatsuoka M (1988) Multivariate Analysis. Macmillan

Turaga P, Chellappa R, Subrahmanian V, Udrea O (2008) Machine recognition of human activities: A survey. IEEE Transactions on Circuits and Systems for Video Technology

18:1473–1488

Veit PT, Cao F, Bouthemy P (2004) Probabilistic parameter-free motion detection. IEEE Conference on Computer Vision and Pattern Recognition (CVPR) pp 715–721

Viola P, Jones M (2004) Robust real-time object detection. International Journal of Computer Vision 57(2):137–154

Wong SF, Kim TK, Cipolla R (2007) Learning motion categories using both semantic and structural informtion. IEEE Conference on Computer Vision and Pattern Recognition (CVPR) pp 1–6

Yeo C, Ahammad P, Ramchandran K, Satry SS (2008) High-speed action recognition and localization in compressed domain videos. IEEE Transactions on Circuits and Systems for Video Technology 18:1006–1015