Linear Regression Motion Analysis for Unsupervised Temporal Segmentation of Human Actions

Simon Jones, Ling Shao
Department of Electronic and Electrical Engineering
The University of Sheffield, Mappin St, Sheffield, S1 3JD, UK
simon.m.jones@sheffield.ac.uk, ling.shao@sheffield.ac.uk

Abstract

One of the biggest difficulties in human action analysis is the temporal complexity and structure of actions. By breaking actions down into smaller temporal pieces, it may be possible to enhance action recognition, or allow unsupervised temporal action clustering. We propose a temporal segmentation system for human action recognition based on person tracking and a novel segmentation algorithm. We apply optical flow, PCA, and linear regression error estimation to human action videos to get a metric, L', that can be used to split an action into several more easily recognised sub-actions. The L' metric can be effectively calculated and is robust. To validate the semantic coherence of the sub-actions, we represent the sub-actions as features using a variant of the Motion History Image and perform action recognition experiments on two popular datasets, the KTH and the MSR2. Our results demonstrate that the algorithm performs well, showing promise for future application in action clustering and action retrieval tasks.

1. Introduction

While existing research on human action recognition concerns itself primarily with temporally pre-segmented actions, real-world recognition systems work with temporally contiguous actions. Many existing works, such as Kläser et al. [10], take a fully supervised approach to this problem, training a model to localise specific actions within a longer video. However, these techniques require training and prior knowledge of the actions to be localised. Additionally, the complexity of this type of localisation scales linearly with the number of action classes to be found. The goal of *unsupervised temporal segmentation*, on the other hand, is to split a video or track into temporally discrete blocks, based on some inherent property of the video (e.g. start/end of linear motion), rather than relying on training data. Unsupervised temporal segmentation could potentially be applied in many scenarios, including video representation for multimedia retrieval, keyframe analysis, unsupervised action categorisation, and efficient action detection. Research on unsupervised temporal segmentation of human actions has been sparse, though there are some notable recent works [25, 20, 15] that have utilised this for action clustering.

One of the biggest prerequisites for accurate temporal segmentation of human actions is to accurately spatially isolate the human from its surroundings. New and effective person detection methods, and tracking works such as [18] and [7], mean that it is increasingly possible to get a tight, predictable bounding box around a person in almost every frame, spatially isolating it from the rest of the scene, and allowing almost pixel-perfect alignment of the person between frames. Such tracking can mitigate background noise, translation and scale variations, as well as camera motion – this facilitates temporal segmentation based purely on the motion of the human.

In this work, we develop an unsupervised temporal segmentation method that, due to its efficiency, could have varied applications in action recognition and action clustering – we propose that it is particularly applicable to surveillance videos. First, we apply an effective human tracker to the human subject, removing any background noise, as well as compensating for translational and scale variations. Secondly, we apply our method for unsupervised segmentation to the human tracks, resulting in a series of sections of self-consistent linear motion.

We consider the primary application of unsupervised temporal segmentation to be temporal clustering of human actions, such as in Turaga et al.[20]. However, as a preliminary measure to prove the effectiveness of our segmentation method, we integrate it into an ac-

tion recognition system. We propose that short temporal segments can be formulated into more effective features for action recognition than the full action, either by using them as independent features in a model such as Bag-of-Words or Naive Bayes Nearest Neighbour (NBNN), or by treating them as states in a time-sequence model such as Hidden Markov Models. We prove this hypothesis through experimentation.

The rest of this paper is as follows. Section 2 gives an overview of works related to this paper. In Section 3 we present our methodology. It is split into three parts, describing person tracking, temporal segmentation and action representation respectively. Section 4 details our experiments on two datasets: KTH and MSR2, demonstrating the validity of our technique. Finally we discuss our findings in Section 5.

2. Literature Review

Action recognition work has historically been divided into that of local and global representations. Local feature-based methods usually rely on a spatio-temporal interest point detector [4], and then a descriptor (such as HOG3D [9]) to describe the areas surrounding those points. These local points are then combined with a statistical representation such as Bag of Words (BoW), or a structural one such as shape contexts [6]. These local features have traditionally shown the most impressive performance on realistic datasets. Global representations, on the other hand, have focused on the overall appearance of the action. An early example of this is the Motion History Image (MHI) [3]. More recently, global representations have focused on keyframes and poses, such as the Bag of Correlated Poses (BoCP) by Wu and Shao [22]. One more popular line of recent research focuses on keypose selection [14], where the idea is to select the most discriminative poses from the full action video for action recognition. These have shown good results on popular datasets such as the KTH [19] and IXMAS [21]. Global representations are typically more sensitive to noise, however, requiring a tight bounding box around the region of interest.

With the recent improvement of person detectors, such as poselets, first published by Bourdev and Malik [2], and person trackers, such as the recent GMCP-tracker by Zamir et al. [18], it is now possible to get relatively stable human tracks from many action datasets. Even for more realistic datasets for which person tracking might not work, such as UCF YouTube [12], bounding box annotations have been published. Because of this, it is now also possible to apply global representation methods even to videos that have significant ego motion, scale and translation variations, and where background subtraction/silhouette extraction is not possible. Recent action recognition works that rely on bounding boxes of persons include Zhen et al. [24] who use embedded motion and structure features within the bounding box and Ji et al. [8], which describes 3D convolutional neural networks for feature extraction from the person tracks.

Compared to the mature field of person tracking, however, the temporal analysis and segmentation of human tracks has seen far less work. Some works, such as Turaga et al. [20] and Zhou et al. [25], have performed unsupervised human action categorisation from temporally unconstrained videos – temporal segmentation forms a necessary part of the clustering process – but these works have not been applied to action recognition, and they are very computationally expensive. Niebles et al. [17] uses a supervised model to extract multiscale temporal segments from human actions and applies this to action recognition. Compared to these methods, our temporal segmentation method is very efficient. It is untrained, based on cost-effective algorithms such as optical flow, and can be calculated faster than real-time on a modern desktop.

3. Methodology

Our method consists of several parts. Firstly, a poselet-based tracking method is applied to spatio-temporally localise human tracks within the video. Each human track is then split into sub-tracks at points of discontinuity in linear motion, and each sub-track is represented using a descriptor based on a modified Motion History Image. Finally each action is classified using these descriptors with NBNN. We describe each of these processes in detail below. A system diagram is shown in Figure 1.

3.1. Person Tracking

For our action representation method to be effective, we first require an accurate person tracker to isolate the person from the rest of the scene, and to spatially align the person in each frame, thus mitigating background motion and scale variations. We implement a simple tracking method based on poselet detection. Poselets are a relatively recent innovation that have so far been applied largely to person detection in images, and have proven quite reliable in a variety of conditions. Here, we apply them in the context of person tracking.

The localisation method is simple. We first pre-process the grayscale video with a 2D Gaussian filter to reduce noise at every frame. Then, we apply a horizontal and vertical Prewitt filter, summing the filtered images together to get an edge intensity image. We have found experimentally that this pre-processing im-

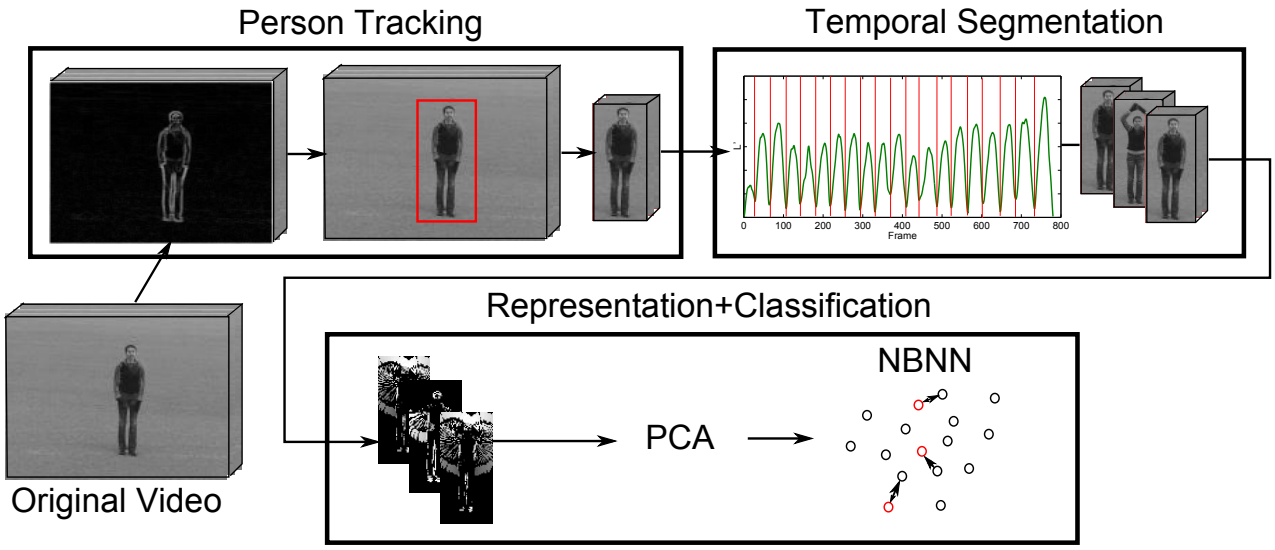

Figure 1. A system diagram capturing the main stages of our action recognition system.

proves the reliability of poselet detection, particularly in low-contrast videos. We then perform poselet detection at every frame, using the code provided online [2].

After poselet detection, we take several post-processing steps to establish a smooth human track. All detected poselets are thresholded by their score to remove inaccurate detections – the threshold τ is static and set empirically. As multiple actors may be present in a scene, we associate different bounding boxes in separate frames into proto-tracks using the KLT point-tracks method described in Everingham et al. [5]. Certain proto-tracks may have missed detections in some frames. We manage this using interpolation. If there is a detection gap of f frames or less between 2 detections, we linearly interpolate the x, y, width, and height values of the bounding boxes separately to estimate the person's position in the missed frames. If there is a gap longer than f frames, we assume that the person is out-of-shot or otherwise occluded, and do not interpolate. We discard any very short tracks of 5 frames or less, as short tracks are more likely to be noisy or contain no useful motion information. Finally, the width and height of the track are increased by 20% in every frame, to ensure that the whole person is captured – this is to compensate for the poselet detector potentially missing an out-stretched arm or the feet of the actor. While our method is simple, we find it is effective for the datasets in our experiments. For more realistic datasets, a different solution might be considered.

3.2. Motion Segmentation

We now wish to break the human track up into temporal segments, each roughly corresponding to a single linear motion performed by the actor. In an action recognition setting, we predict that short temporal segments will be more rate and view invariant than complete actions, and therefore, combined with an appropriate classifier (such as AdaBoost or NBNN) they may give higher accuracy. We also predict that linear motion segmentation will be particularly effective in classifying repetitive or cyclical actions – the action will be split into temporal segments at the same points in the cycle, ensuring temporal alignment between each segment.

In order to find discontinuities in linear motion we consider the most significant motion gradient of the human track. Given a human track h of static height and width, we first extract a series of difference images d: $d_i = h_{i+1} - h_i$. We extract the optical flow from d in the x and y directions between every pair of adjacent frames, to get $optfx$ and $optfy$. To calculate $optfx$ and $optfy$ we use the Lucas-Kanade algorithm [16], as it provided the best efficiency and accuracy trade-off in preliminary tests. We concatenate the pixels of each frame of the track into a time-series of 1D vectors, and perform Principal Component Analysis (PCA) on these vectors, discarding all but the first 2 principal components, so we get $P = \{p_{1,i}p_{2,i}; i = 1, ..., t\}$, where t is the number of frames in the optical flow. We predict that P corresponds to the most significant motions in the video.

The next step is to look for linear discontinuities in time-series P. The R^2 statistic is a measure of how well

a regression model matches the observed data, so it is possible to get a measure of linearity at every point P_m in P by calculating the R^2 statistic of a simple linear regression model on a temporal window of P, centred on P_m. In this model, $P_i, i = m-w, ..., m+w$ are the data points of the regressors p_1 and p_2, and $y = m-w, ..., m+w$ is the dependent variable, where w is a parameter defining the size of the temporal window. We set:

$$\mathbf{X} = \begin{pmatrix} 1 & p_{1,m-w} & p_{2,m-w} \\ 1 & p_{1,m-w+1} & p_{2,m-w+1} \\ \vdots & \vdots & \vdots \\ 1 & p_{1,m+w} & p_{2,m+w} \end{pmatrix}, \quad \mathbf{y} = \begin{pmatrix} m-w \\ m-w+1 \\ \vdots \\ m+w \end{pmatrix} \tag{1}$$

We then apply orthogonal decomposition to X in the normal fashion to get the regression co-efficients β and the predicted values f_i:

$$\mathbf{QR} = \mathbf{X}, \quad \beta = \mathbf{R}^{-1}(\mathbf{Q}^T\mathbf{y}), \quad \begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ f_n \end{pmatrix} = \mathbf{X}\beta \tag{2}$$

From our observed values y_i and predicted values f_i we can calculate the R^2 statistic at each time m to give a measure of how well the linear model matches the data at every data point. A higher value for R^2 corresponds to greater local linearity:

$$R^2 = 1 - \frac{\sum_i (y_i - f_i)^2}{\sum_i (y_i - \overline{y})^2} \tag{3}$$

where $\overline{y}$ is the mean of the observed data. We then get a degree-of-linearity time series $L = L_{m,w}; m = 1, ..., t$ where $L_{m,w}$ is the R^2 statistic for a linear regression around every time point m, with window size w. However, L is not sufficient for accurate temporal segmentation, as in practice the R^2 statistic is highly dependent on w. If w is too small, then L will be noisy – a single outlier could have a strong local effect. If w is too large, however, rapid changes in linear motion will be averaged out and fast actions will not be segmented correctly. One potential solution is to calculate L for every point m for the range of values of w, which we term W, and get an average for M over all W:

$$L' = \left\{ \sum^{w \in W} sL_{m,w};\ m = 1, ..., t, s \propto w \right\} \tag{4}$$

We use weights proportional to w, rather than a simple mean. This is to offset the following: as w grows larger, the average value of M tends to decrease. If we use a simple mean, therefore, larger values of w will be under-represented in L'. In our experiments we set $W = [2, 20]$. When the temporal window overlaps the start or the end of the track, we use a symmetric, mirror-reflected border to get the missing values.

L' can be used directly to perform motion segmentation. The local minima in L' correspond to the break points in linear motion. Local maxima in L', alternatively, correspond roughly to the central frames of the linear motions. The effectiveness of this method can be seen in Figure 2, which shows the graph of L' as applied to two simple cyclical actions from the KTH dataset, handwaving and boxing. The boxing example consists of many fast, short, linear actions, with abrupt changes, whereas the handwaving example has slower, longer actions. These patterns are reflected clearly in the sinusoidal patterns of the L' graphs.

We choose to perform segmentation in two ways to get overlapping temporal segments. We segment between the local minima of L'. In this manner, each segment contains a single linear motion. We additionally segment between the local maxima of L', so each segment contains a transition from one linear motion to another. We denote these two groups of segments S_l and S_t respectively. We combine S_l and S_t in our experiments, as both together perform better than either does alone – together, they capture more of the structure of the action.

3.3. Representation/Recognition

We now describe the representation of the spatio-temporally localised motion segments for classification. We use a variation of the Motion History Image (MHI) for this purpose. The MHI is particularly compatible with our method for two reasons. First, the accurate spatial localisation given by the poselet person tracker removes all scale and translation variations from a person's motion that could potentially distort an MHI. Second, temporal segmentation mitigates spatially overlapping motions. In a full action – particularly cyclical or repetitive actions – motions may spatially overlap, but MHIs can only record the most recent motion in a pixel. If the MHI captures the full action, therefore, it will lose potentially salient information. However, as each temporal segment in our model only has a short linear motion, motion overlap within a segment is minimised, making the MHI a good choice for representation.

Our variation of the MHI is simple. Given a temporal segment v, each frame is scaled to 100x100, so

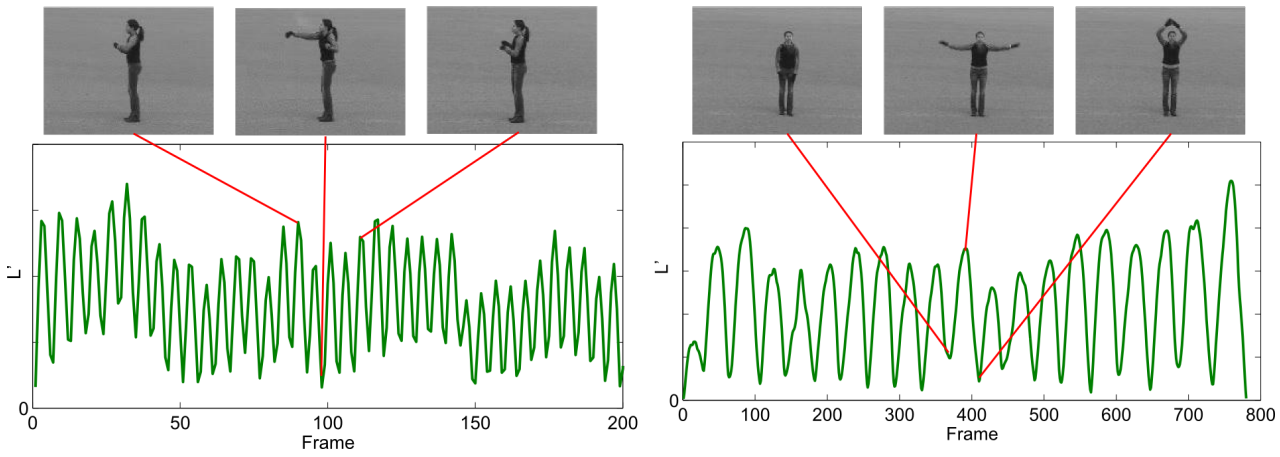

Figure 2. L' applied to two KTH dataset videos, one for boxing and one for handwaving. The peaks of the graph occur in the middle of a linear motion. The valleys occur as the actors' arms change direction.

every MHI will be the same size. We first get a difference video: $v'_i = |v_{i+1} - vi|$. All of the pixels of v' are thresholded to get a binary video v'' – 1 is set for pixels above the threshold, 0 is set for pixels below the threshold. The threshold is set dynamically for each v' to one standard deviation above the mean pixel value. From this binary video, the MHI is calculated as per [4]. Each MHI is reduced by PCA to an N-dimensional vector, and these vectors are then used as features for classification.

At this stage, there are several possible models to for classification, including Bag of Words for efficiency, or Hidden Markov Models to capture the temporal structure between our MHI features. In our experiments, we have opted for NBNN classification due to its superior performance compared to the traditional Bag of Words [1], and its relative simplicity. Further work could investigate the use of temporal representation models in conjunction with L' segmentation.

4. Experiments

In this section we detail the setup and results of our experiments. The goal of our experiments is to validate that our novel temporal segmentation algorithm can split an action in semantically meaningful temporal segments. We show this by demonstrating that our temporal segments can be used to perform accurate recognition – however, we are not concerned with the more complex datasets such as the HMDB [11] or UCF YouTube [12] – these contain unusual body poses unlikely to be detected by our human tracker, and may be too noisy for the temporal segmentation to work well. We propose that our temporal segmentation will find the most practical use in the unsupervised analysis of humans in surveillance videos. With this in mind, we apply our system to the KTH [19] and MSR2 [23] datasets, which, like in surveillance videos, include a static viewpoint and upright body poses.

4.1. Setup

KTH Dataset: This dataset [19] has been perhaps the most popular human action dataset for several years, and as such sees very high accuracies. The actions are single subject, and from a single view point. It includes scale, clothing, and setting variations. To test on this dataset, we use the popular leave-one-person-out cross validation method: our model is trained on all but one actor's actions, and then the model is tested on the left-out actor's actions. This process is repeated for every model in the dataset, and the results averaged over every run.

MSR2: This dataset [23] consists of 3 action classes (boxing, handwaving, handclapping) recorded around a college campus by different actors. There is considerable background noise, and movement during each of the actions. There are 203 actions in total, recorded across 54 different scenes. Here we apply leave-one-scene-out cross validation.

For each experiment, we first convert each video in our datasets to grayscale, then perform person tracking as described above. The poselet code is taken from [4], and is run using the default parameters. We perform temporal segmentation using a variety of methods for comparison as described in Section 4.2. We then represent each temporal segment using our enhanced MHI and perform classification using NBNN. Our experimental machine has two Intel Xeon E5630s, 24GB of memory, and runs 64-bit Ubuntu. All of our code is written in MATLAB for simplicity.

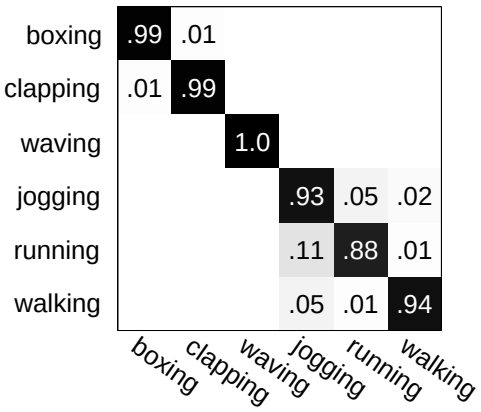

Figure 3. KTH dataset: action class confusion matrix

Method	Accuracy (%)
L' Seg.	**95.5**
No Seg.	89.7
Uni Seg. ($x = 10$)	92.2
Dollar et al. [4]	81.2
Zhen et al. [24]	93.3
Liu and Shah. [13]	94.2

Table 1. KTH dataset: comparison of various methods

4.2. Results

We show our results for various methods of temporal segmentation for the KTH dataset in Table 1. We considered L' based segmentation, uniform segmentation (each track is divided into sections of x frames, where x is chosen to optimise results) and no segmentation. It is clear from this that L' based segmentation gives a better result than any other considered temporal segmentation methods. We also compare our method with several other recent works, showing that our method rivals the state-of-the-art. Figure 2 shows a confusion matrix of the actions - we can see here that the most confused actions are *jogging* and *running*, and to a lesser extent, *walking*. This is expected, as these actions are very similar and difficult to distinguish between.

In Table 2 we show our results for the MSR2 dataset, for L' based segmentation, uniform segmentation (for varying x) and no segmentation. We can see the results here are similar. While it is a more challenging dataset, our temporal segmentation algorithm gives a clear advantage over no temporal segmentation or uniform temporal segmentation. The confusion matrix is shown in Figure 4.

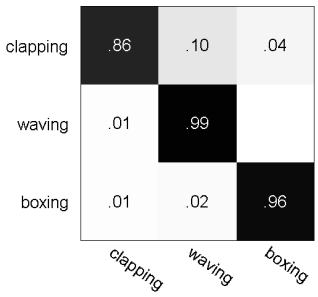

Figure 4. MSR2 dataset: action class confusion matrix

Method	Accuracy (%)
L' Seg.	**94.6**
No Seg.	75.4
Uni Seg. ($x = 3$)	92.1
Uni Seg. ($x = 6$)	89.7
Uni Seg. ($x = 9$)	87.2
Uni Seg. ($x = 12$)	86.2

Table 2. MSR2 dataset: comparison of various methods

5. Discussion

In this paper we have introduced a temporal segmentation method. We have detailed the creation of the L' metric, and how to use this L' metric to break an action into sections of discrete linear motion. Through our experimentation on human action recognition we have demonstrated that our L' metric can be used to split an action up at consistent points, and that the motion segments can be used in combination with an MHI+NBNN classifier to achieve superior action recognition performance. The efficient and accurate temporal segmentation offered by L' can now be applied in fields other than action recognition, such as to improve action clustering or localisation tasks. Our work could be particularly useful in temporally unconstrained action clustering, where it is often necessary to first temporally segment the action in an unsupervised fashion. Splitting human tracks into linear motions may also reduce the complexity of localisation methods, as detections can be made at the motion-level rather than the frame-level.

One significant area for improvement in future work would be to improve our system's applicability to noisier multimedia datasets. Although we envisage that our system, in its current form, can already be applied to real world surveillance videos, the reliance on accurate person tracking makes our system less suitable for multimedia videos such as those found on YouTube. However, as person trackers/spatial segmentation methods grow more robust, it may be possible to

extend or modify our method to work on noisier data.

References

[1] O. Boiman, E. Shechtman, and M. Irani. In Defense of Nearest-Neighbor Based Image Classification. In *Proc. IEEE Conf. Comput. Vision and Pattern Recognition*, pages 1–8, 2008. 5

[2] L. Bourdev and J. Malik. Poselets: Body part detectors trained using 3d human pose annotations. In *IEEE Int. Conf. Comput. Vision*, 2009. 2, 3

[3] J. W. Davis and A. F. Bobick. The Representation and Recognition of Human Movement Using Temporal Templates. In *Proc. IEEE Conf. Comput. Vision and Pattern Recognition*, page 928, 1997. 2

[4] P. Dollar, V. Rabaud, G. Cottrell, and S. Belongie. Behavior Recognition via Sparse Spatio-Temporal Features. *Proc. IEEE Workshop Visual Surveillance and Performance Evaluation Tracking and Surveillance*, pages 65–72, 2005. 2, 5, 6

[5] M. Everingham, J. Sivic, and A. Zisserman. "hello! my name is... buffy" – automatic naming of characters in tv video. In *Proc. British Mach. Vision Conf.*, 2006. 3

[6] M. Grundmann and F. Meier. 3d shape context and distance transform for action recognition. In *Proc. Int. Conf. Pattern Recognition*, pages 1–4, Dec. 2008. 2

[7] J. Henriques, R. Caseiro, P. Martins, and J. Batista. Exploiting the circulant structure of tracking-by-detection with kernels. In *Proc. European Conf. Comput. Vision*, volume 7575, pages 702–715, 2012. 1

[8] S. Ji, W. Xu, M. Yang, and K. Yu. 3d convolutional neural networks for human action recognition. *IEEE Trans. Pattern Anal. Mach. Intell.*, 35(1):221–231, 2013. 2

[9] A. Kläser, M. Marszałek, and C. Schmid. A Spatio-Temporal Descriptor Based on 3D-Gradients. In *Proc. British Mach. Vision Conf.*, pages 995–1004, 2008. 2

[10] A. Kläser, M. Marszalek, C. Schmid, and A. Zisserman. Human Focused Action Localization in Video. In *International Workshop on Sign, Gesture, Activity*, 2010. 1

[11] H. Kuehne and H. Poggio. HMDB: A Large Video Database for Human Motion Recognition. In *IEEE Int. Conf. Comput. Vision*, 2011. 5

[12] J. Liu, J. Luo, and M. Shah. Recognizing Realistic Actions from Videos "in the Wild". In *Proc. IEEE Conf. Comput. Vision and Pattern Recognition*, pages 1996–2003, June 2009. 2, 5

[13] J. Liu and M. Shah. Learning human actions via information maximization. In *Proc. IEEE Conf. Comput. Vision and Pattern Recognition*, pages 1–8, 2008. 6

[14] L. Liu, L. Shao, X. Zhen, and X. Li. Learning discriminative key poses for action recognition. *IEEE Transactions on Cybernetics*, 43(6):1860–1870, Dec 2013. 2

[15] A. López-Méndez, J. Gall, J. Casas, and L. van Gool. Metric learning from poses for temporal clustering of human motion. In *Proc. British Mach. Vision Conf.*, pages 49.1–49.12, 2012. 1

[16] B. D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proc. 7th Int. Joint Conf. Artificial intelligence*, pages 674–679, 1981. 3

[17] J. C. Niebles, C. wei Chen, and L. Fei-fei. Modeling temporal structure of decomposable motion segments for activity classification. In *Proc. European Conf. Comput. Vision*, pages 392–405, 2010. 2

[18] A. Roshan Zamir, A. Dehghan, and M. Shah. Gmcp-tracker: Global multi-object tracking using generalized minimum clique graphs. In *Proc. European Conf. Comput. Vision*, pages 343–356, 2012. 1, 2

[19] C. Schuldt, I. Laptev, and B. Caputo. Recognizing Human Actions: A Local SVM Approach. In *Proc. Int. Conf. Pattern Recognition*, volume 3, pages 32–36, 2004. 2, 5

[20] P. Turaga, A. Veeraraghavan, and R. Chellappa. Unsupervised view and rate invariant clustering of video sequences. *Comput. Vision and Image Understanding*, 113(3):353 – 371, 2009. 1, 2

[21] D. Weinland, R. Ronfard, and E. Boyer. Free Viewpoint Action Recognition Using Motion History Volumes. *Comput. Vision and Image Understanding*, 104(2):249–257, 2006. 2

[22] D. Wu and L. Shao. Silhouette analysis-based action recognition via exploiting human poses. *IEEE Trans. Circuits and Syst. Video Technology*, 23(2):236–243, 2013. 2

[23] J. Yuan, Z. Liu, and Y. Wu. Discriminative video pattern search for efficient action detection. *IEEE Trans. Pattern Anal. Mach. Intell.*, 33(9):1728–1743, 2011. 5

[24] X. Zhen, L. Shao, D. Tao, and X. Li. Embedding motion and structure features for human action recognition. *IEEE Trans. Circuits and Syst. Video Technology*, 23(7):1182–1190. 2, 6

[25] F. Zhou, F. De la Torre, and J. Hodgins. Hierarchical aligned cluster analysis for temporal clustering of human motion. *IEEE Trans. Pattern Anal. Mach. Intell.*, 35(3):582–596, 2013. 1, 2