8th IEEE International Conference on Advanced Video and Signal-Based Surveillance, 2011

Unsupervised Learning of Micro-Action Exemplars using a Product Manifold

Stephen O'Hara and Bruce A. Draper
Colorado State University
Fort Collins, CO 80523
{svohara,draper}@cs.colostate.edu

Abstract

This paper presents a completely unsupervised mechanism for learning micro-actions in continuous video streams. Unlike other works, our method requires no prior knowledge of an expected number of labels (classes), requires no silhouette extraction, is tolerant to minor tracking errors and jitter, and can operate at near real time speed. We show how to construct a set of training "tracklets," how to cluster them using a recently introduced Product Manifold distance measure, and how to perform detection using exemplars learned from the clusters. Further, we show that the system is amenable to incremental learning as anomalous activities are detected in the video stream. We demonstrate performance using the publicly-available ETHZ Livingroom data set.

1. Introduction

Although there is a great deal of research relating to the recognition of human behaviors and actions in video, much of the research to date has focused on the problem of classifying short video segments according to a small, fixed set of labels. Action recognition is hard, and it is reasonable to attempt to simplify the problem using controlled data sets. However, in deference to the no-free-lunch theorem [10], the techniques used to push performance to the highest levels on classification benchmarks may not yield substantial gains in addressing the more general challenges relating to action recognition in less controlled, streaming data sources. In a recent report of the Semantic Description of Human Actions (SDHA) challenge, it was found that a number of methods performed well on action classification, yet none of the submitted techniques could perform sufficiently well at detecting actions in continuous video [9]. We believe that to address the real-world activity recognition needs of applications in surveillance, robotics, video search, or assistive technologies, a different paradigm is needed that stresses unsupervised and incremental learning from continuous video streams.

In this paper, we present a method of unsupervised learning of human micro-actions from long duration videos, based on computing distances between short "tracklets" using a product manifold mapping. There is some variation in the use of the term *micro-action* in the literature. Here, we use it to mean a short-duration, single-entity action that can be recognized, at least by a human observer, with only a few seconds of video. Our method is efficient, can be trained relatively quickly, and can perform detections in near-real-time. We require that the entities of interest be detected and tracked over enough frames to observe any given micro-action, yet our tracklet extraction strategy mitigates minor tracking accuracy issues that are commonly encountered. We do not require any silhouette extraction or part detections of the subjects. We make no assumption on the number of micro-actions an entity may exhibit in any given length of time, and we allow for multiple labels to be applied simultaneously.

Before presenting the details, we first highlight some relevant work and then provide background material on the Product Manifold distance measure used in our method.

1.1. Related Work

A recent survey by Poppe [7] provides a good overview of the Action Recognition body of literature. We will avoid trying to summarize the field, and instead highlight a few recent papers and others we find particularly relevant.

Bag-of-feature methods may employ a variety of space-time interest point detectors, descriptors, and machine learning algorithms to push the state of the art in supervised action classification [1, 8, 11]. Bag-of-features methods often suffer because they require a significant number of design choices and parameter optimizations to work well – which is a time-consuming and often data set-dependent problem. Niebles *et al*. present an unsupervised bag-of-features based model for action recognition, but their system uses expectation maximization over a known number of classes, and furthermore uses a validation stage, requiring labels, to select an optimal feature codebook [5].

Departing from the bag-of-features works are silhouette

978-1-4577-0845-9/11/$26.00

motion methods such as those from Lin *et al*. and Nater *et al*. [2, 4]. Lin employs joint likelihood maximization between the current observation and learned shape-motion prototypes. Lin's method requires supervision and is evaluated for classifying short video clips containing only a single action. Nater is one of the few papers that presents an entirely unsupervised method for learning human behaviors. At a high level, Nater clusters silhouettes and motion patterns and recognizes anomalous activities via outlier thresholding. Nater's approach represents a competing method for achieving many of the same goals we present in this investigation. Our approach is based on manifold geometry, and we demonstrate the use of exemplars for activity detection, while Nater focuses more on anomaly detection.

1.2. Product Manifold Distance

To measure the similarity between short video segments we call tracklets, we employ a product manifold-based distance measure developed by Lui *et al*. [3] (called PM distance henceforth). For the sake of writing a self-contained paper, we provide an overview of the method in this section.

A video can be represented as a stack of sequential images forming a data cube of dimension (x, y, t), where x and y are the width and height of the images and t is the number of frames. This data cube is a 3-mode tensor, and can be factored using the high-order SVD (HOSVD) into a core tensor and three factor matrices. In computing the HOSVD, the tensor, A, is flattened from 3D to 2D along each of the axes, creating three matrices A_1,A_2, and A_3 associated with the following unfoldings: $(x, yt), (y, tx), (t, xy)$. Each A_k is a matrix and can be factored using SVD to generate an orthonormal space associated with each unfolding. Equation 1 shows the relationship between the HOSVD and the SVD of each unfolding.

$$\begin{aligned} A_k &= U^k \Sigma^k {V^k}^T \\ A &= S \times_1 U^1 \times_2 U^2 \times_3 U^3 \end{aligned} \tag{1}$$

Lui modifies the HOSVD to use the right singular vectors V^k instead of the corresponding U^k. Each factor V^k is an orthonormal matrix and can be represented as a point on a corresponding Grassmann manifold. Thus, the data cube of the video becomes three points, one on each of three separate Grassmann manifolds. There exists a product manifold which is the product of the three Grassmann manifolds. Each video is a point in the product manifold structure. An important property is that the geodesic distance between points on a product of Grassmann manifolds is the product of the geodesic distances on each factor manifold.

To compute the PM distance, each video is mapped to three points, one point for each of the Grassmann manifolds associated with the orthogonal decomposition of the three unrollings of the data cube. The distance between the pairs of points on each manifold is calculated using canonical angles, for which there is a closed-form solution. Note that the canonical angle is a representation of distances between vector subspaces, and is generally not a scalar value. The cartesian product of the three canonical angles, from each factor manifold, represents the distance on the product manifold. To generate a scalar number, the chordal distance is computed from the elements of the product.

2. Method

We propose an unsupervised learning method for micro-action recognition based on clustering short duration video clips, called tracklets, that are extracted from entity tracks in training videos. Each tracklet captures the appearance and motion of an entity for a second or two of time. We cluster the tracklets using the Product Manifold distance. In grouping similar tracklets, we discover the repeated micro-actions performed by people (or other entities) in the video. We perform clustering with no foreknowledge of either the expected types or numbers of micro-actions present in the data. The idea is to *discover* the micro-actions, not to force-choice classify the activities into pre-ordained buckets.

From each cluster, we identify a small number of exemplar tracklets that best represent the group. The set of clusters may be given labels by the users of the system, a process we call "Selective Guidance." It is important to note that the system would work just as well with internally generated identifiers. Not all clusters are easily described with a concise label. For those that are easily described, we can apply that label to the cluster's exemplar(s).

The set of exemplars is used in a nearest-neighbor matching strategy to detect and label micro-actions on previously unseen test video. We perform detection on streaming video without any requirement for pre-segmentation of the space-time regions of interest. As an entity being tracked changes behavior, the system will detect the change and apply a new label where appropriate.

At times, a tracklet from the test video may not be a good match to any of the exemplars. In such instances, the system will apply no label to the tracklet, and it will be remembered as a novel detection. The set of novel detections can be evaluated to produce additional exemplars, and thus the system can learn over time, boot-strapped from an initial training set. Further details on the various aspects of our approach are presented below.

2.1. Data

We use the publicly-available ETHZ Living Room data set for our evaluation [4]. We selected this data set because it represents the continuous surveillance problem better than many of the more popular action recognition benchmarks. Many action recognition data sets are designed to

Figure 1. Example of an activity detection from ETHZ Seq1.

support forced-choice classification of pre-segmented video clips. The ETHZ Living Room data, however, provides three video sequences. The first, over 7,000 frames long, is a continuous recording of a person moving about a room and performing a few selected behaviors (walking, sitting, bending down). The first video (Seq1) is intended to allow an unsupervised system to learn the nominal behavior of the room's occupant. The second two videos (Seq2 and Seq3) are shorter, and are used to present novel behaviors, such as falling down or panicked gesticulations, to measure a system's ability to detect anomalous events. Figure 1 shows a sample image from the first video of the data set. For brevity, in the remainder of this paper we refer to this data set as ETHZ.

2.2. Tracks and Tracklets

To generate the tracks on ETHZ video sequences, we perform background subtraction using the median image of the first 2,000 frames as the background model. We use the bounding box of the foreground mask to track the subject in the video. Processing is performed using grayscale imagery.

Action recognition approaches that rely on silhouette extraction [2, 4] can be negatively impacted when the foreground mask is inaccurate. An important advantage to our method is that it processes all pixels within the bounding box, requiring no silhouette mask, and is therefore less sensitive to foreground/background segmentation challenges.

We define a *tracklet* to be a short contiguous section of a track that has been reshaped into a fixed-size data cube of dimension: (x, y, t), where the unit of time, t, is the frame number. The tracklet duration is chosen to be appropriate for capturing the motion of micro-actions, and thus is typically less than a few seconds long. A single track of a person over time will give rise to numerous tracklets, some of which may clearly contain a micro-action, and others may represent transitions between micro-actions and thus have no clear semantic label (see Figure 4 from our results for an example). The size of each frame in the tracklet is kept small in order to capture only large-scale structure, eliminate high-frequency features, and de-emphasize individual appearance. In this investigation, we create tracklets of size (32x32x48).

We employ a sliding window strategy for slicing tracks into tracklets. The bounding box of a track typically varies from frame-to-frame, and so the resulting tracklet can suffer from significant instability that negatively impacts the PM distance computation. To stabilize the tracks, we compute the bounding box that contains the spatial extent of the *entire* tracklet, and we use that box to clip tracklet tiles from corresponding frames in the video. The benefit of this simple stabilization strategy is illustrated in figure 2.

Figure 2. Example tracklet created without (top) and with (bottom) stabilization strategy. Top tracklet uses the bounding rectangles from the track to clip tiles from the source. Bottom uses the full spatial extent of the track within the temporal window to define a single clipping region, and thus stabilizes the images and corrects for minor track drift. In both cases, the clipped tiles are rescaled to fit the fixed tracklet dimensions.

2.3. Clustering and Exemplar Selection

Given a set of tracklets extracted from the training video, we compute the pair-wise PM distances to form a distance matrix. Agglomerative hierarchical clustering, using Ward's linkage, is used to generate a cluster tree. The tree can be cut at a particular linkage threshold value to generate a set of clusters. With no prior knowledge of the expected number of clusters (K), it can be challenging to select the appropriate cut. It is an open question on how best to guage the clustering quality lacking any prior information. However, we have observed that the performance of our method rises quickly as K is increased, and then plateaus at a high level for K greater than approximately ten percent of the training sample size.

To convince ourselves that this is true, we measured the Cluster Accuracy (defined below) against the choice of K, illustrated in Figure 3. Generating this plot requires labeling the training data, but this is not an integral part of our method. Instead, for the experiments described later, we blindly chose values of K equal to 5, 10, 15, and 20 percent of the training set size.

We define Cluster Accuracy following the definition presented by O'Hara *et al*. [6]. Each cluster is assigned the la-

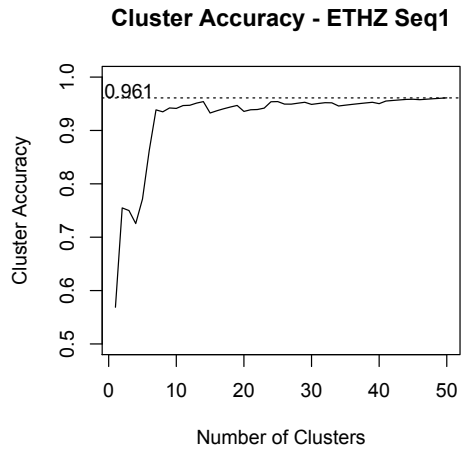

Figure 3. Cluster Accuracy on ETHZ Seq1 tracklets. The sharp rise followed by a long plateau indicates many K values work well.

bel possessed by the majority of its members. Those cluster members not agreeing with the majority label are considered errors. The accuracy is the sum of the errors from all clusters divided by the total number of samples. Equation 2 is the formal definition, where C is the set of k clusters and X_L^k is the set of samples in cluster k with label L.

$$\begin{gathered} C = \{C^1, C^2, \ldots, C^k\} \\ X_L^k = \{x_i | x_i \in C^k \wedge Label(x_i) = L\} \\ \sum_{i=1}^{k} \max_{L \in Labels} |X_L^k| / |C^k| \end{gathered} \tag{2}$$

The PM distance measure is decidedly non-Euclidean, so there is no mean value ("center") to the samples in a cluster. Instead, exemplars (medoids) can be selected from within each cluster that minimize the sum of the distances to the other cluster members. More than one exemplar can be selected from within a cluster by removing the best medoid and repeating the process. Interestingly, we found that pulling two exemplars from clusters that represent "sitting" or "bending" resulted in one of the samples exhibiting the downward aspect of the motion and the other exmplar exhibiting the upward aspect (bending down vs. bending back up, e.g.)

2.4. Detection

After exemplars have been trained, we use them to match against tracklets in the test videos. This process occurs in near-real-time on the streaming video. We compute the K-Nearest-Neighbors using the PM distance between each new tracklet and the exemplars. For the experiments reported herein, unless otherwise mentioned, we use 3 neighbors. Soft weighting is used so that selected exemplars contribute their labels to the new tracklet based on how close they are. A standard gaussian decay is used with σ determined from the distribution of distances in the training samples.

We allow for multiple labels. Each tracklet maintains a bit vector of length equal to the cardinality of the label set. The weighted label vectors from the nearest exemplars are summed component-wise to produce the raw label vector of the new tracklet. A score threshold is applied to each component to generate the label bit vector. It is possible– even desirable–that the label vector will result in all zeros should none of the nearest exemplars be close enough to the sample.

Formally, the scoring computation is shown in Equation 3, where ω_i is the weight based on the PM distance $d(i, x)$ between exemplar i and tracklet x, L_i is the label vector for exemplar i, P is the number of labels, s^p is the component score computed as the weighted sum of the corresponding components from the k nearest exemplars, and L_x is the computed label for tracklet x by comparing the component scores to a constant threshold t.

$$\begin{aligned} \omega_i &= e^{-d(i,x)^2/2\sigma^2} \\ L_i &= (l_i^1, l_i^2, \ldots, l_i^P) \\ s^p &= \sum_{i=1}^{k} \omega_i l_i^p \;, \forall p \in \{1 \ldots P\} \\ L_x &= (s^1 \geq t, s^2 \geq t, \ldots, s^p \geq t) \end{aligned} \tag{3}$$

2.5. Anomalies and Incremental Learning

An anomaly is a tracklet that is too far from the exemplars to produce a non-zero label set. After the initial exemplars have been produced from the training data, we can run the system with a relatively high score threshold in order to generate a set of anomalous samples. We combine the anomalous tracklets with the current exemplar set, and then recompute the clustering over only the combined set (i.e. omitting all of the original training tracklets), yet keeping the K-value the same. In the resulting clusters, we look for any of the anomalous samples that are not grouped in the same clusters with current exemplars. This subset of the anomalous samples is selected to be added to the updated exemplar set, and selective guidance is used to generate labels where appropriate, or to assign the new exemplar to an existing label if it represents a novel aspect of a known micro-action.

3. Results

3.1. Experiment 1

The first experiment demonstrates the variation in performance when selecting different values for K, the number

Figure 4. Example of a tracklet labeled from multiple exemplars. This tracklet captures the transition between multiple states, and is thus correctly described by the unordered label set {walk,site,recline}. Sample frames on right are from the 48 frame, 32x32 pixel tracklet.

of clusters, and various score threshold values, as described previously. We chose four values of K to use in our initial clustering, where we blindly selected a number of clusters equal to 5%, 10%, 15%, and 20% of the number of training tracklets. Having extracted 283 tracklets from a sampling of video Seq1 for training, the values for K were 14, 28, 42, and 56, respectively. We selected one exemplar per cluster.

Figure 5 shows the results. The accuracy is measured in terms of the average $F1$ score between the predicted and ground truth label bit vectors. It is not surprising that having more exemplars leads to better overall performance, yet the performance drop when decreasing from 56 to 42 exemplars is not severe. When using the best score threshold of 0.8, the performance drops by 3% from 56 to 42, and 8% from 56 to 28. This adds support to our belief that performance is not sensistive to the choice of K, as long as K is beyond the steep rising curve, as described earler (see Figure 3).

Figure 4 shows an example of a single tracklet that was given multiple labels. The detection was on a tracklet from Seq2 where the tracklet duration happened to contain the transition between three micro-actions. The advantage of allowing multiple labels is that such interstitial observations may be described as a set of appropriate labels. There are no exemplars that were learned that had more than two labels. This result required the contribution from two or more exemplars that, while different from each other, all had a similarity to the novel tracklet, as measured by the PM distance.

3.2. Experiment 2

The second experiment was performed to guage how well the system can incrementally learn based on anomalous detections. We selected an exemplar set trained from Seq1, and used it to detect anomalous micro-actions from Seq2, which contains never-seen behaviors including falling down, jumping, reclining on the couch, and panicking. Figure 6 shows a set of fourteen new exemplars identified from Seq2 using the procedure described in the Methods section.

After folding in the new exemplars with the original set, we performed detection on Seq3. We repeated this procedure, but reversed the roles of Seq2 and Seq3. The results are shown in Figure 7. There is nearly a 10% performance improvement after incorporating the new exemplars.

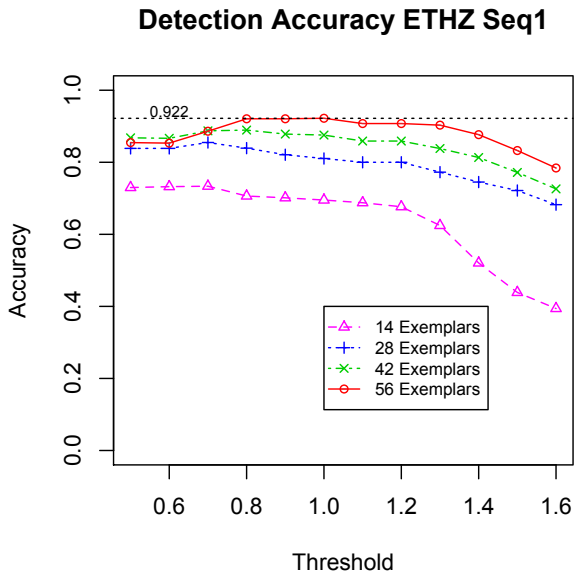

Figure 5. Detection accuracy using different exemplar sets on Seq1. Accuracy is the average F1 score between predicted and ground truth label bit vectors. Threshold is the minimum sum of the weighted bits from the 3 nearest exemplars required to activate the corresponding label bit on the tracklet.

4. Conclusions

We believe that to make progress on the fundamental challenge of human behavior recognition in continuous video, researchers must move away from pre-segmented video clip classification and towards more open-world, incremental learning methods that require a minimum of supervision. We presented a step in this direction by showing how a recently proposed Product Manifold method for measuring similarity between video tensors can be applied to unsupervised, incremental learning of micro-actions.

In addition to those described earlier, our approach has the additional advantage of not requiring a large number of parameters and design choices. This is a clear improvement over many main-stream bag-of-features approaches that require parameter selection and design optimizations for fea-

Figure 6. Fourteen new exemplars were learned from the ETHZ Seq2 video, representing the novel micro-actions of jumping, reclining, falling, and panicking. Images are representative frames from the 48 frame, 32x32 pixel tracklets.

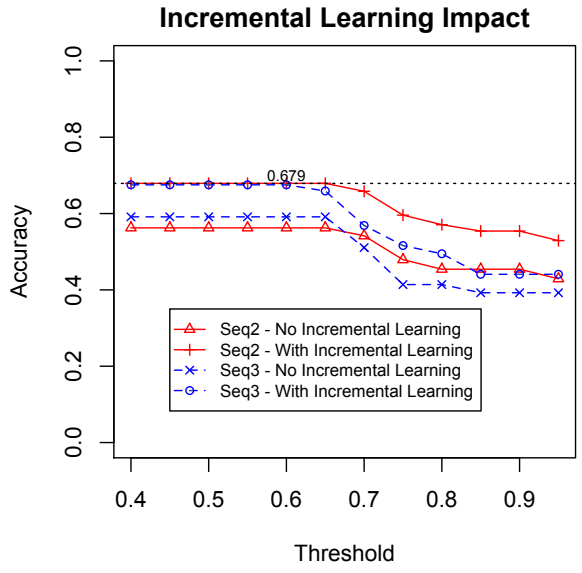

Figure 7. Detection accuracy with and without incremental learning. With incremental learning, we detected anomalous micro-actions from Seq2, using them to update the base set of 56 exemplars for detection on Seq3 (and vice-versa for testing on Seq2 with new exemplars learned from Seq3).

ture detection, feature extraction, dimensionality reduction, codebook size, and so on. As an unsupervised method, we require no extensive training and validation stages. Offline training time required for computing the distance matrix is modest, because the PM distance computation between a pair of tracklets is fast (10's of milliseconds using unoptimized MATLAB code).

We realize we have just scratched the tip of the iceberg. It is an open question on how best to select an appropriate number of exemplars (or clusters) without having prior knowledge of an expected number of behaviors to be observed. We presented a rudimentary method for incrementally updating the set of exemplars, and more sophisticated methods may be required. Future work includes investigation of cluster quality, incremental learning strategies, and the application of micro-action detections to the recognition of longer term events and multi-entity interactions.

5. Acknowledgements

This work was partially supported by DARPA contract W911NF-10-2-0066. We thank Yui Man Lui for many fruitful discussions about manifold geometry and for sharing his source code.

References

[1] A. Kovashka and K. Grauman. Learning a hierarchy of discriminative Space-Time neighborhood features for human action recognition. In *Proc. CVPR*, 2010.

[2] Z. Lin, Z. Jiang, and L. S. Davis. Recognizing actions by shape-motion prototype trees. In *Proc. ICCV*, 2009.

[3] Y. M. Lui, J. R. Beveridge, and M. Kirby. Action classification on product manifolds. In *Proc. CVPR*, 2010.

[4] F. Nater, H. Grabner, and L. V. Gool. Exploiting simple hierarchies for unsupervised human behavior analysis. In *Proc. CVPR*, 2010.

[5] J. C. Niebles, H. Wang, and L. F. Fei. Unsupervised learning of human action categories using spatial-temporal words. *International Journal of Computer Vision*, 79(3):299–318, 2008.

[6] S. O'Hara, Y. M. Lui, and B. A. Draper. Unsupervised learning of human expressions, gestures, and actions. In *Proc. Face and Gesture*, 2011.

[7] R. Poppe. A survey on vision-based human action recognition. *Image and Vision Computing*, 28(6):976–990, 2010.

[8] K. Rapantzikos, Y. Avrithis, and S. Kollias. Dense saliency-based spatiotemporal feature points for action recognition. In *Proc. CVPR*, 2009.

[9] M. S. Ryoo, C. C. Chen, J. K. Aggarwal, and A. Roy-Chowdhury. An overview of contest on semantic description of human activities (SDHA) 2010. In *Proc. ICPR*, 2010.

[10] D. H. Wolpert. The supervised learning no-free-lunch theorems. In *In Proc. 6th Online World Conference on Soft Computing in Industrial Applications*, pages 25–42, 2001.

[11] T. H. Yu, T. K. Kim, and R. Cipolla. Real-time action recognition by spatiotemporal semantic and structural forests. In *Proc. BMVC*, 2010.