Available online at www.sciencedirect.com
SCIENCE DIRECT
Image and Vision Computing 24 (2006) 455–472
image AND vision COMPUTING
www.elsevier.com/locate/imavis

Minimal-latency human action recognition using reliable-inference

James W. Davis *, Ambrish Tyagi

Department of Computer Science and Engineering, Ohio State University, 491 Dreese Lab, 2015 Neil Avenue, Columbus, OH 43210, USA

Received 26 October 2004; received in revised form 3 November 2005; accepted 31 January 2006

Abstract

We present a probabilistic reliable-inference framework to address the issue of rapid detection of human actions with low error rates. The approach determines the shortest video exposures needed for low-latency recognition by sequentially evaluating a series of posterior ratios for different action classes. If a subsequence is deemed unreliable or confusing, additional video frames are incorporated until a reliable classification to a particular action can be made. Results are presented for multiple action classes and subsequence durations, and are compared to alternative probabilistic approaches. The framework provides a means to accurately classify human actions using the least amount of temporal information.

Keywords: Action recognition; Reliable-inference; MAP; Video analysis

1. Introduction

Recognition of human actions is a challenging task that is applicable to various domains including surveillance, video annotation, human–computer interaction, and autonomous robotics. A fundamental question regarding action recognition is 'how much time (or video frames) is actually necessary to identify common human actions'? In the extreme case, people can easily recognize several different actions from looking at just a single picture—one need only flip through the pages of a newspaper or magazine to make this point. Other actions may require seeing additional video frames (perhaps the entire action sequence).

There are several instances of how biological perception is able to identify simple actions/gestures by observing just a small portion of the motion. For example, [22] showed that humans are capable of perceiving and distinguishing simple actions, such as walking, running, and jumping-jacks, from point-light exposures of only 200 ms. Clearly, no periodic information is being exploited. Other instances of such behavior can be seen with animal tricks/performances in response to the trainer's gesture commands. Often the animals are trained on a closed set of gestures and are able to identify their trainer's gesture from just an initial brief exposure of the movement.

A system capable of recognizing human actions from the *smallest* number of video frames would be particularly advantageous to automatic video-based surveillance systems. For instance, consider small unmanned aerial vehicles (UAVs) equipped with video cameras. The UAV's view area is constantly and rapidly changing, and therefore, immediate decisions about the activity in the scene are desirable within a few frames. This is also the case for multi-camera surveillance systems having limited computational processing time scheduled per camera. Even when longer duration video is available, rapid action detection may be particularly helpful in bootstrapping more sophisticated action-specific tracking or recognition approaches.

An important ability for a rapid (low-latency) action recognition system is to *automatically* determine when enough of a video sequence has been observed to permit a reliable classification of the action occurring, as opposed to making forced decisions on short (and potentially unreliable) video exposures or instead waiting for the entire (long) video sequence. This problem becomes even more challenging when the observation could start from any point (temporal offset) during the action sequence (i.e. the observed sequence does not always begin at the expected start pose of the action). For example, consider the case when a camera is switched on to detect actions and there is already an action in progress.

In this paper, we present an approach for quickly recognizing non-nested, temporally constrained human actions from the smallest video exposure possible. The method is

* Corresponding author. Tel.: +1 614 292 1553; fax: +1 614 292 2911.
E-mail addresses: jwdavis@cse.ohio-state.edu (J.W. Davis), tyagia@cse.ohio-state.edu (A. Tyagi).

0262-8856/$ - see front matter © 2006 Elsevier B.V. All rights reserved.
doi:10.1016/j.imavis.2006.01.012

formulated in a probabilistic inference framework and is trained with examples. First, the reliability of an input frame/subsequence is examined with a series of *a posteriori* ratio comparisons to the possible action classes (using Hidden Markov Model (HMM) output likelihoods and action priors). These reliability ratios assess the 'goodness' of an input for inferring a particular action class with respect to the probability of committing an error. If the input is deemed unreliable (not strongly indicative of a particular action), then no action classification takes place. In this case, the input is expanded/extended with additional video frames and then re-examined with the posterior ratios for a reliable-inference. This process is continued until either a reliable action inference is made or there are no more video frames to include (e.g. the person exited the scene). The worst-case scenario is that the entire sequence is evaluated (the default for most other approaches). Therefore, the approach is useful for recognizing actions that have some distinct difference at some point prior to the end of the action sequence. The method keeps tight bounds on the classification error by incorporating additional information for brief and confusing video inputs rather than forcing an early, and likely erroneous, action classification with a fixed-length sequence.

The main advantage of the proposed method is that the system only makes classifications when it believes the input is 'good enough' for discrimination between the possible actions. This is particularly favorable when there is a high cost for making errors and low (or no) cost for passively waiting for more video frames to arrive (advantageous with real-time video). Other popular probabilistic approaches, such as *maximum likelihood* (ML) and *maximum a posteriori* (MAP), perform a *forced-choice* classification for a fixed-length input regardless of the saliency/reliability of the input. To our knowledge, no previous attempts have been made to accurately classify human actions using the least amount of temporal information.

We demonstrate our rapid-and-reliable action recognition approach with sets of common actions having different video exposures of the full action duration. First, we examine the reliability of only a single frame to discriminate different actions, and demonstrate how single frame action classification can be a potential source of high confusion. We then address the reliability of action subsequences (multiple frames). Initially, we examine our method with a set of actions that begin in the same starting pose but gradually diverge into the different actions over time. Our goal here is to determine the *minimal* video exposures from the start frame/pose needed to reliably discriminate the actions. Next, we consider an even more general and difficult scenario with subsequences that are not constrained to begin with the start frame of the original sequence. In each of the experiments, we compare our recognition results with competing classification techniques.

We begin with a review of related methods for detecting and recognizing actions in single frames and video sequences (Section 2). We then formulate our probabilistic reliable-inference approach (Section 3). Next, we compare the approach to posterior analysis and discuss similar techniques that are capable of rejecting uncertain inputs (Section 4). Then, we present experimental evaluations and results with a discussion (Section 5) and provide possible extensions to the framework (Section 6). Lastly, we provide a summary and conclusion of the work (Section 7).

2. Related work on action recognition

The importance of human action recognition is evident by its increased attention in recent years (see surveys in [1,17,37]). Previously, several action recognition techniques have been proposed, but here we briefly mention only a few representative approaches that are relevant to comparing the single- and multiple-frame analysis domains.

For identifying pedestrians in a *single image*, [29] used wavelets to learn a characteristic pedestrian template and employed support vector machines for classification. In [18], a hierarchical coarse-to-fine template approach with edge maps was used to detect pedestrians. In [32], a general unsupervised recognition framework using clustering was employed to recognize several silhouette poses. In the *two frame* category, [23] used two simple properties (dispersedness, area) for discriminating humans and vehicles from image difference results, and maintained consistency over time with a temporal classification histogram. A cascaded AdaBoost architecture was proposed in [35] that used rectangular filters on intensity and difference images for pedestrian detection. None of these approaches attempt to recognize extended temporal actions based on characteristic movement patterns.

Dynamic action recognition methods from *multiple frames* or sequences typically involve analysis of trajectories or templates. For periodic actions such as walking and running, trajectory-based approaches for single and multiple cycle exposures include frequency-based Fourier methods [24], feature-based properties (e.g. stride) [11], spatio-temporal patterns [28], and HMMs [9]. With no part tracking, a different approach is to use spatio-temporal templates derived from the image sequence directly, including generic layered templates [6] and other periodic template representations [10,25,26,30].

Our work here is focused on action recognition from the *shortest-duration* video exposures needed to achieve high recognition rates with low recognition latency. In previous work, we presented initial results of the proposed reliable-inference framework for discriminating actions from single frames (poses) [13] and also provided a hierarchical extension for multiple frames using Motion History Images (MHIs) [12]. In this paper, we extend our prior work to efficiently examine multiple subsequence durations for different actions and incorporate HMM temporal models (instead of hierarchical MHIs) for the probabilistic evaluations. Many of the action recognition approaches described above could be potentially incorporated into our framework, however, we are not aware of any other general approaches related to our goal of reliable, yet minimal-latency, action recognition.

3. Reliable-inference

We base our reliable-inference (RI) approach on the 'key feature' proposal of [20] which states that the success of inferring a world property $\wp$ from a measurable feature f in context C can be formulated as the *a posteriori* probability $p(\wp|f,C)$. In our domain of action recognition, we consider the world property $\wp$ to be an action $\mathcal{A}$ to recognize (e.g. walking). The context C refers to a particular closed-world domain of actions $C=\{\mathcal{A}_1,\mathcal{A}_2,\ldots,\mathcal{A}_n\}$ that can occur in some situation (a closed-world context is an assumption used in most popular ML/MAP recognition techniques). The context-dependent reasoning provides a limited domain of actions for consideration during recognition. For example, if we know the person is traversing the scene, we could possibly limit the context to only locomotory behaviors such as walking and running to greatly reduce the search space of solutions.

According to [20], a 'reliable-inference' of action $\mathcal{A}_i$ from feature f will have a large posterior probability $p(\mathcal{A}_i|f,C)$ and a small probability of error $p(\neg\mathcal{A}_i|f,C)\approx 0$, where $\neg\mathcal{A}_i=(\cup_{j\neq i}\mathcal{A}_j)$. Hence, the *reliability* of feature f for inferring a particular action $\mathcal{A}_i$ can be measured by the ratio of these two probabilities:

$$R_{\text{post}}[\mathcal{A}_i,f]=\frac{p(\mathcal{A}_i|f,C)}{p(\neg\mathcal{A}_i|f,C)}=\frac{p(f|\mathcal{A}_i,C)p(\mathcal{A}_i|C)}{p(f|\neg\mathcal{A}_i,C)p(\neg\mathcal{A}_i|C)}$$
$$=\frac{p(f|\mathcal{A}_i,C)p(\mathcal{A}_i|C)}{\sum_{j\neq i}p(f|\mathcal{A}_j,C)p(\mathcal{A}_j|C)} \quad (1)$$

When $R_{\text{post}}[A_i,f]\gg 1$, the feature f is said to be a highly reliable indicator of $\mathcal{A}_i$. Thus both a large likelihood ratio and context-dependent prior ratio is required. The presence of a large likelihood ratio indicates that the observed feature f arises consistently with the existence of action $\mathcal{A}_i$, but not in its absence. A significant prior ratio states that the action $\mathcal{A}_i$ appears regularly/often.

As we are interested in the reliable-inference of different action classes given *partial/limited* video sequence exposures, we rewrite Eq. (1) for a target action class $\mathcal{A}_i$ and an observed video subsequence $O_{t_1:t_2}$ from time t_1 to t_2 (the new feature f) as:

$$R_{\text{post}}[\mathcal{A}_i,O_{t_1:t_2}]=\frac{p(O_{t_1:t_2}|\mathcal{A}_i,C)p(\mathcal{A}_i|C)}{\sum_{j\neq i}p(O_{t_1:t_2}|\mathcal{A}_j,C)p(\mathcal{A}_j|C)} \quad (2)$$

For subsequence $O_{t_1:t_2}$ to be a reliable indicator of action $\mathcal{A}_i$, we want $R_{\text{post}}[\mathcal{A}_i,O_{t_1:t_2}]\gg 1$. But how large does the R_{post} value need to be for $O_{t_1:t_2}$ to be considered a reliable indicator of $\mathcal{A}_i$? In other words, what is the value of the decision threshold $\lambda_{\mathcal{A}_i}$ such that $R_{\text{post}}[\mathcal{A}_i,O_{t_1:t_2}]>\lambda_{\mathcal{A}_i}$ only for those $O_{t_1:t_2}$ subsequences belonging to class $\mathcal{A}_i$?

We could simply choose the action class in C with the highest R_{post} (the most reliable estimate) for classification, which is equivalent to the MAP classification (see Appendix A), but this can result in an error when the input is unreliable (not enough discriminating information is present in $O_{t_1:t_2}$). Our approach is to select a threshold $\lambda_{\mathcal{A}_i}$ for each action $\mathcal{A}_i$ to classify $O_{t_1:t_2}$ only when it is truly reliable for discriminating the actions. Therefore, if any subsequence $O_{t_1:t_2}$ yields an R_{post} value below the threshold *for all actions* it is 'held' from classification and instead is extended with more video frames (to provide more information) and re-evaluated. This process is continued until a valid R_{post} for some action class is found (a threshold is exceeded) or all input frames have been exhausted.

The R_{post} reliability threshold $\lambda_{\mathcal{A}_i}$ can be learned by selecting the maximum R_{post} value for action class $\mathcal{A}_i$ using the subsequences of all $\neg\mathcal{A}_i$ training examples. Thus, no non-class subsequence (from $\neg\mathcal{A}_i$) produces an R_{post} value for class $\mathcal{A}_i$ that is above $\lambda_{\mathcal{A}_i}$. For any true subsequence of $\mathcal{A}_i$ that happens to produce an R_{post} value below this threshold, it is *not* considered to be an error. Rather, it only states that this subsequence is unreliable for classification to $\mathcal{A}_i$ (in comparison to the other $\neg\mathcal{A}_i$ examples) and that more information/video is required before it should be classified. Only when an R_{post} threshold is exceeded does a classification occur (or when no new frames can be incorporated). If the number of training examples is sufficiently large, a threshold selected for each action class in such a manner as described above will avoid most classification errors during testing with new examples.

The overall approach to recognition is shown in Fig. 1, where each stage labeled 'RI Test' refers to Algorithm 1. Initially, a single input video frame O_{t_1} at time t_1 is evaluated.

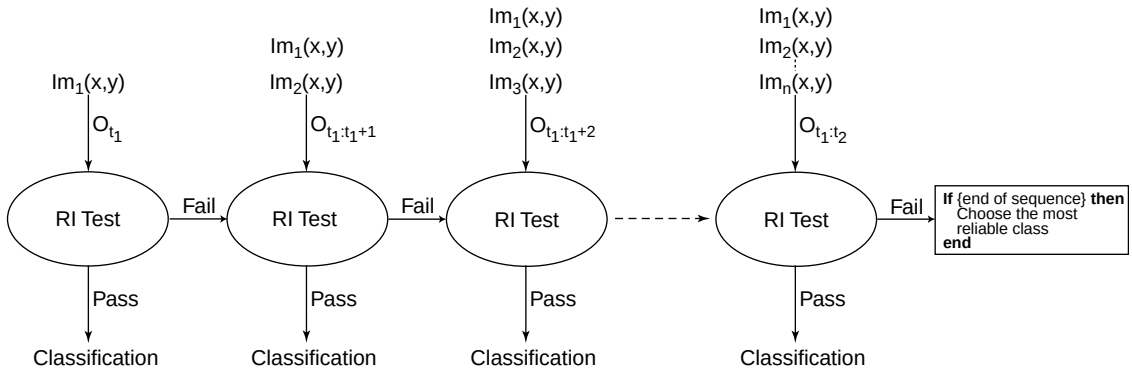


Fig. 1. RI approach to reliable recognition.

We compute the $R_{\text{post}}[\mathcal{A}_i, O_{t_1}]$ value for all actions $\mathcal{A}_i \in C$ and compare each R_{post} value with its decision threshold $\lambda_{\mathcal{A}_i}$. For any action's R_{post} value for O_{t_1} that meets its decision threshold, the action label is placed into a clique of potential classifications. If after examining all actions the resulting clique contains a single action class, then we reliably classify O_{t_1} to that class. In the event that the clique contains more than one class (due to independent λ thresholds for each action), we choose the class within the clique having the highest R_{post} (as it has the most reliable *valid* inference). If the clique is empty after examining all of the actions, we make no commitment to a class and extend the input by incorporating one additional video frame $O_{t_1} \rightarrow O_{t_1:t_1+1}$. Then the R_{post} reliability is examined with the new expanded input $O_{t_1:t_1+1}$ (two frames). The expansion process continues until time t_2, where the first reliable classification to an action can be made (using $O_{t_1:t_2}$) or when no new video frames can be included (in which case the action with the largest R_{post} is selected). We refer to this classification technique as 'thresholded R_{post}' (T-R_{post}).

Algorithm 1. Reliable-inference test

procedure RI TEST$O_{t_1:t_2}$
 Calculate $R_{\text{post}}[\mathcal{A}_i, O_{t_1:t_2}]$ for all classes $\mathcal{A}_i$
 Clique = { }
 for all classes $\mathcal{A}_i$ **do**
  if $R_{\text{post}}[\mathcal{A}_i, O_{t_1:t_2}] > \lambda_{\mathcal{A}_i}$ **then**
   add $\mathcal{A}_i$ to *Clique*
  end if
 end for
 if *Clique* = { } **then**
  Fail
 else
  Select class $\mathcal{A} = \mathcal{A}_i \in$ *Clique* arg max$\{R_{post}[\mathcal{A}_i]\}$
  Pass
 end if
end procedure

3.1. Statistical modeling with HMMs

To evaluate the R_{post} for a video subsequence $O_{t_1:t_2}$ to a particular action class (using Eq. (2)), we require the likelihood probability of $O_{t_1:t_2}$ for each class (i.e. $p(O_{t_1:t_2}|\mathcal{A}_i, C), \forall \mathcal{A}_i \in C$) and the class priors $p(\mathcal{A}_i|C)$. We model the class likelihoods from training data using HMMs and manually assign appropriate context-dependent class priors. Other temporal binding and evaluation approaches, such as Bayesian networks and hierarchical Motion History Images [12], could also be considered. We now present our HMM training and evaluation approach.

A HMM with M states (for an action class $\mathcal{A}_i$) is represented by a three-tuple

$$\Lambda_{\mathcal{A}_i} = (T, B, \pi) \tag{3}$$

where $T = \{t_{pq}\}$ is the state transition probability distribution and t_{pq} denotes the probability of transition from state p to state q ($1 \le p, q \le M$). Next, $B = \{b_q(O_t)\}$ is the observation probability distribution, where $b_q(O_t)$ is the probability of observing frame O_t in state q. Lastly, $\pi = \{\pi_p\}$ is the initial (prior) state distribution having $\sum_p \pi_p = 1$. In the training phase, we learn the HMM parameters for each class from the training data.

The number of HMM states in each experiment is typically determined empirically and intuitively such that each state reflects a natural phase in the action. For example, a four-state HMM can be used to adequately capture the different feet/leg positions during a walk/run cycle (see Fig. 12). We first roughly hand-segment the training data sequences into M temporal phases and estimate the output probability distribution for an image in each state separately. These estimates are used as the starting point (initialization) for the standard forward–backward algorithm [31] to learn the entire parameter set $\Lambda_{\mathcal{A}_i}$ for the HMM. The forward–backward algorithm initialized in such a way will try to maintain the desired logical phases of an action while maximizing the likelihood of the training sequences.

Gaussian Mixture Models (GMMs) are commonly used to model the probability density $b_q(O_t)$ inside each state of a HMM, where the number of components can be selected in an information theoretic manner using the Bayesian Information Criterion (BIC). Both GMMs and BIC were used by [4,21] for model selection in an HMM, but other clustering methods could also be employed (e.g. [5,7]). We model the likelihood of frame O_t (of subsequence $O_{t_1:t_2}$) appearing from a particular state q of an HMM, using a feature vector $\mathbf{f}$ of O_t and a GMM, as

$$b_q(O_t) = p(\mathbf{f}|\theta_q) = \sum_{k=1}^{K} w_k \cdot g_k(\mathbf{f}|\mu_k, \Sigma_k) \tag{4}$$

where $g_k(\mathbf{f}|\mu_k, \Sigma_k)$ is the likelihood of $\mathbf{f}$ (feature representation of O_t) appearing from the kth Gaussian distribution parameterized by the mean μ_k and covariance Σ_k, with mixture weight w_k. For estimating the parameters θ_q, we employ the Expectation Maximization (EM) algorithm [14] that maximizes the class log-likelihood

$$\mathcal{L}(\theta_q|\mathbf{f}^1, \ldots, \mathbf{f}^N) = \sum_{n=1}^{N} \log(p(\mathbf{f}^n|\theta_q)) \tag{5}$$

for N training examples $(\mathbf{f}^1, \ldots, \mathbf{f}^N)$ of frames associated with state q.

Initial values for the means, covariances, and mixture weights in Eq. (4) can be estimated using a K-means clustering of the training samples. As the clustering result can vary depending on the seed values (initial means), the entire EM algorithm is repeated multiple times, each time using a K-means clustering result from a different random seed initialization. We then choose the EM mixture model that produces the maximum class log-likelihood (Eq. (5)).

One issue regarding mixture models is the number of clusters/distributions K needed to model the data. Rather than manually selecting an arbitrary K, we automatically select from models of different K, the model that maximizes the Bayesian

Information Criterion (BIC) [33]. The BIC for a given model parametrization θ_q is computed as

$$\mathrm{BIC}(\theta_q) = 2\mathcal{L}(\theta_q|\mathbf{f}^1,\ldots,\mathbf{f}^N) - P\log(N) \quad (6)$$

where P is the number of independent model parameters to be estimated. In our formulation, we have

$$P = K\left(m + \frac{m^2+m}{2}\right) + (K-1) \quad (7)$$

with K distributions, $(m+(m^2+m)/2)$ independent parameters for each mean and covariance ($m=\dim(\mathbf{f})$), and $(K-1)$ independent mixture weights (constrained to $\sum w_k = 1$). Since the class log-likelihood of the mixture model (Eq. (5)) improves when more parameters are added to the model (i.e. larger K), the term $P\log(N)$ is subtracted from (twice) the class log-likelihood in Eq. (6) to penalize models of increasing complexity. The BIC is maximized in an information theoretic manner for more parsimonious parameterizations. An iterative split-sample training and validation method can be employed where 50% of the training examples are randomly selected and used by K-means/EM to estimate the model parameters, and the remaining 50% of the samples are used to compute the BIC for that model.

To evaluate a new subsequence and calculate its R_{post}, we compute the likelihood probabilities of the given subsequence (with each frame represented by a feature vector) to the different action class HMMs (we use manually assigned class priors). For a given HMM, we compute the likelihood for the subsequence using only the forward pass of the forward–backward learning procedure. The process of adding frames to a subsequence also becomes computationally efficient using the forward pass due to the recursive calculation of the forward probability [31].

4. Comparative evaluation of RI technique

In this section, we first compare the proposed reliable-inference method with direct posterior probability analysis for classification. Next, we describe related techniques for dealing with classification uncertainty.

4.1. Comparison to direct posterior analysis

In the case with temporal data, often the likelihoods (e.g. from an HMM) increase or decrease with the addition of new information. In situations where an input can likely belong to two or more action classes (a confusing input), classification with ML/MAP-based methods is prone to error, as they are forced-choice approaches. The R_{post} ratio, in contrast, reflects the reliability based on the *relative* strength of an action class with respect to all other action classes. Hence, we can simultaneously evaluate when we should classify (and to which action class) and when we should not make a decision.

In Fig. 2, we show the posterior and R_{post} values over time for an action sequence O to classes $\mathcal{A}_1$, $\mathcal{A}_2$, and $\mathcal{A}_3$ (the graphs were generated from a Bend sequence and the HMMs trained for Bend ($\mathcal{A}_1$), Crouch ($\mathcal{A}_2$), and Sit ($\mathcal{A}_3$) from Section 5.2). As shown in Fig. 2(a), the initial portion of the sequence produces very similar posteriors for all three actions (and thus will yield MAP classification errors), but as more information is added, the posteriors begin to diverge and the true class $\mathcal{A}_1$ slowly becomes the most probable class for the input O (the true class). However, in the example shown in Fig. 2(b), the separation of $\mathcal{A}_1$ is more pronounced and much appears much earlier on the R_{post} scale.

In another example (see Fig. 3), we show that determining the class threshold based on multiple examples is more applicable (and useful) for R_{post} than for an a posterior distribution (motivated by the study in [2] for likelihood decisions). In Fig. 3(a), we show (synthetic) distributions for two classes $\mathcal{A}_1$ and $\mathcal{A}_2$ (two 1D Gaussians with standard deviation 15 and means centered at ± 25) from which we sampled data points. In Fig. 3(b), we show the distribution of the data examples on the log R_{post} scale (x-axis) to class $\mathcal{A}_1$ (i.e. log $R_{\text{post}}[\mathcal{A}_1]$). If we choose the previously mentioned T-R_{post} thresholding policy (i.e. selecting the threshold for class $\mathcal{A}_i$ as the maximum R_{post} value of the examples from class

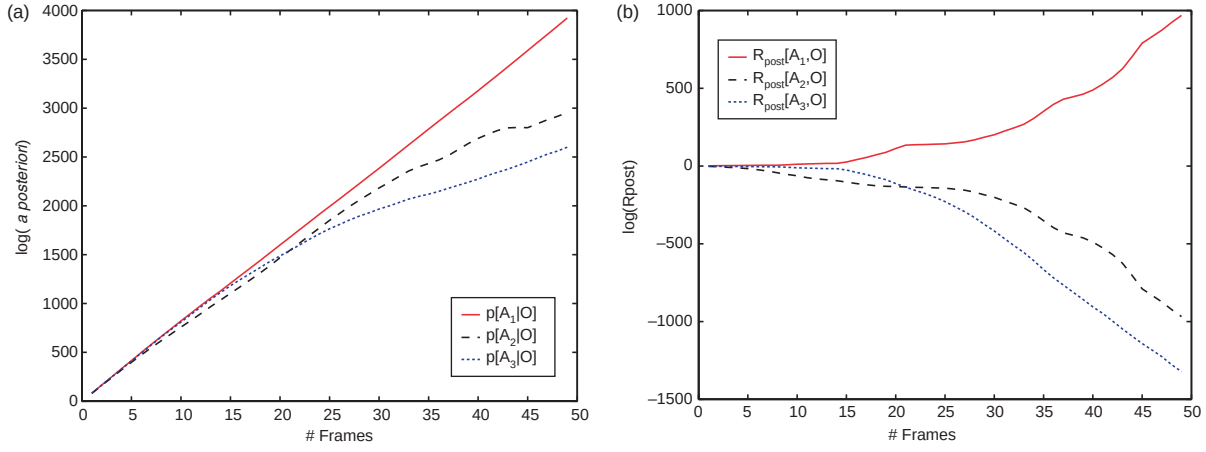


Fig. 2. (a) Posterior and (b) R_{post} values for a video sequence O to the action classes $\mathcal{A}_1$, $\mathcal{A}_2$ and $\mathcal{A}_3$. R_{post} increases the separation between classes better and earlier as compared to the posterior measure.

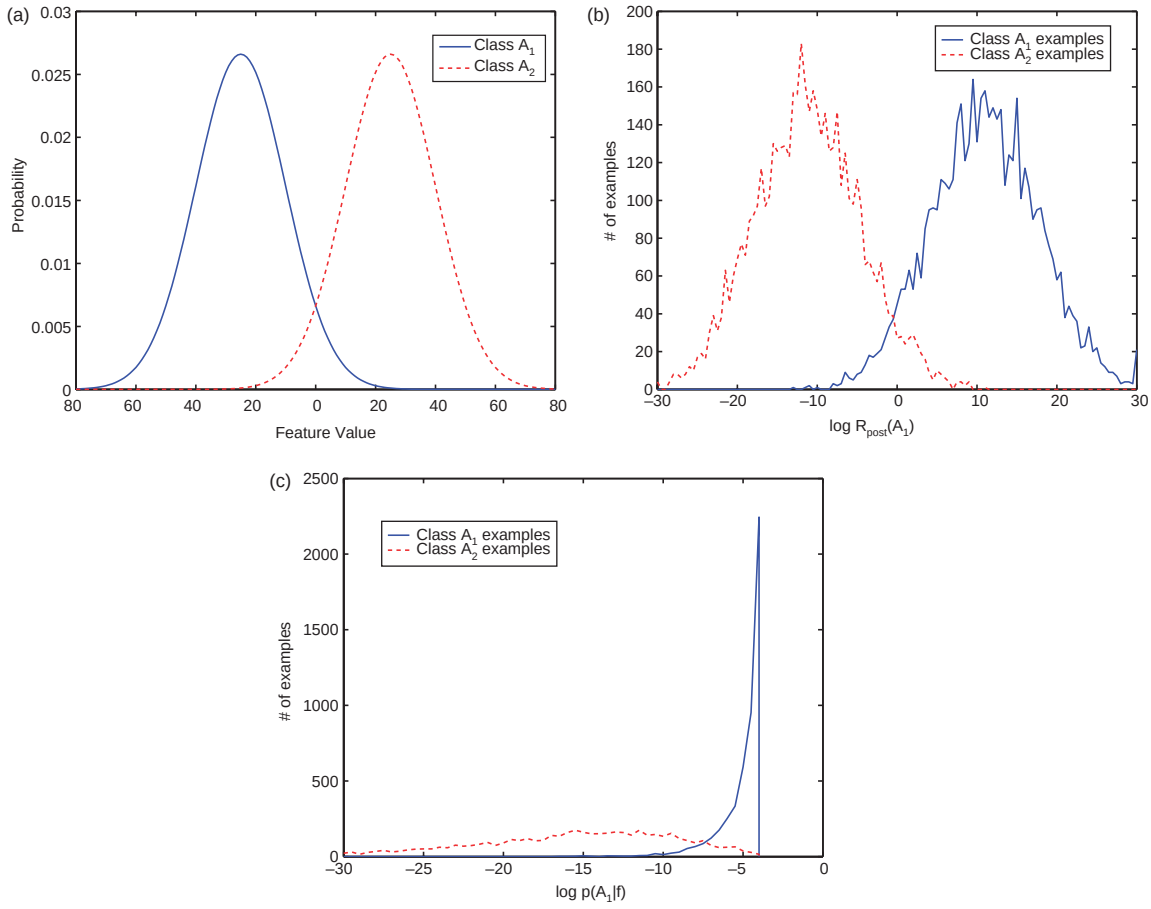


Fig. 3. Distribution comparisons: (a) distributions of class $\mathcal{A}_1$ and $\mathcal{A}_2$ examples; (b) $R_{post}[\mathcal{A}_1]$ distributions; (c) *a posteriori* distributions.

$\neg\mathcal{A}_i$), 49.26% of the class $\mathcal{A}_1$ examples are correctly classified (50.74% are deemed unreliable), while 100% of the class $\mathcal{A}_2$ examples are correctly rejected. A similar set of distributions is shown in Fig. 3(c), but now we use the log *a posteriori* scale to class $\mathcal{A}_1$ on the x-axis. With a similar thresholding strategy (using the maximum posterior value of the class $\mathcal{A}_2$ examples to class $\mathcal{A}_1$ as the threshold), only 1.38% of the class $\mathcal{A}_1$ examples are correctly classified (100% of the class $\mathcal{A}_2$ examples are correctly rejected). This worse classification result in comparison to R_{post} is due to the fact that there is more overlap of class $\mathcal{A}_2$ in class $\mathcal{A}_1$ on the *a posteriori* scale. Even if we instead use the distribution cross-over points in the R_{post} and *a posteriori* scales as the thresholds, 95.14% of the class $\mathcal{A}_1$ and $\mathcal{A}_2$ examples are correctly classified in the R_{post} scale, and a lower 93% of examples for class $\mathcal{A}_1$ and $\mathcal{A}_2$ are correctly classified in the a posterior scale. Other threshold methods could certainly be employed, but their comparison here may not be straightforward.

4.2. Comparison to other classification techniques that handle uncertainty

In this section, we briefly mention and compare a few alternative classification techniques that are capable of non-forced decision making, and hence could be used for reliable-inference.

4.2.1. Sequential probability ratio test

Sequential analysis [36] is a method of statistical inference where one can collect several observations with the aim of improving the classification confidence with each new measurement. In the two-hypothesis sequential probability ratio test (SPRT), the cumulative likelihood ratio of m observations of random variable $\mathbf{x}_{1:m}$ is given by

$$l_m(\mathbf{x}) = \frac{p(\mathbf{x}_{1:m}|\mathcal{A}_1)}{p(\mathbf{x}_{1:m}|\mathcal{A}_2)} \tag{8}$$

and is compared to thresholds $\lambda_{\mathcal{A}_1}$ and $\lambda_{\mathcal{A}_2}$ (where $\lambda_{\mathcal{A}_1} > \lambda_{\mathcal{A}_2}$). If $l_m(\mathbf{x}_{1:m}) > \lambda_{\mathcal{A}_1}$, class $\mathcal{A}_1$ is chosen, or if $l_m(\mathbf{x}_{1:m}) < \lambda_{\mathcal{A}_2}$, class $\mathcal{A}_2$ is chosen. If $\lambda_{\mathcal{A}_2} \leq l_m(\mathbf{x}_{1:m}) \leq \lambda_{\mathcal{A}_1}$ then classification is avoided and new observations are incorporated. One standard method to compute the probabilities with multiple observations (given class $\mathcal{A}$) is to assume independence:

$$p(\mathbf{x}_{1:m}|\mathcal{A}) = \prod_{i=1}^{m} p(\mathbf{x}_i|\mathcal{A}) \tag{9}$$

We could possibly apply the SPRT sequential analysis technique to our reliable-inference task for action recognition. However, instead of multiple measurements of the same random variable (as in traditional SPRT), our measurements are temporally ordered video frames (of an action). Hence, other temporally constrained approaches (e.g. HMM) would be

more appropriate to calculate the probabilities than with Eq. (9). Also, the problem of sequential testing for *multiple* classes ($\mathcal{A}_1 \cdots \mathcal{A}_n$) is relatively more difficult in SPRT than with the two-class case. Several ad hoc methods and simple (non-optimal or asymptotically optimal) extensions of SPRT have been proposed to handle multiple classes [3].

4.2.2. Bayes risk

Another classification method from classical decision theory is to obtain a reliable classification by minimizing the overall Bayes risk [15,34]. To incorporate the idea of confusion in this approach, a 'dummy class' is introduced and a loss function is selected such that the cost associated with a misclassification is higher than the cost of classifying to the dummy class. The method provides an upper bound on the *total* probability of error rather than a bound for each class.

Let the loss function be $\mathcal{L}$, such that $\mathcal{L}(\alpha_i|\mathcal{A}_j)$ describes the loss incurred for classifying to class α_i when the actual classification is $\mathcal{A}_j \in C$. Here, $\alpha_i \in (C \cup \Phi)$ where Φ is the dummy class for classification when the input is confusing. The expected loss or conditional risk for classifying a video subsequence $O_{t_1:t_2}$ can now be defined as

$$R(\alpha_i|O_{t_1:t_2}) = \sum_j \mathcal{L}(\alpha_i|\mathcal{A}_j) P(\mathcal{A}_j|O_{t_1:t_2}) \tag{10}$$

with

$$\mathcal{L}(\mathcal{A}_j|\mathcal{A}_j) = 0 \tag{11}$$

$$\mathcal{L}(\mathcal{A}_i|\mathcal{A}_j) = \beta, \quad i \neq j \tag{12}$$

$$\mathcal{L}(\Phi|\mathcal{A}_j) = \lambda, \quad 0 \leq \lambda \leq \left(\frac{c-1}{c}\beta\right) \tag{13}$$

where c is the number of classes and λ is the penalty for classifying to the dummy class. For the standard zero-one loss formulation [15], $\beta = 1$ in Eq. (12) and (13). The decision rule on Eq. (10) divides the feature space into c acceptance areas and one reject area. Class α_i that minimizes the Bayes risk in Eq. (10) is selected as the final classification.

In comparison, our reliable-inference method makes use of individual class thresholds (as opposed to one single threshold), and hence one can have different bounds on the error for each class. Also for practicality, our framework learns the class thresholds such that it can correctly reject all non-class examples in the training data. Furthermore, the Bayes risk method operates on the posterior probabilities rather than on the more discriminating R_{post} metric.

4.2.3. Cascade architecture

A cascaded architecture, such as one used in [35] with AdaBoost for pedestrian detection, could potentially be adopted for rapid and reliable-inference. The cascade was designed with the goal of maintaining a high detection rate at each level while systematically reducing the false positive rate at each stage as it is traversed. At any level, if a true negative is detected (e.g. a 'non-pedestrian' in [35]) the input is rejected and the processing stops, otherwise the input is continually tested for a true positive in all of the later levels (until the last stage of the cascade is reached or it is classified as a true negative).

Instead of eliminating true negatives at each successive level of the cascade, we could *reverse* the process for reliable-inference and construct the classifiers such that only the reliable inputs (true positives) are classified at each level, and any confusing inputs are examined in the subsequent levels. By adding more information at each new level we could construct a more effective classifier to reduce the confusion as we traverse through the cascade.

Such an architecture will have certain problems if blindly used on action sequences. First, the size of the cascade will grow as the average length of the actions increases (e.g. one level per frame). Second, this approach will have issues when comparing action sequences of different length; hence external warping techniques (such as dynamic time warping) will be required. Finally, this architecture does not correctly handle periodic/repeating actions. However, we note that our proposed approach, as shown in Fig. 1, is a type of cascade that is specifically designed to avoid the aforementioned problems.

5. Experimental evaluation

We evaluated our reliable-inference framework for different video exposure lengths of several common human actions (walking, running, standing, bending-forward, crouching-down, and sitting) of different people and at different views. Each atomic action (e.g. walking) is assigned a particular class, and our experiments are designed to examine the recognition capability of the approach using different types of action subsequences. In many of these scenarios, certain actions have strong similarities that provide ideal testing sets for the proposed reliable-inference approach.

Initially, we examine action discrimination from single frames (at multiple views) for a person walking, running, and standing (Section 5.1). Next, we evaluate the framework's recognition capability for video subsequences. In the first subsequence-based experiment (Section 5.2), we test the framework using multiple examples of Bend, Crouch, and Sit actions with subsequences starting at the first frame of the action (i.e. $O_{1:t}$) and examine how much of the video sequence is necessary to reliably distinguish each action. In the next experiment (Section 5.3), we examine periodic Walk-left, Walk-right, Run-left, and Run-right actions with subsequences that start at any time within the action cycle (i.e. $O_{t_1:t_2}$). We assume equal priors for the action classes, though the approach is general to any set of priors. We note that the framework is not specific for these actions, but is general to any collection of non-nested actions that can be probabilistically modeled.

For all of the experiments, the video sequences were first processed using standard background-subtraction (or thresholding) techniques to create sequences of silhouette images. We represented each silhouette image using a compact subset of similitude moments [19]. For a binary silhouette image I, the similitude moments η_{ij} are derived from the central moments

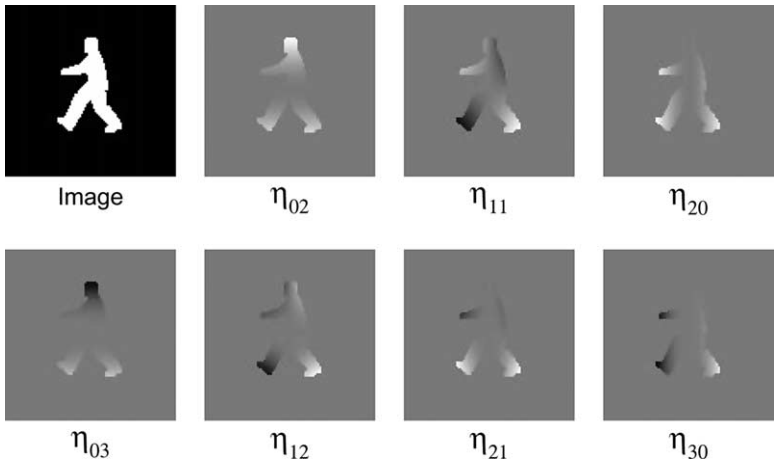


Fig. 4. Similitude moments. Images show the relative weight of each pixel for a particular moment before final aggregation over the entire image.

ν_{ij} and are given by

$$\eta_{ij} = \frac{\nu_{ij}}{(\nu_{00})^{\frac{i+j}{2}+1}} \tag{14}$$

$$\nu_{ij} = \sum_x \sum_y (x-\bar{x})^i (y-\bar{y})^j I(x,y) \tag{15}$$

for orders $(i+j) \geq 2$. These moments produce excellent global shape descriptors for binary (and grayscale) images in a translation- and scale-invariant manner. If rotation invariance is also desired, absolute moment invariants [19] could be employed. In Fig. 4, we visually show how the similitude moments relatively weight different portions of the silhouette. The moment images show the value $(x-\bar{x})^i(y-\bar{y})^j$ for each pixel location (x,y) in the image (these values are summed in the image and scaled to get η_{ij}). We note that other types of features could be employed (e.g. Zernike moments, Principal Components Analysis, Fourier descriptors, etc.[1]). For our experiments, we additionally whitened [15] the similitude feature space to work with uncorrelated feature dimensions.

5.1. Discrimination from a single frame

We first evaluated our RI framework with the action classes Walk, Run, and Stand from different viewpoints to determine the feasibility of using individual poses/frames for discrimination. We examined the individual R_{post} discrimination results for the actions using the T-R_{post} method and compared them to MAP recognition results.

To examine a large number of views for the actions in a controlled manner, we generated synthetic silhouettes from actual motion-captured human actions. We first motion-captured a person performing the Walking, Running, and Standing actions at different efforts/styles using a Vicon-8 motion-capture system. For Walking and Running, one cycle at each pace (slow, medium, fast) was extracted. For Standing, slight variations of two different postures were captured. Maya animation software was then used to create a 3D person model that could be consistently rendered at any desired viewpoint. The motion-capture data was mapped to the 3D body model and rendered (orthographic projection) as silhouettes from multiple viewpoints using OpenGL. Each pose was rendered at 21 different viewpoints separated by 30° horizontal and vertical intervals. In Fig. 5, we show silhouettes of the different actions (at different styles) and the 21 views for a single walking pose. The total number of images for classes Walk, Run, and Stand were 2184, 1512, and 1974, respectively.

The first seven similitude moment features (i.e. orders $2 \leq (i+j) \leq 3$) capturing the variance and skewness of the silhouette were used to represent each rendered image, and the distribution for each action was modeled as a GMM (i.e. a single-state HMM). For each K under consideration in the GMM ($K=2$–24 components, in steps of 2), the Kmeans/EM algorithm was repeated five times (EM itself was limited to 20 iterations). This entire process was repeated for three different split-sample partitions of the class data and the model having the overall largest BIC was selected as the final likelihood model. In Fig. 6(a), we show the BIC values as a function of K for the running data using three different split-sample iterations. The resulting mixture model corresponding to the maximum BIC (at $K=4$) is shown in Fig. 6(b).

As described in Section 3, we selected each class detection threshold λ as the maximum R_{post} value of the non-class training examples. The T-R_{post} recognition results for the actions using these thresholds are reported in Table 1. By design, the RI method does not commit any classification errors on the training data, but the resulting confusion is quite high for two of the classes (71, 61, and 12%, for Walk, Run, and Stand, respectively). Hence, it appears that only a small fraction of poses (at certain views) are distinct in this moment feature space. For comparison, we also report the corresponding recognition results using MAP (with the same trained likelihood models and priors). Though MAP is able to correctly classify most of the poses, it makes several errors (3–15% in

[1] We also tested Principal Components Analysis and Fourier descriptors in several of our experiments. The different feature sets provided similar recognition performance, and thus we chose to use similitude moments due to their conciseness and simplicity.

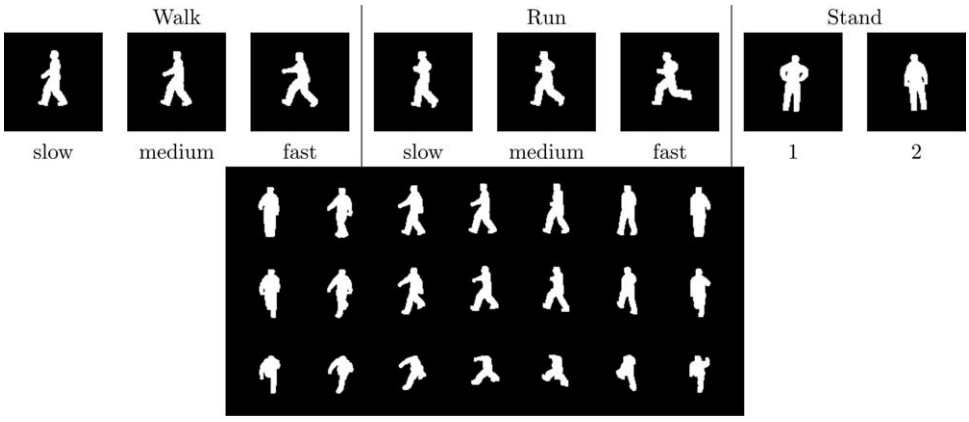


Fig. 5. Example silhouettes for action classes Walk, Run, and Stand. Each class has multiple efforts/styles (top row), and each pose is rendered at 21 different views (bottom image).

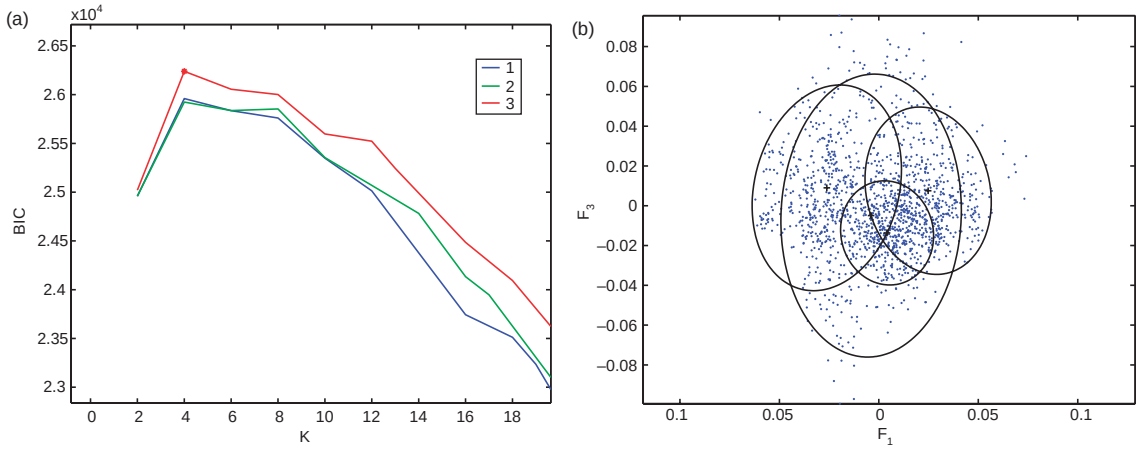


Fig. 6. Likelihood model for running. (a) BIC values for different K using three split-sample iterations. (b) Mixture model (contour plot at 4σ for moments η_{02}, η_{11}) corresponding to the maximum BIC (at $K=4$).

each class) due to its inability to detect and deal with unreliable/confusing inputs.

Overall, single images of actions can be a source of high confusion and thus do not usually provide sufficiently reliable classification results. However, if more frames in a temporal sequence are available, then adding this information should help to improve the recognition performance. Once a sufficient length of an action is seen, only then should a commitment to a class be made. Note that the entire sequence may not need to be examined to make a reliable classification. Our aim in the following experiments was to find the *minimal* exposure lengths of actions required to make reliable classifications.

5.2. Subsequences starting from the initial frame ($O_{1:t}$)

In this experiment, actions Bend, Crouch, and Sit at a side view were examined. Each action starts in a common standing pose (identical initial frame for each action), but becomes more distinct over time from the other actions. A total of 30 video sequences per action were collected, with five different people

Table 1
Recognition rates comparing T-R_{post} and MAP classification using equal class priors

Input	Method	Classification				% Confusion/error
		Walk	Run	Stand	Confusions	
Walk	T-R_{post}	639	0	0	1545	70.74/0.00
	MAP	2129	48	7	–	0.00/2.52
Run	T-R_{post}	0	587	0	925	61.18/0.00
	MAP	223	1280	9	–	0.00/15.34
Stand	T-R_{post}	0	0	1736	238	12.06/0.00
	MAP	91	11	1869	–	0.00/5.32

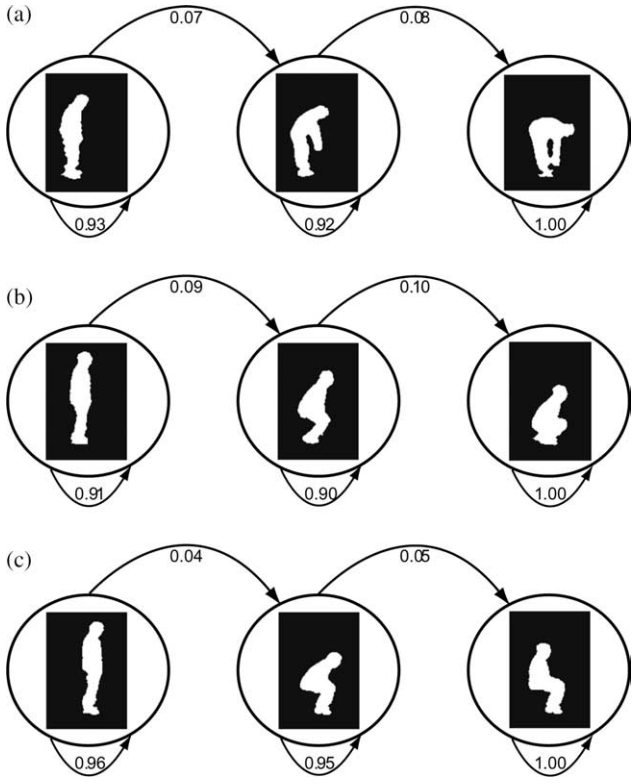


Fig. 7. Three-state HMMs labeled with transition probabilities for actions (a) Bend, (b) Crouch, and (c) Sit.

performing each action six times. Each sequence was trimmed to the start and end frames of the action. Of these, 20 sequences were used to train the system, and the remaining 10 were used for testing. The first seven similitude moment features (Eq. (14)) were used to represent each silhouette image (from background-subtraction). For each action class, we trained a three-state HMM, where the states were selected to represent the initial standing phase, intermediate phase, and final terminating phase of each action. The resulting HMMs for Bend, Crouch, and Sit (and representative silhouettes for each phase/state) are shown in Fig. 7. Since all training and testing subsequences start from the first frame in the sequence, the first HMM state prior was set to $\pi_1 = 1$.

In Table 2 (top), we show the T-R_{post} recognition details for the training sequences, reporting the percentage of subsequences $O_{1:t}$ classified (i.e. subsequences with a valid R_{post} to an action class). We also report the average percent duration of the sequence to the initial classification point $((k+1)/\mathit{length}(O)\times 100$ for sequence O, where k is the length of longest confused subsequence of O) for all sequences that are classified. As shown in the table, Sit was typically the first action to be recognized, where an average of only 19% of the entire sequence length needed to be observed before a reliable class commitment was made. Bend and Crouch each required around 31% of the sequence length before a reliable classification was made. The low percentages of subsequence duration reinforce the fact that only the initial smaller/shorter subsequences were more prone to confusion.

Table 2
T-R_{post} recognition statistics for Bend, Crouch, and Sit actions

Analysis using $O_{1:t}$	Bend	Crouch	Sit
% Subsequences classified (non-confusing)	71.0	71.8	82.4
Avg. % duration to initial classification point	31.4	31.2	18.6
Analysis using O_t	**Bend**	**Crouch**	**Sit**
% Subsequences classified (non-confusing)	66.6	77.1	76.3
Avg. % duration to initial classification point	35.8	26.0	24.2

We additionally compared the T-R_{post} subsequence recognition results with MAP and a thresholded *a posteriori* (T-AP) approach. The T-AP threshold for each action class was set to the maximum *a posteriori* value of all non-class subsequences and only when the posterior probability of a subsequence passed the threshold was a classification made (similar to the T-R_{post} approach, but using individual posteriors rather than the ratios). In Fig. 8, we show the detection counts (true positives) for the three classes, for various subsequence durations (starting from the first frame) of the training data. Both the T-R_{post} and MAP methods correctly classified each full-length sequence. The T-R_{post} and MAP methods also converge to nearly 95% recognition around 60, 80, and 30% of the sequence duration for Bend, Crouch, and Sit, respectively. Though the forced-choice MAP approach achieves a faster recognition convergence than T-R_{post}, MAP makes several classification errors on the shorter subsequences of the training data (4–12% of the total number of subsequences). The T-R_{post} method commits *no* classification errors on the training data by instead waiting for more information/frames to make a reliable classification.

In comparison to T-AP, the plots show a much faster convergence to the correct classification with T-R_{post} (and MAP), where T-AP could not even classify most full-length sequences (see Bend and Crouch) due to high posterior thresholds. When learning the T-AP threshold of class $\mathcal{A}_i$ from negative examples, if any non-class ($\neg\mathcal{A}_i$) example has (1) a much longer duration than the standard sequence length for class $\mathcal{A}_i$ and has (2) a likelihood >1 for each frame of the sequence (possible with continuous density estimates), then the resulting posterior probability for the non-class sequence to $\mathcal{A}_i$ can be larger than the posterior of any shorter true class example. This is the pitfall with the T-AP approach, and the effect can be seen in the Bend and Crouch results where a longer Sit example corrupted the reliability thresholds for both Bend and Crouch. However, the T-R_{post} method employs the *ratio* of class-to-non-class posteriors and thus the effect of sequence length is offset due to the ratio of posteriors to different classes.

In Fig. 9, we show the classification rate and recognition errors for the *testing* sequences (10 sequences per action). Again, the recognition rates of T-R_{post} and MAP are more comparable (MAP converges slightly faster) and better than the worse T-AP approach. There are fewer errors by T-R_{post} as compared to MAP on the new testing sequences, specifically in the shorter sequences. The errors in T-R_{post} are mainly due to the thresholds computed from a limited set of training examples. With T-AP, there were no classification errors (only a few Sit classification commitments were made).

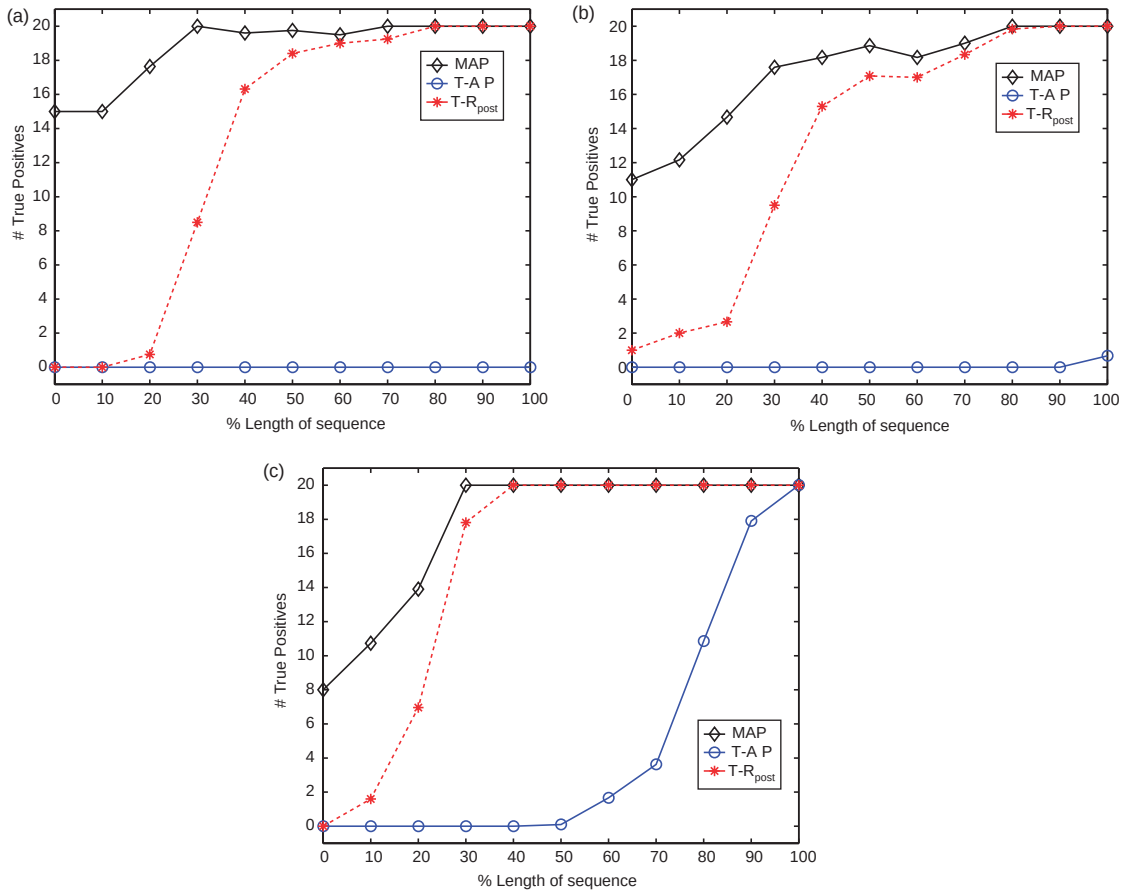


Fig. 8. Number of examples detected using MAP, T-AP, and T-R_{post} approaches for different subsequence lengths of training examples: (a) Bend, (b) Crouch, and (c) Sit.

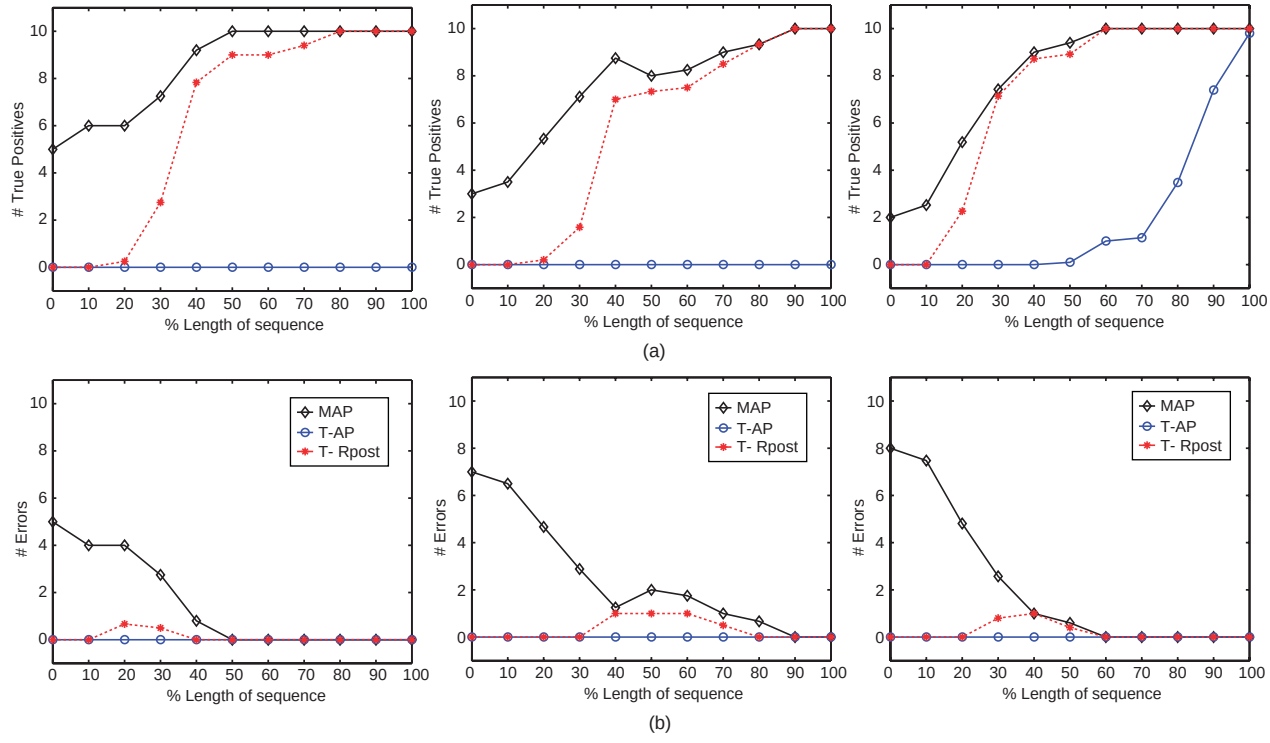


Fig. 9. Plots in (a) show the number of correct classifications and plots in (b) report the number of errors detected using MAP, T-AP, and T-R_{post} approaches for different subsequence lengths of testing examples for Bend, Crouch, and Sit, respectively.

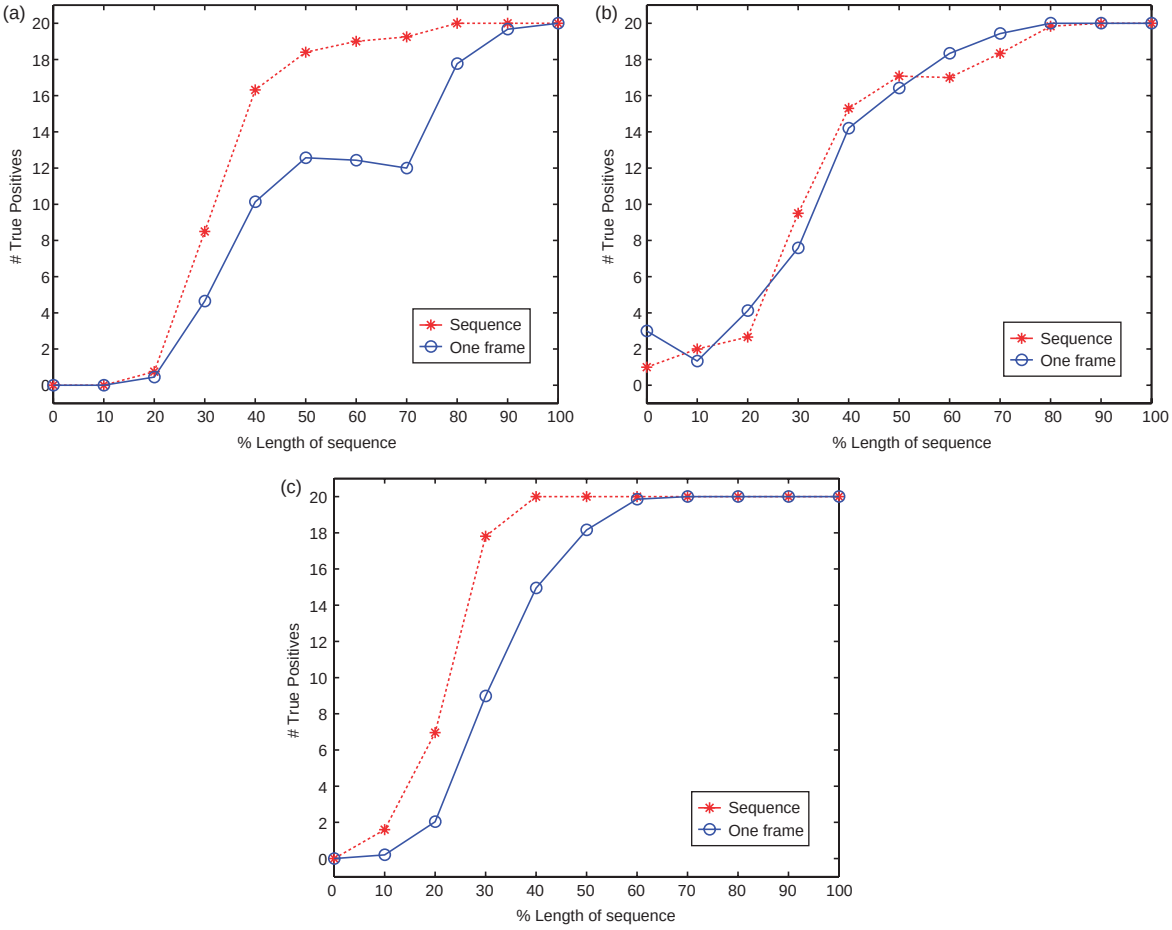


Fig. 10. Number of examples detected with T-R_{post} using single frames and sequences for different subsequence lengths of training examples: (a) Bend, (b) Crouch, and (c) Sit.

A natural question that may arise with the framework is whether the approach is actually making use of the temporal information or if the method is instead only looking/waiting for the first distinguishing frame/pose. To address this question, we compared the previous T-R_{post} results with a new, *limited* T-R_{post} method that uses only the current frame/pose instead of employing the subsequence information. The results using individual frames are reported in Table 2 (bottom), and show that it takes slightly *longer* to reliably classify Bend and Sit from the single-frame examples, thus giving a lower rate of the number of classifiable subsequences. In Fig. 10, we show the detection counts (true positives) obtained from the T-R_{post} recognition method on single frames versus subsequences (for the training sequences). The x-axis in these plots denotes the percentage length of the observed video sequence, and the y-axis shows the number of true detections at that exposure length (or the pose at that point in the sequence). Overall, we again see that the subsequence information provides more information to allow shorter exposures to be recognized quicker.

5.3. Subsequences starting at any frame ($O_{t_1:t_2}$)

In the next experiment, we considered a more difficult scenario of recognizing actions from subsequences, which can start at any point in time *during* the action. We examined new periodic walking and running actions with the aim to rapidly discriminate between Walk-left, Walk-right, Run-left, and Run-right examples recorded at a nearly side view (see example silhouettes in Fig. 11). To make the recognition task even more difficult, we removed the translation component of each person (to focus only on the changing body shape) and used only five moment features to represent the silhouettes extracted from the images (three second-order moments η_{20}, η_{02}, and η_{11} capturing variance, and two third-order moments η_{30} and η_{03} capturing skewness). Approximately 40 video sequences per action were collected, with eight people performing each action multiple times (with a minimum of two cycles in each sequence). From these recorded sequences,

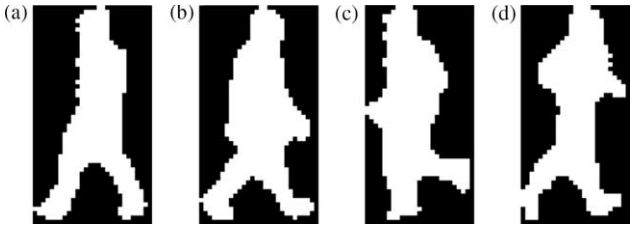


Fig. 11. Silhouettes for (a) Walk-left, (b) Walk-right, (c) Run-left, and (d) Run-right.

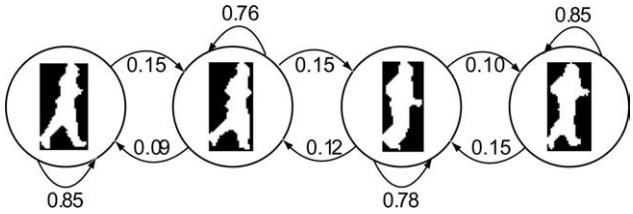


Fig. 12. HMM for Run-right labeled with transition probabilities.

we extracted all one-cycle length sequences such that they could start at any time in the action cycle. Approximately 60% of the sequences were used for training, and the rest were used for testing. For each action class, we initialized and trained a four-state HMM representing four phases in the periodic actions (see the HMM for Run-right in Fig. 12). A typical run/walk action will phase through the states as $1\rightarrow2\rightarrow3\rightarrow4\rightarrow3\rightarrow2\rightarrow1$. The HMM state priors were each set to .25 since the subsequences could start at any point in the action.

In this experiment, we examined all possible subsequences of the action cycle (i.e. $O_{t_1:t_2}$ with $1\leq t_1\leq N_c$ and $1\leq|t_2-t_1+1|\leq N_c$, where N_c is the number of frames in the cycle). This resulted in 835, 830, 569, and 599 training sequences and 559, 551, 346, and 345 testing sequences for Walk-left, Walk-right, Run-left, and Run-right, respectively. The recognition task is analogous to turning on the video camera at any point during the action and examining the video until a reliable-inference can be made.

In Table 3, we show the recognition details of the T-R_{post} recognition method for the training sequences. Both of the Run action classes had more reliable subsequences (92–96% of the total number of subsequences) than the Walk actions (86–91%). For the Run classes, reliable decisions were made within an 11% exposure of the cycle length on average, regardless of the starting point of the subsequences. For the Walk classes, 12–16% of the cycle was required on average for classification.

We compared the T-R_{post} recognition results for the training sequences with the MAP and T-AP approaches. As shown in Fig. 13, both the T-R_{post} and MAP methods correctly classified each full-length sequence. The T-R_{post} and MAP methods also converged to nearly 95% recognition around 50, 50, 60, and 40% of the sequence duration for Walk-left, Walk-right, Run-left, and Run-right, respectively. The forced-choice MAP approach still achieved a faster recognition convergence than T-R_{post}, though it committed some classification errors ($<3\%$). As expected, the T-R_{post} approach committed *no* classification errors on the training data. As in the previous experiment, the T-AP approach showed very poor recognition results.

Table 3
T-R_{post} recognition statistics for Walk-left, Walk-right, Run-left, and Run-right

Analysis using $O_{t_1:t_2}$	Walk-left	Walk-right	Run-left	Run-right
% Subsequences classified (non-confusing)	86.1	90.9	92.4	95.6
% Duration to initial classification point	16.2	12.0	11.2	8.3

In Fig. 14, we show the correct classifications and the number of errors for the *testing* sequences. Fewer recognition errors were committed by T-R_{post} as compared to MAP on the test sequences, and T-AP was unable to classify most of the sequences.

5.4. Discussion of results

In our experiments, we evaluated different temporal scenarios for action recognition. In the first experiment with single frames for actions Walk, Run, and Stand, we showed how a forced classification technique such as MAP will result in several erroneous classifications (3–15%), while the reliable-inference approach can be used to avoid these errors but in turn produces high confusion rates (12–71%) due to the lack of sufficient distinguishing information.

We then experimented with subsequences of actions, where in the first experiment with actions Bend, Crouch, and Sit, we demonstrated how only a short video exposure is sufficient to disambiguate the classes. The proposed T-R_{post} method successfully identified (and ignored) the confusing standing phase in all of the actions (initial 18–30% of each action) and avoided most of the errors committed by MAP. The usefulness of thresholding with the R_{post} metric (i.e. T-R_{post}) over the similarly thresholded *a posteriori* measure (i.e. T-AP) was also demonstrated. Moreover, the advantage of using temporal information versus single frame classification was shown in another experiment with the Bend, Crouch, and Sit dataset.

Finally, similar results were obtained for an even more complex scenario of recognizing walking and running human action classes from different directions where the subsequences were allowed to start at any point of time during the action cycle. Again, most of the confusions were found in the smaller subsequences (8–16% of full cycle duration on an average) using the T-R_{post} method. The competitive MAP and T-AP methods were shown to have higher classification errors and lower detection rates, respectively.

Though the approach was successful with several simple human actions (including Bend, Crouch, Sit, Walk, and Run), it is worthwhile to note certain limitations with the present method. First, we currently use a separate HMM model for each view direction of each action, which for several actions/views will require training a large number of HMMs. This problem is common for most view-based methods, and possible solutions to reduce the number of HMMs could include the use of 3D models or view-invariant features (though at a much higher computational cost). Another issue not addressed with the current framework is the recognition of *nested* actions that contain sub-action phases that are the same as another action class to detect. For example, consider an crouching action class and a 'free-throw' basketball action class (where the player moves to the free-throw line, crouches down, moves/stretches up, and throws the basketball). In this example, a crouching action is nested inside (a part of) the longer free-throw action. Thus during training, an extremely high reliability threshold will be set for the isolated crouching class, as certain subsequences of the free-throw actions appear identical to the crouching actions (causing

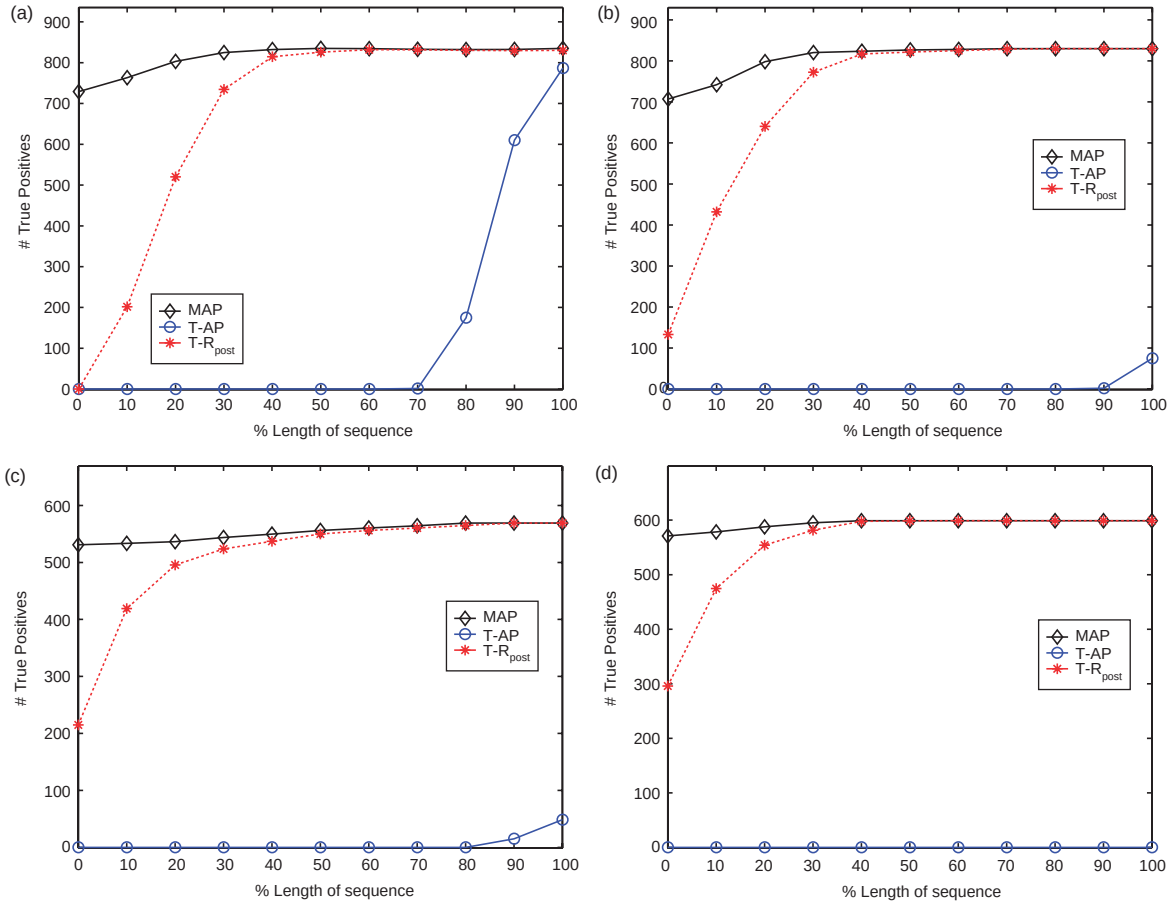


Fig. 13. Number of examples detected using MAP, T-AP, and T-R_{post} approaches for different subsequence lengths of training examples: (a) Walk-left, (b) Walk-right, (c) Run-left, and (d) Run-right.

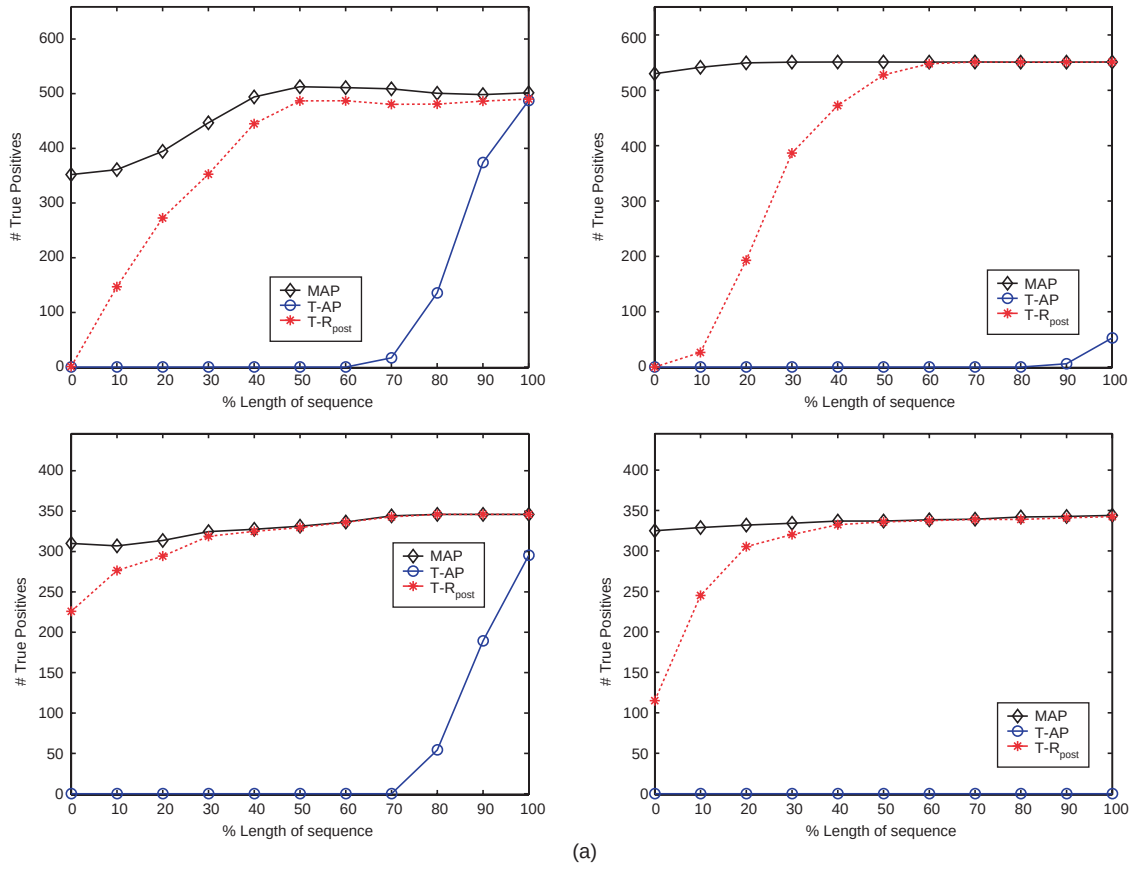
most true crouching examples to be unreliable). However, the free-throw model will only be slightly affected. Lastly, the use of standard HMMs can be applicable to several types of actions, but other more complex activities with phase variability (e.g. performing an activity one of many ways or in different order) and multi-agent interactions may be more appropriately modeled with coupled [8], hierarchical [16], or switching [38] HMMs or Bayesian networks [27]. However, if the activities can be probabilistically modeled, then the reliable-inference approach can be employed.

6. Framework extensions

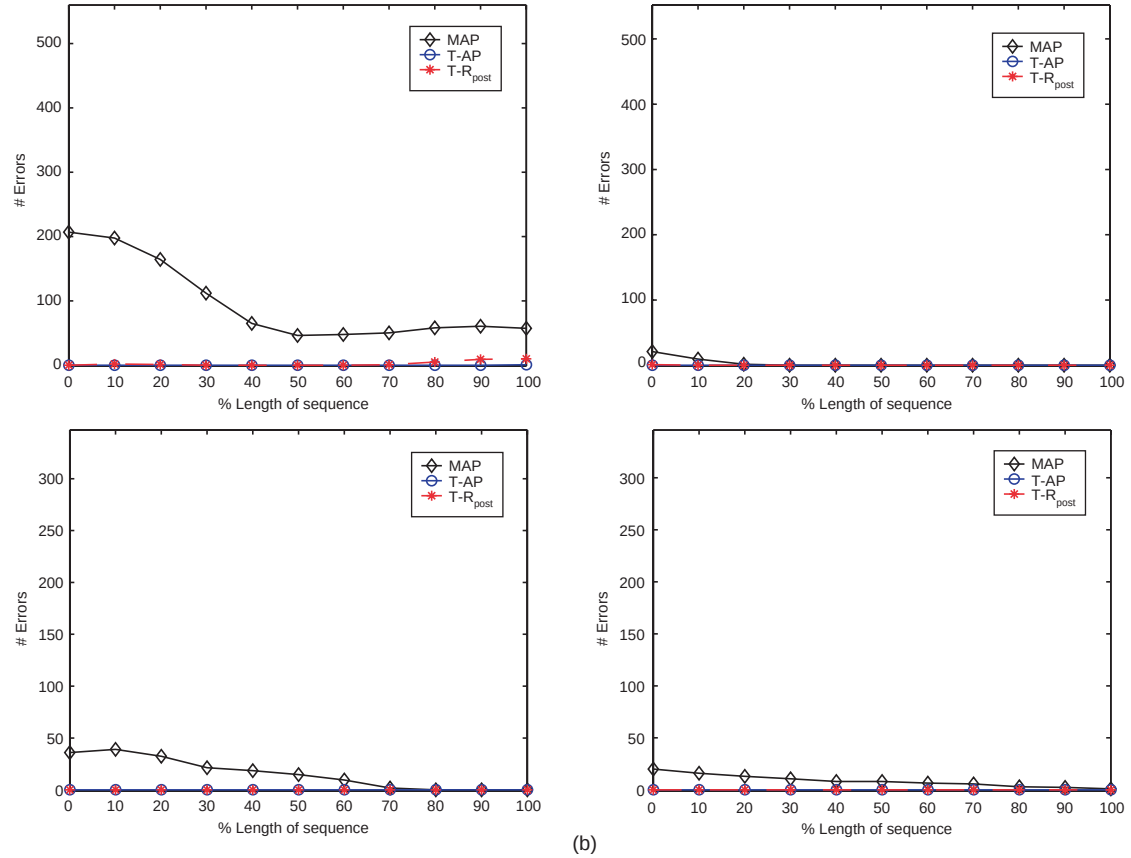
The R_{post} formulation is defined over a closed-world context of actions, but there will be situations when the input does not belong to any of the action classes in the context. In such cases, we should refrain from even calculating the R_{post} values and attempting any classification. An input that violates the expected context should never trigger the RI calculations. A context-checking ability can be incorporated into the framework by having a threshold to which we can simply compare the likelihood of the input to determine if it is truly 'in the context' before attempting a reliable-inference. Furthermore, for an input that is found to be in context, after a reliable decision is assigned we need only to 'follow' (continually verify) this action label as more frames are added rather than computing the R_{post} for *all* classes repeatedly for each new frame. Additionally, any violation during the following stage should reset the entire process (to validate the context). An example state model capturing this extended framework is shown in Fig. 15.

For any new sequence, the likelihood of each frame (independently) is first checked for context validity by comparing it to a likelihood threshold (γ_i) for each action class $\mathcal{A}_i$. The context threshold γ_i for class $\mathcal{A}_i$ can be computed with $\gamma_i = \mu - k\sigma$, where μ and σ are the mean and standard deviation of the likelihood values of the frames belonging to class $\mathcal{A}_i$ and k defines the lower threshold bound (e.g. $k = 2.5$). Once a frame from the sequence comes into context (the likelihood is above at least one γ_i), the RI framework is engaged and used to find the earliest reliable classification point for the incoming video frames (while continually checking for context violations and resetting if necessary). Once a class commitment is made by RI using T-R_{post}, the action is then continually followed by comparing the incoming frames to a new temporally constrained likelihood threshold (τ_i) for the detected action. As before, any violation resets the system to the initial context validation state.

During following, if we compute the compounded likelihoods generated for the new incoming frames using the HMM



(a)



(b)

Fig. 14. Plots in (a) show the number of correct classifications and plots in (b) report the number of errors detected using MAP, T-AP, and T-R_{post} approaches for different subsequence lengths of testing examples for Walk-left, Walk-right, Run-left, and Run-right, respectively.

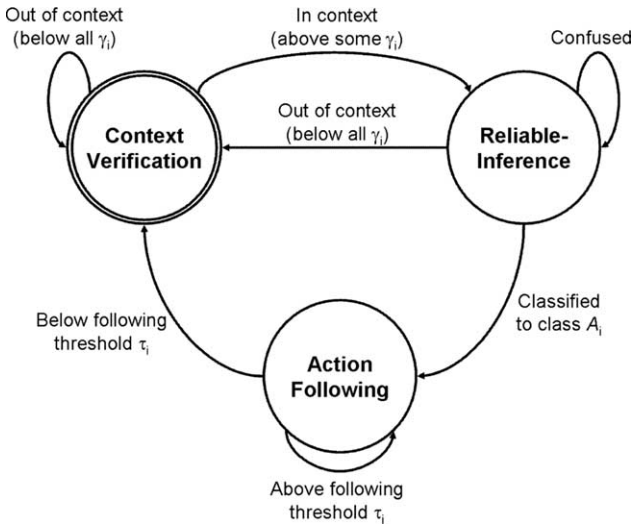


Fig. 15. State model for context verification→reliable-inference→action following.

for the detected action, unfortunately we may require a substantial amount of invalid frames before the entire sequence likelihood falls below a certain threshold (as the initial sequence portion examined in RI may produce a very high likelihood). On the other hand, just computing only the likelihood of each frame individually may help to detect deviations much sooner, but we loose any required temporal relationships of the frames. For example, if we are following a walk action, and the person comes to a stop (in a standing pose), then even though the standing poses will individually have a valid likelihood to the walk model, we desire the following module to identify this change in behavior and stop classifying the action as walking.

One potential solution to the action following task is to compute the individual frame likelihoods, but also penalize the likelihood for each frame by the time duration since the last HMM state transition. For a given HMM and a new frame O_t, we obtain the likelihood value of O_t for each HMM state. We then remove any likelihood values for those states which cannot be reached by a transition from the previously selected state (for O_{t-1}). The frame likelihood for O_t is chosen as the maximum likelihood of the valid states, and O_t is assigned to state s, which produced that maximum valid likelihood. We also retain the memory of the time d spent in state s since transitioning into it and weight the likelihood probability of O_t with t_{ss}^d, where t_{ss} is the self-loop probability of the selected state s. This weighting term ensures that a particular frame/pose only occurs for the expected duration (as learned from training examples). The following threshold τ_i for action class $\mathcal{A}_i$ can be computed in a similar manner as the context threshold, using $\tau_i = \mu - k\sigma$, where now μ and σ are the mean and standard deviations of the *penalized* frame likelihood values of the positive training examples.

7. Summary and conclusion

We presented a probabilistic reliable-inference approach to rapid action recognition. The method 'confirms the reliability' of an input subsequence before making any classification commitment. If the subsequence is deemed unreliable, then more video frames are collected in the hopes to remove the confusion. The proposed framework classifies an input to a particular action $\mathcal{A}$ only if it has a *relatively* higher posterior probability for $\mathcal{A}$ as compared to other all other actions $\neg\mathcal{A}$ in the context C. The measure of relative strength of one class over the others is quantified by the R_{post} metric that forms the ratio of class to non-class *a posteriori* probabilities. A threshold $\lambda_{\mathcal{A}_i}$ is learned for each class $\mathcal{A}_i \in C$ using the $\neg\mathcal{A}$ training data to ensure that all non-class training examples in $\neg\mathcal{A}_i$ are correctly rejected.

To classify a novel (unknown) action subsequence $O_{t_1:t_2}$, first all of the $R_{\text{post}}[\mathcal{A}_i, O_{t_1:t_2}]$ values are calculated $\forall \mathcal{A}_i \in C$. We then select the class with the highest R_{post} from amongst those classes whose R_{post} values are above their respective class thresholds. If a given subsequence is deemed unreliable for all actions (no acceptable R_{post} values), we then examine longer sequences (if possible) to include more information until a reliable decision can be made. The advantage of the proposed approach is that instead of forcing a possibly erroneous classification for brief video exposures or requiring fixed-length or long-duration videos, the method re-evaluates extended subsequences (with additional frames) only until the ambiguity is resolved.

We evaluated our framework with experiments using different video exposure lengths of several common human actions. The experimental results with the method analyzing Walking, Running, and Standing at multiple views from individual frames/poses showed that single images could only be used in a very limited manner to discriminate the actions, and that many frames are unreliable for classification. In the task of reliable action recognition from subsequences, experiments with multiple action sets (Bend, Crouch, Sit, Walk-left, Walk-right, Run-left, Run-right) demonstrated how our thresholded R_{post} method (T-R_{post}) could be used to successfully classify the actions by observing only a small part of the action sequence (on average) while committing fewer errors on test data than with a MAP classification. Furthermore, the use of our recognition method was shown to outperform a similarly thresholded a posterior method (T-AP). In all cases, our method converged to high detection rates while *avoiding* most of the classification errors associated with MAP.

In future work, we plan to investigate alternative threshold selection methods and to further examine the generalization of the models and combined validation/inference/following framework with new test data. We also plan to examine other multi-phase actions, and employ the approach for selecting optimal camera views for discriminating different actions. We expect that rapid-and-reliable action recognition methods, such as the proposed RI framework, will become an area of increased study, and that these approaches will be used to increase the responsiveness of real-time surveillance and interactive systems.

Acknowledgements

This research was supported in part by the National Science Foundation under grant No. 0236653.

Appendix A. Equivalence of MAP and maximum R_{post}

Consider $N+1$ classes $A^*, A_1, \ldots, A_N$ (note: class A^* can be any one of the $N+1$ action classes, but we name it differently for the notational convenience of the proof). The corresponding R_{post} values of feature f to these classes are (in all subsequent cases, $1 \le i \le N$)

$$R_{\text{post}}[A^*,f] = \frac{p(A^*|f)}{\sum_{j=1}^{N} p(A_j|f)} \tag{A1}$$

$$R_{\text{post}}[A_i,f] = \frac{p(A_i|f)}{p(A^*|f) + \sum_{j=1,j\neq i}^{N} p(A_j|f)} \tag{A2}$$

where:

$$p(A^*|f) > 0, \quad p(A_i|f) > 0 \tag{A3}$$

Note that we have dropped the explicit reference to the context C for notational convenience. Given the following relation holds, MAP will choose class A^* as the classification for the given feature f if

$$p(A^*|f) \ge p(A_i|f) \tag{A4}$$

We can manipulate Eq. (A4) as follows:

$$p(A^*|f) \ge p(A_i|f) \tag{A5}$$

$$\frac{p(A^*|f)}{p(A^*|f) + \sum_{j=1}^{N} p(A_j|f)} \ge \frac{p(A_i|f)}{p(A^*|f) + \sum_{j=1}^{N} p(A_j|f)} \tag{A6}$$

$$\frac{p(A^*|f) + \sum_{j=1}^{N} p(A_j|f)}{p(A^*|f)} \le \frac{p(A^*|f) + \sum_{j=1}^{N} p(A_j|f)}{p(A_i|f)} \tag{A7}$$

$$1 + \frac{\sum_{j=1}^{N} p(A_j|f)}{p(A^*|f)} \le 1 + \frac{p(A^*|f) + \sum_{j=1,j\neq i}^{N} p(A_j|f)}{p(A_i|f)} \tag{A8}$$

$$\frac{\sum_{j=1}^{N} p(A_j|f)}{p(A^*|f)} \le \frac{p(A^*|f) + \sum_{j=1,j\neq i}^{N} p(A_j|f)}{p(A_i|f)} \tag{A9}$$

$$\frac{p(A^*|f)}{\sum_{j=1}^{N} p(A_j|f)} \ge \frac{p(A_i|f)}{p(A^*|f) + \sum_{j=1,j\neq i}^{N} p(A_j|f)} \tag{A10}$$

$$R_{\text{post}}[A^*,f] \ge R_{\text{post}}[A_i,f] \tag{A11}$$

As shown above, Eq. (A5) implies Eq. (A11), and hence the maximum R_{post} will also select class A^* as the classification for the given feature f. Therefore, in all cases both MAP and maximum R_{post} result in the same classification.

References

[1] J. Aggarwal, Q. Cai, Human motion analysis: a review, in: Nonrigid and Articulated Motion Workshop IEEE, 1997, pp. 90–102.
[2] L. Arslan, J. Hansen, Likelihood decision boundary estimation between HMM pairs in speech recognition, IEEE Trans. Speech Audio Process. 6 (4) (1998) 410–414.
[3] C.W. Baum, V.V. Veeravalli, A sequential procedure for multihypothesis testing, IEEE Trans. Inform. Theory 40 (6) (1994) 1994–2007.
[4] M. Bicego, V. Murino, Investigating hidden markov models' capabilities in 2D shape classification, IEEE Trans. Pattern Anal. Machine Intell. 26 (2) (2004) 281–286.
[5] A. Biem, A model selection criterion for classification: application to HMM topology optimization, in: Proceedings of the International Conference Doc Analysis and Recognition IEEE, vol. 1, 2003, pp. 104–108.
[6] A. Bobick, J. Davis, The recognition of human movement using temporal templates, IEEE Trans. Pattern Anal. Machine Intell. 23 (3) (2001) 257–267.
[7] M. Brand, V. Kettnaker, Discovery and segmenation of activities in video, IEEE Trans. Pattern Anal. Machine Intell. 22 (8) (2000) 844–851.
[8] M. Brand, N. Oliver, A. Pentland, Coupled Hidden Markov Models for complex action recognition, in: Proceedings of the Computer Vision and Pattern Recognition, 1997, pp. 994–999.
[9] I. Chang, C. Huang, The model-based human body motion analysis system, Image Vis. Comput. 18 (14) (2000) 1067–1083.
[10] R. Cutler, L. Davis, Robust real-time periodic motion detection, analysis, and applications, IEEE Trans. Pattern Anal. Machine Intell. 22 (8) (2000) 781–796.
[11] J. Davis, Visual categorization of children and adult walking styles, in: Proceedings of the International Conference Audio- and Video-based Biometric Person Authentication, 2001, pp. 295–300.
[12] J. Davis, Sequential reliable-inference for rapid detection of human actions, in: Proceedings of the Workshop on Detection and Recognition of Events in Video, 2004.
[13] J. Davis, A. Tyagi, A reliable-inference framework for recognition of human actions, in: Advanced Video and Signal Based Surveillance, IEEE, 2003, pp. 169–176.
[14] A. Dempster, N. Laird, D. Rubin, Maximum likelihood from incomplete data via the EM algorithm,, J. R. Stat. Soc. B 39 (1977) 1–38.
[15] R. Duda, P. Hart, D. Stork, Pattern Classification, Wiley, New York, 2001.
[16] S. Fine, Y. Singer, N. Tishby, The hierarchical hidden markov model: analysis and applications, Mach. Learn. 32 (1) (1998) 41–62.
[17] D. Gavrila, The visual analysis of human movement: a survey, Comput. Vis. Image Understanding 73 (1) (1999) 82–98.
[18] D. Gavrila, Pedestrian detection from a moving vehicle, in: Proceedings of the European Conference Computer Vision, 2000, pp. 37–49.
[19] M. Hu, Visual pattern recognition by moment invariants, IRE Trans. Inform. Theory IT-8 (2) (1962) 179–187.
[20] A. Jepson, W. Richards. What makes a good feature?, in: Spatial Vision in Humans and Robots, Cambridge University Press, 1991, pp. 89–125.
[21] J. Lee, J. Chai, P. Reitsma, J. Hodgins, N. Pollard, Interactive control of avatars animated with human motion data, in: Proceedings of the SIGGRAPH, 2002, pp. 491–500.
[22] G. Johansson, Spatio-temporal differentiation and integration in visual motion perception, Psychol. Res. 38 (1976) 379–393.
[23] A. Lipton, H. Fujiyoshi, R. Patil, Moving target classification and tracking from real-time video, in: Proceedings of the Workshop Applications of Computer Vision, 1998, pp. 8–14.

[24] J. Little, J. Boyd, Recognizing people by their gait: the shape of motion, Videre 1 (2) (1998) 2–32.
[25] F. Liu, R. Picard, Finding periodicity in space and time, in: Proceedings of the International Conference Computer Vision IEEE, 1998, pp. 376–383.
[26] Y. Liu, R. Collins, Y. Tsin, Gait sequence analysis using frieze patterns, in: Proceedings of the European Conference Computer Vision, 2002, pp. 657–671.
[27] N. Moenne-Loccoz, F. Bremond, M. Thonnat, Recurrent bayesian network for the recognition of human behaviors from video, in: Proceedings International Conference Computer Vision Systems, 2003, pp. 68–77.
[28] S. Niyogi, E. Adelson, Analyzing and recognizing walking figures in XYT, in: Proceedings of the Computer Vision and Pattern Recognition IEEE, 1994, pp. 469–474.
[29] M. Oren, C. Papageorgiou, P. Sinha, E. Osuma, T. Poggio, Pedestrian detection using wavelet templates, in: Proceedings of the Computer Vision and Pattern Recognition IEEE, 1997, pp. 193–199.
[30] R. Polana, R. Nelson, Low level recognition of human motion, in: Workshop on Motion of Nonrigid and Articulated Objects, IEEE Computer Society, 1994, pp. 77–82.
[31] L. Rabiner, A tutorial on Hidden Markov Models and selected applications in speech recognition, Proc. IEEE 77 (1989) 257–286.
[32] R. Rosales, S. Sclaroff, Inferring body pose without tracking body parts, in: Proceedings of the Computer Vision and Pattern Recognition, IEEE, 2000, pp. 721–727.
[33] G. Schwarz, Estimating the dimension of a model, Ann. Stat. 6 (2) (1978) 461–464.
[34] C. Therrien, Decision Estimation and Classification, Wiley, New York, 1989.
[35] P. Viola, M. Jones, D. Snow, Detecting pedestrians using patterns of motion and appearance, in: Proceedings of the International Conference Computer Vision, 2003, pp. 734–741.
[36] A. Wald, Sequential Analysis, Wiley, New York, 1947.
[37] L. Wang, W. Hu, T. Tan, Recent developments in human motion analysis, Pattern Recognit. 36 (2003) 585–601.
[38] P. Wang, Q. Ji, Multi-view face tracking with factorial and switching HMM, in: Proceedings Workshop Applications of Computer Vision, 2005, pp. 401–406.