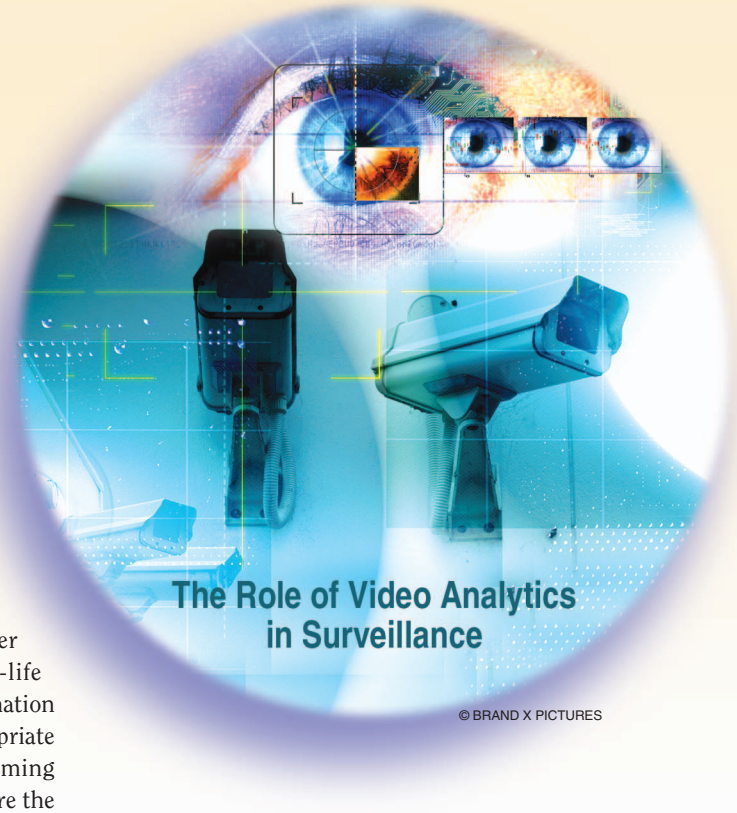
Dimitrios Kosmopoulos and Sotirios P. Chatzis

Robust Visual Behavior Recognition

A framework based on holistic representations and multicamera information fusion



In this article, we propose a novel framework for robust visual behavior understanding, capable of achieving high recognition rates in demanding real-life environments and in almost real time. Our approach is based on the utilization of holistic visual behavior understanding methods, which perform modeling directly at the pixel level. This way, we eliminate the world representation layer that can be a significant source of errors for the modeling algorithms. Our proposed system is based on the utilization of information from multiple cameras, aiming to alleviate the effects of occlusions and other similar artifacts, which are rather common in real-life installations. To effectively exploit the acquired information for the purpose of real-time activity recognition, appropriate methodologies for modeling of sequential data stemming from multiple sources are examined. Moreover, we explore the efficacy of the additional application of semisupervised learning methodologies, in an effort to reduce the cost of model training in a completely supervised fashion. The performance of the examined approaches is thoroughly evaluated under real-life visual behavior understanding scenarios, and the obtained results are compared and discussed.

INTRODUCTION

Event understanding in video sequences is a research field that has rapidly gained momentum over the last few years. This is mainly due to its fundamental applications in automated video indexing, virtual reality, human-computer interaction, assisted living, and smart monitoring. Recently, we have seen an increasing need for assisting and extending the capabilities of human operators in remotely monitored large and complex spaces such as public areas, airports, railway stations, parking lots, and industrial plants.

Digital Object Identifier 10.1109/MSP.2010.937392

Several systems have been presented in the past aiming to cover these needs (see for example the survey [1]); however, the dire fate of most of them has been to remain prototypes deployed in laboratories. To develop visual behavior classification systems that can work in real environments, much more research effort is required towards the resolution of the following problems:

- How can we extract reliable and representative features of tractable dimensionality that will by-pass the error-prone detectors and trackers?
- How can we model highly diverse and complex behaviors that will be more tolerant to noise and outliers?
- How can we exploit camera networks that provide a wider coverage of the scene and redundant data that help solve occlusions and improve accuracy?
- How can we efficiently build reliable behavior models without having to annotate large amounts of data?

In this article, we try to handle all these issues in a unified framework, aiming to tackle all the significant pitfalls that

1053-5888/10/$26.00©2010IEEE

plague existing systems, which despite their consideration as "state of the art," are usually confronted with a big failure when deployed anywhere else than highly controllable laboratory environments. We introduce a novel system for robust visual behavior recognition, capable of achieving in real-time decent recognition rates in real-life installations, based on a holistic representation of the raw input data, and hidden Markov model (HMM)-based statistical pattern recognition methodologies.

EVENT UNDERSTANDING IN VIDEO SEQUENCES IS A RESEARCH FIELD THAT HAS RAPIDLY GAINED MOMENTUM OVER THE LAST FEW YEARS.

More specifically, raw input data are modeled using holistic visual features extracted at the image level, which are further used to associate events and behaviors with temporal patterns. These features bypass the commonly used intermediate representation of the physical world (e.g., objects that have to be tracked) and thus avoid dealing with the very challenging process of tracking. In the sequel, the extracted information is modeled by means of a statistical methodology appropriate for modeling sequential data. We examine various alternatives, and we show that the HMM is the most effective solution for this purpose, as it allows for

- a better handling of outliers (which are rather typical in our setting due to occlusions, illumination changes etc), by being endowed with a student-t observation model
- the effective utilization of information stemming from multicamera configurations, by application of well-studied HMM-based information fusion schemes; this way, we can outbalance the modeling limitations of holistic features regarding the case where occlusions are present, thus allowing for the achievement of high recognition rates in the considered challenging environments.

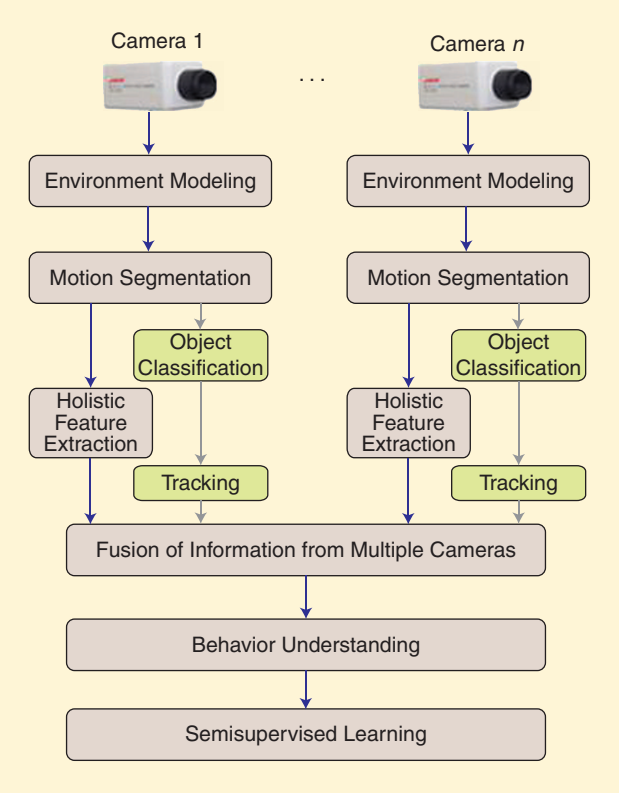
Furthermore, we show that the examined behavior models can be further evolved by additional application of semisupervised learning methods; this is important for reducing the manual annotation effort required for totally supervised model training, hence making installation of our system more cost-effective in real-life settings.

We deploy our system in the premises of a European automobile manufacturer plant under real conditions, and we provide comparative results of modeling several assembly tasks that take place in this challenging real-life environment.

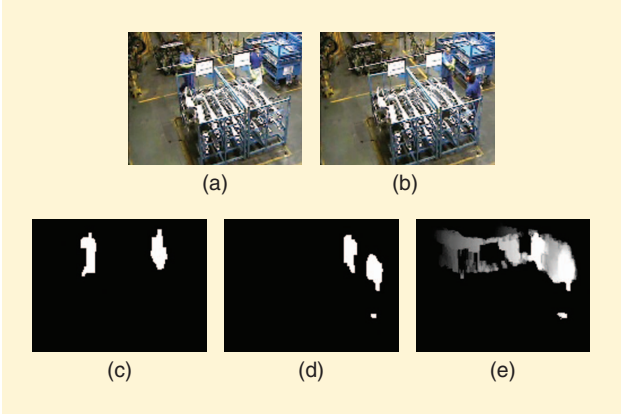
A concise flow diagram of the proposed visual behavior understanding system is presented in Figure 1. The structure of our presentation follows this diagram. The first functional procedure of our system is environment modeling, i.e., the creation of a model of what belongs to the scene, as opposed to the actors or moving objects entering and leaving the scene that are identified in the next step of motion segmentation. This problem has been treated in the literature, e.g., see [2], where the background pixels are modeled by Gaussian mixtures. The moving objects are then identified by calculating their distance from the model.

The next processing step is the extraction of features for the effective representation of the raw input data (followed by a dimensionality reduction step for the obtained feature vectors); we shall elaborate on this procedure in the section "Raw Data Representation: Why Choose Holistic Features Directly at the Pixel Level?" The resulting camera-specific information streams are input to classifiers capable of modeling and recognizing time series, as we shall describe in the section "Why Use HMMs to Model the Extracted Holistic Representation of the Captured Video Sequences?" The recognition results of the deployed system on development data sets may be used to further improve the behavior models in a semisupervised fashion. Moreover, fusion of multiple information streams is possible when the target is viewed by multiple cameras.

In the section "Experimental Evaluation," we provide a thorough experimental evaluation of the proposed system, considering real-life visual behavior recognition scenarios in the context of the assembly lines of a European automobile manufacturer. We compare the performance of the proposed novel approach with popular rival methodologies based on state-of-the-art tracking and person detection algorithms. Additionally, we justify the specific selection of the separate algorithms that comprise the building blocks of our system, by providing empirical evidence regarding the performance we obtain by replacing the selected algorithms with alternative approaches available in the literature.



[FIG1] Architecture of the proposed framework. The error-prone processes of object classification and tracking are bypassed by using holistic scene representation.



[FIG2] (a) and (b) show two keyframes and (c)–(e) show the respective background subtraction images and extracted PCH image.

RAW DATA REPRESENTATION: WHY CHOOSE HOLISTIC FEATURES DIRECTLY AT THE PIXEL LEVEL?

One of the key challenges real-time action recognition systems are confronted with concerns selection of appropriate features for representing the observed raw data. The ideal features should describe different actions accurately, with high discrimination capability, and should be efficiently calculated. Ideally, these features should also provide a hierarchical representation scheme (coarse to fine) so that a desirable, application-wise tradeoff between representation capabilities and computational complexity can be reached. In the following, some popular features and their applicability to behavior recognition tasks are discussed, as well as our proposed representation.

The employment of features directly extracted from the video frames has the significant advantage of obviating the need of detecting and tracking the salient scene objects, a process that is notoriously difficult in cases of occlusions, target deformations, and illumination changes. Thus, by using such an approach, the intermediate levels of semantic complexity, as met in typical bottom-up systems, are completely bypassed (see Figure 1). For this purpose, either local or holistic features (or both [3]) may be used.

An advantage of local descriptors is that their computation does not require static cameras (or a registration process to make the captured video frames comparable); however, real-life installations usually employ static cameras, hence rendering this advantage of local descriptors rather indifferent in the examined context. On the other hand, a major disadvantage of local descriptors is the significant computational burden required for their calculation. Another drawback of local descriptors is that, despite their suitability for extracting the motion patterns of tracked objects within certain regions (e.g., [4] and [5]), they are not suitable for capturing the shape of the moving objects.

Holistic features remedy these drawbacks of local features, while also requiring a much less tedious computational procedure for their extraction. Motion history images and motion energy images are among the first holistic representation methods for behavior recognition [6]. A very positive attribute of such representations is that they can easily capture the history of a task that is being executed. In [7], it was shown that pixel change history (PCH) images are able to capture relevant duration information with better discrimination performance.

The PCH of a pixel is defined as

$$P_{\varsigma,\tau}(x,y,t)=\begin{cases}\min\left(P_{\varsigma,\tau}(x,y,t-1)+\dfrac{255}{\varsigma},255\right) & \\ \qquad\qquad \text{if } D(x,y,t)=1 & \\ \max\left(P_{\varsigma,\tau}(x,y,t-1)-\dfrac{255}{\tau},0\right) & \\ \qquad\qquad \text{otherwise} & \end{cases} \tag{1}$$

where $P_{\varsigma,\tau}(x,y,t)$ is the PCH for a pixel at (x,y), $D(x,y,t)$ is the binary image indicating the foreground region, ς is an accumulation factor, and τ is a decay factor. By setting appropriate values to ς and τ we are able to capture pixel-level changes over time (see Figure 2).

In [7], the moving objects were identified by merely using differencing. A simplistic feature vector including the centroid and the major/minor axes of the overall resulting image was used; however, such a representation suffers from averaging effects in the general case where multiple moving agents are present. Thus, a shape descriptor would be most appropriate to represent the resulting PCH images.

Obviously, when dealing with real-life tasks, methods for extracting the shapes of moving agents, as well as good descriptors for those shapes are needed. Depending on the application requirements, foreground objects (instead of moving objects) may be considered important; foreground objects can be extracted in real time by standard foreground segmentation methods [2]. Zernike, pseudo-Zernike, and Hu moments are among the most popular choices as shape descriptors (see, e.g., [8]). Zernike and pseudo-Zernike moments are very attractive because of their noise resiliency, their reduced information redundancy, and their reconstruction capability.

Based on these observations, raw data representation in our system is conducted by using Zernike moments to extract the valuable information from calculated PCH images; this way, we yield a much more robust representation compared to the simplistic approach of [7]. The complex Zernike moments of order p are defined on a PCH image f as

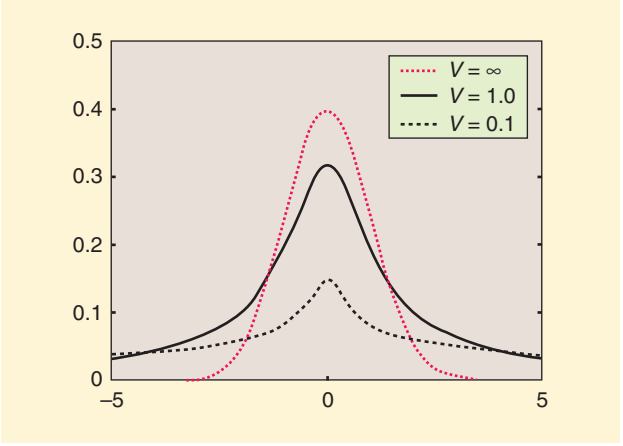
$$A_{pq}=\frac{p+1}{\pi}\int_0^1\int_{-\pi}^{\pi}R_{pq}(r)e^{-jq\theta}f(r,\theta)rdrd\theta, \tag{2}$$

where $r=\sqrt{x^2+y^2}$, and $\theta=\tan^{-1}(\mathrm{y/x})$ and $-1<x,y<1$ (x, y are the image coordinates, with respect to the center, around which the integration is calculated) and

$$R_{pq}(r)=\sum_{s=0}^{\frac{p-q}{2}}(-1)^s\frac{(p-s)!}{s!\left(\frac{p+q}{2}-s\right)!\left(\frac{p-q}{2}-s\right)!}r^{p-2s}, \tag{3}$$

where $p - q$ = even and $0 \leq q \leq p$. Moments of low order hold the coarse information while the ones of higher order hold the fine details. However, the more detailed the region representation is, the more processing power will be demanded, and thus a tradeoff has to be reached considering the specific application requirements.

As one can notice, the resulting representation of the raw captured data entails a high dimensionality of the obtained feature vectors. This could probably give rise to model training issues, affecting both their efficiency and classification performance (curse of dimensionality). Therefore, application of an additional dimensionality reduction step is required. For this purpose, common dimensionality reduction methods, such as principal component analysis (PCA), or linear discriminant analysis [9], may be used. Note, however, that the adoption of a representation based on Zernike moments, allows for the data dimensionality to be reduced merely by not considering moments of higher order.



[FIG3] The student-*t* distribution for various ν values. For $\nu \to \infty$ we yield the Gaussian distribution.

WHY USE HMMs TO MODEL THE EXTRACTED HOLISTIC REPRESENTATION OF THE CAPTURED VIDEO SEQUENCES?

One of the key functionalities of any machine learning model (classifier) suitable for application in visual behavior understanding is the ability to extract the "signature" of a behavior from the captured visual input. The key requirements when designing such a classifier are

1) to support task execution in various time scales, since a task or parts of it may have variable duration
2) to support stochastic processes, because of the task intraclass variability and noise
3) appropriate handling of outliers
4) flexible and effective exploitation of the additional information from multiple data streams.

A popular approach for sequential data modeling that fulfills the above requirements is the HMM (see, e.g., [10]). An HMM entails a Markov chain comprising a number of, say, N states, with each state being coupled with an observation emission distribution. An HMM defines a set of initial probabilities $\{\pi_k\}_{k=1}^{N}$ for each state, and a matrix A of transition probabilities between the states; each state is associated with a number of (emitted) observations $\mathbf{o}$ (input vectors). Gaussian mixture models are typically used for modeling the observation emission densities of the HMM hidden states.

Typically, HMMs are trained under the maximum-likelihood framework, by means of the expectation-maximization (EM) algorithm [10]. The HMM model size, i.e., the number of constituent states and mixture components, can affect model performance and efficiency; for this reason, several criteria have been proposed for the purpose of data-driven HMM model selection (e.g., [11] and [12]). However, for systems that are expected to operate in nearly real time, small models are generally preferable, due to their low number of parameters, hence easier learning, and considerably less computational burden for sequential data classification.

Outliers are expected to appear in data sets obtained from realistic monitoring applications due to illumination changes, unexpected occlusions, and unexpected task variations and may seriously corrupt model training results. The vast popularity of the HMM framework is partly attributed to the fact that it is flexible enough to allow for the replacement of the commonplace Gaussian mixture observation models with other ones that are more tolerant to outliers. As we have recently demonstrated, the adoption of the multivariate student-*t* distribution as the observation model of HMMs allows for the efficient handling of outliers in the context of the HMM framework without compromising overall efficiency [13].

The probability density function (pdf) of a p-dimensional student-*t* distribution with mean vector μ, positive definite inner product matrix Σ, and ν degrees of freedom is given by

$$t(x_t; \mu, \Sigma, \nu) = \frac{\Gamma\left(\frac{\nu + p}{2}\right)|\Sigma|^{-1/2}(\pi\nu)^{-p/2}}{\Gamma(\nu/2)\{1 + d(x_t, \mu; \Sigma)/\nu\}^{(\nu+p)/2}}, \quad (4)$$

where $\Gamma(.)$ denotes the Gamma function and d the Mahalanobis distance. As we observe in Figure 3, the student-*t* distribution has heavier tails compared to the Gaussian, which allows for higher tolerance to outliers. The Gaussian distribution is actually a special case of the student-*t* for $\nu \to \infty$.

Model parameters, including ν, can be automatically estimated by means of a model training algorithm, e.g., under the EM algorithm framework as described in [13].

SEMISUPERVISED LEARNING

Semisupervised learning is a term used to characterize machine learning algorithms designed to exploit both labeled and unlabeled data. The basic notion behind semisupervised learning of probabilistic classifiers is that using a limited sample of labeled data along with a big pool of unlabeled data would offer a good tradeoff between a) using only a limited sample of training data, an approach which is well known to incur a significant overfitting proneness to the trained probabilistic models, thus severely undermining the obtained pattern recognition performance; and b) acquiring large labeled training data sets, which could be extremely resource consuming, or even impossible in real-life applications.

The goal of modeling behaviors and events from holistic features can be severely undermined in real-life applications by the fact that large amounts of manually annotated data are usually required for dependable model training. Furthermore, often

machine learning systems need to be capable of adapting themselves, which is not possible in a strictly supervised learning fashion. Consequently, the application of semisupervised learning may be of significant benefit for holistic representations-based behavior recognition methods. Indeed, semisupervised learning arises as a promising solution for the reduction of the deployment costs of such systems in real-life installations, associated with the reduction of model training costs. For this reason, we incorporate semisupervised refinement of the employed HMM-based classifiers as a functional aspect of our system.

In our system, we employ one of the oldest, simplest, and most successful semisupervised learning approaches for generative probabilistic models, including HMMs, that is self-training. Indeed, self-training is a wrapper method that applies to any existing (complex) classifiers and is often used in real tasks like natural language processing. The basic notion behind the self-training method consists in the simple assumption that one's own high confidence predictions are typically correct. On the basis of this assumption, self-training a set of HMM classifiers involves the following basic steps:

1) Postulate one HMM for each class.
2) Train the models using the available training data points of each class.
3) Label the available unlabeled data points by using the trained HMMs to classify them.
4) Incorporate the unlabeled data points into the training data sets (using their estimated labels).
5) Repeat Steps 2–4 until the estimated labels of the unlabeled data points stop changing between repetitions.

Typical variations of self-training usually reflect different decisions regarding how to incorporate the unlabeled data into the training set. Common alternatives include adding a few most confident unlabeled data points to the training sets, adding all unlabeled data points to the training sets, and adding all unlabeled data points to the training sets, but weighing each one of them by a confidence metric when used in model estimation (e.g., using a homotopy method [14]).

We shall empirically demonstrate the utility and the benefits our system gains from the adoption of the semisupervised learning framework in the experimental section of this article.

EFFECTIVE EXPLOITATION OF INFORMATION CAPTURED FROM MULTICAMERA NETWORKS

One of the weaknesses of holistic image-based methods for behavior recognition is their dependence on the viewpoint, and thus their vulnerability to occlusions. This can be alleviated by deploying multiple cameras so that the occlusions are minimized by appropriately placing the cameras. Each camera input can be used to generate a stream of observations. An appropriate information fusion technique is used after the generation of the observation stream. The ultimate goal of multicamera fusion is to achieve behavior recognition results better than the results that we could attain by using the information obtained by the individual data streams (stemming from different cameras) independently of each other. In the following, we shall survey the most popular fusion methods within the HMM framework, examine their applicability with respect to camera synchronicity and configuration, and consider possible enhancements to make them more tolerant to outliers. The observations from this analysis will form the criterion for our selection of the most suitable HMM-based information fusion scheme to be used in the context of our system.

Existing approaches can be grouped into feature (or early) fusion and late fusion approaches. Feature fusion is the simplest approach; it assumes that the observation streams (sequences of feature vectors as defined in the section "Raw Data Representation: Why Choose Holistic Features Directly at the Pixel Level?") are synchronous. This synchronicity is a valid assumption for cameras that have overlapping fields of view and support synchronization. The related architecture feature-level fusion (FHMM) is displayed in Figure 4(b). Let us denote as s_t the FHMM state emitting the tth observation. Let us consider data deriving from a number of C observation streams, and denote as $\{o_{1t}, \ldots, o_{Ct}\}$ the observations at time t deriving from the available streams. Then, the full observation vector, o_t, considered by the feature fusion approach at time t, is a simple concatenation of the available individual observations

$$o_t = (o'_{ct})'_{c=1\ldots C}. \tag{5}$$

This way, the observation emission probability of the state $s_t = i$ of the fused model, when modeled as a k-component mixture model, yields

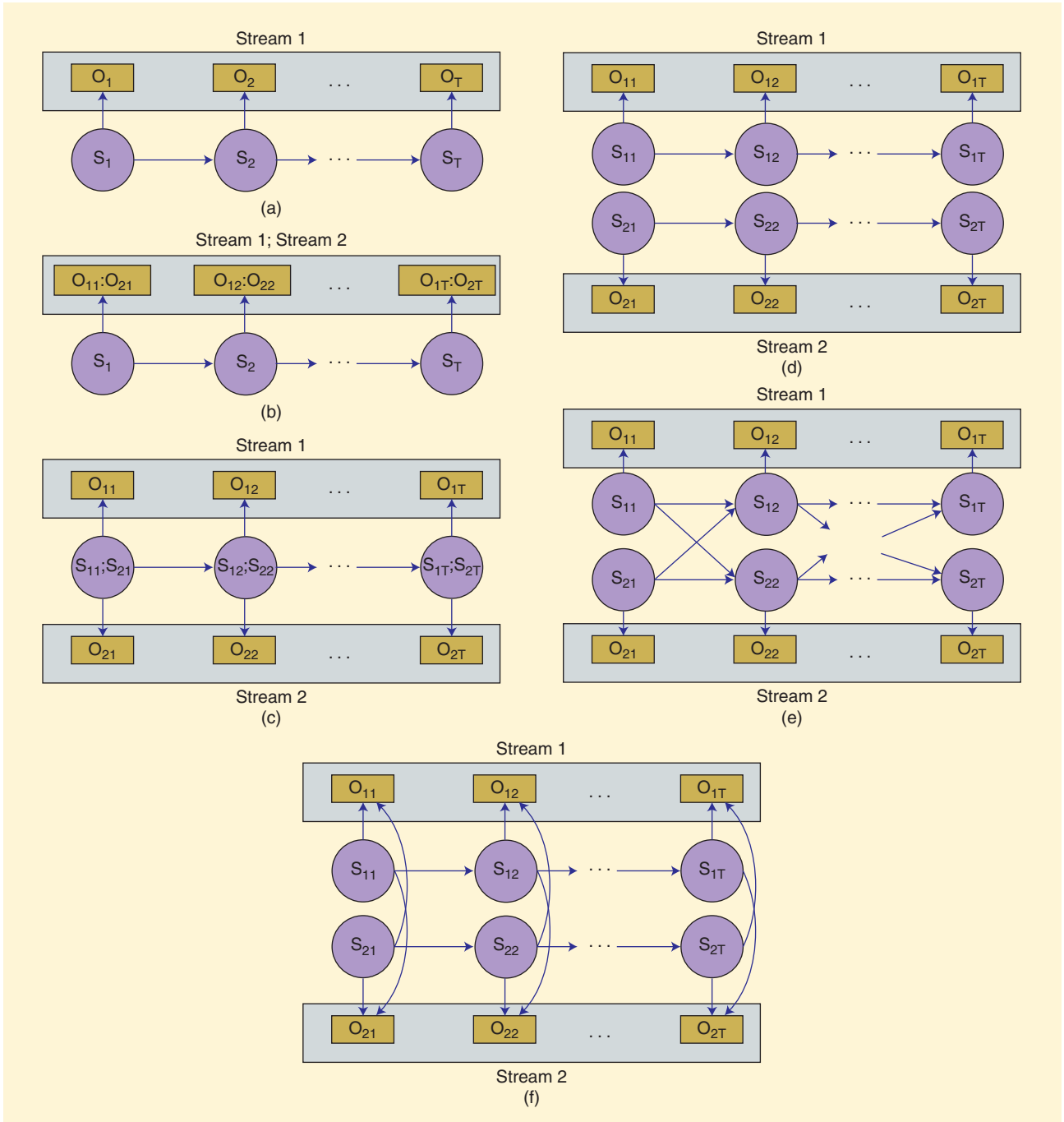
$$P(o_t|s_t = i) = \sum_{k=1}^{K} w_{ik} P(o_t|\theta_{ik}), \tag{6}$$

where w_{ik} denotes the weights of the mixture components, and θ_{ik} are the parameters of the kth component density of the ith model state (e.g., mean and covariance matrix of a Gaussian pdf).

The major limitations of the feature fusion approach lie in the fact that the simple concatenation of observations from different streams leads to high dimensionality and often fails to capture significant statistical dependencies between the different sources of information. Furthermore, it relies heavily on the assumption of a perfect synchronicity of the different data sources (synchronized cameras), which is an assumption that is often difficult to be satisfied.

In the state-synchronous multistream HMM (SHMM) [see Figure 4(c)] the streams are also assumed to be synchronized. Each stream though is modeled using an individual HMM. The postulated stream-wise HMMs share the same state dynamics (initial and transition probabilities of their states). Then, the likelihood of one multistream observation (say, at time t) is computed as the product of the observation likelihood of each constituent stream c raised to an appropriate positive stream weight r_c [15]

$$P(o_t|s_t = i) = \prod_{c=1}^{C} \left[\sum_{k=1}^{K} w_{ikc} P(o_{ct}|\theta_{ikc}) \right]^{r_c}, \tag{7}$$



[FIG4] (a)–(f) Various fusion schemes using the HMM framework for two streams. The s, o stand for the states and the observations, respectively. The first index marks the stream and the second the time.

where w_{ikc} and θ_{ikc} are the parameters of the cth-stream HMM. The weight r_c is associated with the reliability of the information carried by the cth stream. For example, a camera that does not capture the moving target very well due to occlusions should be weighted less. This way, the strong dependency of the feature fusion approach on highly reliable synchronized data streams is relaxed.

Despite its obvious advantages over feature fusion, the SHMM still relies on the assumption of synchronized data streams. Nevertheless, this assumption can be very restrictive in the context of visual behavior understanding. An alternative that assumes that the observation streams are independent of each other is the parallel HMM (PHMM) [16] [see Figure 4(d)]. This HMM-type model can be applied to cameras (or other sensors) that may not be synchronized and may operate at different acquisition rates. A PHMM does also comprise a number of component stream-wise HMMs, independently trained of one another. Similar to the synchronous case, each stream c may

have its own weight r_c depending on the reliability of the source. As a consequence of this construction, the PHMM suffers from the major disadvantage of tending to neglect any dependencies on the state level between the observation streams. Several architectures that ameliorate this issue have been presented in the past; two popular examples are the product HMM [17], and the coupled HMM (CHMM) [18].

The product HMM is used to model streams that exhibit behavior that lies between full synchronization (SHMM) and independent streams (PHMMs). It can be used for multicamera configurations for which the degree of asynchrony can not be easily inferred. Training of product HMMs is conducted by means of a special variant of the EM algorithm, suitable for this model [17]. It has to be noted though that the computational requirements of this model are considerably high.

The CHMM is designed for data comprising only two streams, thus it is not a choice for scalable systems. It assumes dependency of the "current" emitting state of the observations pertaining to each one of the streams on the "previous" state of the other observations stream [see Figure 4(e)]. This can be applied to nonoverlapping camera configurations, where, for example, the target enters the view field of one camera after leaving the view field of another one. However, this very property is the ultimate limitation of this method, which makes it inappropriate for the problem we aim to tackle.

The multistream-fused HMM (MFHMM) is another method recently proposed for multistream data modeling [19] [see Figure 4(f)]. Unlike the product HMM, the connections between the component stream-wise HMMs of this model are chosen based on a probabilistic fusion model, which is optimal according to the maximum entropy principle and a maximum mutual information criterion for selecting dimension-reduction transforms [19]. Specifically, if we consider a set of multistream observations $O = \{o_t\}_{t=1}^{T}$, with $o_t = \{o_{ct}\}_{c=1}^{C}$, and $o^c = \{o_{ct}\}_{t=1}^{T}$, the MFHMM models this data based on the fundamental assumption

$$P(O) = \frac{1}{C}\sum_{c=1}^{C} P(o^c)\prod_{r\neq c} P(o^r|\hat{s}_c), \tag{8}$$

where $\hat{s}_c$ is the estimated hidden sequence of emitting states that corresponds to the cth stream observations, obtained by means of the Viterbi algorithm, $P(o^c)$ is the observation probability of the cth stream-observed sequence, and $P(o^r|\hat{s}_c)$ is the coupling density of the observations from the rth stream with respect to the states of the cth stream model

$$P(o^r|\hat{s}_c) = \prod_{t=1}^{T} P(o_{rt}|\hat{s}_{ct}). \tag{9}$$

The probabilities $P(o_{rt}|\hat{s}_{ct})$ of the MFHMM can be modeled by means of mixtures of Gaussian densities, similar to the state-conditional likelihoods of the stream-wise HMMs. However, if higher tolerance to outliers is needed, student-t mixture models may be used instead of Gaussian mixtures, as also mentioned in the section "Why Use HMMs to Model the Extracted Holistic Representation of the Captured Video Sequences?" this selection can be applied to both the probability models of the stream-wise HMM states and the interstream coupling models of the MFHMM, to further enhance robustness.

Note also that for each possible value, say i, of $\hat{s}_{ct}$, i.e., for each different state of the stream-wise HMMs, a different coupling density model $P(o_{rt}|\hat{s}_{ct} = i)$ has to be postulated. Hence, if we consider K-state stream-wise HMMs, there is a total of K different finite mixture models that must be trained to model the coupling densities $P(o_{rt}|\hat{s}_{ct})$, $\forall r, c$.

We should additionally notice that the training and inference algorithms of the MFHMM are simple, with low computational requirements. EM-based training for the MFHMM is performed as follows: At first, a set of initial stream-wise HMMs is obtained, by means of the standard EM algorithm for HMMs. Subsequently, the state sequences of each stream-wise HMM that generate the corresponding training data are extracted by means of the standard Viterbi algorithm. Using this information, the coupling models comprising the postulated MFHMM can be easily trained by means of the EM algorithm, based on the definition (9). On the other hand, likelihood-based classification is also easy to perform, based on the modeling assumption (8). As we observe, this procedure merely comprises a set of simple likelihood computations with respect to the constituent stream-wise HMMs and coupling models of the postulated MFHMM, as well as an execution of the Viterbi algorithm to obtain the state sequence estimates $\hat{s}_c$.

As we observe, the MFHMM has several desirable properties when regarded in the context of the proposed system for visual behavior understanding based on the following holistic features:

1) State transitions do not necessarily happen simultaneously among the information streams, which makes the method appropriate for both synchronous and asynchronous camera networks.

2) If one of the component HMMs fails due to noise or some other reason, the rest of the constituent HMMs can still work properly.

3) It still retains the crucial information about the interdependencies between the multiple data streams, which CHMMs tend to neglect.

4) It has simple and fast training and inference algorithms.

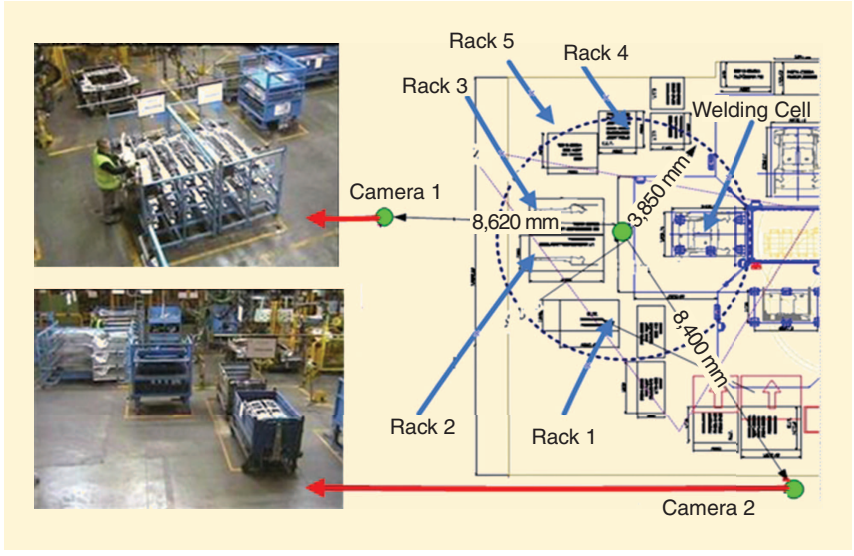
Based on these observations, we select the MFHMM as the method employed by our system to model and classify the obtained multistream sequential data (holistic representations of video frame content). We shall empirically justify the appropriateness of this selection in the experimental section of this article.

EXPERIMENTAL EVALUATION

We have experimentally verified the efficacy of the proposed approach using data obtained from a real assembly line of a European automobile manufacturer. The obtained data sets contain information pertaining to the production process of a real vehicle manufacturing facility. The workflow on this assembly line included tasks of picking several parts from racks and placing them on a designated cell some meters away where welding was performed. Each of the above tasks was regarded as

a class of behavioral patterns that had to be recognized. The information acquired from this procedure could be used for the extraction of production statistics or anomaly detection. Partial or total occlusions due to the racks made the classification task difficult to effect using a single camera; for this reason, two synchronized, partially overlapping views were used.

We evaluated both the efficacy of the proposed system, as well as the appropriateness of its component (building block) algorithms, compared to available alternative algorithms that we could have selected. Specifically, we focused on the investigation of the possible added value gained by the information fusion procedure as well as the adoption of an outlier-tolerant framework. We compared separate processing of the various information sources (streams) to application of fusion methods, and adoption of conventional observation models to the use of outlier-tolerant ones. We also investigated the profits from the application of the semisupervised learning paradigm in the context of our system.



[FIG5] Depiction of a work cell along with the position of the cameras and the Racks 1–5. The recognized behaviors are associated with transferring each part from the respective pallet and putting it on the welding cell. Additionally, the task of welding is also recognized.

SETUP

The behaviors we were aiming to model in the examined application are briefly described in the following:

1) One worker picks Part 1 from Rack 1 and places it on the welding cell.
2) Two workers pick Part 2a from Rack 2 and place it on the welding cell.
3) Two workers pick Part 2b from Rack 3 and place it on the welding cell.
4) A worker picks up Part 3a and Part 3b from Rack 4 and places them on the welding cell.
5) A worker picks up Part 4 from Rack 1 and places it on the welding cell.
6) Two workers pick up Part 5 from Rack 5 and place it on the welding cell.
7) Welding: two workers grab the welding tools and weld the parts together.

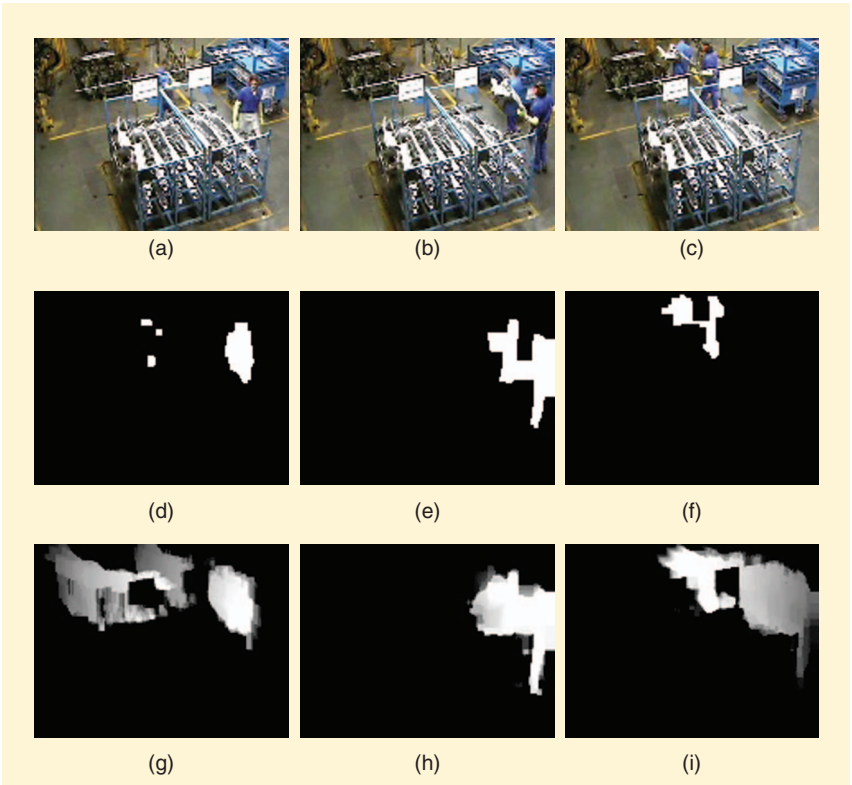
The workspace configuration and the cameras' positioning is given in Figure 5.

For our experiments, we have used two data sets, each one containing 20 segmented sequences representing full assembly cycles. Each cycle included all the seven behaviors. The total number of frames was approximately 80,000 per camera for each data set. In the first data set, the assembly process was rather well structured and was performed strictly by two people. Noisy objects were present (other persons or vehicles) but rather rare. In the second data set, which was acquired several months later, the assembly process was changed in the following sense: a third person was present quite often in the scene. That person was performing tasks in parallel to the tasks executed by the other two workers. This made the second data set much more challenging because the silhouettes got overlayed in a rather random fashion, and, hence, the motion signatures were much more difficult to model.

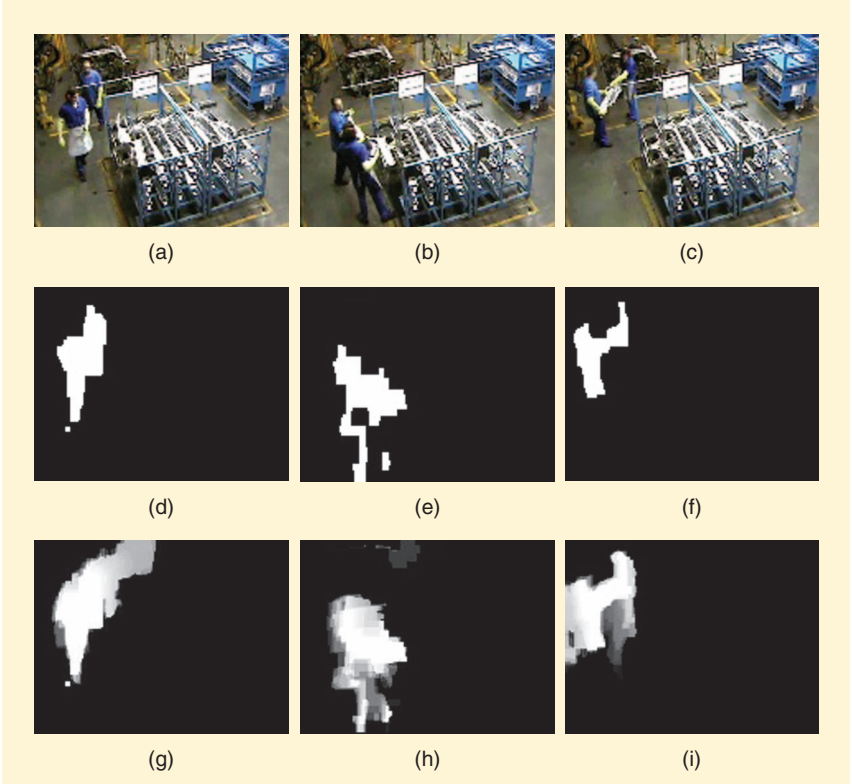
The annotation of the data sets has been done manually. Synchronization of the used IP-cameras was effected by exploiting the timestamps generated by the server our cameras were connected to. This provided a good estimate of timing, without guaranteeing perfect synchronization though, since the cameras were not hardware synchronized and were actually working independently.

In the above tasks we noticed relatively high intraclass similarity as well as high interclass dissimilarity in the PCH images (see, e.g., Figures 6 and 7 for Tasks 2 and 3, respectively). To produce the PCH images we used the blobs calculated from background subtraction. Therefore, we assumed that the motion signatures for each task could be well represented by sequences of holistic features (one feature vector per frame): we used the area, the center of gravity, and the Zernike moments (norm and phase) up to sixth order. The Zernike moments were calculated in down-scaled rectangular regions of interest (approximately 15,000 pixels) to allow for real-time performance (approximately 50-60 frames/s). From this set, we removed the values of four phases which were constant; this way, a good 31-dimensional scene representation was eventually obtained. The resulting dimensionality of the obtained feature vectors allowed for a high efficiency of the employed HMMs, when using relatively moderate model sizes.

For activity recognition we used three-state HMMs with a single mixture component per state to model each of the seven tasks described above; this was a good tradeoff between performance and efficiency. In all cases, we employed full covariance matrices for the adopted observation (mixture)



[FIG6] Key frames for the Task 2, Camera 2: (a)–(c) acquired images, (d)–(f) background subtraction, and (g)–(i) PCH images.



[FIG7] Key frames for the Task 3, Camera 2: (a)–(c) acquired images, (d)–(f) background subtraction, and (g)–(i) PCH images.

models. We trained all our models using the EM algorithm.

The stream weights employed by the considered fused models were selected based on the reliability of the individual streams. More specifically, the weight factors were roughly proportional to the classification accuracies of each individual stream (see, e.g., [19]). For more dependable results, in our experiments we used cross-validation, by repeating the employed training algorithms several times, where in each repetition all scenarios were considered except for one used for testing (leave-one-out cross-validation).

The reported accuracies were calculated as the percentages of the behavior instances that were correctly classified (for all seven tasks over all 20 scenarios—140 instances in total per data set).

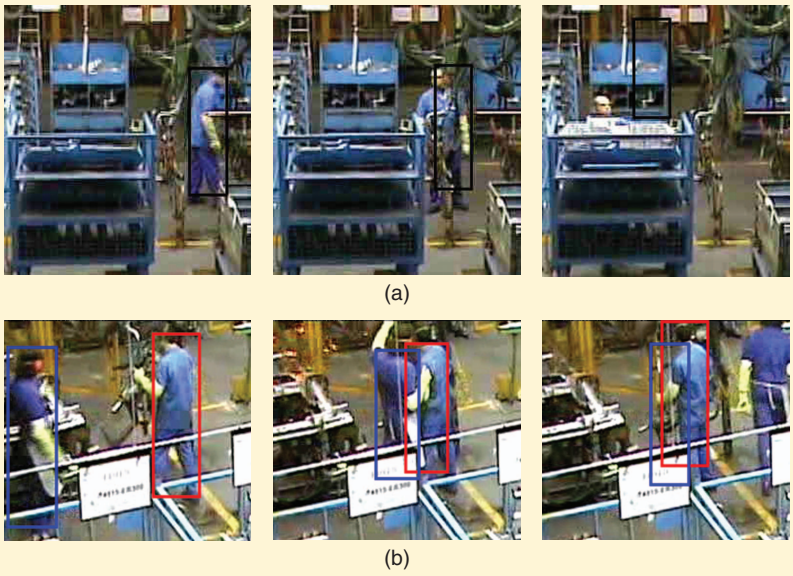
CHALLENGES FOR STATE-OF-THE-ART OBJECT-BASED METHODS

To showcase the merits of the proposed system, we initially performed experiments using popular methods that rely on object-based representations. In particular, we have tested a tracker and a prominent person detector.

The tracker was based on standard particle filtering and the employed features were the color histogram and the edges of the blobs corresponding to the human figure. Each human was represented by a rectangle (x and y position, width, and height). The measurement probability for each sample was calculated based on how well the sample fitted the model. We used the Bhattacharyya distance for histogram comparison and an exponential function to calculate the edge distance from the rectangles. More details can be found in our previous work [20]. We used no more than 100 particles to have a performance close to real time. As expected, all our study cases were extremely challenging and the tracker was losing the target very often. Therefore, no behavior recognition was meaningful. Expectably enough, the most frequent errors were due to the occlusions caused either by the racks, or by other workers due to their similar appearance (see Figure 8). The target deformations, the background clutter, and the illumination changes made the problem even more challenging. As we

found empirically, it would still remain rather unrealistic to expect consistent tracking for a whole scenario under such adverse conditions, even if we were employing more advanced tracking methods or more complex visual features.

Given the experienced unreliability of the tracking algorithms, independently seeking persons in each frame arose as a more appropriate approach. To this end, we employed a popular person detector, specifically the one presented in [21]. Its operation is based on the idea that the distribution of local intensity gradients or edge directions can often characterize local object appearance and shape relatively well, even without precise knowledge of the corresponding gradient or edge positions. Hence, by using a grid of overlapping cells and a classifier, we can decide for the existence of humans in each cell. In our experiments, we used the implementation provided by the authors of [21]; the detected rectangles were transformed to vectors using the same Zernike moments implementation as the one we used in our system. For our experimentations, we used the first available data set; the trained HMMs were employing Gaussian observation models. The accuracy was 17.85% and 56.42% for Stream 1 and 2, respectively. Occlusions were far more frequent in Stream 1 than in Stream 2, and that was reflected in the results, thus explaining the significant difference in the obtained recognition rates on the two streams. Several false positives were also observed, obviously due to the high background clutter in the scene.

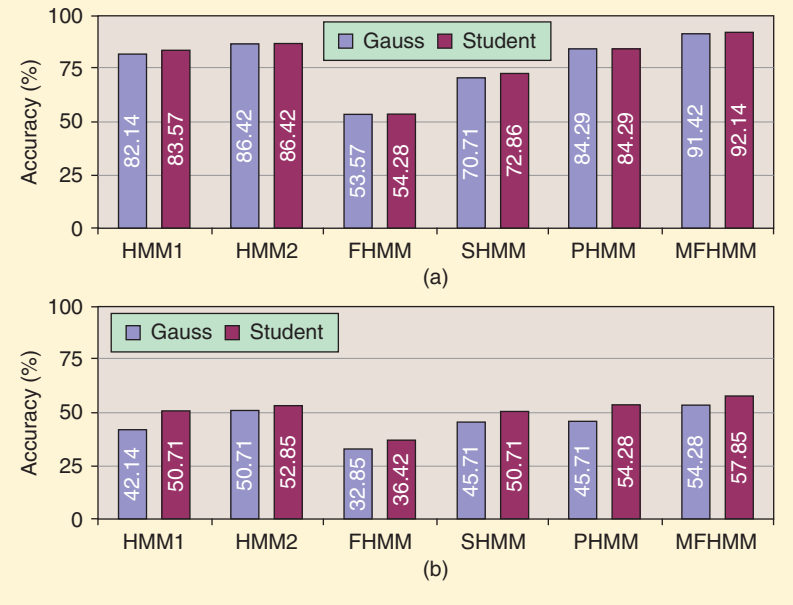


[FIG8] Typical examples of tracking failure. (a) The tracker fails as the worker gets occluded by the rack. (b) A tracker is misled by the occurrence of another worker with similar appearance.

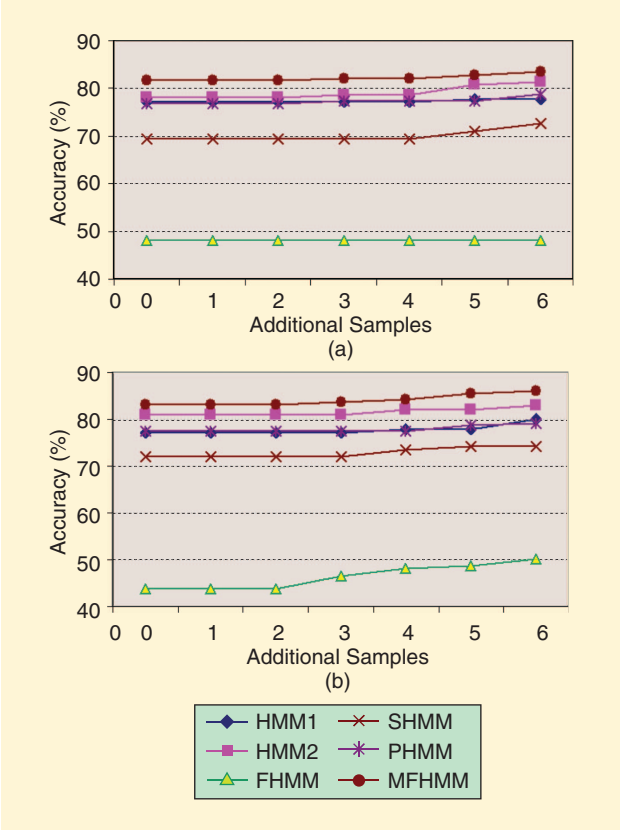
SUPERVISED LEARNING

Further, we evaluated the proposed approach in the same experimental setting. We considered application of the mainstream HMM for individual stream modeling, as well as various HMM fusion approaches, particularly the FHMM, SHMM, PHMM, and MFHMM. We experimented with the Gaussian observation model as well as with the multivariate student-t model. For the mixture model representing the interstream interactions in the context of the MFHMM, we used mixture models comprising two components.

The obtained results are given in Figure 9(a) and (b) for the first and second data sets, respectively. It becomes obvious that the sequences of holistic features and the respective HMMs represented much more successfully the well-structured assembly process (Data Set 1) than the less structured one (Data Set 2). The yielded representation is illustrated in Figures 6 and 7 for examples of the tasks two and three, from the Stream (camera) 2 of Data Set 1; the disambiguation capacity attained by using holistic representations based on PCH images is obvious. The cameras were positioned with the goal to provide complementary views. The



[FIG9] Success rates obtained using i) individual HMMs to model information from Stream 1 (HMM1); ii) individual HMMs to model information from Stream 2 (HMM2); iii) FHMM; iv) state-synchronous HMMs (SHMMs); v) parallel HMMs (PHMMs); and vi) multistream fused HMMs (MFHMM). (a) Two workers and well-defined processes. (b) Three workers with relaxed process constraints.



[FIG10] Success rates of semisupervised learning incorporating one, two, three, four, five, and six unlabeled samples of the unlabeled data into the training set using i) individual HMMs to model information from Stream 1 (HMM1); ii) individual HMMs to model information from Stream 2 (HMM2); iii) FHMMs; iv) SHMMs; v) PHMMs; and vi) MFHMMs. (a) Using the Gauss observation model. (b) Using the student-*t* observation model.

viewpoint of Camera 1 generally provided better differentiation between Tasks 4 and 6, while the performance was not as good for the other tasks, especially when trying to differentiate Task 1 from Task 5, due to occlusions. On the other hand, Camera 2 was not so heavily occluded, and apart from Tasks 4 and 6, which sometimes could look similar, it provided better views. This explains the classification performances when using the individual streams in a separate manner.

Information fusion provided results that were at least as accurate as the best stream-wise model, when implemented in the form of an MFHMM. This is due to the fact that the state interdependencies were successfully captured by the MFHMM, while no strict synchronicity was assumed. This was a significant advantage in the context of our experimental setup, since our cameras were not hardware-synchronized.

The PHMM provided accuracy slightly inferior or better, compared to the best individual stream model. However, the PHMM assumes that the streams are completely asynchronous, thus no state interdependecies could be exploited. As a consequence, the results were inferior to the MFHMM.

The accuracy deteriorated significantly when we assumed perfect synchronization by using SHMMs, or even worse when considering feature-level fusion. Obviously, such an assumption was not valid in our setup. Additionally, as expected, feature fusion was unable to exploit the correlation of the different sources, in contrast to the other methods.

Finally, the employment of the student-*t* HMM provided some extra accuracy, and thus proved its utility in visual behavior recognition applications, where outlier robustness is always of interest. The improvement was more visible in the second data set where much more noise was present.

We would like to mention the case of the PHMM in the first experiment, which gave lower accuracy than HMM2. By setting the weight of the Stream 2 to 1 and the weight of Stream 1 to 0 (instead of using the rough stream reliability as explained in the subsection "Setup") we would be yielding exactly the results of HMM2 (no fusion would be then actually affected). This outcome is justified by the fact that the PHMM employs directly the models HMM1 and HMM2. The same remark does not apply to fusion schemes which assume state coupling (e.g., the SHMM), because the respective stream-wise models generally differ from HMM1 and HMM2.

SEMISUPERVISED LEARNING

Finally, to assess the utility of semisupervised learning in the context of the considered holistic visual behavior understanding framework, we repeated one of our previous experiments under the semisupervised learning paradigm. Specifically, we considered the case of the first assembly process (two workers). We started from models trained with half our data set, further incorporating an additional 5–30% of the available samples (1–6 sequences/class) of the available "unlabeled" (test) set to conduct semisupervised learning, measuring the effect of this procedure on the obtained classification performance. For our investigations, we limited ourselves to the application of a simple self-training algorithm, with all the unlabeled samples being used having the same weight ($\lambda = 0.8$) in the model training procedure.

In Figure 10, we illustrate the obtained results. As we observe, the application of the semisupervised approach offers an increase in model performance. This is even more apparent in the case of the student-*t* HMMs, which appear to work considerably better when an additional semisupervised training procedure is employed for their learning. This behavior can be attributed to the relatively small sizes of the data sets used for model training in our experiments. Clearly, however, it does also highlight the advanced capabilities of the student-*t* observation model in data modeling and pattern recognition applications under adverse, real-world settings (in terms of the contamination of the observable data with noise and outliers).

DISCUSSION

We have experimentally explored the challenges posed to object-based methods, such as tracking or person detectors, when considering visual behavior recognition in real, complex scenes. As demonstrated, the performance of such state-of-the-art approaches was significantly lower compared to the rates obtained by the proposed system. This is mainly attributed to the

fact that the targets were often partially occluded and deformed, hence unrecognizable by these methods. However, this was not the case for our novel system, which even under this setting was still capable of providing reliable behavior signatures.

As we showed, holistic scene representation is very well applicable in monitoring and classifying rather structured processes, such as the production tasks in an assembly line. Partial occlusions or deformations were proven to be less than a problem, as long as they occur in a statistically consistent way, allowing for them to be learned by behavioral models.

An important limitation of holistic scene representation concerns the fact that no detailed object-based descriptions are possible, like those obtainable when assuming successful detection and tracking.

However, such an expectation from an automatic visual recognition system may be way too unrealistic in real-life settings. Furthermore, we saw that the less structured the modeled visual process is, the less accurate behavior recognition should be expected to be.

Moreover, holistic representations do not support, by their nature, object models that could be possibly used for target disambiguation or hypothesis evaluation. This results in higher dependency on the view point; yet, we saw that this problem can be addressed to a certain extent when multiple cameras are available, providing better views and thus giving higher task differentiation capabilities. Indeed, a flexible fusion approach, such as the MFHMM, which is capable of exploiting the correlation of information in multiple cameras, can significantly enhance the performance.

We have also seen that the employment of outlier-tolerant methods, such as the student-t observation model, appears to add significant value to holistic-based behavior recognition approaches, by mitigating the effect of illumination changes or appearance of unexpected irrelevant objects in the observed scenes.

Finally, as empirically indicated in our experimental section, a good deal of the effort imposed by the entailed labor intensive task of raw data annotation can be alleviated by using semisupervised learning.

ACKNOWLEDGMENTS

This research was partially funded by the EU project SCOVIS FP7-216465 (www.scovis.eu). The authors would also like to thank Athanasios Voulodimos for annotating the videos.

AUTHORS

Dimitrios Kosmopoulos (dkosmo@iit.demokritos.gr) received the B.Eng. degree in electrical and computer engineering from the National Technical University of Athens in 1997 and the Ph.D. degree from the same institution in 2002. Currently, he is with the Institute of Informatics and Telecommunications in the National Center for Scientific Research "Demokritos" in Athens, Greece. He also collaborates with the National Technical University of Athens, the University of Central Greece, and the Technical Educational Institute of Athens. His current research interests are in the field of computer vision, robotics, and signal processing. He has published more than 50 papers in these fields and has participated in several industrial and scientific projects as developer, consultant, or technical coordinator.

Sotirios P. Chatzis (soteri0s@me.com) received the M.Eng. degree in electrical and computer engineering from the National Technical University of Athens in 2005 and the Ph.D. degree in machine learning in 2008, from the same institution. From 2009 to 2010, he was a postdoctoral fellow with the University of Miami, United States. Currently, he is with the Department of Electrical and Electronic Engineering, Imperial College London. His current research interests are in the field of statistical machine learning with a focus on sparse Bayesian classifiers, transfer learning, reinforcement learning, preference learning, and their applications to long-term human-robot interaction.

REFERENCES

[1] W. Hu, T. Tan, L. Wang, and S. Maybank, "A survey on visual surveillance of object motion and behaviors," *IEEE Trans. Systems, Man, Cybern. C*, vol. 34, no. 3, pp. 334–352, 2004.

[2] C. Stauffer, W. Eric, and W. E. L. Grimson, "Learning patterns of activity using real-time tracking," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 22, no. 8, pp. 747–757, 2000.

[3] X. Sun, M. Chen, and A. Hauptmann, "Action recognition via local descriptors and holistic features," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2009, pp. 58–65.

[4] G. Willems, T. Tuytelaars, and L. Gool, "An efficient dense and scale-invariant spatio-temporal interest point detector," in *Proc. 10th European Conf. Computer Vision*. Berlin: Springer-Verlag, 2008, pp. 650–663.

[5] E. Shechtman and M. Irani, "Space-time behavior-based correlation—or—how to tell if two underlying motion fields are similar without computing them?" *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 29, no. 11, pp. 2045–2056, 2007.

[6] J. W. Davis and A. F. Bobick, "The representation and recognition of action using temporal templates," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 1997, pp. 928–934.

[7] T. Xiang and S. Gong, "Beyond tracking: Modelling activity and understanding behaviour," *Int. J. Comput. Vis.*, vol. 67, no. 1, pp. 21–51, 2006.

[8] J. Flusser, B. Zitova, and T. Suk, *Moment Functions in Image Analysis: Theory and Applications*. Hoboken, NJ: Wiley, 2009.

[9] G. J. McLachlan, *Discriminant Analysis and Statistical Pattern Recognition*. Hoboken, NJ : Wiley, 2004.

[10] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proc. IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[11] C. Li and G. Biswas, "Temporal pattern generation using hidden Markov model based unsupervised classification," in *Advances in Intelligent Data Analysis (Proc. 3rd Int. Symp., IDA-99, Amsterdam, The Netherlands, Aug. 1999)* (Lecture Notes in Computer Science, vol. 1642). New York: Springer-Verlag, 1999, pp. 245–256.

[12] M. Ostendorf and H. Singer, "HMM topology design using maximum likelihood successive state splitting," *Comput. Speech Lang.*, vol. 11, no. 1, pp. 17–41, 1997.

[13] S. P. Chatzis, D. I. Kosmopoulos, and T. A. Varvarigou, "Robust sequential data modeling using an outlier tolerant hidden Markov model," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 31, no. 9, pp. 1657–1669, 2009.

[14] S. Ji, L. T. Watson, and L. Carin, "Semisupervised learning of hidden Markov models via a homotopy method," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 31, no. 2, pp. 275–287, 2009.

[15] S. Dupont and J. Luettin, "Audio-visual speech modeling for continuous speech recognition," *IEEE Trans. Multimedia*, vol. 2, no. 3, pp. 141–151, Sept. 2000.

[16] C. Chenand, J. Liang, H. Zhao, H. Hu, and J. Tian, "Factorial HMM and parallel HMM for gait recognition," *IEEE Trans. Systems, Man, Cybern. C*, vol. 39, no. 1, pp. 114–123, 2009.

[17] J. Luettin, G. Potamianos, and C. Neti, "Asynchronous stream modeling for large vocabulary audio-visual speech recognition," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, 2001, pp. 169–172.

[18] A. Nefian, L. Liang, X. Pi, L. Xiaoxiang, C. Mao, and K. Murphy, "A coupled HMM for audio-visual speech recognition," in *Proc. IEEE Int. Conf. Acoustics, Speech, and Signal Processing*, 2002, vol. 2, pp. 2013–2016.

[19] Z. Zeng, J. Tu, B. Pianfetti, and T. Huang, "Audio-visual affective expression recognition through multistream fused HMM," *IEEE Trans. Multimedia*, vol. 10, no. 4, pp. 570–577, June 2008.

[20] A. Makris, D. Kosmopoulos, S. Perantonis, and S. Theodoridis, "Hierarchical feature fusion for visual tracking," in *Proc. IEEE Int. Conf. Image Processing*, 2007, pp. VI:289–292.

[21] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, 2005, pp. 886–893.

[SP]