Hierarchical Dirichlet Processes for Unsupervised Online Multi-View Action Perception using Temporal Self-Similarity Features

Mahesh Venkata Krishna, Marco Körner and Joachim Denzler
Chair for Computer Vision, Friedrich Schiller University Jena, Germany
Email: {mahesh.vk, marco.koerner, joachim.denzler}@uni-jena.de

***Abstract*—In various real-world applications of distributed and multi-view vision systems, ability to learn unseen actions in an online fashion is paramount, as most of the actions are not known or sufficient training data is not available at design time. We propose a novel approach which combines the unsupervised learning capabilities of *Hierarchical Dirichlet Processes (HDP)* with *Temporal Self-Similarity Maps (SSM)* representations, which have been shown to be suitable for aggregating multi-view information without further model knowledge. Furthermore, the HDP model, being almost completely data-driven, provides us with a system that works almost "out-of-the-box". Various experiments performed on the extensive JAR-AIBO dataset show promising results, with clustering accuracies up to 60% for a 56-class problem.**

I. Introduction and Recent Work

Activity extraction and recognition from video data is an important area of research in computer vision. This is motivated by myriads of possible applications, including surveillance, smart environments, geriatric care, and many others. 3-dimensional representations of the real-world become possible with the ubiquitous presence of visual sensors in various places of day-to-day life. It is thus imperative that viable solutions be found to analyze the video data from *multiple views*, as this helps to overcome problems induced by ambiguities or self-occlusions. On the other hand, the usage of camera systems by non-technical users has increased, and one can no longer depend on the availability of good training labels. In addition, in many cases, the cost of labeling the data is prohibitive and an unsupervised, online learning approach is called for, which is commonly referred to as *activity perception* [1], in order to distinguish it from classical action recognition tasks.

Despite its practical significance, there exist only a small number of well-developed approaches to tackle this task. Most of the existing works on activity recognition focus on the supervised training-testing paradigm (*cf.* [2], [3]). They are either based on interest-point/object tracking and 3D reconstruction or on view-independent features. The work of Huynh and Schiele [4] deals with the problem of unsupervised activity extraction by representing activities as automatically learned projection sub-spaces (using a PCA-like method) and using distances from these sub-spaces as the basis for classification of new data. A major shortcoming of this approach is that, in the case of large number of activities (for example more than 50, as we have in our experiments), the distances between the sub-spaces can be small and the accuracy would suffer as a result. In addition, this method is not suitable for online learning.

Although there are some approaches which apply *Hierarchical Dirichlet Processes (HDP)* (or broadly *Probabilistic Topic Models (PTM)*) on the problem of activity classification, like [1], [5]–[8], they largely deal with the 2D case instead of incorporating multi-view knowledge. Furthermore, they mostly rely on motion features like local optical flow, which tend to be noisy and often not suitable for various applications like human action classification. Nevertheless, the power of the HDPs to model co-occurrence statistics and infer rules from these makes it a very useful modeling tool.

We propose to tackle this problem through a novel combination of HDPs and *Temporal Self-Similarity Maps* (SSM) recently proposed for action recognition tasks by Körner *et al.* [9]. The main contributions of this paper are two fold:

i) We introduce a novel joint framework using SSMs for data representation and HDPs for latent action extraction.
ii) We use this framework to perform unsupervised action perception in the context of multi-view video data, showing its applicability.

Additionally, we also affirm the statement made by Körner *et al.* [9], that SSMs based on Fourier features form a good representation of video data which is robust to view-point changes, even for action perception scenarios.

The remainder of this paper is structured as follows: in Sect. II, we will introduce HDPs and motivate their use for unsupervised machine learning tasks. Subsequently, SSMs will be briefly described in Sect. III. A description of the combined system follows in Sect. IV and the results of various experiments performed on a large-scale dataset will be reported in Sect. V.

II. Hierarchical Dirichlet Processes

The *Hierarchical Dirichlet Processes (HDP)* [10] model is a tool to cluster grouped data according to word co-occurrences. In HDP models and other non-parametric *generative* models, the calculation of parameters for the prior distributions are avoided, unlike the supervised, *discriminative* models, and the latent distributions are *inferred* depending on the data and a small number of hyper-parameters.

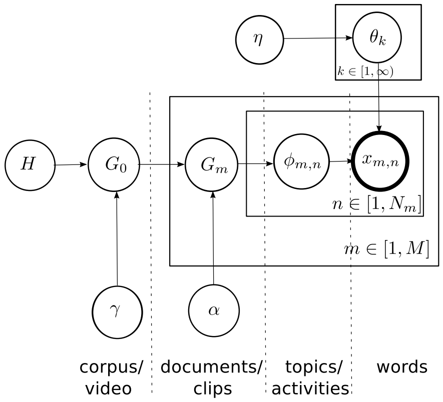


Figure 1: Hierarchical Dirichlet Processes. The latent mixtures $\phi_{m,n}$ are to be estimated from the observed words $x_{m,n}$. The topics are denoted by θ_k.

Fig. 1 shows the basic HDP model, which clusters grouped data in a hierarchical manner. HDPs assume that the data is ordered in a hierarchical manner, with a *corpus* forming the complete dataset, smaller clips forming the *documents* (say M in number), and features extracted from these clips forming the *words* (denoted as $x_{m,n}$, $n \in [1, N_m]$). Given this data as one *Bag of Words* histogram for each clip, the HDP model assigns topics (*i.e,*activities in our scenario) to the feature words that occur together.

Generative models view the data as being generated by random processes governed by latent distributions. In the case of HDPs, two layers of generative processes are assumed, one at the corpus level, denoted by G_0, and one on the other layer for each clip, denoted by G_m. Using the hyperparameters α and γ, we can formulate the generating process as

$$\begin{aligned} G_0 \,|\, \gamma, H &\sim DP(\gamma, H) \;, \\ G_m \,|\, \alpha, G_0 &\sim DP(\alpha, G_0) \quad \text{for } m \in [1, M] \;. \end{aligned} \tag{1}$$

Here, H is the base distribution, which is a *Dirichlet distribution* in our case (as conjugate prior, *cf.* [10]). From these clip-wise *Dirichlet Processes*, we assume that *topic mixtures* $\phi_{m,n}$ are generated, which in turn define priors over the observed words. Thus, the remaining part of the so-called *Chinese Restaurant Franchise formulation* is given by

$$\begin{aligned} \theta_k &\sim P(\eta) \quad \text{for } k \in [1, \infty) \;, \\ \phi_{m,n} \,|\, \alpha, G_m &\sim G_m \quad \text{for } m \in [1, M] \text{ and } n \in [1, N_m] \;, \\ x_{m,n} \,|\, \phi_{m,n}, \theta_k &\sim F(\theta_{\phi_{m,n}}) \;. \end{aligned} \tag{2}$$

Here, M is the number of clips, N_m is the number of words in clip m, and θ_k represent the pool of topics from which the topic mixtures draw words. $F(\cdot)$ and $P(\cdot)$ are parametrized distributions, and we set them as Dirichlet distributions.

Given the feature words $X = \{x_{m,n}\}$, the task of *Bayesian inference* is to estimate the topic mixtures $\phi_{m,n}$, which we interpret as individual actions in our context. This task is not simple and has no closed-form solution. To manage this computation, we resort to the widely used *Markov Chain Monte Carlo* methods [11], [12], especially the *Gibbs sampler*. The conditional distribution of the latent word-topic association,

$$p(\phi_{m,n} = k \,|\, X, \alpha, \gamma, \eta, \theta, H) \propto \left(n_{m,k}^{\neg m,n} + \alpha\theta_k\right) \cdot \frac{n_{k,t}^{\neg m,n} + \eta}{n_k^{\neg m,n} + V \cdot \eta} \;, \tag{3}$$

can be used for sampling, where $n_{m,k}, n_{k,t}$, and n_k represent count statistics of the word-topic associations, topic-document associations, and the topic-wise word counts, respectively. The superscript $\neg m, n$ indicates that the present word $x_{m,n}$ is to be excluded from these statistics and V is the size of the dictionary. The term containing count statistics in the Eq. (3) shows that the probability that a new word is assigned to a topic is proportional to the number of words already assigned to the topic. This shows the clustering property of the HDPs. The term with the hyperparameters α, γ, and η defines the probability that a new topic will be formed. Hence, these hyperparameters, especially η, can be used to control the number of topics the model extracts from the video.

In an unsupervised situation, suitable hyperparameter values may not be known in advance. HDPs provide a way to sample these parameters within the iterations as

$$\eta \;\sim\; \text{Dir}(n_1, \ldots, n_k, \gamma) \;. \tag{4}$$

For more a more detailed description of HDPs and sampling schemes for the remaining hyperparameters, α and γ, we refer to the work of Teh *et al.* [10] and Heinrich [13].

HDPs have some important properties which can be useful in our case:

i) HDPs are independent of the actual data representation. This makes the framework easily adaptable to a variety of application situations, like changes in the number of views, feature extraction procedures *etc.*

ii) Further, they are independent of the distances in feature spaces. This is basically due to the fact that they see different words and not the distances between them. This gives a huge advantage when we think of scalability in terms of number of action classes. Then, the ability of the system to distinguish actions from one-another is limited only by the representation and not by the model.

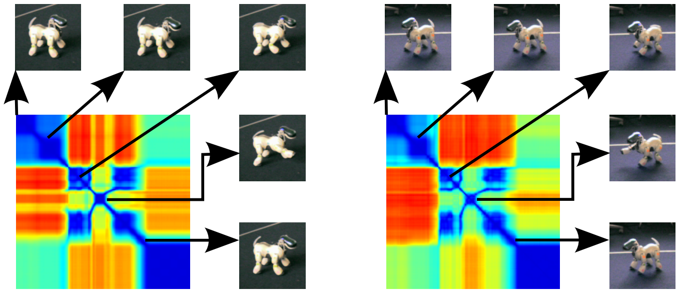
iii) The implementational extensions of sampling procedures proposed by Wang and Blei [14] and Yao *et al.* [15], which provide modified inference procedures for HDPs, enable us to apply HDPs for online learning scenarios.

These desirable properties make HDPs an invaluable tool for unsupervised action extraction.

III. Temporal Self-Similarity Maps

When using multiple cameras or a combination of other kinds of sensors, their data have to be aggregated in order to make use of their information gain. Most often, reconstruction-based methods are applied to tackle this problem. Nevertheless, these methods are very time and space consuming and require a very accurate calibration, otherwise they may lose or even hallucinate information.

In order to overcome this problem, we make use of the concept of temporal self-similarity, as proposed by Körner *et al.*



(a) Camera 0

(b) Camera 3

Figure 2: Two SSMs obtained for a robot dog performing an `stand_kickright` action captured from different viewpoints. Action primitives induce similar local structures in the corresponding SSM even under changes of viewpoint, illumination, or image quality. [9]

[9]. They suggest to encode changes of dynamic systems by constructing so-called *Temporal Self-Similarity Maps (SSM)*

$$\boldsymbol{S}^{\boldsymbol{I}_{1:N}}_{f,d} = \left[d(f(I_i), f(I_j))\right]_{i,j} \in \mathbb{R}^{N\times N}, \tag{5}$$

where each matrix element represent the frame-wise difference $d(\cdot,\cdot)$ of low-level features $f(\cdot)$ extracted from individual frames of an input image sequence $\boldsymbol{I}_{1:N} = \{I_1, \ldots, I_N\}$.

As can be seen in Fig. 2, particular atomic actions induce specific pattern structures within these SSMs. It has been shown that SSM are very robust under viewpoint changes.

Furthermore, the specific invariants of the underlying low-level features $f(\cdot)$—*e.g.* intensity values, Histograms of Oriented Gradients (HOG), or truncated Fourier descriptors [9]—are preserved while constructing SSMs. Therefore, one can assume, that the particular local structures of SSMs are specific for certain action primitives. In order to exploit this observation, high-level features are extracted from the SSM. Körner *et al.* [9] proposed to use SIFT descriptors [16] extracted along the main diagonal in order to obtain a temporal-scale invariant description of these structures. Converting this collection of individual descriptors extracted from distinct view-specific SSMs into a joint *Bag of Self-Similarity Words (BoW)* representation leads to a single fixed-size feature vector describing the whole action observed by multiple cameras.

IV. Unsupervised Multi-View Action Learning

As proposed by Körner *et al.* [9], multi-view SSM features extracted following the scheme presented in Sect. III can be used for view-independent or multi-view action recognition. In order to tackle the problem of action perception, we use these features together with the HDP framework introduced in Sect. II to create an unsupervised online learning system, as visualized in Fig. 3.

In our framework, we first compute Fourier descriptors for each frame, which magnitudes are then used to construct temporal SSMs as described in Sect. III. As we have seen before, temporal SSMs can be a very good representation for multi-view video data. But when used directly to form a dictionary and a *Bag of Self-Similarity Words* representation, owing to the large dimensionality of the feature-space, they often result in a huge vocabulary size and result in over-fitting by the HDP model. Furthermore, effects of noise can be a serious problem in such situations. To overcome these problems, we use a further higher level of representation, *i.e,* use of an extra clustering step to merge visual features into prototypes which form the entries in the dictionary. Here, the goal is to represent related feature vectors like those that are close to each other in the feature space as one entry in the dictionary.

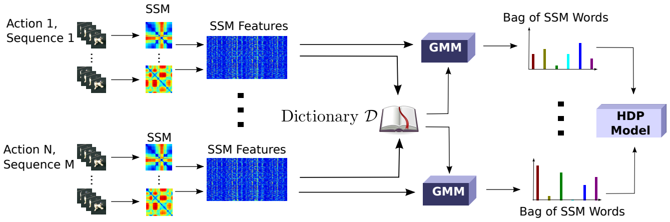


Figure 3: Complete outline of the approach for unsupervised online action perception in multi-view settings.

Similar to the approach used by Körner *et al.* [9], this can be achieved by *Gaussian Mixture Models (GMM)* [17] or other clustering approaches like k-means. The advantage of using GMM for clustering is in their ability to suppress noisy features and also to produce a rather compact, yet discriminative representation of the data.

Once the dictionary is formed and Bag of Self-Similarity Words histograms are extracted, the HDP modeling step is applied. The inference algorithm follows as described in Sect. II. The result is a mapping of actions to words present in each of the histograms. Then, for each histogram, we see the most probable action through voting among the words belonging to the histogram. This gives us the actions presented in each clip.

For an online learning setup, we can simply add another clip and instead of re-running the whole inference step, we only infer for the new clip (similar to Yao *et al.* [15]), and see which of the learned actions is present in the clip and if a new action is discovered, it is added to the list of known actions.

V. Experimental Evaluation

A. Dataset

In order to evaluate the performance of our proposed framework, we performed various experiments on the JAR-Aibo multi-view dataset [18], which was designed for benchmarking model-free action recognition systems.

The recording details, as described by the creators of the dataset, are as follows. Six interconnected and synchronized VGA cameras were equally distributed around a rectangular scene. 56 actions performed by Sony ERS-7 Aibo robot dogs where recorded 7 to 10 times per action, resulting in 526 labeled clips in total for each view. For studying the effects of view changes and to see the improvements made by increasing the number of views on the performances of algorithms, various combinations of the available cameras have been considered. Thus, in our experiments we used i) all views individually, ii) all 20 possible combinations of 3 views, and iii) the combination of all 6 views.

Fig. 4 shows some sample images from the dataset. As can be seen, the color impressions as well as the illumination

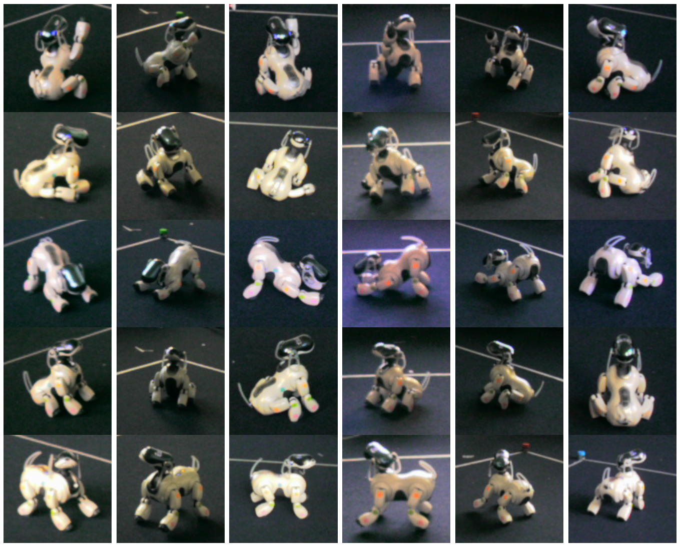


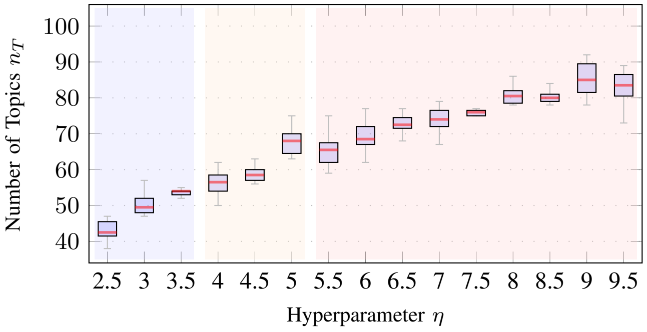
Figure 4: Example images from the dataset. Each column represents one camera view, each row show one Aibo action exemplar frame (`sit_greeting`, `sit_scootright`, `sit_stretch`, `sit_yes`, `stand_dance`).

conditions vary between the cameras. Additionally, the bodies of the recorded robot dogs are nearly untextured and specular. This, added with the relatively large number of actions compared to other datasets, make it challenging a dataset to work with. The challenge is further compounded by the fact that the actions are actually combinations of pose and atomic actions. For example, `lie_wave` and `stand_wave` are regarded as two different actions, while in feature spaces they are generally placed very close together. We will next see some specific details on how we represent this data for use in the HDP model.

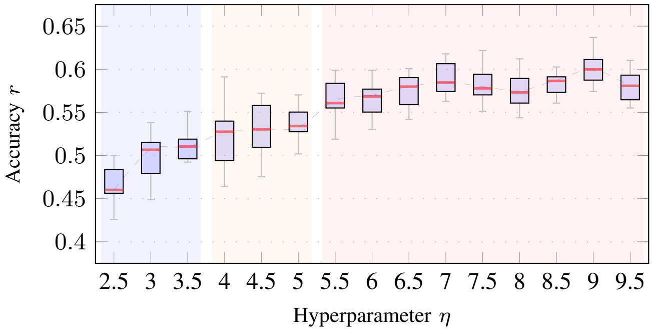
B. *Experimental Setup*

In order to create the test data for our experiments, we followed the suggestions made by Körner *et al.* [9]. We created SSMs using truncated Fourier descriptors as low-level features and normalized cross-correlation as similarity measure. Thereafter, we extracted 128-dimensional SIFT descriptors for all view-specific SSMs. These descriptors obtained for all views were further transformed into a joint 512-dimensional *Bag of Self-Similarity Words* representation using the GMM clustering scheme. The image processing part of our experiments was implemented in C++ using the OpenCV and NICE[1] libraries for feature extraction and classification purposes, respectively, while the HDP modeling was realized in Matlab using the standard toolboxes. We performed our experiments on a standard desktop computer equipped with a Intel(R) Core(TM) i7 960 CPU running at 3.2 GHz and 24 GB of RAM. For each activity extraction experiment with 500 Gibbs iterations for the HDP model, we needed about 40 minutes to process 526 clips, each approximately of 10 to 12 seconds.

For evaluation, we associated the topics extracted by the HDPs to the ground truth labels by voting among the topics. As the dataset includes 56 pose-action combinations, the evaluation was done as in the case of a 56-class classification problem, with the topic-label associations treated as classification results. Then, we used the accuracy measure to evaluate performance, due to its intuitiveness and as the ground truth labels give the optimum description of the scene.



(a)



(b)

Figure 5: Influence of varying the value of the hyperparameter η on (a) the number of topics extracted from the data presented to the HDP and (b) the clustering accuracy. While the number of topics grows almost linearly (up to $\eta = 8.0$) with increasing value of η, the resulting classification accuracy saturates beyond a certain point, as the model starts to overfit the presented data.

C. *Results*

We tested our approach wrt. four main aspects and goals: i) the overall accuracy of the system, tested against the ground-truth, ii) the view independence of temporal SSMs, iii) to study the gain in multi-view activity perception as against monocular perception, and finally iv) to see how increasing the number of views improves performance.

1) First Experiment—Overall Accuracy: To demonstrate the accuracy of the action extraction, we used the joint BoW histograms of all 6 views. Furthermore, due to the element of randomness in the Bayesian inference step, we repeated each experiment 10 times and compiled the results. To study the effects of the hyperparameter η, we varied the values and observed the clustering accuracy based on the overall number of topics (*i.e,*actions) that were extracted correctly.

Intuitively, the number of topics extracted by the HDP would directly affect the action classification accuracy. Here, too small a number would under-represent the actions presented by the data, which causes joining of similar actions into one. On the other hand, a too large number of topics would result in over-fitting, which leads to consequent errors, as differences

[1] Git repository: https://github.com/cvjena/nice-core.git

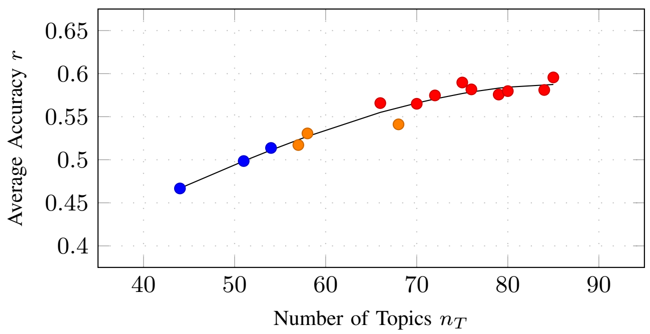


Figure 6: Relationship between the number of topics extracted from the data and the clustering accuracy. The colors of the data points correspond to the shading in Fig. 5. After a linear growth until a certain number of topics, the model does not benefit from introducing more action topics.

in execution of actions would be taken into account. One could expect that for good performance, the number of topics extracted should be close to, or even slightly larger, than the actual number of actions. When evaluating our experiments, we found that this is indeed the case, as can bee seen in Fig. 6. In more detail, Fig. 5(a) shows the accuracy, and Fig. 5(b) shows the variation in the number of actions, as the hyperparameter η is varied from 2.5 to 9.5. The accuracy of our approach is up to 60% when η is set to 7.0 or 9.0. To put this result in perspective, state-of-the-art supervised learning methods using *Gaussian Process (GP)* classification [19] achieve accuracies up to 87.8%, as reported in [9].

We can further note that, as the number of extracted actions crosses the actual number of actions, the accuracy improves significantly. Thereafter, the change is minimal. The reason is that, in HDPs, the number of actions extracted does not increase indefinitely based on larger values of η. Beyond a certain point (here, $\eta = 8.0$), the hyperparameter has little effect on the number of actions, as the hyperparameter itself gets resampled depending on the data. This makes our framework almost parameter-free.

2) Second Experiment—View Changes: Another interesting aspect is to see the merits of our representation in terms of robustness to view changes. For this, we considered joint BoW histograms using all possible combinations of three views each, which results in 20 combinations. Again, the experiments were conducted 10 times for each view-combination, while the value of the hyperparameter η was fixed at 7.0. Fig. 7(a) and Fig. 7(b) show our results. The accuracies are similar over almost all view combinations, showing that irrespective of the views used, the algorithm performs on the same level. The few cases when the results vary correspond to the view combinations that covered the field either poorly or very well. This shows that temporal SSMs are a good representation of multi-view video data without the need for camera calibration or 3D reconstruction.

3) Third Experiment—Monocular Perception: Next, we study the case of monocular vision. For this purpose we individually considered each of the 6 views and compared the results. Fig. 8(a) and Fig. 8(b) show the accuracy and number of actions for the individual views. We again set the hyperparameter η to 7.0 and repeated the experiments 10 times.

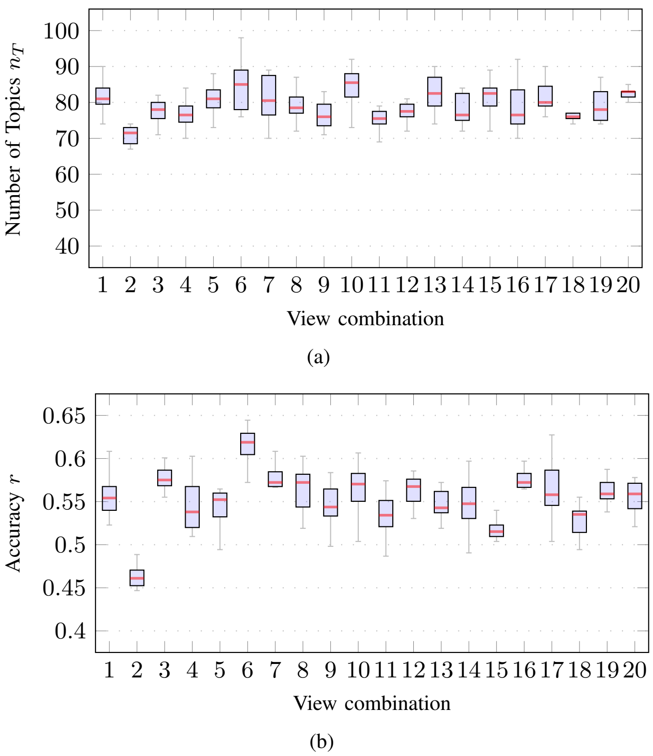


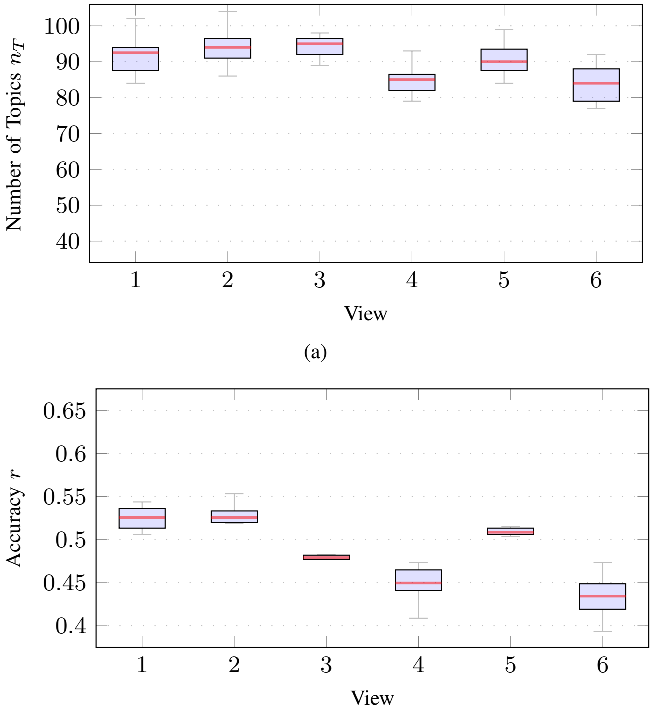
Figure 7: Results of experiments individually performed on all 20 possible 3-fold subsets of available camera views and fixed hyperparameter $\eta = 7.0$. Since the ambiguities of actions increases with less observing cameras, the number of topics extracted by the HDP grows and results in inferior accuracies. Note that the number of topics is in general constant for all 3-fold subsets.

Clearly, some views are better suited to observe some subtle movements, and the overall performance is worse than the previous multi-view cases. The number of actions follows the same pattern.

4) Fourth Experiment—From One View to Six Views: Finally, we compare the clustering accuracy as we increase the number of views. This is to verify that the additional information, that the new views provide, help our system to make better decisions regarding activities. Fig. 9 shows the performance of the algorithm as the number of views are increased. As we go from 1 view to 2 views, the improvement in area coverage is not very large, whereas, going from 2 to 3 views, the additional view provides much better coverage, and the boost in performance is quite sharp. As the number of views is further increased, the redundancy increases, making improvement steps smaller.

VI. Conclusion and Summary

We presented a novel approach combining the HDP models with the temporal SSMs to perform unsupervised multi-view action recognition, with online learning possibilities. It was evaluated using the challenging and extensive JAR-Aibo dataset and the results presented in the last section validated the performance and applicability of our approach. Evaluating it as a normal 56-class problem, we achieved accuracy rates up to 60%. Also, we showed the view-point independent nature of



(a)

(b)

Figure 8: Results of experiments performed on individual camera views and fixed hyperparameter $\eta = 7.0$. (a) As articulated actions observed from single cameras show the highest amount of ambiguities and self-occlusions, the number of topics extracted from the presented data is maximal in this case. (b) Consequently, the classification accuracy drops significantly.

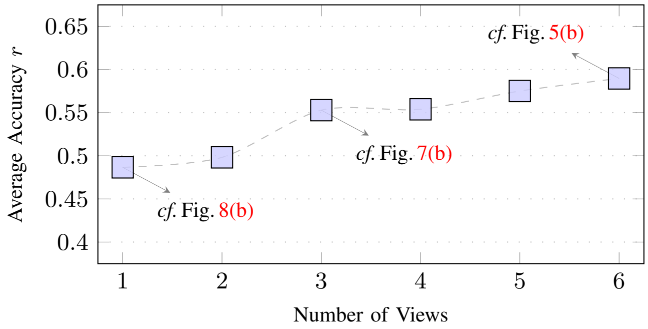


Figure 9: Improvement in performance when increasing the number of observing camera views. While using three cameras results in a significant gain of the accuracy, adding more cameras in general does not affect the recognition rates, as the amount of redundancy is increased.

our representation, as accuracy and number of actions extracted remained quite uniform across view-point changes. Furthermore, for 526 clips, we obtained runtimes of just 40 minutes, showing that an optimized implementation can easily yield real-time performance.

One of the shortcomings of the present state of implementation is, that in the case of complex videos with multiple actions happening, the basic global feature extraction will not suffice. For such a case, we can extend our framework to detect each foreground object through motion based segmentation and then compute SSMs for each object individually. Also, some changes could be made in the framework to include some supervision, so that if a few videos with labels or camera calibration data are available, we should be able to use them. A possible way of doing this could be to use a refinement step after the HDP model has completed the inference step.

Acknowledgements

The authors thank Christian Wojek and Stefan Saur for useful discussions and suggestions. Mahesh Venkata Krishna is supported by Carl Zeiss AG through the “Pro-Excellenz” scholarship of the Federal State of Thuringia, Germany.

References

[1] X. Wang, X. Ma, and W. Grimson, “Unsupervised activity perception in crowded and complicated scenes using hierarchical bayesian models,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 3, pp. 539–555, 2009.

[2] R. Poppe, “A survey on vision-based human action recognition,” *Image and Vision Computing*, vol. 28, no. 6, pp. 976–990, 2010.

[3] J. K. Aggarwal and M. S. Ryoo, “Human activity analysis: A review,” *ACM Computing Surveys*, vol. 43, no. 3, pp. 16:1–16:43, 2011.

[4] T. Huỳnh and B. Schiele, “Unsupervised discovery of structure in activity data using multiple eigenspaces,” in *Proceedings of the 2nd International Conference on Location- and Context-Awareness*, 2006, pp. 151–167.

[5] D. Kuettel, M. D. Breitenstein, L. V. Gool, and V. Ferrari, “What is going on? discovering spatio-temporal dependencies in dynamic scenes,” in *Proceedings of the 13th IEEE Conference on Computer Vision and Pattern Recognition*, 2010, pp. 1951–1958.

[6] T. Hospedales, S. Gong, and T. Xiang, “A markov clustering topic model for mining behaviour in video,” in *Proceedings of the 12th IEEE International Conference on Computer Vision*, 2009, pp. 1165–1172.

[7] T. Hospedales, J. Li, S. Gong, and T. Xiang, “Identifying rare and subtle behaviors: A weakly supervised joint topic model,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 33, no. 12, pp. 2451 –2464, 2011.

[8] J. Li, S. Gong, and T. Xiang, “Learning behavioural context,” *International Journal of Computer Vision*, vol. 97, no. 3, pp. 276–304, 2012.

[9] M. Körner and J. Denzler, “Temporal self-similarity for appearance-based action recognition in multi-view setups,” in *Proceedings of the 15th International Conference on Computer Analysis of Images and Patterns (CAIP)*, 2013.

[10] Y. Teh, M. Jordan, M. Beal, and D. Blei, “Hierarchical Dirichlet Processes,” *Journal of the American Statistical Association*, pp. 1566–1581, 2006.

[11] C. Andrieu, N. de Freitas, A. Doucet, and M. I. Jordan, “An introduction to mcmc for machine learning,” *Machine Learning*, vol. 50, no. 1, pp. 5–43, 2003.

[12] W. Gilks, S. Richardson, and D. Spiegelhalter, *Markov Chain Monte Carlo in Practice: Interdisciplinary Statistics*. Chapman and Hall/CRC, 1995.

[13] G. Heinrich, “Infinite LDA – implementing the HDP with minimum code complexity,” Fraunhofer IGD, Tech. Rep. TN2011/1, 2011. [Online]. Available: http://arbylon.net/publications/ilda.pdf

[14] C. Wang and D. M. Blei, “A split-merge mcmc algorithm for the hierarchical dirichlet process,” *ArXiv e-prints, arXiv:1201.1657 [stat.ML]*, 2012.

[15] L. Yao, D. Mimno, and A. McCallum, “Efficient methods for topic model inference on streaming document collections,” in *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2009, pp. 937–946.

[16] D. G. Lowe, “Distinctive image features from scale-invariant keypoints,” *International Journal on Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[17] J. Winn, A. Criminisi, and T. Minka, “Object categorization by learned universal visual dictionary,” in *Proceedings of the 10th IEEE International Conference on Computer Vision*, vol. 2, 2005, pp. 1800–1807.

[18] M. Körner and J. Denzler, “JAR-AIBO: A multi-view dataset for evaluation of model-free action recognition systems,” in *Proceedings of the 1st International Workshop on Social Behaviour Analysis (SBA)*, 2013.

[19] C. E. Rasmussen and C. K. I. Williams, *Gaussian Processes for Machine Learning*. MIT Press, 2006.