Indian Classical Dance classification by learning dance pose bases

Article · January 2012

DOI: 10.1109/WACV.2012.6163050

3 authors:

Soumitra Samanta
Indian Statistical Institute
14 PUBLICATIONS **159** CITATIONS
SEE PROFILE

Pulak Purkait
Toshiba Research Europe Limited
41 PUBLICATIONS **745** CITATIONS
SEE PROFILE

Bhabatosh Chanda
Indian Statistical Institute
225 PUBLICATIONS **3,863** CITATIONS
SEE PROFILE

Indian Classical Dance Classification by Learning Dance Pose Bases

Soumitra Samanta, Pulak Purkait and Bhabatosh Chanda
Electronics and Communication Sciences Unit
Indian Statistical Institute, Kolkata, India
soumitramath39@gmail.com, pulak_r@isical.ac.in and chanda@isical.ac.in

Abstract

In this paper, we address an interesting application of computer vision technique, namely classification of Indian Classical Dance (ICD). With the best of our knowledge, the problem has not been addressed so far in computer vision domain. To deal with this problem, we use a sparse representation based dictionary learning technique. First, we represent each frame of a dance video by a pose descriptor based on histogram of oriented optical flow (HOOF), in a hierarchical manner. The pose basis is learned using an on-line dictionary learning technique. Finally each video is represented sparsely as a dance descriptor by pooling pose descriptor of all the frames. In this work, dance videos are classified using support vector machine (SVM) with intersection kernel. Our contribution here are two folds. First, to address dance classification as a new problem in computer vision and second, to present a new action descriptor to represent a dance video which overcomes the problem of the "Bags-of-Words" model. We have tested our algorithm on our own ICD dataset created from the videos collected from YouTube. An accuracy of 86.67% is achieved on this dataset. Since we have proposed a new action descriptor too, we have tested our algorithm on well known KTH dataset. The performance of the system is comparable to the state-of-the-art.

1. Introduction

In recent years, the field of computer vision has seen tremendous progress in classification problem. In this paper, we address a new category classification problem in computer vision domain which is a dance style classification in particular and human activity recognition in general. Here, we concentrate on Indian Classical Dance (ICD). During the last two decades, people have tried to develop different algorithms for human activity analysis [2, 22] for wide applications in the area of surveillance, patient monitoring and many more. Most of the works have been reported on classifying human activity from videos. Recently researchers are trying to classify an activity from a single image [29, 19, 30]. In the video based activity recognition, people have tried with different human activities like walking, jogging, running, boxing, hand waving, hand clapping, pointing, digging and carrying for a single actor [2, 12]. There are a few works on group activities also [10, 23]. With the best of our knowledge, no one has addressed the dance classification problem so far at least in computer vision domain. Due to the increase in multimedia data access through the internet, multimedia data specially video data indexing becomes more and more important. Not only in the retrieval but also for digitization of cultural heritage, this can be an interesting problem. It can be used to analyze a particular dance language. We believe that, in future this will attract more interest of computer vision community.

The earliest civilizations discovered in the Indian subcontinent are those of Mohenjo Daro and Harappa in the Indus valley, and are dated about 6000 B.C. [20]. It would appear that by that time dance had achieved a considerable measure of discipline and it is likely, but not certain, that it was connected with religion. In any case it must have played some important role in the society, for one of the finds at Mohenjo Daro was a beautiful little statuette of dancing girl. Indian classical dance is one of the oldest dance traditions associated with any of the world's major religions.

Indian classical dance is the gesture of all the body parts. The analysis of various gesture in the ICD is truly remarkable. Each class of dance has its own gesture. Depending on the gesture of different body parts we can classify the different ICD. However, during dance performance due to occlusion, it becomes difficult to capture all the gestures with the help of existing technology. In this study, we consider a high level feature representation of dance video exploiting mainly motion information to handle the above dance classification problem. Here, we address three oldest Indian dance classes namely Bharatnatyam, Kathak and Odissi. Bharatnatyam is one of most popular ICD from southern part of the India. Kathak and Odissi are from northern and eastern part of India respectively.

In general ICD classification is a human activity classification problem. There are several attempts to recognize the human activities from video [24, 2]. Aggarwal *et al*. [2] classify the human activity recognition in two classes namely, single-layered approach and hierarchical approach. In single layer approach, activities are recognized directly from videos, while in hierarchical approach, an action is divided into sub-actions [8]. The action is represented by classifying it into sub-actions. Wang *et al*. [26] have used topic model to model the human activity. They represent a video by Bag-of-Wards representation. Later, they have used a model which is popular in object recognition community, called Hidden Conditional Random Field (HCRF) [27]. They model human action by flexible constellation of parts conditioned on image observations and learn the model parameters in max-margin framework and named it max-margin hidden conditional random field (MMHCRF).

Some researchers use space time features to classify the human action. Blank *et al*. represent the human action as three dimensional shapes included by the silhouettes in the space-time volume [3]. They use space-time features such as local space-time saliency, action dynamics, shape structure and orientation to classify the action. In [24], they recognise human action based on space-time locally adaptive regression kernels and the matrix cosine similarity measure. Klaser *et al*. localize the action in each frame by obtaining generic spatio temporal human tracks [9]. They have used sliding window classifier to detect specific human actions.

Fengjun *et al*. model the human activity as a collection of 2D synthetic human poses rendered from a wide range of viewpoints [15]. For an input video sequence, they match each frame with synthetic poses and track the same by Viterbi algorithm. On the other hand, in a dance video, there are lots of variations in poses. So, to model each dance pose is largely an unexplored research area. In [23], they classified human activities into three categories: atomic action, composite action, and interaction. They use context-free grammar (CFG) to represent the complex human action.

Our contribution in this paper are in two folds. First, we present ICD classification problem as a new application of computer vision. Secondly, to solve this problem, we propose a new dance descriptor (or action descriptor). We build a new dataset of ICD which may be helpful in computer vision research community for further research in this domain and will be available soon. Rest of the paper is organized as follows: Proposed methodology is described in section 2. In section 3, we show the experimental results and finally, section 4 concludes the paper.

2. Proposed Methodology

In our method, we first represent each frame of a ICD video using pose descriptor. An over-complete dictionary is then learned from the visual words consisting of some sequence of pose descriptors using a well known on-line dictionary learning technique [17]. Based on the learned dictionary, we sparsely represent each video words and finally build a sparse descriptor for dance. In the following subsection, we discuss the pose descriptor.

2.1. Pose descriptor:

Motion information [7, 11, 14, 6] is one of the dominant feature in human activity analysis from a video data. We use motion and oriented gradient information to build a pose descriptor of each frame [21]. To get a pose descriptor, we calculate optical flow $[\overrightarrow{OF}]$ and the gradient $[\overrightarrow{G}]$, as shown in Figure 1(a) and 1(b) respectively.

We combine these two vector fields to get the final motion matrix $[\overrightarrow{R}]$ of the corresponding frame as follows:

$$[\overrightarrow{R}] = [\overrightarrow{G}] * [\overrightarrow{OF}] \tag{1}$$

where binary operation $(*)$ represents the element wise multiplication of $\overrightarrow{G}$ and $\overrightarrow{OF}$. i.e.,

$$R_x = G_x \times OF_x \tag{2}$$

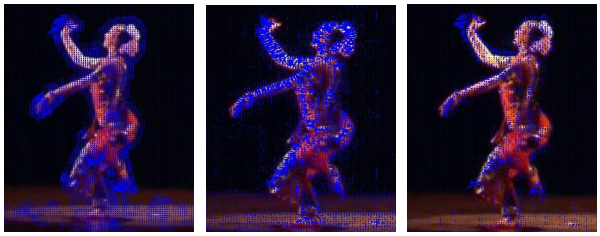
and

$$R_y = G_y \times OF_y \tag{3}$$

Hence,

$$\theta = \tan^{-1}(\frac{R_y}{R_x}) \tag{4}$$

The above equation clearly explains that, the weighted optical flow captures the information mostly at the outline of the actor present in the video frame and not from background shown in Figure 1(c). We calculate the feature in a hierarchical fashion. For that, the image (corresponding to a video frame) is divided into 4^l parts in l^{th} $(l = 0, 1, ...)$ layer. Angle histogram (calculated by Eq. 4) of each parts is calculated in L bins. For l^{th} layer, we calculate the histogram of each parts and concatenate them to form a vector of length $(L \times 4^l)$. In each layer, we normalize the vector by L_1 norm. For the details, please go through [21]. For our experiment, we take the value of l as 0, 1 and 2, and L = 8. Finally, concatenate all the vectors of all layers and get a $168 = (8 + 4 \times 8 + 16 \times 8)$ dimensional feature vector, which is the pose descriptor shown in Figure 2. For i^{th} training dance video V_i $(i = 1 : M)$ having the number of frames N_i, we get $(N_i - 1)$ pose descriptors of each 168 dimension as $f_{i,1}, f_{i,2}, ..., f_{i,N_i-1}$. We learn an over-complete dictionary from the set of all pose descriptors $F = \{f_{i,j} : i = 1 : M, j = 1 : N_i\}$ using online dictionary learning method, discussed in the next section.



(a) (b) (c)

Figure 1. (a): Optical Flow filed($\bar{OF}$) (b): Gradient field($\bar{G}$) (c): Final motion filed(($\bar{R}$))

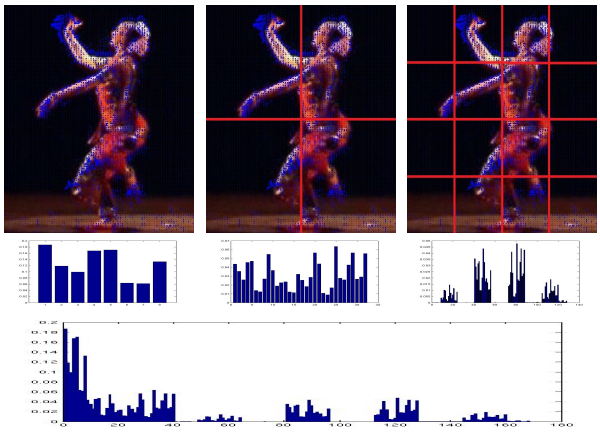


Figure 2. Pose deacriptor: Calculation in each layer

After representing each frame of a video by a pose descriptor our goal is to make a fixed length representative vector that can summarizes the whole video so that we may use a conventional SVM for classification task. This process can often be broken down into two steps [4]: (1) a coding step, which performs a point-wise transformation of the frame descriptors into a representation better adapted to the task, and (2) a pooling step, which summarizes the coded features over larger neighborhoods.

In this section, we mainly focus on the following steps:

1. **Coding:** Input mid-level pose feature descriptors are locally transformed block-wise into some compact representation. The code is typically a vector with binary (vector quantization) or continuous (HOG, sparse coding) entries, obtained by decomposing the original feature using some code-book, or dictionary.

2. **Temporal pooling:** The codes associated with local frame features are pooled over some image neighborhood. The codes within each cell are summarized by a single "semi-local" feature vector, common examples being the average of the codes (average pooling) or their maximum (max pooling).

In this work, we propose a sparse representation of spatio-temporal (space-time) visual word as the sequence of frames and a representing vector that summarizes a video by building a temporal pyramid and max pooling on each cell. Details are given below:

2.2. Learning Space-Temporal 3D dictionary (Coding)

Given a set of features described in the last section for each frame of a video, we build a spatio-temporal dictionary. As a video is represented by a sequence of frames, we take subsequences of fixed size from the whole sequence of frames and call it a "visual word". We divide each video into some overlapping subsequence of frames, i.e. concatenation of visual words. Then each video consists of some visual words and total number of visual words in videos are different as their sizes differ. We learn a dictionary on those "visual words" that can represent any "visual words" in a video by a single code. There are several existing techniques to learn the dictionaries. Among them "Bag-of-Words" is the most popular in this context. Yang *et al*. [28] proposed a sparse coding technique to learn a dictionary of SIFT features for object classification. Later Boureau *et al*. [4] also described in detail the usefulness of sparse coding for generating a dictionary for classification. The concept of "Bag-of-Words" was to quantize the feature space into some hard-clusters and cluster centers were represented as words. In sparse representation, the main concept is to generate a dictionary from which a word can be generated by a linear combination of few words from dictionary.

2.3. Summarizing a video by Pooling

Spatial pooling techniques are the most popular and well studied techniques for image classification [13, 28]. In linear Spatial Pyramid Matching (SPM) technique, images are divided into cells in hierarchical structure and on each cell a linear pooling or a max pooling is used to summarize the cells into a single representative vector. Being motivated by the good experimental result [13, 28] of SPM and through extensive study on [4], we build a temporal pyramid on a video in a hierarchical manner. A max pooling is done on each of the cells consisting of some visual words represented by sparse codes. In [13], it is shown that for pyramid matching kernel which is simply a weighted sum of histogram intersections, we can use simply a single intersection kernel of concatenated responses of individual cells.

Let a video V is represented by a sequence of low-level frame descriptors f_j for j^{th} frame identified with their indices $j = 1, \ldots, N-1$. Divide the whole sequence into some overlapping subsequence of frames as shown in Figure 3. Frame descriptors of all the frames of each of the subsequences are concatenated in order to represents a visual word. Then the whole video can be represented by a sequence of visual words $vf_i, i = 1, \ldots, \mathcal{N}$. We build a temporal pyramid on top of the visual word sequence. Let

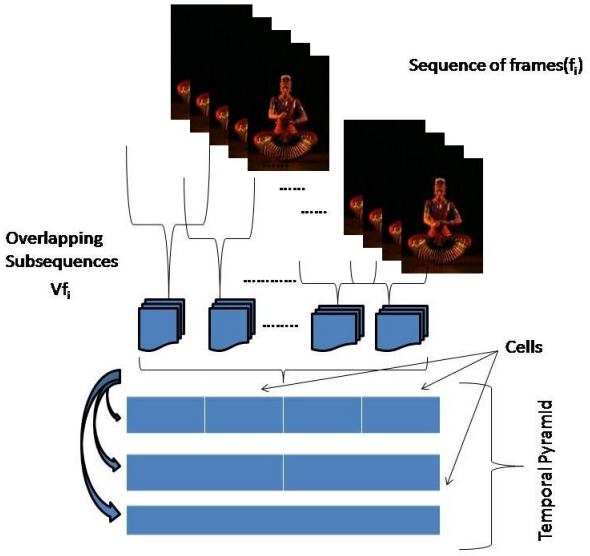


Figure 3. Illustrates the architecture of our algorithm based on Temporal Pyramid matching. The whole video sequence is divided into overlapping visual words vf_i Sparse coding measures the responses of each local descriptor to the dictionary's "visual words". These responses are pooled across different temporal locations over different temporal scales.

there be $\mathcal{M}$ cells/regions of interests on the video (e.g., the $15 = 8 + 4 + 2 + 1$ cells of a four-level temporal pyramid), with $\mathcal{N}_m$ denoting the number of frames/indices within region m. Let f and g denote some coding and pooling operators, respectively. The vector z representing the whole video is obtained by sequentially coding, pooling over all regions, and concatenating:

$$\alpha_i = f(vf_i), i = 1, \ldots, \mathcal{N} \quad (5)$$

$$h_m = g(\{\alpha_i\}_{i \in \mathcal{N}_m}), m = 1, \ldots, \mathcal{M} \quad (6)$$

$$z^T = [h_1^T, h_2^T, \ldots, h_{\mathcal{M}}^T] \quad (7)$$

In the usual bag-of-features framework [13], f minimizes the distance to a code-book, usually learned by an unsupervised algorithm (e.g., K-means), and g computes the average over the pooling region:

$$\alpha_i \in \{0,1\}^K, \alpha_{i,j} = 1 \text{ iff } j = arg\min_{k \leq K} \|vf_i - d_k\|_2^2 \quad (8)$$

$$h_m = \frac{1}{|\mathcal{N}_m|} \sum_{i \in \mathcal{N}_m} \alpha_i \quad (9)$$

where d_k denotes the k-th codeword in the code-book. Note that averaging and using uniform weighting is equivalent (up to a constant multiplier) to using histograms with weights inversely proportional to the area of the pooling regions.

Van Gemert et al. [25] have obtained improvements in object classification by replacing hard quantization by soft quantization:

$$\alpha_{i,j} = \frac{\exp(-\beta \|vf_i - d_j\|_2^2}{\sum_{k=1}^{K} \exp(-\beta \|vf_i - d_k\|_2^2)} \quad (10)$$

where β is a parameter that controls the softness of the soft assignment (hard assignment is the limit when $\beta \to \infty$). This amounts to coding as in the E-step of the expectation-maximization algorithm to learn a Gaussian mixture model, using codewords of the dictionary as centers.

Sparse coding [5, 16] uses a linear combination of a small number of codewords to approximate the vf_i. We use an online dictionary learning algorithm [17] to generate our code-book so that each visual word can be approximate by a sparse combination of code-book words. We use the idea of soft-assignment of target sample visual word with the constrained that it can be approximated by linear combination of a few visual words. Then we use max pooling on each of the cells to get near-local responses.

$$\alpha_i = arg\min_{\alpha} L(\alpha, \mathcal{D}) \simeq \|vf_i - \mathcal{D}\alpha\|_2^2 + \lambda \|\alpha\|_1 \quad (11)$$

$$h_{m,j} = \max_{i \in \mathcal{N}_m} \alpha_{i,j}, forj = 1, \ldots, K \quad (12)$$

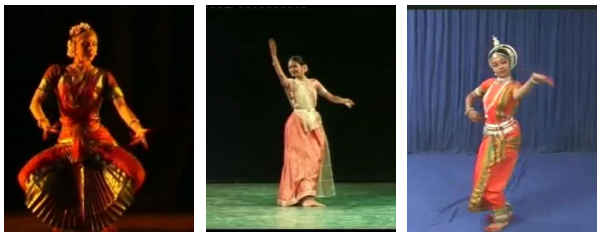
where $\|\alpha\|_1$ denotes the l_1 norm of α, λ is a parameter that controls the sparsity, and $\mathcal{D}$ is a dictionary trained by minimizing the average of $L(\alpha_i, \mathcal{D})$ over all samples of visual words in the training videos, alternatively over $\mathcal{D}$ and α_i. We have used SPAMS software [1] to train the dictionary and sparse representation of each visual words.

2.4. video descriptor

After pooling on each of the cells a pyramid matching kernel is used. As described in detail in [13], it can be represented by the intersection kernel concatenating the individual responses of each cell. So our case the summarization of a video or the video descriptor can be written as $V^T = [h_1^T, h_2^T, \ldots, h_{\mathcal{M}}^T]$. We use a one to one SVM with intersection kernel to learn those video descriptors for classification task.

3. Experimental results

Since this is the first attempt to classify ICD in computer vision domain, there is no standard dataset available for evaluation. To test our algorithm, we have created a dataset from the videos downloaded from YouTube video library. Primarily, the dataset consists of the three oldest and most popular ICDs, Bharatnatyam, Kathak and Odissi. The dataset is manually labeled by dance experts. In our dataset, each class contains 30 video clips with different resolutions (max. 400×350). The maximum duration of a



(a) (b) (c)

Figure 4. Indian classical dances :(a): Bharatnatyam (b): Kathak and (c): Odissi.

particular video clip is around 25 second. There is no common dancer between any two classes although within a class some videos have the same dancer. Some sample frames of each class are shown in Figure 4 for illustration. Most of the video clips are from real-life stage performance meaning thereby, the video clips are having lots of variations in terms of lighting condition, clothing, camera position, background, occlusion etc. Moreover, videos are captured from different distances which incorporate an additional variation in terms of zooming. All the said variations make the classification a challenging task.

In this experiment, we use two-third of the data for training and the remaining for testing. A dictionary of visual words (concatenation of 168 dimensional pose vector extracted from each frames of the training videos) is learned using a well-known on-line dictionary learning algorithm [1, 17]. To represent an ICD video, we use max pooling of sparse representation of visual words based on the learned dictionary as discussed in the Section 2.3. Finally, a kernel SVM [18] with intersection kernel is used for classification. Hyper-parameters of the SVM is fine tuned employing a cross validation algorithm.

Interestingly, we get an average classification accuracy of 86.67%. A confusion matrix of the same is presented in Table 1. From the table, one can notice that the Kathak is giving the best classification accuracy (of 90.00%) when compared with Bharatnatyam (83.33%) and Odissi (86.67%). Moreover, confusion between Bharatnatyam and Odissi is much higher than that between Kathak and either of the former ones. The confusion matrix thus indicates the fact that there is a strong correlation between the poses of Bharatnatyam and Odissi as compared to the Kathak. In reality also, there are some poses which are common for both Bharatnatyam and Odissi. Since the proposed algorithm is based on a dictionary learning algorithm, in Table 2, we compare our results with the most popular bag-of-words model [26]. Table 2 shows that the proposed algorithm outperforms bag-of-words model. Moreover, it is significantly better than another related algorithm proposed in [21]. Where, in [21] they starts with a large vocabulary of poses (visual words) and derives a refined and

	bharatnatyam	kathak	odissi
bharatnatyam	83.33	3.33	13.33
kathak	6.67	90.00	3.33
odissi	10.00	3.33	86.67

Table 1. Confusion matrix of ICD dataset (in average accuracy (%))

Method	Average accuracy (%)
Key Pose [21]	78.50
Bag-of-Words [26]	82.30
Proposed method	86.67

Table 2. Compare with others model on ICD dataset.

	boxing	handclapping	handwaving	jogging	running	W
boxing	93	2	4	0	0	1
handclapping	0	97	3	0	0	0
handwaving	1	0	99	0	0	0
jogging	0	0	0	88	8	4
running	1	0	0	18	77	4
walking(W)	0	0	0	1	2	97

Table 3. Confusion matrix of KTH dataset (in average accuracy (%))

compact code book of key poses using centrality measure of graph connectivity. They have used a meaningful threshold on centrality measure that selects key poses for each action type. To represent a pose descriptor, they have used HOOF feature.

To see the efficacy of the proposed action descriptor for a more common action recognition problem, we have also tested our algorithm on KTH action dataset. This dataset consists of six different types of human actions: boxing, hand clapping, hand waving, jogging, running and walking. In KTH, 25 different persons performed each action in four different conditions (outdoors, outdoors with scale variation, outdoors with different clothes, and indoors). We apply the proposed algorithm on KTH database in the manner it has been done for ICD classification and get an average classification accuracy of 91.83%. A confusion matrix is shown in Table 3 which is comparable to the state-of-the art. The table establishes that the proposed descriptor is suitable for general human action recognition task.

4. Conclusion

We have presented have a new application of computer vision, i.e., classification of Indian classical dance (ICD). Beside presenting a novel classification problem, we pro-

pose a new action descriptor. The proposed descriptor not only classify ICD efficiently but also classify a common human actions. The results on KTH datbase proves the same. Since this is the first attempt of its kind, the dataset on ICD is relatively small. Though the dataset is small, the variation in terms of number of person involved, clothing, acquisition condition etc. makes the problem challenging. We are going to make our dataset available soon. We have a plan to increase the number of classes and number of videos in each class.

References

[1] http://www.di.ens.fr/willow/spams/.

[2] J. K. Aggarwal and M. S. Ryoo. Human activity analysis: A review. *ACM Computing Surveys(To appear)*, 2011.

[3] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri. Actions as space-time shapes. In *ICCV*, volume 2, pages 1395–1402, October 2005.

[4] Y.-L. Boureau, F. Bach, Y. LeCun, and J. Ponce. Learning mid-level features for recognition. In *CVPR*, pages 2559–2566, June 2010.

[5] A. M. Bruckstein, D. L. Donoho, and M. Elad. From sparse solutions of systems of equations to sparse modeling of signals and images. *SIAM Review*, 51(1):34–81, January 2009.

[6] R. Chaudhry, A. Ravichandran, G. Hager, and R. Vidal. Histograms of oriented optical flow and binet-cauchy kernels on nonlinear dynamical systems for the recognition of human actions. In *CVPR*, pages 1932–1939, June 2009.

[7] A. A. Efros, A. C. Berg, G. P. Mori, and J. Malik. Recognizing action at a distance. In *ICCV*, pages 726–733, October 2003.

[8] A. Gupta, P. Srinivasan, J. Shi, and L. S. Davis. Understanding videos, constructing plots: Learning a visually grounded storyline model from annotated videos. In *CVPR*, 2009.

[9] A. Kläser, M. Marszałek, C. Schmid, and A. Zisserman. Human focused action localization in video. In *International Workshop on Sign, Gesture, Activity*, 2010.

[10] T. Lan, Y. Wang, W. Yang, and G. Mori. Beyond actions: Discriminative models for contextual group activities. In *NIPS*, 2010.

[11] I. Laptev. On space-time interest points. *International Journal of Computer Vision*, 64(2):107–123, 2005.

[12] I. Laptev and T. Lindeberg. Velocity adaptation of space-time interest points. In *ICPR*, pages 52–56, September 2004.

[13] S. Lazebnik, C. Schmid, and J. Ponce. Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories. In *CVPR*, volume 2, pages 2169–2178, October 2006.

[14] B. D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *7th IJCAI*, pages 674–679, 1981.

[15] F. Lv and R. Nevatia. Single view human action recognition using key pose matching and viterbi path searching. In *CVPR*, pages 1–8, June 2007.

[16] J. Mairal, F. Bach, J. Ponce, and G. Sapiro. Online dictionary learning for sparse coding. In *ICML*, Montreal, Canada, June 2009.

[17] J. Mairal, F. Bach, J. Ponce, and G. Sapiro. Online learning for matrix factorization and sparse coding. *Journal of Machine Learning Research*, 11(9):19–68, 2010.

[18] S. Maji, A. C. Berg, and J. Malik. Classification using intersection kernel support vector machines is efficient. In *CVPR*, pages 1–8, June 2008.

[19] S. Maji, L. Bourdev, and J. Malik. Action recognition from a distributed representation of pose and appearance. In *CVPR*, pages 3177–3184, June 2011.

[20] R. Massey. India's dances. In *India's Dances*, 2004.

[21] S. Mukherjee, S. K. Biswas, and D. P. Mukherjee. Recognizing human action at a distance in video by key poses. *IEEE Transactions on Circuits and Systems for Video Technology*, 21(9):1228–1241, 2011.

[22] N. Nayak, R. Sethi, B. Song, and A. Roy-Chowdhury. Motion pattern analysis for modeling and recognition of complex human activities. *Visual Analysis of Humans: Looking at People, Springer*, 2011.

[23] M. S. Ryoo and J. K. Aggarwal. Recognition of composite human activities through context-free grammar based representation. In *CVPR*, pages 1709 – 1718, October 2006.

[24] H. Seo and P. Milanfar. Action recognition from one example. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 33(5):867–882, 2011.

[25] J. Van Gemert, C. Veenman, A. Smeulders, and J.-M. Geusebroek. Visual word ambiguity. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(7):1271–1283, July 2009.

[26] Y. Wang and G. Mori. Human action recognition by semi-latent topic models. *IEEE Trans. on Pattern Analysis and Machine Intelligence Special Issue on Probabilistic Graphical Models in Computer Vision*, 31(10):1762–1774, 2009.

[27] Y. Wang and G. Mori. Hidden part models for human action recognition: Probabilistic vs. max-margin. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 33(7):1310–1323, 2011.

[28] J. Yang, K. Yu, Y. Gong, and T. Huang. Linear spatial pyramid matching using sparse coding for image classification. In *CVPR*, pages 1794–1801, June 2009.

[29] W. Yang, Y. Wang, and G. Mori. Recognizing human actions from still images with latent poses. In *CVPR*, pages 2030–2037, June 2010.

[30] B. Yao, A. Khosla, and L. Fei-Fei. Classifying actions and measuring action similarity by modeling the mutual context of objects and human poses. In *ICML*, Bellevue, USA, June 2011.