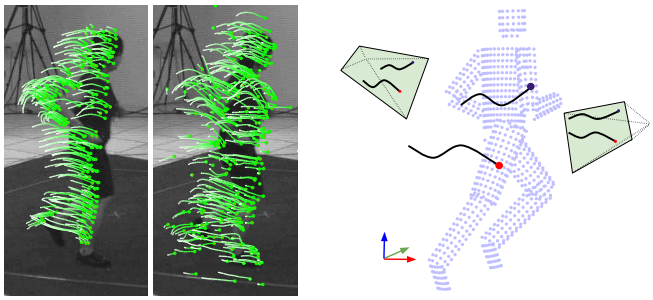
Unlabelled 3D Motion Examples Improve Cross-View Action Recognition

Ankur Gupta
ankgupta@cs.ubc.ca
Alireza Shafaei
shafaei@cs.ubc.ca
James J. Little
little@cs.ubc.ca
Robert J. Woodham
woodham@cs.ubc.ca

Department of Computer Science
University of British Columbia
Vancouver, Canada



(a) (b)

Figure 1: (a) We exploit the visual similarity between mocap-generated trajectories (left) and dense trajectories (right) to improve cross-view action recognition. (b) For mocap-trajectories, we can easily obtain corresponding features (i.e., descriptors for trajectories that originate from the same 3D point) in two views. We use these pairs of features to learn the transformation function for viewpoint change.

1 Overview

A view-invariant representation of human motion is crucial for effective action recognition. However, most view-invariant representations require either tracking of body parts or multi-view video data for learning which may not be a practical approach in many real-life scenarios. We describe a view-independent model for human action which is flexible, action-independent, and requires no multi-view video data or additional labelling effort.

We present a novel method for cross-view action recognition. Using a large collection of motion capture data we synthesize mocap-trajectory features from multiple viewpoints. Features originating from the same 3D point on the surface correspond, and this allows us to learn a feature transformation function for viewpoint change. Given this function, we can "hallucinate" the action descriptors of a video for different viewing angles. We use these hallucinated examples as additional training data to make our model view-invariant. We demonstrate the effectiveness of our approach on the unsupervised scenario of the INRIA IXMAS dataset.

2 Methodology

The approach has three steps:

Generating training data We adapt the mocap trajectory generation pipeline of Gupta *et al*. [1], which uses a human model with cylindrical primitives (see Figure 1(b)). Each limb consists of a collection of points that are placed on a 3D surface. Given a camera viewpoint, these points are projected under orthographic projection and tracked for L(=15) consecutive frames to generate trajectory descriptors similar to the dense-trajectories of Wang *et al*. [3]. The resulting displacement vectors are then used to generate trajectory features. Given two arbitrary viewpoints, we can find a correspondence between features that originate from the same point on the surface (see Figure 1(b)).

Learning the transformation function We quantize the mocap trajectory features using a fixed codebook $\mathcal{C}$ of size n. Given a source camera elevation angle θ and relative change in viewpoint given by $\Delta = (\delta\theta, \delta\phi)$, we define the training set $\mathcal{D}_\theta^\Delta = \{(f_i, g_i)\}_1^m$ to be the set of m pairs $(f,g) \in \mathcal{C} \times \mathcal{C}$, where f_i and g_i are the codewords for two corresponding trajectory features.

Given the training data $\mathcal{D}_\theta^\Delta$, we can learn a joint probability mass function $\mathrm{P}(F,G)$ which captures the probability of having feature pairs (f_i, g_i) in $\mathcal{D}_\theta^\Delta$. We calculate the empirical probability by counting the co-occurrences of (f_i, g_i) in $\mathcal{D}_\theta^\Delta$ followed by normalization. After observing an instance of codeword f_i in the source view, $\mathrm{P}(G|F = f_i)$ allows us to infer the possible outcomes in the target view.

Synthesizing cross-view descriptors Given a BoW descriptor of an action, we wish to synthesize another descriptor for a viewpoint $\Delta = (\delta\theta, \delta\phi)$ away from the original view. Let $\mathbf{x} = [x_1, \ldots, x_n]^\mathrm{T}$ be the BoW descriptor in the source view, and $\mathbf{y} = [y_1, \ldots, y_n]^\mathrm{T}$ be the descriptor we want to estimate. Using the probabilistic mapping between the codewords across views, we return an expected transformed descriptor

$$\bar{\mathbf{y}} = [\mathbb{E}[y_1], \ldots, \mathbb{E}[y_n]]^\mathrm{T} \text{ and } \mathbb{E}[y_j] = \sum_{i=1}^{n} x_i \cdot \mathrm{P}(G = f_j | F = f_i)$$

By organizing $\mathrm{P}(G|F)$ in the form of a matrix (say N) where the i-th row is the categorical distribution $\mathrm{P}(G|F = f_i)$, we can rewrite the above formulation as a matrix multiplication $\bar{\mathbf{y}} = \mathrm{N}^\mathrm{T}\mathbf{x}$. We further l_2 normalize $\bar{\mathbf{y}}$ to make it consistent with the original descriptor.

Method	Average accuracy
Ours	**71.7%**
nCTE based matching [1]	67.4%
w/o aug.	62.1%
Hankelets [2]	56.4%

Table 1: Average accuracy for action recognition over all view pairs of the INRIA IXMAS dataset. Given the training data from one viewing angle, the task is to recognize actions from a previously unseen viewpoint. We compare with other state-of-the-art methods. *w/o aug.* is our baseline without any data augmentation.

3 Experiments

To test our method we use the INRIA IXMAS dataset which has short view clips of 10 actors performing 11 activities (3 trials each) captured from 5 diverse angles. To learn the mapping between codewords, we generate mocap trajectories from multiple viewpoints and quantize them using the same codebook $\mathcal{C}$. We also quantize the viewpoints into 18 bins.

We synthesize multiple descriptors per training examples (one per viewpoint change), as described above, to augment our original training data. We train an SVM with χ^2 kernel using one-vs-all strategy. The main results are summarized in Table 1. Our code is publicly available: `http://cs.ubc.ca/research/motion-view-translation/`.

[1] Ankur Gupta, Julieta Martinez, James J. Little, and Robert J. Woodham. 3D Pose from Motion for Cross-view Action Recognition via Non-linear Circulant Temporal Encoding. In *CVPR*, 2014.

[2] Binlong Li, Octavia I. Camps, and Mario Sznaier. Cross-view Activity Recognition using Hankelets. In *CVPR*, 2012.

[3] Heng Wang, Alexander Klaser, Cordelia Schmid, and Cheng-Lin Liu. Action recognition by dense trajectories. In *CVPR*, 2011.