Does Human Action Recognition Benefit from Pose Estimation?

Angela Yao[1]
yaoa@vision.ee.ethz.ch
Juergen Gall[1]
gall@vision.ee.ethz.ch
Gabriele Fanelli[1]
fanelli@vision.ee.ethz.ch
Luc Van Gool[1,2]
vangool@vision.ee.ethz.ch

[1] Computer Vision Laboratory
ETH Zurich, Switzerland
[2] IBBT, ESAT-PSI
K.U. Leuven, Belgium

Abstract

Early works on human action recognition focused on tracking and classifying articulated body motions. Such methods required accurate localisation of body parts, which is a difficult task, particularly under realistic imaging conditions. As such, recent trends have shifted towards the use of more abstract, low-level appearance features such as spatio-temporal interest points. Motivated by the recent progress in pose estimation, we feel that pose-based action recognition systems warrant a second look. In this paper, we address the question of whether pose estimation is useful for action recognition or if it is better to train a classifier only on low-level appearance features drawn from video data. We compare *pose-based*, *appearance-based* and *combined* pose and appearance features for action recognition in a home-monitoring scenario. Our experiments show that pose-based features outperform low-level appearance features, even when heavily corrupted by noise, suggesting that pose estimation is beneficial for the action recognition task.

1 Introduction

Human action recognition is an active research topic within the computer vision community. Development has been driven by the potential for many applications such as human-computer interaction, content-based video indexing, intelligent surveillance, and assisted living. Some of the earliest works in action recognition focused on tracking body parts and classifying the joint movements [6, 12, 32]. These *pose-based approaches* stem directly from the definition of an action as a sequence of articulated poses and are the most straightforward. However, they require accurate tracking of body parts, which is a notoriously challenging task in its own right. As recent trends in action recognition have shifted towards analysis in natural and unconstrained videos, such as sequences from feature films [17], broadcast sports [21] and Youtube [19], efforts have moved from high-level modelling of the human body to directly classifying actions with abstract and low-level appearance features [6, 7, 13, 16, 24, 31] in *appearance-based approaches*.

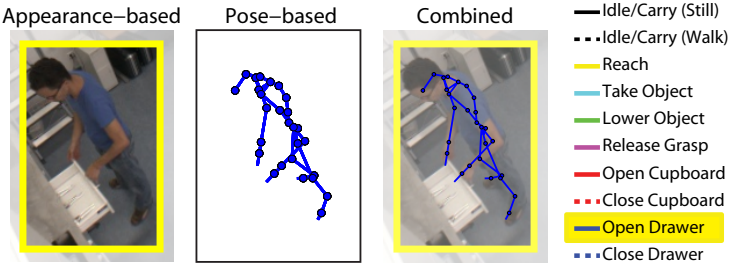


Figure 1: We address the question of whether it is useful to perform pose estimation for the task of action recognition by comparing the use of appearance-based features, pose-based features and combined appearance- and pose-based features.

Appearance-based methods require little to no high-level processing and can bypass the difficulties of pose estimation. They can also take some contextual information into account (*e.g.*, background), since features are not restricted to the human body. And despite having to deal with great intra-class variations, such as human appearance, background clutter and differing viewpoints, appearance-based systems are applicable in scenarios where pose estimation is difficult (*e.g.* monocular views) or even impossible (*e.g.* very low resolutions [7]).

Pose-based action recognition approaches have received little attention in recent years due to the inherent difficulty of extracting human pose, particularly under realistic imaging conditions. But despite requiring more initial processing, these approaches have several advantages. First, pose representations suffer little of the intra-class variances that plague appearance-based systems. In particular, 3D skeleton poses are viewpoint and appearance invariant, such that actions vary less from actor to actor. Secondly, using pose representations greatly simplifies the learning for the action recognition itself, since the relevant high-level information has already been extracted. Given the great progress in pose estimation over the past few years [1, 9, 10, 18, 26], we feel that pose-based action recognition systems warrant a second look. More importantly, we address the question of whether it is useful to perform pose estimation for the action recognition task or if it is better for the classifier to identify the necessary information only from low-level appearance features drawn from video data.

In this work, we compare appearance and pose-based features for action recognition as depicted in Fig. 1. Pose-based features are derived from articulated 3D joint information, while we label as appearance-based any feature which can be extracted from video data without explicit articulated human body modelling. We apply the same action classifier [33] to the two different sets of features and investigate their combination into a single system.

2 Related Work

Early works in recognising human motions relied on recovering the articulated poses from each frame and then linking either the poses or pose-derived features into sequences. Pose information was typically obtained from moving light displays [12], motion capture systems [5] or segmentation [23, 32]. The sequences themselves were then classified through exemplar matching [12, 23, 32] or with state-space models such as HMMs [5].

An alternative line of work models the entire body as a single entity, using silhouettes or visual hulls [2, 4, 20, 29, 30]. These works are sometimes referred to as pose-based approaches, in reference to the extracted silhouettes of the human body. However, we consider

silhouettes to be a specialised appearance feature, since it offers little interpretation of the individual body parts, and categorise these works as appearance-based approaches.

To avoid articulated tracking or segmentation, recent works have shifted towards the use of local, low-level appearance features such as Gabor filter responses [13, 24] and optical flow [7]. Lately, spatio-temporal interest points have become especially popular, *e.g.* cuboids [6], 3D Harris corners [16] and 3D Hessians [31]. As extensions of their 2D counterparts used in object detection, their usage follows a traditional object detection approach. After interest point detection at multiple scales, feature descriptors are computed, clustered, and assigned to a code-book to be used in some bag-of-words representation [6, 17, 19].

In an attempt to bring back the "human" to human action recognition, works such as [10, 14, 28, 33] have tried to couple person detectors with the action recognition task. Even though these works still fall under the appearance-based class, they focus on features that are related to the human pose. In particular, we were inspired by [10], which used action recognition to help human pose estimation. Here, we pose the inverse question of whether pose estimation can be beneficial for action recognition.

3 Methods

For classifying the actions, we use the Hough-transform voting method of [33], which provided state-of-the-art results while being easily adaptable to use different features. In [33], a *Hough forest* [8] was trained to learn a mapping between appearance-based feature patches and corresponding votes in an action Hough space. Each tree T in the forest is constructed from a set of patches $\{\mathscr{A}_i = (\mathscr{I}_i, c_i, d_i)\}$ randomly sampled from the training sequences. $\mathscr{A}_i$ is a 3D spatio-temporal patch sampled from a normalized track. $\mathscr{I}_i = \left(I_i^1, I_i^2, ..., I_i^F\right)$ are the F feature channels extracted at patch i, c_i is the action label ($c_i \in C$) and d_i is the temporal displacement of the patch centre to the action centre in the sequence.

Non-leaf nodes in a tree store a binary test; during training, the test associated to a node is selected among a large number of randomly generated tests so that the resulting split of the training patches maximises a desired measure. In [33], there is a random selection between a measure of class-label uncertainty and centre offset uncertainty. The process iterates until a leaf is created, either from reaching a maximum depth or from having too few patches remaining. Leaves store the proportion of patches per class which reached the leaf L ($p_c{}^L$) and the patches' respective displacement vectors ($D_c{}^L$).

Our tests compare pixels at locations p and $q \in \mathbb{R}^3$ (within the spatio-temporal patch) in feature channel f, with an offset τ:

$$t(f;p,q;\tau) = \begin{cases} 0 & \text{if } I^f(p) < I^f(q) + \tau \\ 1 & \text{otherwise} \end{cases} \quad (1)$$

At classification time, patches are densely extracted from the test track and passed through all trees in the forest. The patches are split according to the binary tests stored in the non-leaf nodes and, depending on the reached leaf, cast votes proportional to p_c for the action label and the temporal centre of each class c.

We use the publicly available source code[1] and apply it directly to our appearance-based features experiments. We then modified the code to accept pose-based features (see Section 3.2) as well as combined features (see Section 3.3).

[1] http://www.vision.ee.ethz.ch/~yaoa

(a) colour (b) dense optical flow (c) spatial gradients (d) temporal gradient

Figure 2: Appearance-based features. *(a)* Colour in the Lab colour space. *(b)* Dense optical flow in x and y. *(c)* Spatial gradients in x and y. *(d)* Temporal gradient.

3.1 Appearance-based features

As per [33], we use low-level appearance-based features, such as colour (Fig. 2*(a)*), dense optical flow [9] (Fig. 2*(b)*), and spatio-temporal gradients (Fig. 2*(c,d)*). While more sophisticated spatio-temporal features exist in the literature, we omit them from our experimentation as [33] showed that the above-mentioned low-level features achieve comparable results.

3.2 Pose-based features

One of the biggest challenges of using posed-based features is that semantically similar motions may not necessarily be numerically similar [15, 22]. As such, we do not directly compare 3D skeleton joints in space and time. Instead, we use relational pose features describing geometric relations between specific joints in a single pose or a short sequence of poses. Relational pose features, introduced in [22], have been used for indexing and retrieval of motion capture data; we modify a subset of them for use in the random forest framework.

Let $p_{j_i,t} \in \mathbb{R}^3$ and $v_{j_i,t} \in \mathbb{R}^3$ be the 3D location and velocity of joint j_i at time t. The joint distance feature F^{jd} (see Fig. 3*(a)*) is defined as the Euclidean distance between joints j_1 and j_2 at time t_1 and t_2 respectively:

$$F^{jd}(j_1, j_2; t_1, t_2) = \|p_{j_1,t_1} - p_{j_2,t_2}\|, \tag{2}$$

If $t_1 = t_2$, then F^{jd} is the distance between two joints in a single pose; if $t_1 \neq t_2$, then F^{jd} would encode distances between joints separated by time.

The plane feature F^{pl} (see Fig. 3*(b)*) is defined as

$$F^{pl}(j_1, j_2, j_3, j_4; t_1, t_2) = \text{dist}\Big(p_{j_1,t_1}, \langle p_{j_2,t_2}, p_{j_3,t_2}, p_{j_4,t_2}\rangle\Big), \tag{3}$$

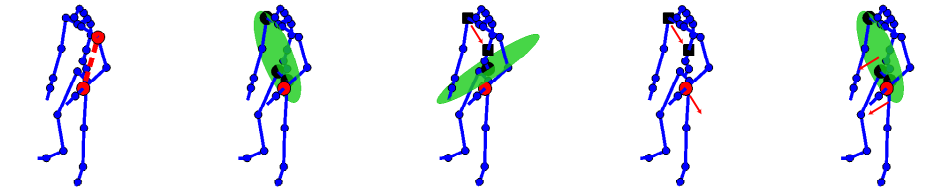
where $\langle p_{j_2}, p_{j_3}, p_{j_4}\rangle$ indicates the plane spanned by p_{j_2}, p_{j_3}, p_{j_4}, and $\text{dist}(p_j, \langle\cdot\rangle)$ is the distance from point p_j to the plane $\langle\cdot\rangle$. Similarly, the normal plane feature F^{np} (see Fig. 3*(c)*) is defined as

$$F^{np}(j_1, j_2, j_3, j_4; t_1, t_2) = \text{dist}\Big(p_{j_1,t_1}, \langle p_{j_2,t_2}, p_{j_3,t_2}, p_{j_4,t_2}\rangle_n\Big), \tag{4}$$

where $\langle p_{j_2}, p_{j_3}, p_{j_4}\rangle_n$ indicates the plane with normal vector $p_{j_2} - p_{j_3}$ passing through p_{j_4}.

The velocity feature F^{ve} (see Fig. 3*(d)*) is defined as the component of v_{j_1,t_1} along the direction of $p_{j_2} - p_{j_3}$ at time t_2:

$$F^{ve}(j_1, j_2, j_3; t_1, t_2) = \frac{v_{j_1,t_1} \cdot (p_{j_2,t_2} - p_{j_3,t_2})}{\|(p_{j_2,t_2} - p_{j_3,t_2})\|}. \tag{5}$$



(a) joint distance (b) plane (c) normal plane (d) velocity (e) normal velocity

Figure 3: Pose-based features. *(a)* Euclidean distance between two joints (red). *(b)* Plane feature: distance between a joint (red) and a plane (defined by three joints - black). *(c)* Normal plane feature: same as plane feature, but the plane is defined by its normal (direction of two joints - black squares) and a joint (black circle). *(d)* Velocity feature: velocity component of a joint (red) in the direction of two joints (black). *(e)* Normal velocity feature: velocity component of a joint in normal to the plane defined by three other joints (black).

Similarly, the normal velocity feature F^{nv} (see Fig. 3*(e)*) is defined as the component of v_{j_1,t_1} in the direction of the normal vector of the plane spanned by p_{j_2}, p_{j_3} and p_{j_4} at time t_2:

$$F^{nv}(j_1,j_2,j_3,j_4;t_1,t_2) = v_{j_1,t_1} \cdot \hat{n}_{\langle p_{j_2,t_2}, p_{j_3,t_2}, p_{j_4,t_2} \rangle}, \tag{6}$$

where $\hat{n}_{\langle\cdot\rangle}$ is the unit normal vector of the plane $\langle\cdot\rangle$.

Any of the above features can be easily incorporated into the random forest framework by sampling "pose patches", *i.e.* $\{\mathscr{P}_i = (P_i, V_i, c_i, d_i)\}$, where P_i and V_i are consecutive frames of skeleton location and velocity respectively and modifying the binary tests (see Eq. (1)):

$$t(f;j_1,..,j_n;t_1,t_2;\tau) = \begin{cases} 0 & \text{if } F^f(j_1,..,j_n;t_1,t_2) < \tau \\ 1 & \text{otherwise} \end{cases}, \tag{7}$$

where f, $j_1 ... j_n$, t_1, t_2, τ are randomly selected pose-based feature types, joints, times and thresholds respectively.

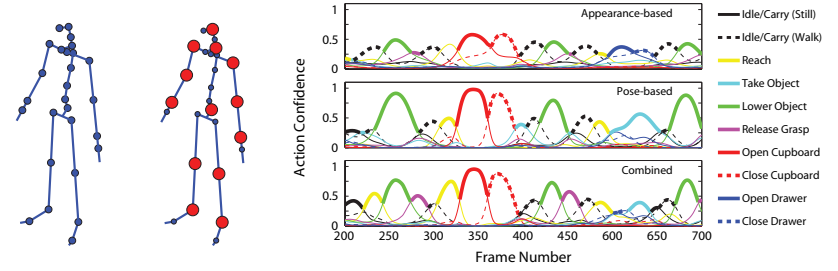
3.3 Combined features

For combining appearance and pose-based features, we created combined patches $\{\mathscr{C}_i = (\mathscr{A}_i, \mathscr{P}_i)\}$ with both appearance and pose information. Note that $\mathscr{A}_i$ are patches sampled in space and time, while $\mathscr{P}_i$ are poses sampled in time only[2]. When generating the binary tests, we randomly determine whether to use appearance or pose features. In this way, the classifier automatically selects the most relevant features.

3.4 Dataset and experimentation

We focused our comparison on a home-monitoring scenario and used the TUM kitchen dataset [27], with multi-view video data and corresponding motion capture of actors setting a table. The actions are relatively realistic, subtler, and thus more challenging than standard benchmarks [2, 25]. The fixed camera views eliminate much of the problems associated with background variance for appearance-based methods. Even though we used the provided 3D joint positions, these were determined by a markerless motion capture system [1], *i.e.*, not measured directly from markers, exemplifying state-of-the-art pose estimation results.

[2] the same time as $\mathscr{A}_i$



(a) Full skeleton (b) Reduced skeleton (c) Action confidences

Figure 4: *(a)* 27-joint full skeleton. *(b)* 13-joint reduced skeleton (in red). *(c)* Normalised action confidences for appearance-based, pose-based, and combined features for frames 200-700 of episode *0-8*. In general, action confidences are higher for pose features than appearance features; given that they are also more accurate (see Table 1), this suggests that pose-based features are more discriminative than appearance-based features.

For each type of feature, we trained a forest of 15 trees of depth 15 each, generating 20000 random tests at all nodes. Of the 20 episodes in the dataset, we used *0-2,0-8,0-4,0-6,0-10,0-11,1-6* for testing and the remaining 13 episodes for training, from which we extracted 40 or less instances per action class. We normalised the output of the Hough forest into a confidence score of each action over time, such that all actions at any time sum up to 1. To maintain consistency with [10, 27], the two other works using this dataset, we employed action recognition labels for the left hand as ground truth. As per [10], we also split the idle/carry class according to whether the subject is walking or standing.

For appearance-based features, we generated silhouettes using background subtraction and thus extracted bounding boxes which we linked into tracks. For each training instance, we randomly selected 1200 patches of size $15 \times 15 \times 5$. We trained independent forests for each camera view and then used a classifier-combination strategy (max-rule) to combine the outputs from the multiple views as per [10].

For each type of pose-based feature, we trained a Hough forest on the full 3D skeletons provided (without limb endpoints) [27] as well as a reduced set of 13 joints (see Fig. 4*(a,b)*). To simulate less-than-optimal pose estimations, we also tested the classifier on joint data corrupted by Gaussian noise. For all pose-based feature experiments, we used 200 "pose patches" per training instance[3], with a time duration of 5 frames.

For the combined features, all settings were kept the same as for the appearance-based features. When randomly generating the binary tests, half were for appearance features and half for pose features, leaving the classifier to determine the optimal test and feature type.

4 Results

4.1 Appearance-based features

Using the appearance features described in Section 3.1, we achieved a combined performance of 0.698. A sample of the normalised classifier output is shown in Fig. 4*(c)* and

[3]The possible number of unique "pose patches" is much smaller than that of appearance-based features patches.

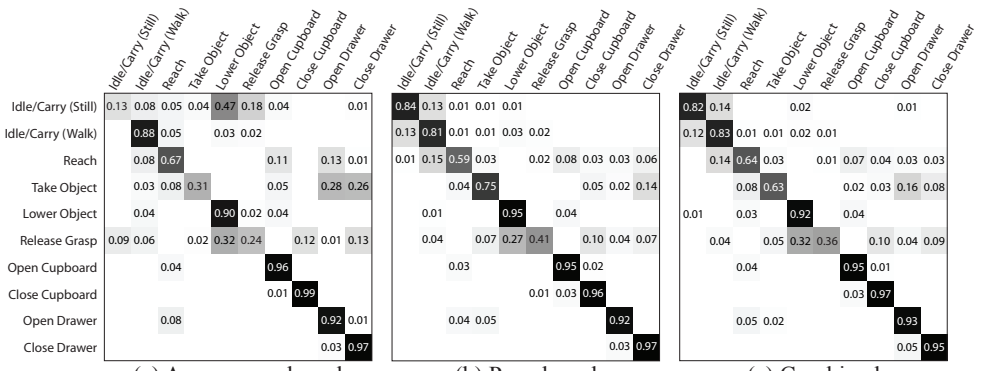


Figure 5: Confusion matrices for appearance-based, pose-based and combined features for action recognition, with mean classification rates of 0.698, 0.815 and 0.801 respectively. While "release grasp" is the most confused action in all cases, classes like "take object" or "idle/carry (still)" significantly improve with the introduction of pose-based features.

a confusion matrix for the individual classes is shown in Fig. 5*(a)*. Of the different features tested, colour was selected most often in the binary tests assigned to the nodes (43%, Fig 8*(a)*) in comparison to gradients (36%) and optical flow (21%).

4.2 Pose-based features

All pose-based features outperformed the appearance-based features by 7-10%. Of the different types of pose-based features tested on the full skeleton, velocity features and plane features have comparable results, slightly higher than that of the joint distance (see Table 1). Combining all three feature types yielded the best result of 0.815, with the confusion matrix shown in Fig. 5*(b)*. For the reduced skeleton, results are all slightly lower than or equal to that of the full skeleton. The slight performance loss is probably not only due to the reduced number of joints but also due to the changed distribution of the joints on the skeleton (*e.g.* spine and hip in Fig. 4*(a,b)*). When combining several features, the performance does not improve by much and is sometimes even lower than that of the best feature (see Table 1); this behaviour has been observed for random forests when the number of redundant feature channels increases [□]. When all features are used together, the selection of the features at the nodes is nearly equal (Fig. 8*(b)*).

When using only joint-distance features, we examined which of the joints were selected at the nodes according to the different actions (Fig. 6). While different joints are favoured for the different actions, they are not immediately intuitive (*e.g.* joints from the legs or feet are not always selected for walking), suggesting that joints not associated with the limbs performing the action can also encode some information for discriminating the various actions.

Pose Features	Full skeleton (27 joints)	Reduced Skeleton (13 joints)
joint distance (Fig. 3 *(a)*)	0.777	0.733
plane features (Fig. 3 *(b)* & *(c)*)	0.802	0.787
velocity features (Fig. 3 *(d)* & *(e)*)	0.803	**0.803**
joint distance & plane features	0.784	0.769
joint distance & velocity features	0.800	0.774
plane features & velocity features	0.804	0.773
all features	**0.815**	0.776

Table 1: Classification performance with different pose-based features.

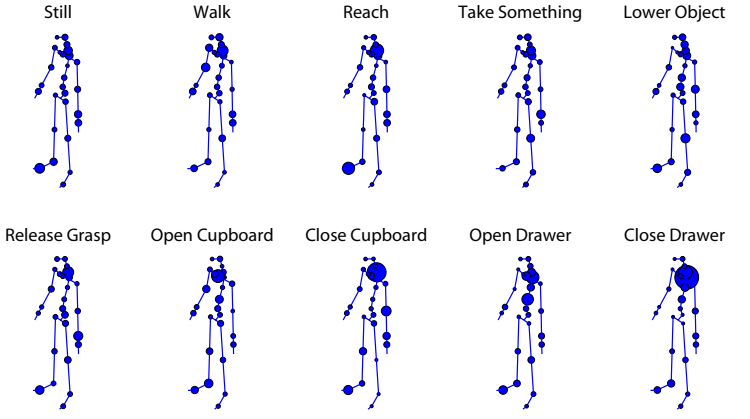


Figure 6: Joints selected by binary tests assigned to nodes of the Hough forest, where size of the plotted joint on the stick figure corresponds to frequency of selection. Note that each test uses two joints as the relational pose feature.

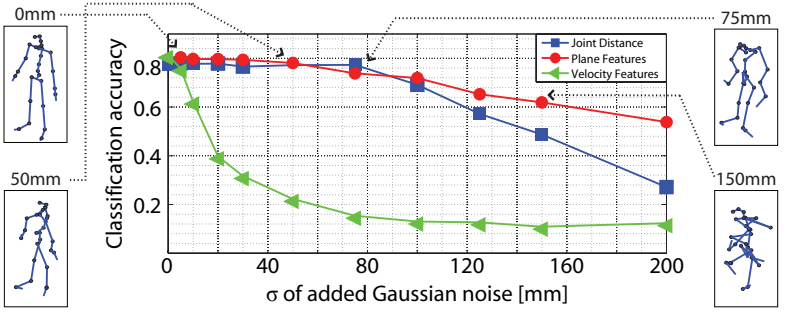
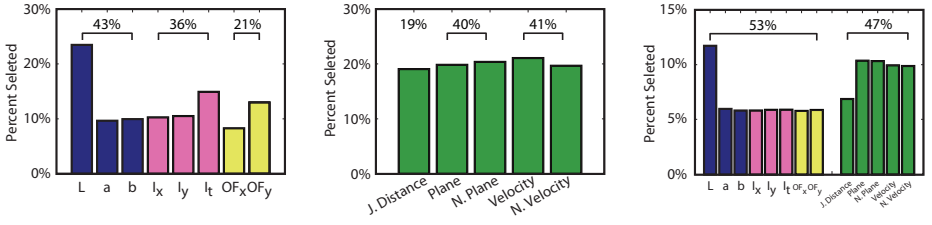


Figure 7: Classification accuracy as a function of the Gaussian noise added to the joint 3D locations. Velocity features are badly affected, while the other pose-based features slowly degrade in performance. The skeletons on the sides visualise the amount of noise added.

Finally, we tested the robustness of the pose-based features by corrupting the test joint data with Gaussian noise to simulate errors in the extracted poses; classification accuracy versus noise is plotted in Fig. 7. Performance of velocity features drops off quickly; joint distance and plane features, however, are more robust and maintain almost the same performance until around 75mm of added noise on each joint. At 100mm of added noise, performance is about equal to that of the appearance-based features.

4.3 Combined appearance- and pose-based features

We found no improvements after combining the appearance-based and pose-based features and achieve a mean classification of 0.801. The confusion matrix for combined outputs are shown in Fig. 5*(c)*. Looking at the normalised classifier output (Fig. 4*(c)*), we see that the output is almost the same as that of the pose-based classifier, suggesting a strong favouring of the pose-based features. When looking at the features selected at the nodes of trees (Fig. 8*(c)*), however, appearance features are still selected 53% of the time, suggesting a high



(a) Appearance-based (b) Pose-based (c) Combined

Figure 8: Feature selection. *(a)* In the appearance-based classifier, colour features (L,a,b) were selected at 43% of the nodes, gradient features (I_x, I_y, I_t) at 36% and optical flow (OF_x, OF_y) at 21%. *(b)* In the pose-based classifier, joint distance was selected at 19% of the nodes, plane features at 40% and velocity features at 41%. *(c)* In the combined classifier, appearance features were selected at 53% of the nodes and pose-based features at 47%.

redundancy in the pose and appearance features. One may still expect improvement from the combination, since the image appearance covers more than the human pose (Fig.1). Fig. 5, however, reveals that the action classes which involve interaction with the environment (cupboard or drawer) are already accurately estimated by the pose features.

5 Conclusion

In this paper, we raised the question of whether it is useful to perform pose estimation for the action recognition task or if it is better for the classifier to identify the necessary information from low-level appearance features drawn from video data. Our results showed that, using the same classifier on the same dataset, pose-based features outperform appearance features. While pose-based action recognition is often criticised for requiring extensive preprocessing for accurate segmentation and tracking of the limbs, we have shown that this is not necessarily the case. Even with high levels of noise (up to 100mm of additive Gaussian noise), the pose-based features either matched or outperformed appearance-based features, indicating that perfect pose estimates are not necessary.

On the other hand, appearance features are more versatile to use than pose features and can be applied in many cases in which poses cannot be extracted. In addition, appearance-based features are capable of encoding contextual information (*e.g.* the appearance of the cupboards and drawers) which are missing from the poses alone. We believe that a combination of appearance and pose features would be most ideal when actions cannot be classified by the pose alone though this was not the case in our experiments. However, the question remains whether contextual information should be better learned from low-level or from high-level information extracted from the data. From looking at the most confused action class ("release grasp"), we observe that actions are often defined on high level information which is very difficult to learn from low-level features directly.

Acknowledgements This research was supported by funding from the Swiss National Foundation NCCR project IM2 and EU projects RADHAR and TANGO. Angela Yao was also supported by funding from NSERC Canada.

References

[1] J. Bandouch and M. Beetz. Tracking humans interacting with the environment using efficient hierarchical sampling and layered observation models. In *Int. Workshop on Human-Computer Interaction*, 2009.

[2] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri. Actions as space-time shapes. In *ICCV*, 2005.

[3] A. Bobick and J. Davis. The recognition of human movement using temporal templates. *TPAMI*, 23(3):257–267, 2001.

[4] T. Brox, A. Bruhn, N. Papenberg, and J. Weickert. High accuracy optical flow estimation based on a theory for warping. In *ECCV*, 2004.

[5] L. Campbell and A. Bobick. Recognition of human body motion using phase space constraints. In *ICCV*, 1995.

[6] P. Dollar, V. Rabaud, G. Cottrell, and S. Belongie. Behavior recognition via sparse spatio-temporal features. In *Int. Workshop on Visual Surveillance and Performance Evaluation of Tracking and Surveillance (VS-PETS)*, 2005.

[7] A. Efros, A. Berg, G. Mori, and J. Malik. Recognizing action at a distance. In *ICCV*, 2003.

[8] J. Gall and V. Lempitsky. Class-specific hough forests for object detection. In *CVPR*, 2009.

[9] J. Gall, B. Rosenhahn, T. Brox, and H.-P. Seidel. Optimization and filtering for human motion capture – a multi-layer framework. *IJCV*, 87:75–92, 2010.

[10] J. Gall, A. Yao, and L. Van Gool. 2d action recognition serves 3d human pose estimation. In *ECCV*, 2010.

[11] M. Gashler, C. Giraud-Carrier, and T. (2008) Martinez. Decision tree ensemble: Small heterogeneous is better than large homogeneous. In *ICML*, 2008.

[12] D. Gavrila and L. Davis. Towards 3-d model-based tracking and recognition of human movement: a multi-view approach. In *Int. Workshop on Face and Gesture Rec.*, 1995.

[13] H. Jhuang, T. Serre, L. Wolf, and T. Poggio. A biologically inspired system for action recognition. In *ICCV*, 2007.

[14] A. Kläser, M. Marszałek, C. Schmid, and A. Zisserman. Human focused action localization in video. In *Int. Workshop on Sign, Gesture, and Activity (SGA)*, 2010.

[15] L. Kovar and M. Gleicher. Automated extraction and parameterization of motions in large data sets. *ACM Trans. Graph.*, 23:559–568, 2004.

[16] I. Laptev and T. Lindeberg. Space-time interest points. In *ICCV*, 2003.

[17] I. Laptev, M. Marszałek, C. Schmid, and B. Rozenfeld. Learning realistic human actions from movies. In *CVPR*, 2008.

[18] R. Li, T.P. Tian, S. Sclaroff, and M.H. Yang. 3d human motion tracking with a coordinated mixture of factor analyzers. *IJCV*, 87:170–190, 2010.

[19] J.G. Liu, J.B. Luo, and M. Shah. Recognizing realistic actions from videos "in the wild". In *CVPR*, 2009.

[20] F. Lv and R. Nevatia. Single view human action recognition using key pose matching and viterbi path searching. In *CVPR*, 2007.

[21] M.D.Rodriguez, J. Ahmed, and M. Shah. Action mach: A spatio-temporal maximum average correlation height filter for action recognition. In *CVPR*, 2008.

[22] M. Müller, T. Röder, and M. Clausen. Efficient content-based retrieval of motion capture data. *ACM Trans. Graph.*, 24:677–685, 2005.

[23] C. Rao, A. Yilmaz, and M. Shah. View-invariant representation and recognition of actions. *IJCV*, 50(2):203–226, 2002.

[24] K. Schindler and L. Van Gool. Action snippets: How many frames does human action recognition require? In *CVPR*, 2008.

[25] C. Schuldt, I. Laptev, and B. Caputo. Recognizing human actions: A local svm approach. In *ICPR*, 2004.

[26] G. Taylor, L. Sigal, D. Fleet, and G. Hinton. Dynamical binary latent variable models for 3d human pose tracking. In *CVPR*, 2010.

[27] M. Tenorth, J. Bandouch, and M. Beetz. The TUM kitchen data set of everyday manipulation activities for motion tracking and action recognition. In *IEEE Workshop on THEMIS*, 2009.

[28] C. Thurau and V. Hlavac. Pose primitive based human action recognition in videos or still images. In *CVPR*, 2008.

[29] D. Weinland and E. Boyer. Action recognition using exemplar-based embedding. In *CVPR*, 2008.

[30] D. Weinland, E. Boyer, and R. Ronfard. Action recognition from arbitrary views using 3d exemplars. In *ICCV*, 2007.

[31] G. Willems, J.H. Becker, T. Tuytelaars, and L. Van Gool. Exemplar-based action recognition in video. In *BMVC*, 2009.

[32] Y. Yacoob and M.J. Black. Parameterized modeling and recognition of activities. *CVIU*, 73(2):232–247, 1999.

[33] A. Yao, J. Gall, and L. Van Gool. A hough transform-based voting framework for action recognition. In *CVPR*, 2010.