Posture Recognition with a Top-view Camera

Hu, N.; Englebienne, G.; Kröse, B.

DOI
10.1109/IROS.2013.6696657

Publication date
2013

Document Version
Final published version

Published in
2013 IEEE/RSJ International Conference on Intelligent Robots and Systems: IROS 2013: Tokyo, Japan, 3-7 November 2013

Posture Recognition with a Top-view Camera

Ninghang Hu[1], Gwenn Englebienne[1], and Ben Kröse[1,2]

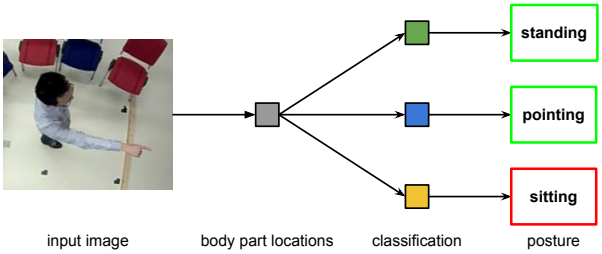
***Abstract*— We describe a system that recognizes human postures with heavy self-occlusion. In particular, we address posture recognition in a robot assisted-living scenario, where the environment is equipped with a top-view camera for monitoring human activities. This setup is very useful because top-view cameras lead to accurate localization and limited inter-occlusion between persons, but conversely they suffer from body parts being frequently self-occluded. The conventional way of posture recognition relies on good estimation of body part positions, which turns out to be unstable in the top-view due to occlusion and foreshortening. In our approach, we learn a *posture descriptor* for each specific posture category. The posture descriptor encodes how well the person in the image can be 'explained' by the model. The postures are subsequently recognized from the matching scores returned by the posture descriptors. We select the state-of-the-art approach of pose estimation as our posture descriptor. The results show that our method is able to correctly classify 79.7% of the test sample, which outperforms the conventional approach by over 23%.**

I. INTRODUCTION

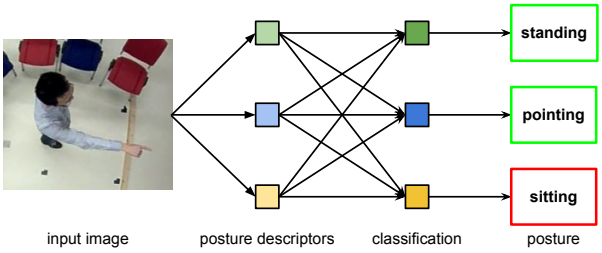
Human posture recognition is one of the most important tasks for human-robot interactions (HRI), as it provides a solid base for human activity recognition [1], [2], [3]. There are many papers on recognizing human posture with robot sensors or ambient cameras in 2D [4], 2.5D (RGB+Depth) [5] and 3D [6]. Most of the 2D approaches observe humans from a side-view, however, and recognizing human posture from the top-view still remains a challenging and unsolved problem.

For the purposes of this paper, *posture recognition* is defined as the process of assigning semantic posture labels to people in an image, *e.g.* whether people are standing, sitting, bending or pointing. In contrast, *pose estimation* is to estimate the configuration of the body parts [7], which focuses on getting the accurate body part locations rather than on posture labeling. Similarly, *pose* refers to a configuration of the body parts, and *posture* refers to a category of poses that bare the same semantic label.

In this paper, we present a system that recognizes human postures from the still images captured by a top-view camera. An overview of our system is shown in Fig. 1b. Our interest in this problem stems from a robot assisted-living scenario, where we use ceiling-mounted cameras as part of a domestic monitoring system to inform the robot on the human activities. Compared with robot-mounted sensors, top-view cameras give a good overview of the overall scene and a large amount of information about the person. Besides, the top-view cameras also provide a better estimation of the human locations and allow for far less inter-occlusion between persons when compared to side-views and robot-mounted sensors. Top-view cameras do, however, suffer more from a different form of occlusion than side-view cameras, *i.e.* self-occlusion.



(a) traditional approach



(b) our approach

Fig. 1. A comparison of the (a) traditional approach and (b) our approach. The traditional approach classifies posture categories based on the estimated body part locations. In contrast, our proposed system uses the matching scores from the *posture descriptors*. By combining the matching scores from all posture descriptors into a single feature vector, we apply a binary classifier to determine whether the image belongs to a certain posture category or not.

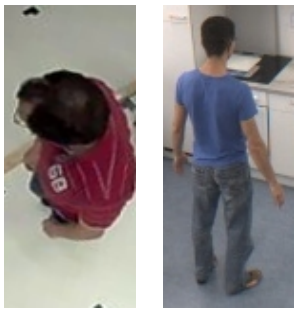
We distinguish between two types of occlusion: inter-occlusion and self-occlusion. Inter-occlusion refers to an object being blocked by another, *e.g.* when the view of a person is partially blocked by the person in front. In contrast, self-occlusion means that the object is occluded by itself, *e.g.* the limbs are occluded by the head and the torso. The two types of occlusions both happen in the side-view and the top-view, but at different levels. The inter-occlusion is more frequent in the side-view because other persons also stand at the same height level. In contrast, self-occlusion is more severe in the top-view (see Fig. 2). Most literature on posture recognition addresses the side-view and neglects the problem of the top-view.

In this paper, we focus on recognizing human postures

The research has received funding from the European Union's Seventh Framework Programme (FP7/2007-2013) under grant agreement No. 287624, and partly from the SIA project BALANCE-IT.

[1] N. Hu, G. Englebienne, and B. Kröse are with Intelligent System Lab Amsterdam, University of Amsterdam, 1098XH Amsterdam, The Netherlands {n.hu,g.englebienne,b.j.a.krose} @ uva.nl

[2] B. Kröse is also with the Amsterdam University of Applied Science



(a) top-view (b) side-view

Fig. 2. Posture recognition from the top-view (a) is a more challenging task than from the side-view (b) due to the severe self-occlusion.

under the severe self-occlusion seen in top-view images. The conventional approach is to firstly estimate the human pose configuration, and then classifies postures based on the body part positions [8]. In top-view images, people are largely self-occluded. With little information about the body part locations, recovering an articulated pose from these images is already a quite difficult task even for human annotators, let alone to further derive the posture category based on the ambiguous body part locations.

Recent work shows that, when the joint positions are accurately known, the best performance in posture recognition is obtained from the 3D joint positions [8]. In our approach, we recognize the human posture without explicitly knowing the exact location of body parts, and we will show that, in the case of heavy self-occlusion, this approach outperforms joint position based posture recognition. Unlike the conventional approach which classifies postures based on the body part locations, our idea is to use *posture descriptors* instead for classification. A posture descriptor provides a mapping from image features to the matching score of a posture category. Given a new test image, each posture descriptor gives a matching score that measures how well the person can be explained by that posture descriptor. For example, the standing posture descriptor returns a higher value when applied to standing people, and lower values on the others. Note that the posture categories overlap. For instance, a standing person may be also pointing. Our posture descriptors encode such attributes in a natural way by enabling multiple data labels to be applied to a single image. Fig. 1 compares our proposed system with the conventional approach.

In this work, we address the following research questions:

1) *Is 2D pose competitive with 3D pose for posture recognition?* Posture recognition from (perfect) 3D pose has been shown to outperform appearance-based approaches. We show that the performance of posture recognition with 2D pose is virtually identical to 3D pose, including for top-view projections.
2) *How accurately can we obtain 2D pose from top-view images?* To investigate this, we apply a state-of-the-art 2D pose estimation algorithm to the top-view images. We show that the performance is generally very low, but the specific models that are trained on a particular posture category perform comparably better.
3) *How accurately can we recognize posture from imperfect 2D pose, and how does this performance compare to our proposed model?* We show that our proposed model based on posture descriptors significantly outperforms the baseline, which consists of two state-of-the-art approaches.

II. RELATED WORK

Previous work on human posture recognition is mostly based on the images taken from the side-view. The top-view, which has been extensively used in domestic monitoring, receives surprisingly little attention.

Only recently did researchers start to work on the top-view to classify human postures [9], [10]. These approaches use the silhouettes of humans, which are extracted by background subtraction and represented as a vector of features. The features include the height-width ratio, the position, and the polar histograms of the silhouettes. These approaches rely on accurate foreground-background segmentation, which is difficult to obtain in practice due to noise, the change of lighting conditions or incorrect segmentation of the foreground blobs.

The more conventional method of posture recognition relies on side-view images to perform pose estimation and then predicts posture categories based on the estimated articulated pose. The state-of-the-art approach estimates body part locations using the Histogram of Oriented Gradient (HOG) features [11], and fits a human skeleton model to still images. In the human skeleton model, the joints of articulations are represented as body part detectors, and two joints are positioned in a way that the deformation costs are minimized.

To perform posture recognition, [8] assumes that the body part locations are known and transforms the 3D body part locations into a feature vector of geometric distances. The postures are then recognized using a random forest. The results are compared with the approach in [3], where a Hough Forest [12] was trained to learn the mapping from appearance patches to action labels. The results show that the pose-based distance features outperform the appearance-based features.

From the top-view, the body parts become largely self-occluded, which makes conventional approaches less suitable. It is very difficult to estimate the body part locations accurately from top-view images, and the resulting posture recognition performance is substandard. To solve this problem, we perform posture recognition with the matching scores from [11] instead of using the estimated poses. In this way, the exact body part locations need not to be extracted accurately for recognizing the postures.

III. APPROACH

Our system consists of two parts. First the image is transformed into a vector of posture scores by using the posture descriptors. These posture scores are then used as features by a posture classifier, which returns the final posture label.

A. Posture Descriptor

The posture descriptor is a component that transforms the input image into a vector of features that can be used for posture recognition. Normally, the posture descriptor consists of body part locations [8] or transformed low-level features [13]. In this paper, we capture the posture descriptor at a higher level. Each posture descriptor is a measurement of how likely the input image belongs to a certain posture category. Specifically, we adopt the posture descriptor from the state-of-the-art approach in human pose estimation [11], where the poses are estimated by finding the optimal skeleton configuration with respect to the local body part detection. Similar to the structure of a Support Vector Machine (SVM), each posture descriptor returns a matching score along with the estimated body part positions. Since the body part positions are often unknown due to the occlusion, we disregard them and use only the matching score to perform posture recognition in our approach.

We now formulate the problem and give a brief introduction to the posture descriptor. For more details, please refer to [11].

Let I be an input image, and k is a posture category that follows $k \in \{1, ..., K\}$. Given the input image, each posture descriptor gives a matching score $S_k(I)$ by maximizing the energy function $Q_k(I, l, t)$ over all possible body part locations L and all types of the body parts T

$$S_k(I) = \max_{l \in L, t \in T} Q_k(I, l, t) \tag{1}$$

where l is a vector of body part locations in the discretized image space and t is a vector of type assignments over all the body parts.

Solving a general problem of (1) takes exponential time. But when Q_k are computed within a tree structure, the non-maximum suppression of the function can be computed efficiently using dynamic programming [14]. We define a tree structure following the human skeleton, where the vertices V of the tree are the body parts and the edges E are the pair-wise connections between the vertices.

We write the energy function of the tree structure as

$$Q_k(I, l, t) = \sum_{i \in V} \omega_i^{t_i} \cdot \phi(I, l_i) + \sum_{ij \in E} \omega_{ij}^{t_i t_j} \cdot \psi(l_i, l_j) + S(t) \tag{2}$$

where $\omega_i^{t_i} \cdot \phi(I, l_i)$ is a linear filter of the body parts. It gives high scores if the image at location l_i looks like the type t_i of the i^{th} body part. The second term $\omega_{ij}^{t_i t_j} \cdot \psi(l_i, l_j)$ is a quadratic spring model that makes connections between two body parts with a spatial deformation cost. $S(t)$ is the bias that models the prior of seeing a particular type as well as the prior of seeing the pair-wise type combination. The term of the bias is formulated as

$$S(t) = \sum_{i \in V} b_i^{t_i} + \sum_{ij \in E} b_{ij}^{t_i, t_j} \tag{3}$$

Note that (2) is a linear equation that is parameterized by ω and b, therefore it can be rewritten as

$$Q_k(I, l, t) = \beta_k \cdot \Phi(I, l, t) \tag{4}$$

where β is the concatenation of ω and b. Knowing β, we are able to solve $S_k(I)$ in polynomial time [11].

The parameter β can be learned from the training data within a structured-SVM framework [15]. Note that $S_k(I)$ is bounded by $Q_k(I, l, t)$ with respect to all combinations of l and t, therefore the constraint equation of the SVM can be drawn as: a) $Q_k(I, l, t)$ needs to be larger than or equal to 1 on all positive examples. b) For all negative samples, $Q_k(I, l, t)$ should be smaller than or equal to -1 with respect to all possible l and t. Under such constraint, we would like to maximize the margin between two classes, which is a typical optimization problem that can be solved by using quadratic programming (QP) [16].

B. Posture Recognition

We learn a separate posture descriptor with respect to each of the posture categories by selecting and training the descriptors on specific subsets of the training data. The posture descriptors estimate the body part locations in the image and simultaneously generate a score associated to the best approximation of the pose articulation. We note that the quality of generated part positions is extremely low due to the severe self-occlusion. Rather than using the positions, we use the corresponding scores for posture recognition. After applying the set of posture descriptors to the input image, we get a vector of scores $\{S_1(I), ..., S_K(I)\}$ from the descriptors. The score reflects the confidence of that image belonging to a certain posture category. One straight-forward way of recognizing posture from these scores would be to apply non-maximum suppression over the scores. However, the scores cannot be guaranteed to have the same scale and are, therefore, not comparable to each other. Moreover, the output of multiple descriptors may be informative of a posture, so that it makes sense to combine them.

Our solution is to treat the descriptor scores as a vector of features in a classification problem. We compute a classification result $P_k(I) = \Psi_k(S_1(I), ..., S_K(I))$, which could, in general, be a binary label or a probabilistic measure of the predicted label. For the purposes of this work, we used a standard SVM [17] with Gaussian kernel.

IV. EXPERIMENT AND RESULTS

In this section, we evaluate both the conventional approach and the proposed approach in the context of the top-view. We firstly describe the two datasets that are used for evaluation. We conducted three experiments, each of which gives answers to the one of the research questions introduced in Section I.

A. Data

1) TUM Kitchen Dataset: The first dataset that we use is the publicly available TUM Kitchen Dataset [18]. The dataset is recorded in a home-monitoring scenario where the actor performs daily activities in a kitchen. The dataset consists of 10 typical posture classes, including standing,

walking, reaching, taking objects, etc. The postures have been annotated for each of the frames. The dataset also provides the ground-truth body part locations in 3D, so that we can freely project these points to any camera view that we want.

The TUM Kitchen Dataset also contains image sequences that are captured with four cameras. However, like most benchmark datasets [19], [20], the TUM Kitchen Dataset contains only the side-view images. We therefore collect our own dataset to be able to evaluate in the top view scenarios.

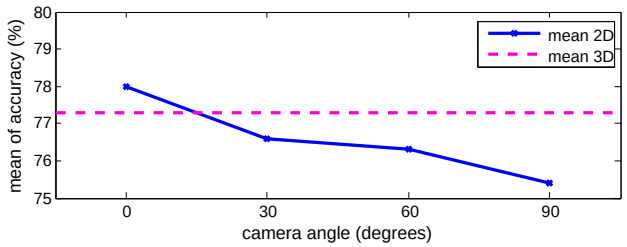
2) Our Dataset: The dataset is recorded with an omni-directional camera that is mounted on the ceiling. The persons in the frames are seen from the top-view and the body parts strongly occlude each other (see Fig. 5). To get the ground-truth body part locations, we mounted a Kinect sensor to capture the side view of the person, and we apply OpenNI skeleton tracking on the Kinect data. From the depth image, we use the skeleton tracker to generate a human skeleton that consists of 15 joint points, *i.e.* head, neck, torso, shoulders, ankles, hips, knees and feet. Since both the Kinect sensor and the omni-directional camera are calibrated within the same world coordinate system, we are able to project these joint points from the coordinate system of the Kinect sensor onto the omni-directional image plane. These projected points in 2D are manually corrected for errors, and they are used as the ground-truth body part locations for training.

The dataset contains 8 videos, and each of them has about 3000 frames. We annotated the posture labels every 10 frames (about 1 second),and the labels are as follows: standing, bending, sitting, pointing, stretching, and walking. Note that in our dataset one frame can be associated with multiple posture labels, *e.g.* a person may be standing and pointing at the same time.

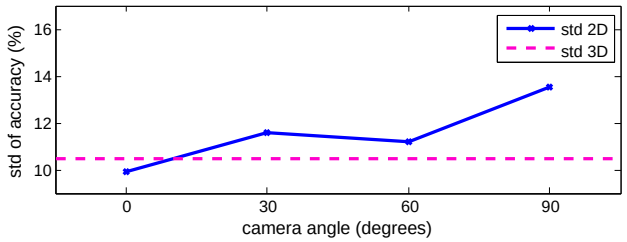
Next, we introduce the three experiments that we conducted. In the first experiment, we evaluate on the TUM Kitchen Dataset as their ground-truth pose are well annotated in 3D. In contrast, 3D poses in our dataset are less accurate as they are annotated in an automatic way using the Kinect. For the second and third experiments, we use our own dataset because the TUM Kitchen Dataset contains only persons with the side view.

B. Is 2D pose competitive with 3D pose for posture recognition?

Our first experiment is to evaluate the performance of posture recognition with respect to different camera angles. In this experiment, we use the TUM Kitchen dataset because it allows for easy comparison with the state-of-the-art 3D-based posture recognition approach, and also because the ground-truth locations in 3D are more accurate, compared with the points detected by Kinect in our own dataset. Following the work of [8], firstly we compute the geometric distance between the 3D body part locations. The geometric distances are computed within a certain temporal window, in such a way that the temporal changes of the body part



(a) mean performance



(b) standard deviation

Fig. 3. Performance of posture recognition with 3D locations and with different 2D projections based on the TUM Kitchen dataset. (a) shows the mean of the performance, and (b) shows the standard deviation. In the case of 2D, the performance drops gradually as the camera angles changes from the side-view ($0°$) to the top view ($90°$).

locations are also encoded. We apply these distance measurements as the features, and the postures are recognized by using the Random Forest [12] classifier. Furthermore, we manually define a set of mock cameras that captures different views of the humans. Following the positioning these cameras, we project the 3D body part locations onto the image plane, and then we evaluate the system in a 2D space.

We set the camera angles from $0°$ (side-view) to $90°$ (top-view) with the step-size of $30°$. For this experiment, we use the posture recognition approach as described in [8]. Fig. 3 demonstrates the classification rate of postures over different camera angles. The results show that recognizing a posture becomes more difficult with the increasing camera angles. Notably, the mean drops by over 2% when the camera shifts from the side-view to the top-view. Also, we note that the side-view (2D) outperforms the 3D, which is rather surprising as projecting from 3D to 2D results in data loss. We infer that the data loss here contains mostly the noise in 3D. After projection, the 2D points in the side-view still hold the most discriminative information which can facilitate posture recognition. It is analogical to applying noise reduction using Principle Component Analysis (PCA), which reduces the dimension of the data from 3D to 2D.

This experiment shows that top-view is a more difficult task compared with the side-view. Again, the approaches are evaluated based on the ground-truth locations. In practice, however, getting the correct body part locations is already a very challenging task by itself. Next, we evaluate the state-of-the-art pose estimation approach on our top-view data to see how well the 2D pose can be estimated from the top-view.

C. How accurately can we obtain 2D pose from top-view images?

In this experiment, we evaluate on our own dataset to see how well the state-of-the-art approach can estimate body part locations from the top-view images. We randomly select 10% samples per posture category as the test set, and the rest are kept as the data for training. The positive training examples are the top-view images together with the associated body part locations. The negative training examples are taken from the INTRA dataset [21], which contains random background images with no person. To generate more positive training images, we mirrored and added slight rotation to the training examples.

We adopt the state-of-the-art approach [11] for estimating body articulations. We use the Histogram of Oriented Gradient (HOG) as our image features. We evaluate the system with the standard evaluation criteria of pose estimation, *i.e.* the probability of a correct pose (PCP). The PCP computes the percentage of correctly localized body parts. The results on the test set are shown in Fig. 4. The performance in general is rather bad, which is mainly caused by the self-occlusion. The single descriptor in the graph refers to the posture descriptor that is trained on all the data instead of a specific posture category. We compare the results of the posture descriptors with the single descriptor. We show that the posture descriptor always outperforms the single descriptor when evaluated on its specific posture class. This is because the single descriptor tries to model all the data which bare large variation over different posture classes. Note that the posture descriptor performs much better on its own posture category than on the others. It exhibits notable potential of distinguishing among posture categories using the posture descriptors, which can be very helpful for posture recognition.

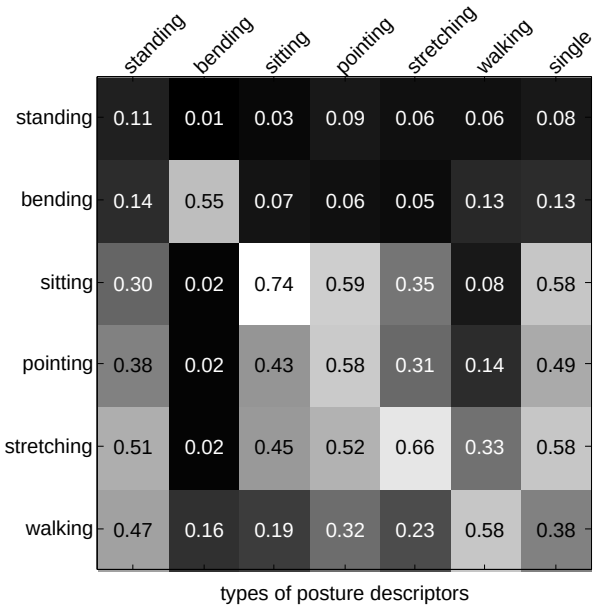


Fig. 4. The PCP performance of body part estimation over posture descriptors (columns) and posture classes of the test data (rows) on the top-view dataset. The first six posture descriptors are trained with the specific posture data. In contrast, the last "single" posture descriptor is trained on the mixing of all the training data, and therefore it is a general model that learned from all the data regardless of the posture categories. We show that when the posture descriptors are evaluated on its own posture category, the results (diagonal) always outperform the "single" model (last column).

D. How accurately can we recognize posture from imperfect 2D pose, and how does this performance compare to our proposed model?

This section compares the performance of posture recognition between our proposed system and the baseline approach on our top-view dataset.

In our approach, we adopt the method from [11] as our posture descriptor. The posture descriptor is learned from each of the posture categories. For classification, we train a Support Vector Machine (SVM) [17] with the RBF kernel per posture class. Using the matching scores from all posture descriptors, the SVM gives a binary decision on the posture label.

To compare with our proposed system, we form the baseline approach by combining two state-of-the-art approaches in pose estimation and posture recognition. Specifically, we follow the approach of [11] for pose estimation. We learn a single descriptor over all the data. Then we use the single descriptor to estimate body part locations. After that, we follow [8] to extract the geometric features from the 2D body part locations, and we infer the posture labels using random forest.

Note that the geometric features in [8] are extracted within a short sequence of frames, therefore the temporal information are encoded in the baseline approach. In contrast, our proposed system is evaluated on still images, and we believe the performance can be further improved by adding temporal filtering to our current system. This is left as future work.

The performance of posture recognition is shown in Table I. The results show that our approach outperforms the conventional approach on all posture categories, and the average performance is better than the conventional approach by over 23%. In particular, the performance is improved by 69% on the bending data. This is because when people are bending, occlusion is more severe compared with the other postures, *e.g.* the limbs are most likely to be fully occluded by the torso when bending. Estimating the body part locations from these missing limbs becomes an extremely difficult task. Benefiting from the posture descriptors, our approach does not require the body part locations to be correctly localized and therefore our system still shows very high performance on the bending data. From our results, we believe our system is more robust to the self-occlusion as we do not rely on the body part locations which are rather unstable when estimated under the top-view. Moreover, we believe our system can be further improved after adding the temporal information. Finally, we show some sample postures recognized by our system in Fig. 5 which gives an illustration of our results.

TABLE I

RESULTS OF POSTURE RECOGNITION: F-SCORE

	standing	bending	sitting	pointing	stretching	walking	avg.
Baseline [8]+[11]	93.65	20.69	93.51	43.87	25.53	60.43	56.28
Our approach	**95.53**	**89.00**	**96.83**	**62.69**	**58.90**	**75.53**	**79.75**

Fig. 5. Results of the posture recognition based on our top-view dataset. The example images are randomly sampled from the testing results. The text on the left indicates the ground-truth posture label of the images in the row. Postures that are correctly recognized are in green rectangles, and postures are in red rectangles if wrong labels are predicted.

V. CONCLUSION

In this paper, we proposed a novel method to classify human postures from the top-view cameras. Using the posture descriptors, we get a vector of matching scores, and we use the scores for posture recognition instead of the conventional way which use the body part locations. The results show that leveraging the posture descriptors provides superior classification results in images with self-occlusion. We believe the posture descriptors can be further leveraged by enabling temporal filtering for activity recognition.

REFERENCES

[1] D. F. Fouhey, V. Delaitre, A. Gupta, A. A. Efros, I. Laptev, and J. Sivic, "People watching: Human actions as a cue for single-view geometry," in *ECCV*, 2012.

[2] B. Yao and L. Fei-Fei, "Modeling mutual context of object and human pose in human-object interaction activities," in *CVPR*, 2010, pp. 17–24.

[3] A. Yao, J. Gall, and L. Van Gool, "A hough transform-based voting framework for action recognition," in *CVPR*, 2010, pp. 2061–2068.

[4] J. P. Wachs, D. Goshorn, and M. Kölsch, "Recognizing human postures and poses in monocular still images," in *IPCV*, 2009.

[5] E. Weng and L. Fu, "On-line human action recognition by combining joint tracking and key pose recognition," in *IROS*, 2012, pp. 4112–4117.

[6] N. Zouba, B. Boulay, F. Bremond, and M. Thonnat, "Monitoring activities of daily living of elderly based on 3d key human postures," *Cognitive Vision*, pp. 37–50, 2008.

[7] L. Sigal, "Human pose estimation," *Encyclopedia of Computer Vision*, 2011.

[8] A. Yao, J. Gall, G. Fanelli, and L. Van Gool, "Does human action recognition benefit from pose estimation?" in *BMVC*, 2011.

[9] Q. Lin, C. Zhou, S. Wang, and X. Xu, "Human behavior understanding via top-view vision," *International Conference on Control Automation and Systems*, vol. 3, pp. 184–190, 2012.

[10] S. Weerachai and M. Mizukawa, "Human behavior recognition via top-view vision for intelligent space," in *International Conference on Control Automation and Systems*, 2010, pp. 1687–1690.

[11] Y. Yang and D. Ramanan, "Articulated pose estimation with flexible mixtures-of-parts," in *CVPR*, 2011, pp. 1385–1392.

[12] J. Gall and V. Lempitsky, "Class-specific hough forests for object detection," in *CVPR*, 2009, pp. 1022–1029.

[13] N. Ikizler and P. Duygulu, "Histogram of oriented rectangles: A new pose descriptor for human action recognition," *Image and Vision Computing*, vol. 27, no. 10, pp. 1515–1526, 2009.

[14] P. F. Felzenszwalb and D. P. Huttenlocher, "Pictorial structures for object recognition," *IJCV*, vol. 61, no. 1, pp. 55–79, 2005.

[15] T. Finley and T. Joachims, "Training structural svms when exact inference is intractable," in *Proc. of the 25th International Conference on Machine learning*. ACM, 2008, pp. 304–311.

[16] C. Hsu, C. Chang, C. Lin, *et al.*, "A practical guide to support vector classification," 2003.

[17] C.-C. Chang and C.-J. Lin, "LIBSVM: A library for support vector machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, pp. 27:1–27:27, 2011, software available at http://www.csie.ntu.edu.tw/ cjlin/libsvm.

[18] M. Tenorth, J. Bandouch, and M. Beetz, "The tum kitchen data set of everyday manipulation activities for motion tracking and action recognition," in *ICCV workshop on Computer Vision*, 2009, pp. 1089–1096.

[19] C. Schuldt, I. Laptev, and B. Caputo, "Recognizing human actions: A local svm approach," in *ICPR*, vol. 3, 2004, pp. 32–36.

[20] M. Blank, L. Gorelick, E. Shechtman, M. Irani, and R. Basri, "Actions as space-time shapes," in *ICCV*, vol. 2, 2005, pp. 1395–1402.

[21] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *CVPR*, vol. 1, 2005, pp. 886–893.