Human Pose as Context for Object Detection

Abhilash Srikantha[1,2]
abhilash.srikantha@tue.mpg.de
Juergen Gall[1]
gall@iai.uni-bonn.de

[1] University of Bonn, Germany
[2] MPI for Intelligent Systems, Tuebingen, Germany

Abstract

Detecting small objects in images is a challenging problem particularly when they are often occluded by hands or other body parts. Recently, joint modelling of human pose and objects has been proposed to improve both pose estimation as well as object detection. These approaches, however, focus on explicit interaction with an object and lack the flexibility to combine both modalities when interaction is not obvious. We therefore propose to use human pose as an additional context information for object detection. To this end, we represent an object category by a binary star model and train regression forests that localize parts of an object for each modality separately. Predictions of the two modalities are then combined to detect the bounding box of the object. We evaluate our approach on three challenging datasets which vary in the amount of object interactions and the quality of automatically extracted human poses.

1 Introduction

Object detection has seen considerable success, but the case of medium and small sized everyday objects still remains an open problem [16]. Although such objects appear at low image resolutions, they often occur in the context of human interactions. However, this introduces new challenges as objects are heavily occluded and undergo large pose and appearance variations during the process. Nevertheless, the context of human-object interactions can be incorporated as has been proposed by recent methods [2, 10, 11, 12, 17, 19]. For instance, [2] extends a deformable part model (DPM) [5] to model spatial relations between body parts and parts of objects. This approach, however, only works well for images showing the instant of human-object interaction, *i.e.*, when a human is closely in contact with an object. For images without an interaction, pose and objects are independently modelled, *e.g.*, by having several models including either object or pose, or both together. In such cases, the additional information from human context is therefore not exploited.

In this work, we propose an approach that includes human pose as an additional context for object detection. Our approach is not limited to images showing explicit human-object interactions, but also works for general images where pose can be inferred. For instance, a pose related to emptying a tin indicates that a tin opener might be close although the person does not use the tin opener at this moment. To this end, we model objects by a part based model and predict locations of parts from both image and pose data using regression forests.

In our experiments, we show that jointly modelling human and object as in [2] leads to suboptimal performance for object detection. On the other hand, our approach has the

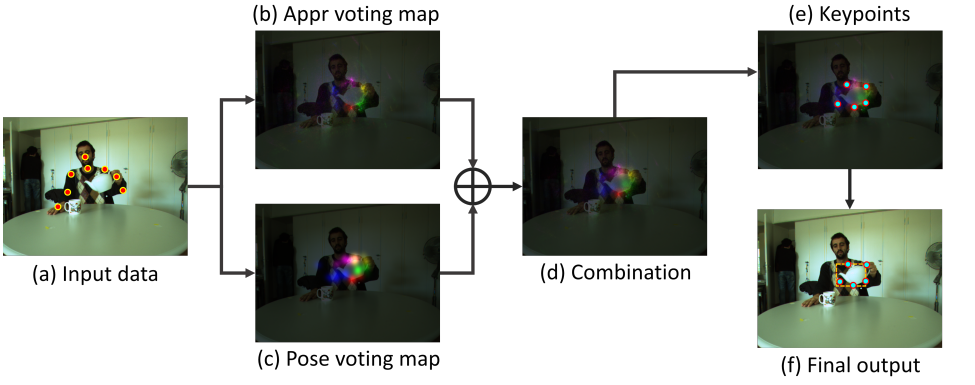

Figure 1: Detecting teapots: (a) Input is an image and automatically extracted human pose. (b) Object keypoint unaries based on appearance features and (c) Keypoint unaries based on human pose features. Note the reduced keypoint localization capability. (d) Linear combination of unaries. (e) Inferring keypoints using the pictorial structures model. (f) Regressing object bounding box using the inferred keypoints.

flexibility to incorporate potential gains from either modality. We evaluate our approach on three datasets [2, 12, 14] that have varying quality of automatically extracted 2d or 3d human pose. Overall, we show that human pose can be successfully used to improve object detection performance. Further, we investigate the effect of various human pose estimation techniques on object detection accuracy. An outline of the approach is presented in Figure 1.

2 Related Work

Combining humans and objects together to address various problems in computer vision has received considerable attention in the recent past. [3] builds a discriminative model for action classification by reasoning about spatial co-occurrences of body parts and objects. In [13] a weakly supervised approach is proposed for action classification that does not require annotations of objects and humans in training images. Human context has also been used to deduce object functionality either by inferred [20] or by hypothesised human pose [11].

As for methods relating to object detection, [10] proposes a generative model that combines body part trajectories and object appearance. However, it uses strictly handcrafted metrics to tap human motion information which can be difficult to adapt to realistic actions. [19] learns a discriminative random field model by representing body part location priors as nodes and spatial relations between body parts and objects as edges. However, mixture models are treated independently resulting in poor performance for complex data. In this regard, [17] introduces a coarse-to-fine hierarchical grammar for a more concise representation of mixture models. Introducing phraselets, [2] extends a DPM [6] to improve the quality of mixtures by clustering training examples based on their relative locations. The method reports state-of-the-art results for jointly reasoning about pose estimation, action classification and object detection.

3 Object Detection

As illustrated in Figure 1(f), we represent an object by a set of descriptive keypoints $\mathcal{K} = \{\mathbf{k}_i\}$ where $\mathbf{k}_i$ encodes the image location of the i^{th} keypoint. As in the pictorial structures model [6], the spatial relations between them are defined by a directed graph E and the prior on any keypoint configuration is given by

$$p(\mathcal{K}) = \prod_{i,j \in E} \psi_{ij}(\mathbf{k}_i, \mathbf{k}_j), \tag{1}$$

where the binary potentials $\psi_{ij}(\mathbf{k}_i, \mathbf{k}_j)$ model spatial relations between two keypoints $\mathbf{k}_i$ and $\mathbf{k}_j$. Given an observation $\mathcal{D}$, an optimal configuration is estimated by the maximum of the posterior distribution

$$\begin{aligned} p(\mathcal{K}|\mathcal{D}) &\propto p(\mathcal{D}|\mathcal{K}) \cdot p(\mathcal{K}) \\ &\propto \prod_i \phi_i(\mathbf{k}_i) \cdot \prod_{i,j \in E} \psi_{ij}(\mathbf{k}_i, \mathbf{k}_j) \end{aligned} \tag{2}$$

While we use 3-mixture Gaussians as binary potentials to model relative keypoint offsets in the star structured graph E for efficient inference as in [6], our work focuses on extracting more discriminative unary potentials $\phi_i(\mathbf{k}_i)$ derived from appearance and human pose features. The unary potentials will be discussed in Section 4.

Since we adhere to the PASCAL-VOC protocol for evaluation, we have to predict a bounding box (x_1, y_1, x_2, y_2) from the inferred keypoint configuration $\mathcal{K}$. To this end, we use a mixture of linear least squares regressors and predict each parameter of the bounding box independently. For the regression, we normalize keypoint locations such that the mean becomes zero and variance one. We use a mixture of 3 regressors, each of which is trained on a cluster of training data. As for the feature vector for clustering, we add the aspect ratio to the normalized keypoints resulting in a $(2|\mathcal{K}|+1)$ dimensional vector. The aspect ratio is calculated using the smallest rectangle enclosing all keypoints.

The inference procedure results in multiple overlapping detections for each object instance. We therefore use a greedy approach to eliminate redundant detections. Given an image, we get a set of detected bounding boxes and their respective scores $p(\mathcal{K}|\mathcal{D})$. The set is sorted according to the score and all bounding boxes that have an intersection-over-union (IoU) ratio over 0.5 with a higher scoring bounding box are discarded.

4 Keypoint Regressors

The unary potentials $\phi_i(\mathbf{k}_i)$ in Eqn (2) are modelled by probabilities over keypoint location $\mathbf{k}_i$. The probabilities are estimated from two modalities, namely the object appearance $\mathcal{D}_A$ and the human pose $\mathcal{D}_P$, *i.e.*

$$\phi_i(\mathbf{k}_i) = p(\mathbf{k}_i|\mathcal{D}_A, \mathcal{D}_P). \tag{3}$$

As random forests are used as regressors, we introduce them briefly in Section 4.1. Sections 4.2, 4.3 and 4.4 then present unary potentials based on individual features and their combination.

4.1 Random Forests

As in [8], we use random forests for object detection. However, instead of voting for the center of the bounding box we use random forests to predict keypoints of an object and infer the object bounding box as described in Section 3. A tree T in a forest $\mathcal{T}$ is built from a random subset $\mathbb{D}$ of the training data. For each training image, features $F_{\mathcal{D}}$ are extracted. For training a tree, the set $\mathbb{D}$ is recursively divided into two subsets $\mathbb{D}_0$ and $\mathbb{D}_1$ using a binary split function $\zeta^*(F_{\mathcal{D}}) \rightarrow \{0,1\}$. The split function, which maximizes the information gain $g(\zeta)$, is chosen from a pool of randomly generated split functions:

$$\zeta^* = \underset{\zeta}{\operatorname{argmax}}\, g(\zeta) \tag{4}$$

$$g(\zeta) = H(\mathbb{D}) - \sum_{s\in\{0,1\}} \frac{|\mathbb{D}_s(\zeta)|}{|\mathbb{D}|} H(\mathbb{D}_s(\zeta)), \tag{5}$$

where H is randomly chosen to be the class entropy or the squared error of the predicted mean [8]. The best split function is stored at the node and the training continues recursively until the maximum depth of the tree is reached or the number of samples in a node falls below a threshold. Incoming training data $\mathbb{D}$ is stored at the leaves.

4.2 Appearance Features

We first consider the case when keypoints are predicted from image data. In this case, an observation $\mathcal{D}$ consists of a set of image patches. The image features $F_{\mathcal{D}}$ are similar to [8], *i.e.*, they consist of 15 feature channels: 6 color channels obtained by the Lab color space processed by a 5×5 min- and max- filter and 9 gradient features obtained by 9 HOG bins using a 5×5 cell and soft binning.

To train a forest for each keypoint, patches are sampled from training images where patches within a radius of 100 pixels are considered as positive examples and as negative examples otherwise. Each patch is further augmented with a binary class label c and in case of a positive patch the scale s of the object and the offset $\mathbf{d}$ to the keypoint is also stored. The splitting functions used are pixel comparisons as in [8]:

$$\zeta_\gamma(F_{\mathcal{D}}) = \begin{cases} 0 & \text{if} \quad F^l_{\mathcal{D}}(\mathbf{p}) - F^l_{\mathcal{D}}(\mathbf{q}) < \tau\,, \\ 1 & \text{otherwise} \end{cases} \tag{6}$$

where parameters $\gamma = (\mathbf{p}, \mathbf{q}, l, \tau)$ are described by coordinates $\mathbf{p}$ and $\mathbf{q}$ within the patch, the selected feature $l \in \{1, 2, \cdots, 15\}$ and a threshold τ. For selecting the splitting functions, we use either the entropy for classification or the squared error of the mean predictor for regression in Eqn (5):

$$\begin{aligned} H_{clas}(\mathbb{D}) &= -\sum_c p(c|\mathbb{D})\ \log(p(c|\mathbb{D})) \\ H_{regr}(\mathbb{D}) &= \frac{1}{|\mathbb{D}^+|} \sum_{\mathcal{D}\in\mathbb{D}^+} \left\| \mathbf{d}_{\mathcal{D}} - \frac{1}{|\mathbb{D}^+|} \sum_{\mathcal{D}\in\mathbb{D}^+} \mathbf{d}_{\mathcal{D}} \right\|^2 \end{aligned} \tag{7}$$

where $\mathbb{D}^+$ is the set of positive patches. At the leaves, class probabilities $p(c|L)$, distributions of the offset vectors with respect to a quantized scale $\hat{s}$ and keypoint class c, *i.e.* $p(\mathbf{d}|c, \hat{s}, L)$,

are stored. The unary potential based on appearance for a given scale $\hat{s}$ is then defined by

$$\phi_i^A(\mathbf{k}_i,\hat{s}) = \sum_{\mathbf{y}\in\Omega} \frac{1}{|\mathcal{T}_i|} \sum_{T\in\mathcal{T}_i} p(\mathbf{k}_i-\mathbf{y}|c,\hat{s},L_T)\cdot p(c|L_T), \tag{8}$$

where $\mathcal{T}_i$ is the forest trained for the i^{th} keypoint and Ω is a set of locations in the image.

In contrast to [8], we do not scale training examples to a fixed object size since this requires performing object detection over several scaled versions of the test image. Instead, we store the scale of the objects in training images in the leaves and process a test image at the resolution as is. The unaries $\phi_i^A(\mathbf{k}_i,\hat{s})$ are therefore modelled for pixel location $\mathbf{k}_i$ and scale $\hat{s}$. The keypoint configuration $\mathcal{K}$ is then inferred as per Eqn (2) for each scale independently.

4.3 Human Pose Features

When the keypoints are predicted from automatically extracted 2d or 3d human pose, the pose features $F_\mathcal{P}$ are based on joint locations $\mathbf{j}_m$ as in [18], *i.e.*, for all joint combinations the Euclidean distance between two joints is computed and for all quadruples of joints the normal plane feature and the velocity feature are used.

To train a forest for each keypoint, training images with the object of interest are considered as positive examples and as negative examples otherwise. For each image, pose is augmented with a binary class label c. The positive examples are further augmented with scale s of the object and offsets $\mathbf{d}_m$ from all joints to the keypoint. The splitting functions are defined by

$$\zeta_\gamma(F_\mathcal{P}) = \begin{cases} 0 & \text{if} \quad f_\mathcal{P} < \tau\,, \\ 1 & \text{otherwise} \end{cases} \tag{9}$$

where $f_\mathcal{P}$ is a randomly chosen pose feature. The splitting functions are selected as in Eqn (7).

Besides class probabilities $p(c|L)$, the distributions of offset vectors with respect to a quantized scale $\hat{s}$ and keypoint class c for each joint m, *i.e.*, $p_m(\mathbf{d}_m|c,\hat{s},L)$, are stored at the leaves. The unary potential based on pose for a given scale $\hat{s}$ is then defined by

$$\phi_i^P(\mathbf{k}_i,\hat{s}) = \sum_{m=1}^{M} \frac{1}{|\mathcal{T}_i|} \sum_{T\in\mathcal{T}_i} p_m(\mathbf{k}_i-\mathbf{j}_m|c,\hat{s},L_T)\cdot p(c|L_T), \tag{10}$$

where $\mathcal{T}_i$ is the forests trained for the i^{th} keypoint and M is the number of joints.

4.4 Combining Appearance and Pose

The unary potential in Eqn (2) is a linear combination of the filtered unaries discussed in Sections 4.2 and 4.3:

$$\phi_i(\mathbf{k}_i,\hat{s}) = \left(K(\sigma_A) * \phi_i^A(\mathbf{k}_i,\hat{s})\right) + \alpha\left(K(\sigma_P) * \phi_i^P(\mathbf{k}_i,\hat{s})\right) \tag{11}$$

where $*$ represents the convolution operation and σ is the standard deviation for the Gaussian blur kernel K. Since the human pose can only provide a rough prior for the location of an object class but is insufficient for accurate object localization, $\sigma_P > \sigma_A$. The parameters α, σ_A and σ_P are estimated by cross-validation.

5 Experiments

We evaluate the proposed approach on three datasets: ETHZ-Activity [7], CAD-120 [12] and MPII-Cooking [14]. Human pose is inferred in all three datasets using different methods. ETHZ-Activity uses a model based method to extract 3d joint locations of the upper body, CAD-120 uses the OpenNI SDK to extract 3d full body joint locations and MPII-Cooking uses a pictorial structure model to extract the 2d joint locations for the arms. The datasets collectively represent a rich variety of human-object interactions, *e.g.* elementary interactions are captured in ETHZ-Activity, multi-object interactions in MPII-Cooking and CAD-120 also captures varying viewpoints. There is also a diversity in objects ranging from large to small and from rigid to deformable. The amount of occlusion also varies and the objects are sometimes barely visible. Figure 2 shows some cropped representative images. We manually labelled[1] 5 keypoints for each object in the three datasets for every 10^{th} frame of the training data.

For evaluation, we use the PASCAL-VOC measure [4] that considers a detected bounding box as true positive when the IoU ratio with the groundtruth bounding box exceeds 0.5. Multiple detections overlapping with a true positive are counted as false positives. We report area under the precision-recall curve (AUC) where the precision at any recall level r is replaced by the maximum precision measured at recall levels exceeding r as in [4].

We present implementation details in Section 5.1 followed by the evaluation on the three datasets in Sections 5.2– 5.4.

5.1 Implementation Details

Random Forests: A forest consists of 4 trees with a maximum depth of 25. A tree based on appearance features is trained with 100,000 positive and negative 16×16 sized image patches each and contains at least 20 samples in a leaf. At each node, a pool for splitting functions is generated by randomly choosing 10 thresholds τ and 100 combinations for other parameters in γ. A tree based on human pose features is trained with all positive and negative examples and contains at least 10 samples in a leaf. The pool of splitting functions is generated by randomly choosing 80 parameters and 8 thresholds. The binary potentials in Eqn (2) are modelled by a mixture of 3 Gaussians.

Setting parameters: The proposed method has three parameters as per Eqn (11). We set these parameters by grid search on the validation dataset which was obtained by splitting the training data in half. Generally, we found that the parameters are stable across several splits of the same dataset. In case of several splits, we therefore estimate the parameters on the first split and use the same for the rest.

5.2 MPII Cooking Dataset

The dataset contains two cooking activities performed by 12 actors. Since the dataset does not provide bounding box annotations for objects, we use the object classes that have been used in [15] for object discovery. We take all frames where the objects are annotated. This gives a subset of the dataset [14]. On this dataset, we perform a 7 fold cross validation as

[1] Annotations can be found at http://ps.is.tue.mpg.de/person/srikantha

Table 1: AUC measures for the MPII dataset.

class	Appr.	Pose	Gall [8]	Desai [4]	Concat.	PoseObject	Comb.
bowl	0.25	0.15	0.17	0.07	0.02	0.15	**0.27**
bread	0.50	0.45	0.30	0.20	0.13	0.29	**0.60**
pan	0.20	0.20	**0.34**	0.22	0.14	0.21	0.23
plate	0.51	0.48	**0.54**	0.22	0.49	0.42	0.51
grater	0.13	0.02	**0.15**	0.03	0.03	0.13	0.14
squeezer	0.33	0.22	**0.35**	0.07	0.21	0.33	**0.35**
tin	**0.16**	0.07	0.11	0.14	0.05	0.03	**0.16**
spiceholder	**1.00**	0.15	**1.00**	0.60	0.92	0.15	**1.00**
average	0.38	0.22	0.37	0.19	0.25	0.21	**0.41**

in [14]. While training the proposed method on one split took 40hrs on a 6-core 3.2GHz machine, running [2] took 72hrs in the same setup.

The AUC measure for each object class averaged over all 7 splits are given in Table 1. We first compare our approach based on appearance and pose features (*Comb*), which is described in Section 4.4, to only one of the two modalities, namely appearance (*Appr*) and pose (*Pose*), which are described in Section 4.2 and Section 4.3, respectively. Although the pose features perform worse than the appearance features, the combination improves the accuracy of the appearance features. While we train a forest separately for each modality, we also compare to an approach where a single forest is trained on a concatenation of appearance and pose features (*Concat*). In this case both splitting functions Eqn (6) and Eqn (9) are used in a single tree. The accuracy of this approach, however, drops sharply in contrast to the appearance features.

We also compare our method to the two most related approaches. In [8], Hough forests are used for object detection. While our approach uses a star model for keypoints for the objects as described in Section 3, [8] uses a star model for a single keypoint. When comparing it with our approach using only appearance features, we observe that the multi-keypoint is only slightly better than the single-keypoint setup. The method [2] combines human pose estimation and object detection. We train the method on the training data with estimated human pose and annotated keypoints for the objects. The approach actually performs worse than the pose features. We therefore also implemented the approach using random forests (*PoseObject*) by using appearance based features and using the joints of the human pose as additional keypoints. The results are also worse than the pose features. In order to analyse if the reduced accuracy stems from the additional pose estimation, which is not performed by our approach since we use the estimated human poses provided by the dataset [14], we evaluated the impact of the chosen pose estimation method for our approach. We therefore trained a pose estimator [1] on the separate training set for pose estimation [14] and estimated the poses on both our training and test data. Using the poses estimated by the approach [1] did not change the object detection accuracy, which remained 0.41. This indicates that it is not the pose estimation that results in a poor performance, but the combination of objects and pose as proposed in [2] is not flexible enough to model object-pose relations that are not limited to the moment of an interaction.

We additionally investigated if it is important that the poses for training and testing are estimated by the same method for human pose estimation. We therefore used our approach originally trained on the poses provided by [14] and only replace the poses for the test data. When using [1] for estimating the human pose on the test data, the accuracy slightly drops from 0.41 to 0.40, showing that the approach can be trained and tested with different methods for human pose estimation. We also used the human poses obtained by [2] on the test data.

Table 2: AUC measures for the ETHZ Activity dataset.

class	Appr.	Pose	Gall [8]	Desai [7]	Concat.	Comb.
brush	0.37	0.10	0.24	**0.51**	0.20	0.46
calculator	0.98	0.70	**1.00**	0.84	0.32	0.98
camera	0.77	0.80	0.74	0.79	0.72	**0.93**
headphone	0.42	0.43	0.25	**0.64**	0.13	0.47
marker	**0.09**	0.02	0.02	0.08	0.06	**0.09**
mug	0.25	0.13	0.30	**0.54**	0.05	0.30
phone	**0.33**	0.02	0.05	0.07	0.01	**0.33**
puncher	0.74	0.08	**0.78**	0.64	0.30	0.76
remote	0.24	0.05	**0.33**	0.10	0.15	0.29
roller	0.45	0.08	0.48	**0.68**	0.14	0.51
teapot	0.42	0.36	**0.51**	0.46	0.36	0.42
videogame	0.48	0.12	0.40	**0.63**	0.42	0.52
average	0.46	0.24	0.42	0.50	0.23	**0.51**

Even in this case, the accuracy of 0.39 is still better than using the appearance features only.

5.3 ETHZ Activity Dataset

The ETHZ-activity dataset contains 143 sequences where 6 subjects interact with 12 different objects. For evaluation, we perform 6 fold cross validation for each of the 12 objects. As a preprocessing stage, we normalize all images for lighting conditions using [9].

The results are reported in Table 2. They are similar to the MPII cooking dataset. The appearance features outperform the pose features, but the pose features perform better for the classes *camera* and *headphone*. Our proposed combination outperforms each of the modalities and the concatenation of the two features. The method [8] performs in average worse than the PS model with appearance features. The approach [7] performs better for this dataset and achieves a higher accuracy than the pose or appearance features, but our combination still performs better on average.

5.4 CAD 120 Dataset

The CAD-120 dataset contains 120 sequences of 10 different high level activities performed by 4 subjects. For evaluation, we perform a 4 fold cross validation for each of the 10 objects. It must be noted that while most classes have a sufficient amount of training data, it is not the case with classes *book* and *remote* resulting in all object detectors to fail. Also, the human pose extracted from OpenNI SDK not only has noisy joint locations specially for hands and legs, but also consists of missing joints due to low detection confidence or frequent occlusion. Missing joint locations are handled by assigning them to a default value of zero.

The results are reported in Table 3. The pose features perform poorly on the dataset due to low quality of estimated human poses. In particular, arms are often wrongly estimated as shown in Figure 2. Nevertheless, using pose features in addition to the appearance features improves accuracy. As on other datasets, the method [8] performs worse than the keypoint approach with appearance features on average. The accuracy of the approach [7] is similar to the accuracy of the concatenated features, which is lower than our approach with appearance features.

Table 3: AUC measures for the CAD-120 dataset.

class	Appr.	Pose	Gall [5]	Desai [4]	Concat.	Comb.
book	0.00	0.00	0.00	**0.03**	0.00	0.00
bowl	**0.69**	0.17	0.68	0.17	0.48	**0.69**
box	**0.60**	0.10	0.55	0.03	0.27	**0.60**
cloth	0.03	0.00	0.02	**0.12**	0.00	0.03
cup	0.24	0.02	**0.26**	0.12	0.03	0.24
medicinebox	0.35	0.17	0.32	0.69	0.39	**0.40**
microwave	0.15	0.15	0.13	**0.30**	0.10	0.20
milk	0.75	0.30	0.71	0.61	0.69	0.75
plate	**0.25**	0.02	0.26	0.03	0.03	**0.25**
remote	0.00	0.00	0.00	0.00	0.00	0.00
average	0.31	0.09	0.29	0.21	0.20	**0.32**

6 Conclusion

In this work we have presented an approach that combines two modalities, namely image appearance and human pose, for object detection. We have evaluated the approach on three challenging datasets that contain small objects that are often occluded during human-object interaction. Our experiments not only showed that human pose improves an appearance based object detector irrespective of the underlying pose estimation technique, but also that the proposed combination of a separate forest for each modality outperforms the concatenation of features or a joint model for human pose estimation and object detection.

Acknowledgements: We thank Umar Iqbal for providing human poses on the MPII-Cooking dataset and DFG Emmy Noether program (GA 1927/1-1) for providing financial support.

References

[1] Matthias Dantone, Juergen Gall, Christian Leistner, and Luc Van Gool. Human pose estimation using body parts dependent joint regressors. In *CVPR*, pages 3041–3048. IEEE, 2013.

[2] Chaitanya Desai and Deva Ramanan. Detecting actions, poses, and objects with relational phraselets. In *ECCV*, pages 158–172. Springer, 2012.

[3] Chaitanya Desai, Deva Ramanan, and Charless Fowlkes. Discriminative models for static human-object interactions. In *CVPRW*, pages 9–16. IEEE, 2010.

[4] Mark Everingham, Luc Van Gool, Christopher K.I. Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, pages 303–338, 2010.

[5] Pedro Felzenszwalb, David McAllester, and Deva Ramanan. A discriminatively trained, multiscale, deformable part model. In *CVPR*, pages 1–8. IEEE, 2008.

[6] Pedro F Felzenszwalb and Daniel P Huttenlocher. Pictorial structures for object recognition. *IJCV*, pages 55–79, 2005.

[7] Juergen Gall, Andrea Fossati, and Luc Van Gool. Functional categorization of objects using real-time markerless motion capture. In *CVPR*, pages 1969–1976. IEEE, 2011.

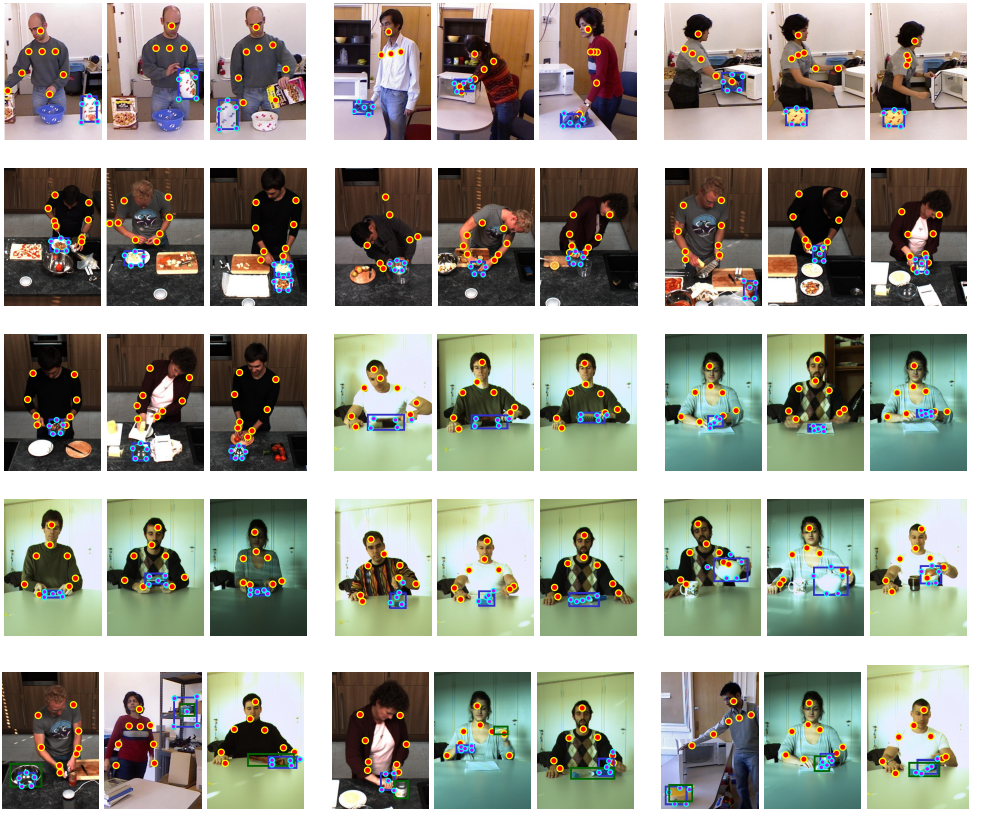

Figure 2: Qualitative results showing input human pose and most confident inferred bounding boxes as per Eqn (11). **Top four rows:** Successful detections are shown for classes *Milkbox, Cloth, Bowl* from CAD-120; *Plate, Squeezer, Tin, Bowl* from MPII-Cooking and *Brush, Marker, Videogame, Roller, Teapot* from ETHZ-Activity. **Last row:** Failed detections due to scale problems, occlusions and faulty bounding box regression. Groundtruth bounding boxes are shown in green.

[8] Juergen Gall, Angela Yao, Nima Razavi, Luc Van Gool, and Victor Lempitsky. Hough forests for object detection, tracking, and action recognition. *PAMI*, pages 2188–2202, 2011.

[9] Pascal Getreuer. Automatic Color Enhancement (ACE) and its fast implementation. *Image Processing On Line*, pages 266–277, 2012.

[10] Abhinav Gupta and Larry S Davis. Objects in action: An approach for combining action understanding and object perception. In *CVPR*, pages 1–8. IEEE, 2007.

[11] Yun Jiang and Ashutosh Saxena. Hallucinating humans for learning robotic placement of objects. In *Experimental Robotics*, pages 921–937. Springer, 2013.

[12] Hema Swetha Koppula, Rudhir Gupta, and Ashutosh Saxena. Learning human activities and object affordances from rgb-d videos. *IJRR*, pages 951–970, 2013.

[13] Alessandro Prest, Cordelia Schmid, and Vittorio Ferrari. Weakly supervised learning of interactions between humans and objects. *PAMI*, pages 601–614, 2012.

[14] Marcus Rohrbach, Sikandar Amin, Mykhaylo Andriluka, and Bernt Schiele. A database for fine grained activity detection of cooking activities. In *CVPR*, pages 1194–1201. IEEE, 2012.

[15] Abhilash Srikantha and Juergen Gall. Discovering object classes from activities. In *ECCV*, pages 415–430. Springer, 2014.

[16] Baochen Sun and Kate Saenko. From virtual to reality: Fast adaptation of virtual object detectors to real domains. In *BMVC*, 2014.

[17] Min Sun and Silvio Savarese. Articulated part-based model for joint object detection and pose estimation. In *ICCV*, pages 723–730. IEEE, 2011.

[18] Angela Yao, Juergen Gall, and Luc Van Gool. Coupled action recognition and pose estimation from multiple views. *IJCV*, pages 16–37, 2012.

[19] Bangpeng Yao and Li Fei-Fei. Modeling mutual context of object and human pose in human-object interaction activities. In *CVPR*, pages 17–24. IEEE, 2010.

[20] Bangpeng Yao, Jiayuan Ma, and Li Fei-Fei. Discovering object functionality. In *ICCV*, pages 2512–2519. IEEE, 2013.