Viewpoints and Keypoints

Shubham Tulsiani and Jitendra Malik
University of California, Berkeley - Berkeley, CA 94720
{shubhtuls,malik}@eecs.berkeley.edu

Abstract

We characterize the problem of pose estimation for rigid objects in terms of determining viewpoint to explain coarse pose and keypoint prediction to capture the finer details. We address both these tasks in two different settings - the constrained setting with known bounding boxes and the more challenging detection setting where the aim is to simultaneously detect and correctly estimate pose of objects. We present Convolutional Neural Network based architectures for these and demonstrate that leveraging viewpoint estimates can substantially improve local appearance based keypoint predictions. In addition to achieving significant improvements over state-of-the-art in the above tasks, we analyze the error modes and effect of object characteristics on performance to guide future efforts towards this goal.

1. Introduction

There are two ways in which one can describe the pose of the car in Figure 1 - either via its viewpoint or via specifying the locations of a fixed set of keypoints. The former characterization provides a global perspective about the object whereas the latter provides a more local one. In this work, we aim to reliably predict both these representations of pose for objects.

Our overall approach is motivated by the theory of global precedence - that humans perceive the global structure before the fine level local details [27]. It was also noted by Koenderink and van Doorn [22] that viewpoint determines appearance and several works have shown that larger wholes improve the discrimination performance of parts [31, 26, 29]. Inspired by this philosophy, we propose an algorithm which first estimates viewpoint for the target object and leverages the predicted viewpoint to improve the local appearance based keypoint predictions.

Viewpoint is manifested in a 2D image by the spatial relationships among the different features of the object. Convolutional Neural Network (CNN) [9, 24] based methods which can implicitly capture and hierarchically build on such relations are therefore suitable candidates for viewpoint prediction.

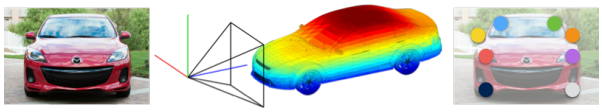

Figure 1: Alternate characterizations of pose in terms of viewpoint and keypoint locations

A robot which merely knows that a cup exists but cannot find its handle will not be able to grasp it. Towards the goal of developing a finer understanding of objects, we tackle the task of predicting keypoints by modeling appearances at multiple scales - a fine scale appearance model, while prone to false positives can localize accurately and a coarser scale appearance model is more robust to mis-localizations. Note that merely reasoning over local appearance is not sufficient to solve the task of keypoint prediction. For example, the notion of the 'front wheel' assumes its meaning in context of the whole bicycle. The local appearance of the patch might also correspond to the 'back wheel' - it is because we know the bicycle is front facing that we are able to disambiguate. Motivated by this, we use the viewpoint predicted by our system to improve the local appearance based keypoint predictions.

Our proposed algorithm, as illustrated in Figure 2 has the following components -

Viewpoint Prediction : We formulate the problem of viewpoint prediction as predicting three euler angles (azimuth, elevation and cyclorotation) corresponding to the instance. We train a CNN based architecture which can implicitly capture and aggregate local evidences for predicting the euler angles to obtain a viewpoint estimate.

Local Appearance based Keypoint Activation : We propose a fully convolutional CNN based architecture to model local part appearance. We capture the appearance at multiple scales and combine the CNN responses across scales to obtain a resulting heatmap which corresponds to a spatial log-likelihood distribution for each keypoint.

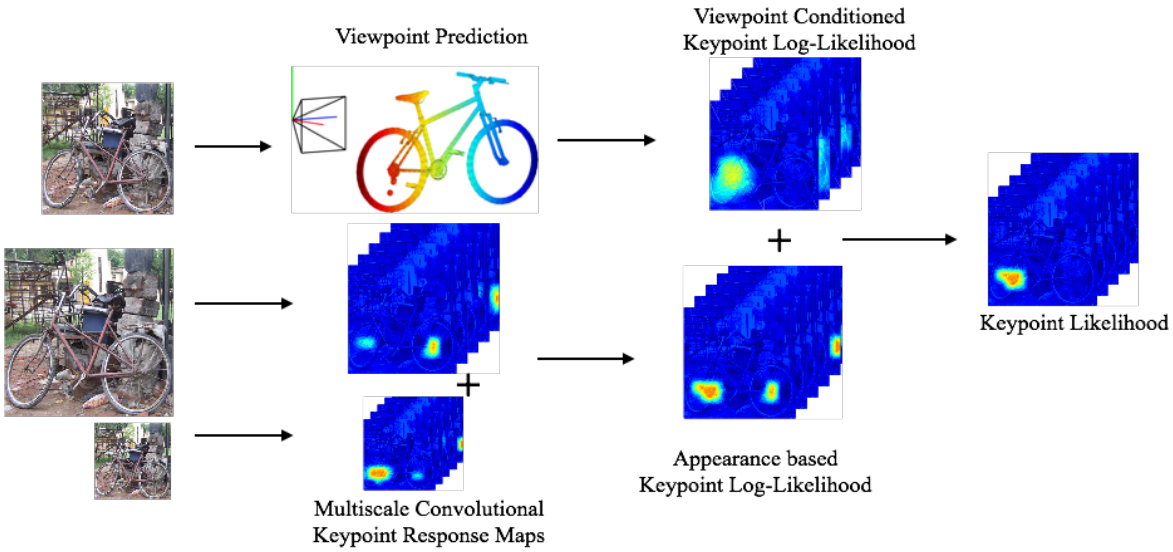

Figure 2: Overview of our approach. To recover an estimate of the global pose, we use a CNN based architecture to predict viewpoint. For each keypoint, a spatial likelihood map is obtained via combining multiscale convolutional response maps and it is then combined with a likelihood conditioned on predicted viewpoint to obtain our final predictions.

Viewpoint Conditioned Keypoint Likelihood : We propose a viewpoint conditioned keypoint likelihood, implemented as a non-parametric mixture of gaussians, to model the probability distribution of keypoints given the viewpoint prediction. We combine it with the appearance based likelihood computed above to obtain our keypoint predictions.

Keypoint prediction methods have traditionally been evaluated assuming ground-truth boxes as input [1, 21, 25]. This means that the evaluation setting is quite different from the conditions under which these methods would be used - in conjunction with imprecisely localized object detections. Yang and Ramanan [38] argued for the importance of this task for human pose estimation and introduced an evaluation criterion which we adapt to generic object categories. To the best of our knowledge, we are the first to empirically evaluate the applicability of a keypoint prediction algorithm not restricted to a specific object category in this challenging setting.

Furthermore, inspired by the analysis of the detection methods presented by Hoeim *et al*. [18], we present an analysis of our algorithm's failure modes as well as the impact of object characteristics on the algorithm's performance.

2. Related Work

Viewpoint Prediction: Recently, CNNs [9, 24] have been shown to outperform Deformable Part Model (DPM) [8] based methods for recognition tasks [11, 6, 23]. Whereas DPMs explicitly model part appearances and their deformations, the CNN architecture allows such relations to be captured implicitly using a hierarchical convolutional structure. Girshick *et al*. [12] argued that DPMs could also be thought as a specific instantiation of CNNs and therefore training an end-to-end CNN for the corresponding task should outperform a method which instead explicitly models part appearances and relations.

This result is particularly applicable to viewpoint estimation where the prominent approaches, from the initial instance based methods [19] to current state-of-the-art [37, 30] explicitly model local appearances and aggregate evidence to infer viewpoint. Pepik *et al*. [30] extend DPMs to 3D to model part appearances and rely on these to infer pose and Xiang *et al*. [37] introduce a separate DPM component corresponding to each viewpoint. Ghodrati *et al*. [10] differ from the explicit part-based methodology, using a fixed global descriptor to estimate viewpoint. We build on both these approaches by using a method which, while using a global descriptor, can implicitly capture part appearances.

Keypoint Prediction: Keypoint Prediction can be classified into two settings - a) 'Keypoint Localization' where the task is to find keypoints for objects with known bounding boxes and b) 'Keypoint Detection' where bounding box is unknown. This problem has been particularly well studied for humans - tracing back from classic model-based approaches for video [28, 17] to more recent pictorial structure based approaches [38] on challenging single image based real world datasets like LSP[21] or MPII Human Pose [1]. Recently Toshev *et al*. [36] demonstrated that CNN based models can successfully be used for keypoint prediction

for humans and Tompson *et al*. [35] significantly improved upon these results using a purely convolutional approach. These evaluations, however, are restricted to keypoint localizations. A more general task of keypoint detection without assuming ground truth box annotations was also recently introduced for humans by Yang and Ramanan [38] and Gkioxari *et al*. [14, 13] evaluated their keypoint prediction algorithm in this setting.

For generic object categories, annotations for keypoints on the challenging PASCAL VOC dataset [7] were introduced by Bourdev *et al*. [4]. Though similar annotations or fitted CAD models have been successfully used to train better object detection systems [3] as well as for simultaneous object detection and viewpoint estimation [30], the task of keypoint prediction has largely been unaddressed for generic object categories. Long *et al*. [25] recently evaluated keypoint localization results across all PASCAL categories but, to the best of our knowledge, the more general setting of keypoint detection for generic object categories has not yet been explored.

Previous works [32, 15, 16] have also jointly tackled the problem of keypoint detection and pose estimation. While these are perhaps the closest to ours in terms of goals, they differ markedly in methodology - they explicitly aggregate local evidence for pose estimation and have either been restricted to a specific object category [15, 16] or use instance model based matching [32]. Long *et al*. [25], on the other hand share many commonalities with our methodology for the task of keypoint prediction - convolutional keypoint detections augmented with global priors to predict keypoints. However, we show that we can significantly improve their results by combining multiscale convolutional predictions from a trained CNN with a more principled, viewpoint estimation based global model. Both [16, 25] only evaluate keypoint localization performance whereas we also evaluate our method in the setting of keypoint detection.

3. Viewpoint Estimation

3.1. Formulation

We formulate the global pose estimation for rigid categories as predicting the viewpoint wrt to a canonical pose. This is equivalent to determining the three euler angles corresponding to azimuth (ϕ), elevation(φ) and cyclo-rotation(ψ). We frame the task of predicting the euler angles as a classification problem where the classes $\{1, \dots N_\theta\}$ correspond to N_θ disjoint angular bins. We note that the euler angles, and therefore every viewpoint, can be equivalently described by a rotation matrix. We will use the notion of viewpoints, euler angles and rotation matrices interchangeably.

3.2. Network Architecture and Training

Let N_c be the number of object classes, N_a be number of angles to be predicted per instance. The number of output units per class is $N_a * N_\theta$ resulting in a total of $N_c * N_a * N_\theta$ outputs. We adopt an approach similar to Girshick *et al*. [11] and finetune a CNN model whose weights are initialized from a model pretrained on the Imagenet [5] classification task. We experimented with the architectures from Krizhevsky *et al*. [23] (denoted as TNet) and Simonyan *et al*. [33] (denoted as ONet). The architecture of our network is the same as the corresponding pre-trained network with an additional fully-connected layer having $N_c * N_a * N_\theta$ output units. We provide an alternate detailed visualization of the network architecture in the supplementary material.

Instead of training a separate CNN for each class, we implement a loss layer that selectively considers the $N_a * N_\theta$ outputs corresponding the class of the training instance and computes a logistic loss for each of the angle predictions. This allows us to train a CNN which can jointly predict viewpoint for all classes, thus enabling learning a shared feature representation across all categories. We use the Caffe framework [20] to train and extract features from the CNN described above. We augment the training data with jittered ground-truth bounding boxes that overlap with the annotated bounding box with IoU > 0.7. Xiang *et al*. [37] provide annotations for (ϕ, φ, ψ) corresponding to all the instances in the PASCAL VOC 2012 detection train, validation set as well as for ImageNet images. We use the PASCAL train set and the ImageNet annotations to train the network described above and use the PASCAL VOC 2012 validation set annotations to evaluate our performance.

4. Viewpoint Informed Keypoint Prediction

As we noted earlier, parts assume their meaning in context of the whole. Thus, in addition to local appearance, we should take into account the global context. To operationalize this observation, we propose a two-component approach to keypoint prediction.

4.1. Multiscale Convolutional Response Maps

We use CNN based architectures to learn the appearance of keypoints across an object class. Using a fully convolutional architecture allows us to capture local appearance in a more hierarchical and robust way than HOG feature based models while still allowing for efficient inference by sharing computations across evaluations at different spatial locations in the same image.

Let C denote the set of classes, K_c denote the set of keypoints for class c and $N_c = |K_c|$. The total number of keypoints N_{kp} is therefore $\sum_{c \in C} N_c$. We train a fully convolutional network with an input size (384×384) such that the channels in its last layer correspond to the keypoints i.e.

we use a loss which forces the channels in the last layer to only fire at positions which correspond to the locations of the respective keypoint. The CNN architecture we use has the convolutional layers from ONet followed by an additional convolution layer with the output size $12 \times 12 \times N_{kp}$ such that each channel of the output corresponds to a specific keypoint of a particular class.

The architecture enforces that the receptive field of an output unit in the location (i, j) has a centre corresponding to $(32 * i, 32 * j)$ in the input image. For each training instance with annotated keypoints with locations $\{(x_k, y_k)|k \in K_c\}$, we construct a target response map T with $T(k_i, k_j, k) = 1$ and zero otherwise (where (k_i, k_j) is the index of the unit with its receptive field's centre closest to the annotated keypoint). For each keypoint, this is similar to training with multiple classification examples per image centered at the repective fields of output units, akin to the formulation used for object detection by Szegedy *et al.* [34]. Similar to the details described in section 3.2, we use a loss layer that only selects the channels corresponding to the instance class and implements a euclidean loss between the output and the target map, thus enabling us to jointly train a single network to predict keypoints for all classes. We train using the annotations from Bourdev *et al.* [4] and use ground truth and jittered boxes as training examples.

The above network captures the appearance of the keypoints at a particular scale. A coarser scale would be more robust to false positives as it captures more context but would not be able to localize well. In order to benefit from the predictions at a coarser level, without compromising localization, we propose using a multiscale ensemble of networks. We therefore train another network with exactly the same architecture with a smaller input size (192×192) and a smaller output size $6 \times 6 \times N_{kp}$. We upsample the outputs of the smaller network and linearly combine them with the outputs of the larger network to get a spatial log-likelihood response map $L(\cdot, \cdot, k)$ for each keypoint k.

4.2. Viewpoint Conditioned Keypoint Likelihood

If we know that a particular car is left-facing, we'd expect its left wheels to be visible but not the right wheels. In addition to the ability to predict visibility, we'd also have a strong intuition about the approximate locations of the keypoints. If the problem setting was restricted to a particular instance, the exact locations of the keypoints could be inferred geometrically from the exact global pose. However, the two assumptions that would allow this approach do not hold true - we have to deal with different instances of the object category and our inferred global pose would only be approximate. To counter this, we propose a non-parametric solution - we would expect the keypoints of a given instance to lie at positions similar to other training instances whose global pose is close to the predicted global pose for the given instance.

Let the training instances for class c be denoted by $\{R^i, \{(x_k^i, y_k^i)|k \in K_c\}\}$ where R_i is the rotation matrix and $\{(x_k^i, y_k^i)|k \in K_c\}$ the annotated keypoints corresponding to the i_{th} instance. Let R be the predicted rotation matrix corresponding to which we want a prior for keypoint locations denoted by P st $P(i, j, k)$ indicates the likelihood of keypoint k being present at location (i, j). Let $\Delta(R_1, R_2) = \frac{\|log(R_1^T R_2)\|_F}{\sqrt{2}}$ denote the geodesic distance between rotation matrices R_1, R_2 and $N(R) = \{i|\Delta(R, R_i) < \frac{\pi}{6}\}$ represent the the training instances whose viewpoint is close to the predicted viewpoint. Our non-parametric global pose conditional likelihood (P) is defined as a mixture of gaussians and we combine it with the local appearance likelihood (L) to get keypoint locations as follows -

$$P(\cdot, \cdot, k) = \frac{1}{|N(R)|} \sum_{i \in N(R)} \mathcal{N}((x_k^i, y_k^i), \sigma I) \quad (1)$$

$$(x_k, y_k) = \underset{y,x}{argmax}\ log(P(x, y, k)) + L(x, y, k) \quad (2)$$

Note that all the coordinates above are normalized by warping the instance bounding box to a fixed size (12×12) and we choose $\sigma = 2$.

5. Experiments : Viewpoint Prediction

In this section, we use the the PASCAL3D+ [37] annotations to evaluate the viewpoint estimation performance of our approach in the two different settings described below -

5.1. Viewpoint Estimation with Ground Truth box

To analyze the performance of our viewpoint estimation method independent of factors like mis-localization, we first tackle the task of estimating the viewpoint of an object with known bounds. Let $\Delta(R_1, R_2) = \frac{\|log(R_1^T R_2)\|_F}{\sqrt{2}}$ denote the geodesic distance function over the manifold of rotation matrices. $\Delta(R_{gt}, R_{pred})$ captures the difference between ground truth viewpoint R_{gt} and predicted viewpoint R_{pred}. We use two complementary metrics for evaluation -

- **Median Error :** The common confusions for the task of viewpoint estimation often are predictions which are far apart (eg. left facing vs right facing car) and the median error $(MedErr)$ is a widely use metric that is robust to these if a significant fraction of the estimates are accurate.

- **Accuracy at θ :** A small median error does not necessarily imply accurate estimates for all instances, a complementary performance measure is the fraction of instances whose predicted viewpoint is within a fixed threshold of the target viewpoint. We denote this metric by Acc_θ where θ is the threshold. We use $\theta = \frac{\pi}{6}$.

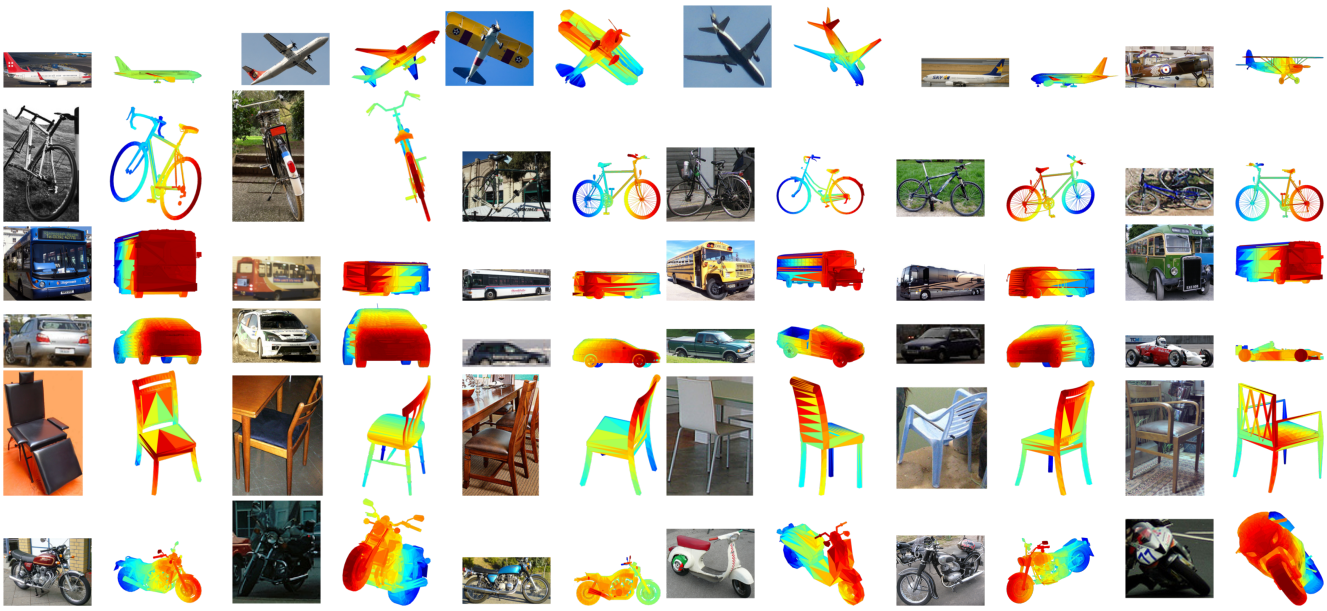

Figure 3: Viewpoint predictions for unoccluded groundtruth instances using our algorithm. The columns show 15th, 30th, 45th, 60th, 75th and 90th percentile instances respectively in terms of the error. We visualize the predictions by rendering a 3D model using our predicted viewpoint.

	aero	bike	boat	bottle	bus	car	chair	table	mbike	sofa	train	tv	mean
$Acc_{\frac{\pi}{6}}$ (Pool5-TNet)	0.27	0.18	0.36	0.81	0.71	0.36	0.52	0.52	0.38	0.67	0.7	0.71	0.52
$Acc_{\frac{\pi}{6}}$ (fc7-TNet)	0.5	0.44	0.39	0.88	0.81	0.7	0.39	0.38	0.48	0.44	0.78	0.65	0.57
$Acc_{\frac{\pi}{6}}$ (ours-TNet)	0.78	0.74	0.49	**0.93**	0.94	**0.90**	0.65	**0.67**	0.83	0.67	0.79	0.76	0.76
$Acc_{\frac{\pi}{6}}$ (ours-ONet)	**0.81**	**0.77**	**0.59**	**0.93**	**0.98**	0.89	**0.80**	0.62	**0.88**	**0.82**	**0.80**	**0.80**	**0.81**
$MedErr$ (Pool5-TNet)	42.6	52.3	46.3	18.5	17.5	45.6	28.6	27.7	37	25.9	20.6	21.5	32
$MedErr$(fc7-TNet)	29.8	40.3	49.5	13.5	7.6	13.6	45.5	38.7	31.4	38.5	9.9	22.6	28.4
$MedErr$(ours-TNet)	14.7	18.6	31.2	13.5	6.3	**8.8**	17.7	17.4	17.6	15.1	8.9	17.8	15.6
$MedErr$(ours-ONet)	**13.8**	**17.7**	**21.3**	**12.9**	**5.8**	9.1	**14.8**	**15.2**	**14.7**	**13.7**	**8.7**	**15.4**	**13.6**

Table 1: Viewpoint Estimation with Ground Truth box

Recently, Ghodrati *et al*. [10] achieved results comparable to state-of-the art by using a linear classifier over layer 5 features of TNet. We denote this method as 'Pool5-TNet' and implement it as a baseline. To study the effect of end-to-end training of the CNN architecture, we use a linear classifier on top of the fc7 layer of TNet as another baseline (denoted as 'fc7-TNet'). With the aim of analyzing viewpoint estimation independently, the evaluations were restricted only to objects marked as non-occluded and non-truncated and we defer the study of the effects of occlusion/truncation in this setting to section 7.1. The performance of our method and comparisons to the baseline are shown in Table 2. The results clearly demonstrate that end-to-end training improves results and that our method with the TNet architecture performs significantly better than the 'Pool5-TNet' method used in [10]. We also observe a significant improvement by using the ONet architecture and only use this architecture for further experiments/analysis. In figure 3, we show our predictions sorted in terms of the error and it can be seen that the predictions for most categories are reliable even at the 90th percentile.

5.2. Detection and Viewpoint Estimation

Xiang *et al*. [37] introduced the AVP metric to measure advances in the task of viewpoint estimation in the setting where localizations are not known a priori. The metric is similar to the AP criterion used for PASCAL VOC detection except that each detection candidate has an associated viewpoint and the detection is labeled correct if it has a correct predicted viewpoint bin as well as a correct localization

(bounding box IoU > 0.5). Xiang *et al*. [37] also compared to Pepik *et al*. [30] on the AVP metric using various viewpoint bin sizes and Ghodrati *et al*. [10] also showed comparable results on the metric. To evaluate our method, we obtain detections from RCNN [11] using MCG [2] object proposals and augment them with a pose predicted using the corresponding detection's bounding box. We note that there are two issues with the AVP metric - it only evaluates the prediction for the azimuth (ϕ) angle and discretizes viewpoint instead of treating it continuously. Therefore, we also introduce two additional evaluation metrics which follow the IoU > 0.5 criteria for localization but modify the criteria for assigning a viewpoint prediction to be correct as follows -

- $AVP_\theta : \delta(\phi_{gt}, \phi_{pred}) < \theta$
- $ARP_\theta : \Delta(R_{gt}, R_{pred}) < \theta$

Note that ARP_θ requires the prediction of all euler angles instead of just ϕ and therefore, is a stricter metric.

The performance of our CNN based approach for viewpoint prediction is shown in Table 2 and it can be seen that we significantly outperform the state-of-the-art methods across all categories. While it is not possible to compare our pose estimation performance independent of detection with DPM based methods like [37, 30], an indirect comparison results from the analysis using ground truth boxes where we demonstrate that our pose estimation approach is an improvement over [10] which in turn performs similar to [37, 30] while using similar detectors.

	AVP				$AVP_{\frac{\pi}{6}}$	$ARP_{\frac{\pi}{6}}$
Number of bins	**4**	**8**	**16**	**24**	**-**	**-**
Xiang *et al*. [37]	19.5	18.7	15.6	12.1	-	-
Pepik *et al*. [30]	23.8	21.5	17.3	13.6	-	-
Ghodrati *et al*. [10]	24.1	22.3	17.3	13.7	-	-
ours	**49.1**	**44.5**	**36.0**	**31.1**	50.7	46.5

Table 2: Mean performance of our approach for various metrics. We report the performance for individual classes with the supplementary material

6. Experiments : Keypoint Prediction

The task of keypoint prediction is commonly studied in the setting with known location of the object but some methods, restricted to specific categories like 'people' recently evaluated their performance in the more general detection setting. We extend these metrics to generic categories and evaluate our predictions in both the settings using the following metrics proposed by Yang and Ramanan [38] -

- PCK (Keypoint Localization) : For each annotated instance, the algorithm predicts a location for each keypoint and a groundtruth keypoint is said to have been found correctly if the corresponding prediction lies within $\alpha * max(h, w)$ of the annotated keypoint with the corresponding object's dimension being (h, w). For each keypoint, we measure the fraction of objects where it was found correctly.

- APK (Keypoint Detection) : A keypoint candidate is deemed correct if it lies within $\alpha * max(h, w)$ of a groundtruth keypoint. Each keypoint hypothesis has an associated score and the area under the precision-recall curve is used as the evaluation criterion.

We use the keypoint annotations from [4] and use the PASCAL VOC train set for training and the validation set images for evaluation.

6.1. Keypoint Localization

The performance of our system and comparison to [25] are shown in Table 3. We denote by 'conv6' ('conv12') the predictions using only the 6×6 (12×12) output size network, by 'conv6+conv12' the predictions using the multiscale convolutional response and by 'conv6+conv12+pLikelihood' the predictions using our full system. Our baseline system ('conv6+conv12') performs much better than [25], indicating the importance of end-to-end training and multiscale response maps. We also see that incorporating the viewpoint conditioned likelihood induces a significant performance gain.

6.2. Keypoint Detection

Given an image, we use RCNN [11] combined with MCG [2] object proposals to obtain detection candidates, each comprising of a class label and location. We then predict keypoints on each candidate using our system and score each keypoint hypothesis by linearly combining the keypoint log-likelihood score and the object detection system score. Our results for the task of keypoint detection are summarized in Table 4. The pose conditioned likelihood consistently improves the local appearance based predictions. Though the task of keypoint detection on PASCAL VOC has not yet been analyzed for categories other than person, we believe our results of 33.2% mean APK with a reasonably strict threshold indicate a promising start.

The above results support our three main assertions - a global prior obtained in the form of a viewpoint conditioned likelihood improves the local appearance based predictions, that end-to-end trained CNNs can effectively model part appearances and combining responses from multiple scales significantly improves performance.

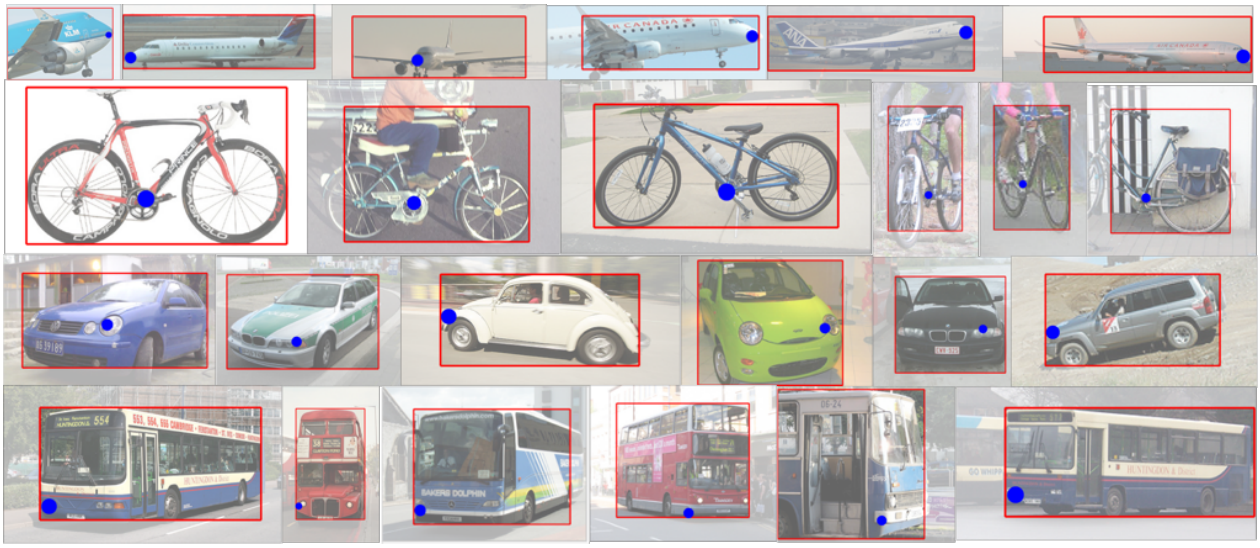

Figure 4: Visualization of keypoints predicted in the detection setting. We visualize every 15^{th} detection, sorted by score, for 'Nosetip' of aeroplanes, 'Crankcentre' of bicycles, 'Left Headlight' of cars and 'Right Base' of buses.

PCK[$\alpha = 0.1$]	**aero**	**bike**	**boat**	**bottle**	**bus**	**car**	**chair**	**table**	**mbike**	**sofa**	**train**	**tv**	**mean**
Long *et al.* [25]	53.7	60.9	33.8	72.9	70.4	55.7	18.5	22.9	52.9	38.3	53.3	49.2	48.5
conv6 (coarse scale)	51.4	62.4	37.8	65.1	60.1	59.9	34.8	31.8	53.6	44	52.3	41.1	49.5
conv12 (fine scale)	54.9	66.8	32.6	60.2	80.5	59.3	35.1	37.8	58	41.6	59.3	53.8	53.3
conv6+conv12	61.9	74.6	43.6	72.8	84.3	70.0	45.0	44.8	66.7	51.2	66.8	56.8	61.5
conv6+conv12+pLikelihood	**66.0**	**77.8**	**52.1**	**83.8**	**88.7**	**81.3**	**65.0**	**47.3**	**68.3**	**58.8**	**72.0**	**65.1**	**68.8**

Table 3: Keypoint Localization

6.3. Generalization to Articulated Pose

While the focus of our work is pose prediction for rigid objects, we note that our multiscale convolutional response based approach is also applicable for articulated pose estimation. To demonstrate this, we trained our convolutional response map system to detect keypoints for the category 'person' in PASCAL VOC 2012 and achieved an APK = 0.22 which is a significant improvement compared to the state-of-the-art method [13] which achieves APK = 0.15. We refer the reader to [13] for further details on the evaluation metrics for the task of articulated pose estimation.

7. Analysis

An understanding of failure cases and effect of object characteristics on performance can often suggest insights for future directions. Hoeim *et al.* [18] suggested some excellent diagnostics for object detection systems and we adapt those for the task of pose estimation. We evaluate our system's output for both the task of viewpoint prediction as well as keypoint prediction but restrict our analysis to the setting with known bounding boxes - this enables us to analyze our pose estimation method independent of the detection system. We denote as 'large objects' the top third of instances and by 'small objects' the bottom third

Setting	**Mean Error**	**Mean Accuracy**
Default	13.5	0.81
Small Objects	15.1	0.75
Large Objects	12.7	0.87
Occluded Objects	19.9	0.65

Table 5: Object characteristics vs viewpoint prediction error

Setting	**Accuracy**
Error$< \frac{\pi}{9}$	83.7
$\frac{\pi}{9} <$Error $< \frac{2\pi}{9}$	5.7
Error$> \frac{\pi}{9}$ & Error$(\pi - flip) < \frac{\pi}{9}$	5.8
Error$> \frac{\pi}{9}$ & Error$(z - ref) < \frac{\pi}{9}$	6.5
Other	2.9

Table 6: Analysis of error modes for viewpoint prediction

APK[$\alpha = 0.1$]	aero	bike	boat	bottle	bus	car	chair	table	mbike	sofa	train	tv	mean
conv6+conv12	41.9	47.1	15.4	29.0	58.2	37.1	11.2	8.1	40.7	25.0	36.9	25.5	31.3
conv6+conv12+pLikelihood	**44.9**	**48.3**	**17.0**	**30.0**	**60.8**	**40.7**	**14.6**	**8.6**	**42.8**	**25.7**	**38.3**	**26.2**	**33.2**

Table 4: Keypoint Detection

PCK[$\alpha = 0.1$]	aero	bike	boat	bottle	bus	car	chair	table	mbike	sofa	train	tv	mean
Default	66.0	77.8	52.1	83.8	88.7	81.3	65.0	47.3	68.3	58.8	72.0	65.1	68.8
Occluded Objects	55.2	56.6	38.7	68.8	64.4	62.8	48.1	40.5	53.1	59.6	68.6	47.3	55.3
Small Objects	51.6	66.4	48.1	81.2	85	67.4	57.4	48.2	57.9	53.8	57.4	56.8	60.9
Large Objects	74.6	87.4	57.2	86.3	90.9	90.6	65.1	37.7	76.1	68.5	74.1	65.3	72.8
left/right	71.1	80.2	53.4	84.4	90.9	84.1	74.7	49.2	69.8	63.4	75.0	68.2	72.0
PCK[$\alpha = 0.2$]	79.9	88.7	69.1	95.2	92	88.3	79.6	67.5	87.3	72.2	82.2	78.1	81.7

Table 7: Analysis of Keypoint Prediction

of instances. The label 'occluded' describes all the objects marked as truncated or occluded according to the PASCAL VOC annotations. We summarize our observations below.

7.1. Viewpoint Prediction

Object Characteristics : Table 5 shows the effect of object characteristics by reporting the mean across the classes of the median viewpoint error and accuracy. We can see that the method performs worse for occluded objects. There is also a significant difference between the performance for small and large objects - while such error trends are acceptable in the robotic setting where ambiguity for the farther objects is tolerable, one may need to capture more context to perform well without higher resolution input.

Error Modes: Since it is difficult to characterize error modes for generic rotations, we restrict the analysis to only the predicted azimuth. Assuming the image plane to be XY, we denote by $Z - ref$ the pose for the instance reflected along the XY plane and by $\pi - flip$ a rotation of π along the Z axis. Table 6 reports the percentage of instances whose predicted pose corresponds to various modes. We observe that these error modes are equally common and that only about 3% of the errors are not explained by these.

Note that we exclude 'diningtable' and 'bottle' categories from the above analysis due to small number of unoccluded instances and insignificant variations respectively.

7.2. Keypoint Prediction

We use the PCK metric (section 6.2) to characterize our algorithm's performance for various settings. Our results using the full method (local appearance combined with viewpoint conditioned likelihood) are reported in Table 7. We report the analysis using various components (single scale prediction, purely local appearance etc.) in the supplementary material.

Object Characteristics : The effect of object characteristics is similar to the viewpoint prediction setting - occluded objects are not handled well and there is a significant performance gap between small and large objects.

Error Modes : In the 'left/right' setting, we label a prediction to be correct if it was in the vicinity of the corresponding or the laterally inverted keypoint. Surprisingly, the performance is similar to the base performance - indicating that laterally symmetric keypoints are not a significant error mode. The difference between the base performance and $PCK[\alpha = 0.2]$ analyzes the inaccurate localizations which we find to be the main source of error.

8. Conclusion

We have presented an algorithm which leverages CNN architectures to predict viewpoint, and combines multiscale appearance with a viewpoint conditioned likelihood to predict keypoints. We demonstrated that our approach significantly improve state-of-the-art in settings with and without annotated bounding boxes for both viewpoint and keypoint prediction tasks. We also present evaluations for the keypoint detection setting alongwith a detailed ablation study of our performance on various tasks and hope that these will contribute towards progress on the task of pose estimation for generic objects. We will make our code and trained models publicly available.

Acknowledgements

The authors would like to thank Abhishek Kar, João Carreira and Saurabh Gupta for their valuable comments. This work was supported in part by NSF Award IIS-1212798 and ONR MURI - N00014-10-1-0933 and the Berkeley Graduate Fellowship. We gratefully acknowledge NVIDIA corporation for the donation of Tesla GPUs for this research.

References

[1] M. Andriluka, L. Pishchulin, P. Gehler, and B. Schiele. 2d human pose estimation: New benchmark and state of the art analysis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2014. 2

[2] P. Arbeláez, J. Pont-Tuset, J. Barron, F. Marques, and J. Malik. Multiscale combinatorial grouping. In *Computer Vision and Pattern Recognition*, 2014. 6

[3] H. Azizpour and I. Laptev. Object detection using strongly-supervised deformable part models. In *Proceedings of the 12th European Conference on Computer Vision - Volume Part I*, ECCV'12, pages 836–849, Berlin, Heidelberg, 2012. Springer-Verlag. 3

[4] L. Bourdev, S. Maji, T. Brox, and J. Malik. Detecting people using mutually consistent poselet activations. In *European Conference on Computer Vision (ECCV)*, 2010. 3, 4, 6

[5] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009. 3

[6] J. Donahue, Y. Jia, O. Vinyals, J. Hoffman, N. Zhang, E. Tzeng, and T. Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. *arXiv preprint arXiv:1310.1531*, 2013. 2

[7] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html. 3

[8] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan. Object detection with discriminatively trained part-based models. *TPAMI*, 2010. 2

[9] K. Fukushima. Neocognitron: A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, 36:193–202, 1980. 1, 2

[10] A. Ghodrati, M. Pedersoli, and T. Tuytelaars. Is 2d information enough for viewpoint estimation? In *Proceedings of the British Machine Vision Conference*. BMVA Press, 2014. 2, 5, 6

[11] R. Girshick, J. Donahue, T. Darrell, and J. Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 2014. 2, 3, 6

[12] R. B. Girshick, F. N. Iandola, T. Darrell, and J. Malik. Deformable part models are convolutional neural networks. *CoRR*, abs/1409.5403, 2014. 2

[13] G. Gkioxari, B. Hariharan, R. Girshick, and J. Malik. R-cnns for pose estimation and action detection. *CoRR*, abs/1406.5212, 2014. 3, 7

[14] G. Gkioxari, B. Hariharan, R. Girshick, and J. Malik. Using k-poselets for detecting people and localizing their keypoints. In *Computer Vision and Pattern Recognition (CVPR)*, 2014. 3

[15] D. Glasner, M. Galun, S. Alpert, R. Basri, and G. Shakhnarovich. Viewpoint-aware object detection and pose estimation. In *IEEE International Conference on Computer Vision (ICCV)*, pages 1275–1282, 2011. 3

[16] M. Hejrati and D. Ramanan. Analyzing 3d objects in cluttered images. In P. Bartlett, F. Pereira, C. Burges, L. Bottou, and K. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 602–610. 2012. 3

[17] D. Hogg. Model-based vision: a program to see a walking person. *Image and Vision Computing*, 1(1):5 – 20, 1983. 2

[18] D. Hoiem, Y. Chodpathumwan, and Q. Dai. Diagnosing error in object detectors. In *Computer Vision–ECCV 2012*, pages 340–353. Springer Berlin Heidelberg, 2012. 2, 7

[19] D. P. Huttenlocher and S. Ullman. Recognizing solid objects by alignment with an image. *International Journal of Computer Vision*, 5(2):195–212, 1990. 2

[20] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell. Caffe: Convolutional architecture for fast feature embedding. *arXiv preprint arXiv:1408.5093*, 2014. 3

[21] S. Johnson and M. Everingham. Clustered pose and nonlinear appearance models for human pose estimation. In *Proceedings of the British Machine Vision Conference*, 2010. doi:10.5244/C.24.12. 2

[22] J. J. Koenderink and A. J. van Doorn. The internal representation of solid shape with respect to vision. *Biological cybernetics*, 32(4):211–216, 1979. 1

[23] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, 2012. 2, 3

[24] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation applied to handwritten zip code recognition. *Neural Comput.*, 1(4):541–551, Dec. 1989. 1, 2

[25] J. Long, N. Zhang, and T. Darrell. Do convnets learn correspondence? In *NIPS*, 2014. 2, 3, 6, 7

[26] J. McClelland and J. Miller. Structural factors in figure perception. *Perception & Psychophysics*, 26(3):221–229, 1979. 1

[27] D. Navon. Forest before trees: The precedence of global features in visual perception. 1977. 1

[28] J. O'Rourke and N. Badler. Model-based image analysis of human motion using constraint propagation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, PAMI-2(6):522–536, Nov 1980. 2

[29] S. E. Palmer and N. M. Bucher. Configural effects in perceived pointing of ambiguous triangles. *Journal of Experimental Psychology: Human Perception and Performance*, 7(1):88, 1981. 1

[30] B. Pepik, M. Stark, P. Gehler, and B. Schiele. Teaching 3d geometry to deformable part models. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2012. 2, 3, 6

[31] J. R. Pomerantz, L. C. Sager, and R. J. Stoever. Perception of wholes and of their component parts: Some configural superiority effects. *Journal of Experimental Psychology-human Perception and Performance*, 3:422–435, 1977. 1

[32] S. Savarese and L. Fei-Fei. 3d generic object categorization, localization and pose estimation. In *IEEE International Conference on Computer Vision (ICCV)*, 2007. 3

[33] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014. 3

[34] C. Szegedy, A. Toshev, and D. Erhan. Deep neural networks for object detection. In *Advances in Neural Information Processing Systems*, pages 2553–2561, 2013. 4

[35] J. Tompson, A. Jain, Y. LeCun, and C. Bregler. Joint training of a convolutional network and a graphical model for human pose estimation. *CoRR*, abs/1406.2984, 2014. 2

[36] A. Toshev and C. Szegedy. Deeppose: Human pose estimation via deep neural networks. In *2014 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2014, Columbus, OH, USA, June 23-28, 2014*, pages 1653–1660. IEEE, 2014. 2

[37] Y. Xiang, R. Mottaghi, and S. Savarese. Beyond pascal: A benchmark for 3d object detection in the wild. In *WACV*, 2014. 2, 3, 4, 5, 6

[38] Y. Yang and D. Ramanan. Articulated pose estimation with flexible mixtures-of-parts. In *Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition*, CVPR '11, pages 1385–1392, Washington, DC, USA, 2011. IEEE Computer Society. 2, 3, 6