Learning Models for Actions and Person-Object Interactions with Transfer to Question Answering

Arun Mallya(✉) and Svetlana Lazebnik

University of Illinois at Urbana-Champaign, Champaign, USA
{amallya2,slazebni}@illinois.edu

Abstract. This paper proposes deep convolutional network models that utilize local and global context to make human activity label predictions in still images, achieving state-of-the-art performance on two recent datasets with hundreds of labels each. We use multiple instance learning to handle the lack of supervision on the level of individual person instances, and weighted loss to handle unbalanced training data. Further, we show how specialized features trained on these datasets can be used to improve accuracy on the Visual Question Answering (VQA) task, in the form of multiple choice fill-in-the-blank questions (Visual Madlibs). Specifically, we tackle two types of questions on person activity and person-object relationship and show improvements over generic features trained on the ImageNet classification task

Keywords: Activity prediction · Deep networks · Visual Question Answering

1 Introduction

The task of Visual Question Answering (VQA) has recently garnered a lot of interest with multiple datasets [1–3] and systems [4–10] being proposed. Many of these systems rely on features extracted from deep convolutional neural networks (CNNs) pre-trained on the ImageNet classification task [11], with or without fine-tuning on the VQA dataset at hand. However, questions in VQA datasets tend to cover a wide variety of concepts such as the presence or absence of objects, counting, brand name recognition, emotion, activity, scene recognition and more. Generic ImageNet-trained networks are insufficiently well tailored for such open-ended tasks, and the VQA datasets themselves are currently too small to provide adequate training data for all types of visual content that are covered in their questions.

Fortunately, we are also seeing the release of valuable datasets targeting specific tasks such as scene recognition [12], age, gender, and emotion classification [13,14], human action recognition [15–17], etc. To better understand and answer questions about an image, we should draw on the knowledge from these specialized datasets. Given a specific question type, we should be able to choose

B. Leibe et al. (Eds.): ECCV 2016, Part I, LNCS 9905, pp. 414–428, 2016.
DOI: 10.1007/978-3-319-46448-0_25

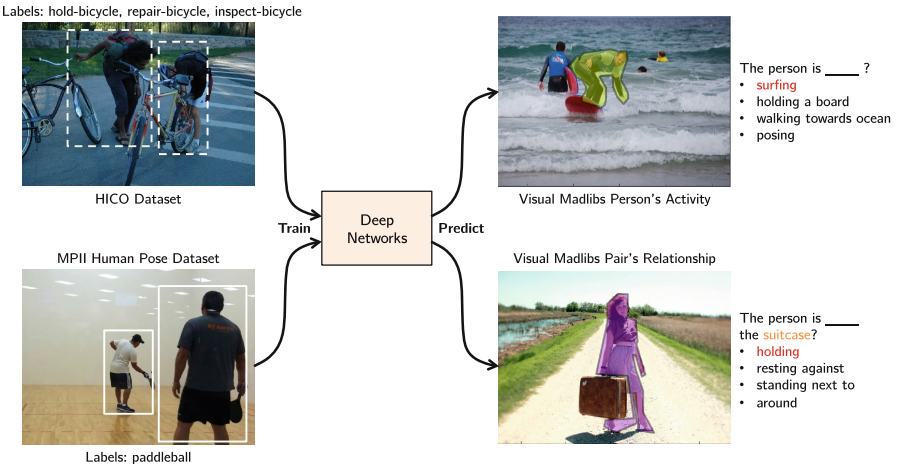


Fig. 1. We train CNNs on the HICO and MPII datasets to predict human activity labels. Our networks fuse features from the full image and the person bounding boxes, which are provided in the MPII dataset and detected in the HICO dataset. We then use these networks to answer two types of multiple choice questions from the MadLibs dataset – about a person's activity, and the relationship between a person and an object.

features from appropriate expert models or networks. In this paper, we show that transferring expert knowledge from a network trained on human activity prediction can not only improve question answering performance, but also help interpret the model's decisions. We train deep networks on the HICO [16] and MPII [17] datasets to predict human activity labels and apply these networks to answer two types of multiple choice fill-in-the-blank questions from the Madlibs dataset [3] on person activity and person-object relationships (Fig. 1). Our contributions are as follows:

1. We propose simple CNN models for predicting human activity labels by fusing features from a person bounding box and global context from the whole image. At training time, the person boxes are provided in the MPII dataset and must be automatically detected in HICO. Our CNN architecture is described in Sect. 3.
2. At training time, we use Multiple Instance Learning (MIL) to handle the lack of full person instance-label supervision and weighted loss to handle the unbalanced training data. The resulting models beat the previous state-of-the-art on the respective datasets, as shown in Sect. 4.
3. We transfer our models to VQA with the help of a standard image-text embedding (canonical correlation analysis or CCA) and show improved accuracy on MadLibs activity and person-object interaction questions in Sect. 5.

2 Related Work

There exist many datasets for action recognition in still images, including the older PASCAL VOC [18] and Stanford 40 Actions [19], and newer MPII Human Pose Dataset [17], COCO-A [20] and *Humans Interacting with Common Objects* (HICO) dataset [16]. The number of actions in some of the newer datasets is an order of magnitude larger than in the older ones, allowing us to learn vocabularies fit for general VQA. The HICO dataset is currently the largest, consisting of nearly 50000 images belonging to 600 human-object interaction categories. Each category in the HICO dataset is composed of a verb-object pair, with objects belonging to the 80 object categories from the MS COCO dataset [21]. On the other hand, the MPII dataset comprises humans performing 393 different activities including walking, running, skating, etc. in which they do not necessarily interact with objects. In this work, we train CNNs with simple architectures on HICO and MPII datasets, and show that they outperform the previous state-of-the-art models.

One limitation of the HICO dataset is that it provides labels for the image as a whole, instead of associating them with specific ground truth person instances. We disambiguate activity label assignment over the people in the image with the help of *Multiple Instance Learning* (MIL) [22], which has been widely used for recognition problems with weakly or incompletely labeled training data [23–26]. In the MIL framework, instead of receiving a set of individually labeled 'instances', the learner receives a set of 'bags,' each of which is labeled negative if all the instances inside it are negative, and labeled positive if it contains at least one positive instance. In this work, we treat each person bounding box as an 'instance' and the image, which contains one or more people in it, as a 'bag'. The exact formulation of our learning procedure is explained in Sect. 3.2.

To recognize a person's activity, we want to use not only the evidence from that person's bounding box, but also some kind of broader contextual information from the image. Previous work suggests the use of latent context boxes [27], multiresolution or zoom-out features [28,29] and complex 2-D recurrent structures [28]. In particular, Gkioxari *et al.* [27] have recently proposed an R*CNN network that chooses a second latent box that overlaps the bounding box of the person and provides the strongest evidence of a particular action being performed. They also proposed a simpler model, the Scene-RCNN, that uses the entire image instead of a chosen box. We explored using latent boxes but found their performance to be lacking on datasets with hundreds of labels, possibly due to overfitting and the infeasibility of thoroughly sampling latent boxes during training. Similarly, we could not obtain good results with multiresolution features owing to overfitting. Instead, we get surprisingly good results with a simpler architecture combining features from the entire image and the bounding box of the person under consideration, outperforming both R*CNN and Scene-RCNN.

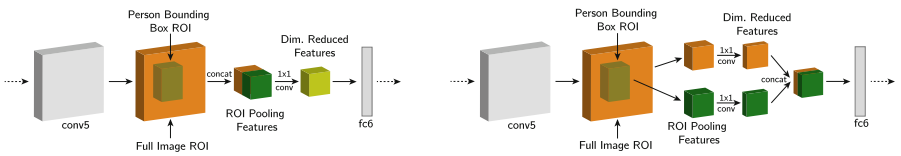
3 Action Recognition Method

3.1 Network Architecture

Our network is based on the *Fast RCNN* [30] architecture with VGG-16 [31]. Fast RCNN includes a new adaptive max pooling layer, referred to as the ROI pooling layer, that replaces the standard max pooling layer (*pool5*) after the set of the first five convolutional layers. This layer takes in a list of bounding boxes, referred to as Regions Of Interest (ROI) and outputs a set of fixed-size feature maps for each input ROI that are then fed to the fully connected layers. During the forward pass of our network, we use two ROIs for each person instance in the image: the tight bounding box of the person, and the full image (we also experimented with using an expanded person bounding box instead of the full image, but found the full image to always work better). The ROI Pooling layer produces a feature of 512 channels and spatial size 7×7 for each ROI. The *fc6* layer of the VGG-16 network expects a feature of size $512 \times 7 \times 7$.

We explore two ways of combining the two ROI features: through stacking and dimensionality reduction (Fig. 2). In the first, referred to as Fusion-1, we stack features from the bounding box and the entire image along the channel dimension and obtain a feature of size $1024 \times 7 \times 7$. A convolutional layer of filter size 1×1 is used to perform dimensionality reduction of channels from 1024 to 512, while keeping the spatial size the same. In the second, referred to as Fusion-2, we first perform dimensionality reduction on the two ROI features individually to reduce the number of channels from 512 to 256 each, and then stack the outputs to obtain an input of size $512 \times 7 \times 7$ for the *fc6* layer.

Our architecture differs from R*CNN and Scene-RCNN [27] in two major ways. First, unlike R*CNN, we do not explicitly try to find a box or set of boxes that provide support for a particular label. Second, while R*CNN and Scene-RCNN independently perform prediction using the two features and then average them, we combine features before prediction. The results presented in Sect. 4 confirm that our "early" fusion strategy gives better performance. Further, our architecture is faster than R*CNN because it does not need to sample boxes during training and testing.



(a) Fusion-1 (b) Fusion-2

Fig. 2. Our networks extract ROI features [30] of dimension $512 \times 7 \times 7$ from both the person bounding box and the full image. The resulting feature is fed into the *fc6* layer of the VGG-16 network. (a) Fusion-1: The two ROI features are stacked and a 1×1 convolution is used for dimensionality reduction. (b) Fusion-2: Each ROI feature is separately reduced using 1×1 convolutions, and the outputs are then stacked.

3.2 Multiple Instance Learning for Label Prediction

In the HICO dataset, if at least one of the people in the image is performing an action, the label is marked as positive for the image. As our architecture makes predictions with respect to a person bounding box, we treat the assignment of labels to different people as latent variables and try to infer the assignment during end-to-end training of the network. For an image I, let B be the set of all person bounding boxes in the image. Using our network described above which takes as input an image I and a person bounding box $b \in B$, we obtain the score of an action a for the image as follows:

$$\text{score}(a; I) = \max_{b \in B} \text{score}(a; b, I) \tag{1}$$

where $\text{score}(a; b, I)$ is the score of action a for the person b in image I. The predicted label for the action can be obtained by passing the score through a logistic sigmoid or softmax unit as required. The max operator enforces the constraint that if a particular action label is active for a given image, then at least one person in the image is performing that action, and when a particular action label is inactive for a given image, then no person in the image is performing the action. During the forward pass, the score and thus the label for the image are predicted using the above relationship. The predicted label is compared to the groundtruth label in order to compute the loss and gradients for backpropagation.

3.3 Weighted Loss Function

Mostajabi *et al.* [29] showed that use of an asymmetric weighted loss helps greatly in the case of an unbalanced dataset. For the HICO dataset, we have to learn 600 independent classifiers per image and this makes for a highly unbalanced scenario, with the number of negative examples greatly outnumbering the positive examples, even for the most populous categories (an average negative to positive ratio of 6000:1, worst case of 38116:1). We thus compute a weighted cross-entropy loss in which positive examples are weighted by a factor of w_p and negative examples by a factor of w_n. Given a training sample (I, B, y) consisting of an image I, set of person bounding boxes or detections B, and ground truth action label vector $y \in \{0, 1\}^C$ for C independent classes, the network produces probabilities of actions being present in the image by passing predictions through a sigmoid activation unit. For any given training sample, the training loss on network prediction $\hat{y}$ is thus given by

$$\text{loss}(I, B, y) = \sum_{i=1}^{C} w_p^i \cdot y^i \cdot \log(\hat{y}^i) + w_n^i \cdot (1 - y^i) \cdot \log(1 - \hat{y}^i) \tag{2}$$

In our experiments, we set $w_p = 10$ and $w_n = 1$ for all classes for simplicity.

4 Activity Prediction Experiments

Datasets. We train and test our system on two different activity classification datasets: HICO [16] and the MPII Human Pose Dataset [17]. The HICO dataset contains labels for 600 human-object interaction activities, any number of which might be simultaneously active for a given image. Labels are provided at the image level even though each image might contain multiple person instances, each performing the same or different activities. The labels can thus be thought of as an aggregate over labels of each person instance in the image. As the person bounding boxes are not provided with the HICO dataset, we run the Faster-RCNN detector [32] with the default confidence threshold of 0.8 on all the train and test images. The obtained person bounding boxes are thus not perfect and might have wrong or missing annotations. The HICO training set contains 38,116 images and the test set contains 9,658 images. The training set is highly unbalanced with 51 out of 600 categories having just 1 positive example.

The MPII dataset contains labels for 393 actions. Unlike in HICO, each image only has a single label together with one or more annotated person instances. All person instances inside an image are assumed to be performing the same task. Ground truth bounding boxes are available for each instance in the training set, so we do not need to use MIL can take advantage of the extra training data available by training on each person instance separately. On the test set, however, only a single point inside the bounding box is provided for each instance, so we run the Faster-RCNN detector to detect people. The training set consists of 15,200 images and 22,900 person instances and the test set has 5,709 images. Similar to HICO, the training set is unbalanced and the number of positive examples for a label ranges from 3 to 476 instances.

HICO Results. On the HICO dataset, we compare the networks described in the previous section with VGG-16 networks trained on just the person bounding boxes and just the full image, as well as with R*CNN and Scene-RCNN. For the latter two, we use the authors' implementation [27]. For all the networks, except the R*CNN, we use a learning rate of 10^{-5}, decayed by a factor of 0.1 every 30000 iterations. For the R*CNN, we use the recommended setting from [27] of a learning rate of 10^{-4}, with a lower and upper intersection over union (IoU) bound for secondary regions of 0.2 and 0.75 and sample 10 secondary regions per person bounding box during a single training pass. We train all networks for 60000 iterations with a momentum of 0.9. Further, all networks are finetuned till the *conv3* layer as in previous work [27,30]. We use a batch size of 10 images, resize images to a maximum size of 640 pixels, and sample a maximum of 6 person bounding boxes per image in order to fit the network in the GPU memory during training with MIL. Consistent with [28,33,34], we initialize our models with weights from the ImageNet-trained VGG-16.

Table 1 presents our comparison. As HICO is fairly new, the only published baseline [16] uses the AlexNet [35] (Table 1a). Using the VGG-16 network improves upon AlexNet by 10 mAP (first line of Table 1b). The VGG-16 network that uses just the person bounding box to make predictions with MIL performs

Table 1. Performance of various networks on the HICO person-activity dataset. Note that usage of the Bounding Box (Bbox) necessitates the usage of Multiple Instance Learning (MIL).

	Method	Full Im.	Bbox	MIL	Wtd. loss	mAP
(a)	AlexNet+SVM [16]	✓				19.4
(b)	VGG-16, full image	✓				29.4
	VGG-16, bounding box		✓	✓		14.6
	VGG-16, R*CNN		✓	✓		28.5
	VGG-16, Scene-RCNN	✓	✓	✓		29.0
(c)	Fusion-1	✓	✓	✓		33.6
	Fusion-1, weighted loss	✓	✓	✓	✓	36.0
	Fusion-2	✓	✓	✓		33.8
	Fusion-2, weighted loss	✓	✓	✓	✓	**36.1**

poorly with only 14.6 mAP (second line of Table 1b). This is not entirely surprising since the object that the person is interacting with is often not inside that person's bounding box. More surprisingly, the R*CNN architecture, which tries to find secondary boxes to support the person box, performs slightly worse than the full-image VGG network. One possible reason for this is that R*CNN has to use MIL twice during training: once for finding the secondary box for an instance, and then again while aggregating over the multiple person instances in the image. Since R*CNN samples only 10 boxes per person instance during each pass of training (same as in [27]), finding the right box for each of the 600 actions might be difficult. The Scene-RCNN, which uses the entire image as the secondary box, needs to do MIL just once, and performs marginally better than R*CNN. Another possible reason why both R*CNN and Scene-RCNN cannot outperform a full-image network is that they attempt to predict action scores independently from the person box and the secondary box before summing them. As we can see from the poor results of our bounding-box-only model (second line of Table 1b), such prediction is hard.

With our fusion networks, we immediately see improvements over the full-image network (Table 1c). The weighted loss, which penalizes mistakes on positive examples more heavily as described in Sect. 3.2, helps push the mAP higher by about 2.5 mAP for both our networks. The Fusion-2 network, which performs dimensionality reduction before local and global feature concatenation, has a slight edge probably due to lower number of parameters (Fusion-1 has 1024×512 parameters for dimensionality reduction and Fusion-2 has $2 \times 512 \times 256$, lesser by a factor of 2).

MPII Results. On the MPII dataset, we compare our networks with previously published baselines from Pischulin *et al.* [17] and Gkioxari *et al.* [27]. Our networks are trained with a learning rate of 10^{-4} with a decay of 0.1 every 12000 iterations, for 40000 iterations. We only finetune till the $fc6$ layer due to the

Table 2. Results on the MPII test set (obtained by submitting our output files by email to the authors of [17]).

Method	mAP
Dense Trajectory + Pose [17]	5.5
VGG-16, R*CNN [27]	26.7
Fusion-1, label per ground truth person instance	32.06
Fusion-2, label per ground truth person instance	**32.24**
Fusion-1, MIL over ground truth person instances	31.68
Fusion-2, MIL over ground truth person instances	31.89
Fusion-2, label per detected person instance	32.02
Fusion-2, MIL over detected person instances	31.81

smaller amount of training data than in HICO. We do not use the weighted loss on this dataset, as we did not find it to make a difference.

Table 2 shows the MPII results. The trend is similar to that in Table 1: our fusion networks outperform previous methods, with Fusion-2 having a lead over Fusion-1. Recall that the MPII training set comes with ground truth person instances, which gives us a chance to examine the effect of MIL. If we assume that the assignment of labels to the people in the image is unknown and use the MIL framework, we see a small dip in performance as opposed to assuming that the label applies to each person in the image (last two rows of Table 2). The latter gives us more training data along with full supervision and improves over MIL by around 0.4 mAP. We also tried training the network with detected person bounding boxes instead of groundtruth boxes and found that the performance was very similar, indicating that groundtruth boxes may not be necessary if there is no ambiguity in assignment of labels.

Qualitative Results. Figure 3 displays some of the predictions of our best-performing network on the HICO dataset. In spite of the lack of explicit supervision of which labels map onto a specific person instance, the network learns to reasonably assign labels to the correct person instance. It is interesting to note a few minor mistakes made by the network: in the top left example, the network confuses the tower in the background for a clock tower, and assigns the label 'no_interaction-clock' to one of the people. In the middle example of the second row, there is a false person detection (marked in red) due to the reflection in the glass, but it does not get an activity prediction since the highest-scoring label has confidence less than 0.5.

Figure 4 shows some of the failures of our system on the HICO dataset. Unusual use-cases of an object such as swinging around a backpack can confuse the deep network into misclassifying the object as in the leftmost image. Since our system relies on detected people, we can either miss or produce false positives, or label the wrong instances as shown in the middle image. Lastly, one drawback of the weakly supervised MIL framework is that it is unable to distinguish labels

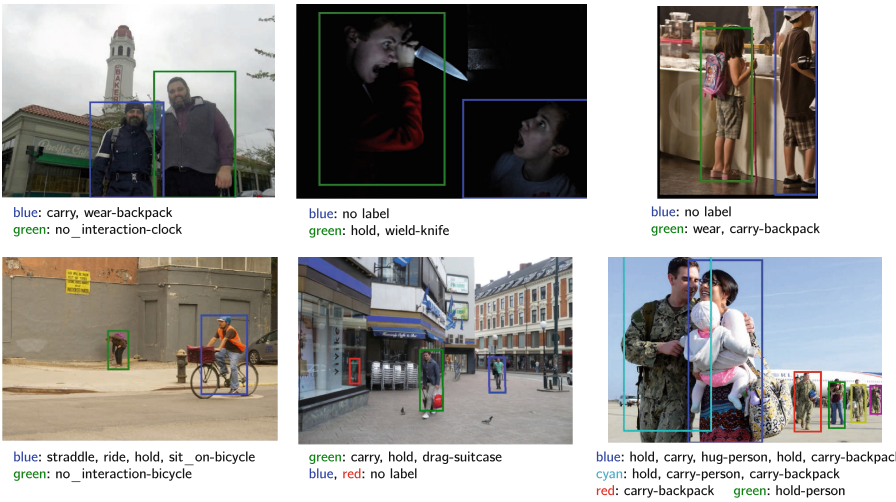


Fig. 3. Predictions of our Fusion-2 model on the HICO test set. Detected person instances are marked in different colors and corresponding action labels are given underneath. (Color figure online)

Fig. 4. Failure examples on HICO. Incorrect classification of objects/actions, wrong interacting person detection, and inability to assign labels to correct person instances due to weak supervision and sampling are common issues.

in a crowded scenario, especially when the crowd occurs only in specific settings such as sports games (right image).

5 Visual Question Answering Results

Dataset and Tasks. In this section, we evaluate the performance of features extracted by our networks on two types of questions from the Madlibs dataset [3] that specifically target people's activities and their interactions with objects. The first type, 'Person's Activity,' asks us to choose an option that best describes the activity of the indicated person/people, while the second type, 'Pair's Relationship,' asks us to describe the relationship between the indicated person and object(s). The indicated people and objects come from ground truth annotations

on the MS COCO dataset [21], from which MadLibs is derived, so there is no need to perform any automatic detection. The prompt is fixed for all questions of a particular type: 'The person/people is/are ____' and 'The person/people is/are ____ the object(s)'.

The training data for MadLibs consists of questions paired with correct answers. There are 26528 and 30640 training examples for the activity and relationship questions, respectively (the total number of distinct images is only about 10 K, but a single image can give rise to multiple questions centered on different person and object instances). In the test data, each question contains four possible answer choices, of which one is the correct answer (or best answer, in case of confusing options). Depending on the way the distractor options are selected, test questions are divided into two categories, Easy and Hard. The test sets for the activity and relationship types have 6501 and 7595 questions respectively, and each comes with Easy and Hard distractor options. Hard options are often quite confusing, with even humans disagreeing on the correct answer. Thus, the performance on filtered hard questions, on which human annotators agree with the 'correct' answer at least 50 % of the times, is also measured. Since MadLibs does not provide a set of multiple choice questions for validation, we created our own validation set of Easy questions by taking 10 % of the training images and following the distractor generation procedure of [3].

Models and Baselines. Similarly to [3], we use normalized Canonical Correlation Analysis (nCCA) [36] to learn a joint embedding space to which the image and the choice features are mapped. Given a question, we select the choice that has the highest cosine similarity with the image features in the joint embedding space as the predicted answer.

On the text side, we represent each of the choices by the average of the 300-dimensional word2vec features [37] of the words in the choice. In the case that a word is out of the vocabulary provided by [38], we represent it with all zeros.

On the image side, we compare performance obtained with three types of features. The first is obtained by passing the entire image, resized to 224×224 pixels, through the vanilla (ImageNet-trained) VGG-16 network and extracting the $fc7$ activations. This serves as the baseline, similar to the original work of Yu *et al.* [3]. The second type of feature is obtained by passing the entire image through our activity prediction network that uses full image inputs. We compare both the $fc7$ activations (of length 4096) and the class label activations (of length 600). The third type of feature is extracted by our Fusion-2 architecture (as detailed in Sect. 3). As our MadLibs question types target one or more specific people in the image, we feed in the person bounding boxes as ROIs to our network (for the relationship questions, we ignore the object bounding box). In the case that a particular question targets multiple people, we perform max pooling over the class label activations of the distinct people to obtain a single feature vector. Note that we found it necessary to use the class label activations before passing them through the logistic sigmoid/softmax as the squashing saturated the scores too close to 0 or 1.

Table 3. Performance of different visual features on Activity and Relationship MadLibs questions (Fil. H. ≡ Filtered Hard). See text for discussion.

Dataset:Network	-	Feature	Person's activity			Pair's relationship		
			Easy	Hard	Fil. H.	Easy	Hard	Fil. H.
ImageNet:VGG-19 [3]	-	fc7	80.7	65.4	68.8	63.0	54.3	57.6
ImageNet:VGG-16	-	fc7	80.79	65.14	67.73	71.45	51.47	56.28
HICO:VGG-16, Full Im.	-	cls_score	86.03	68.74	72.06	77.25	54.10	59.77
HICO:VGG-16, Full Im.	-	fc7	86.54	69.14	72.39	77.96	55.76	61.03
HICO:Fusion-2	-	cls_score	86.66	70.05	73.46	78.29	55.52	61.39
MPII:Fusion-2	-	cls_score	83.23	68.11	70.89	72.81	52.75	57.68
HICO+MPII:Fusion-2	-	cls_score	**87.57**	**71.13**	**74.45**	**78.50**	**56.17**	**62.06**

To train the nCCA model, we used the toolbox of Klein et al. [39]. We set the CCA regularization parameter using the validation sets we created, resulting in values of 0.01 and 0.001 for the $fc7$ and class score features respectively. Our learned nCCA embedding space has dimensionality of 300 (same as the dimensionality of word2vec).

Question Answering Performance. The first two rows of Table 3 contain the accuracies from the vanilla VGG baseline of Yu et al. [3] and our reproduction. Some of our numbers deviate from those of [3], probably owing to the different features used (VGG-16 v/s VGG-19), CCA toolboxes, and hyperparameter selection procedures. From the second row of Table 3, using the vanilla VGG features gives an accuracy of 80.79 % and 71.45 % on the Easy Person Activity and Easy Pair Relationship questions respectively. By extracting features from our full-image network trained on the HICO dataset, we obtain gains of around 6–7 % on the Easy questions (rows 3–4). It is interesting to note that the 600-dimensional class label features give performance comparable to the 4096-dimensional $fc7$ features. Next, features from our Fusion-2 network trained on HICO (row 5) help improve the performance further. The Fusion-2 network trained on the smaller MPII dataset (row 6) gives considerably weaker performance. Nevertheless, we obtain our best performance by concatenating class label predictions from both HICO and MPII (last row of Table 3), since some of the MPII categories are complementary to those of HICO, especially in the cases when a person is not interacting with any object. Compared to our baseline (row 2), we obtain an improvement of 6.8 % on the Easy Activity task, and 7.5 % on the Easy Relationship task. For the Hard Activity task, our improvements are 6 % and 6.7 % on the unfiltered and filtered questions, and for the Hard Relationship task, our improvements are 4.7 % and 5.8 % on the unfiltered and filtered questions respectively.

Qualitative Results. Figure 5 shows a range of correctly answered multiple choice questions using our best-performing features. By examining top labels predicted by our network, we can gain intuition into the choices of our model as these are easily interpretable unlike $fc7$ features of the VGG-16 network.

Fig. 5. Correctly answered questions of the person activity type (first two rows) and person-object relationship type (last row). The subjects of the questions are highlighted in each image. The left column below each image shows the answer choices, with the correct choice marked in red. The right column shows the activity labels and scores predicted by our best network. (Color figure online)

In fact, our top predicted labels often align very closely to the correct answer choice. In the top left image of Fig. 5, the question targets multiple people and the label scores max pooled over the people correctly predict the activity of sitting at and eating at the dining table. In the middle image of the first row, the question targets the skateboarder. Accordingly, our network gives a high score for skateboard-related activities, and a much lower score for the bicyclist in the background. In the rightmost image in the first row, our network also correctly predicts the labels for 'ride, straddle-horse' along with 'wear, carry-backpack' (which is not one of the choices). The middle and right images in the middle row show that our predictions change depending on the target bounding

Fig. 6. Failure examples. The correct choice is marked in red, and the predicted answer in blue. Failure modes mainly belong to three classes as illustrated (left to right): correct predictions but unfamiliar object ('picture'); incorrect predictions ('dog' missed); and a mix of the first two, i.e., partly correct predictions and unfamiliar setting. (Color figure online)

box: the 'hold-book' label has a much higher probability for the boy on the right, even though the network was trained using weak supervision and MIL, as detailed in Sect. 3.

Figure 6 displays some of the common failure modes of our system. In the leftmost image, even though the predicted activity labels are correct, the target object of the question ('picture') is absent from the HICO and MPII datasets so the labels offer no useful information for answering the question. The network can also make wrong predictions, as in the middle image. In the rightmost image, the choices are rather hard and confusing as the person is indeed holding onto a kite as well as a surfboard in an activity best described as 'parasurfing' or 'windsurfing'.

6 Conclusion

In this paper, we developed effective models exploiting local and global context to make person-centric activity predictions and showed how Multiple Instance Learning could be used to train these models with weak supervision. Even though we used a simple global contextual representation, we obtained state-of-the-art performance on two different datasets, outperforming more complex models like R*CNN. In future work, we hope to further explore more sophisticated contextual models and find better ways to train them on our target datasets, which feature hundreds of class labels with highly unbalanced label distributions.

We have also shown how transferring the knowledge from models trained on specialized activity datasets can improve performance on VQA tasks. While we demonstrated this on fairly narrow question types, we envision a more general-purpose system that would have access to many more input features such as person attributes, detected objects, scene information, etc. and appropriately combine them based on the question and image provided.

Acknowledgments. This material is based upon work supported by the National Science Foundation under grants CIF-1302438, IIS-1563727, Xerox UAC, and the Sloan Foundation. We would to thank Licheng Yu for his help with the MadLibs dataset.

References

1. Malinowski, M., Fritz, M.: A multi-world approach to question answering about real-world scenes based on uncertain input. In: NIPS (2014)
2. Antol, S., Agrawal, A., Lu, J., Mitchell, M., Batra, D., Zitnick, C.L., Parikh, D.: VQA: Visual Question Answering. In: ICCV (2015)
3. Yu, L., Park, E., Berg, A.C., Berg, T.L.: Visual Madlibs: fill in the blank image generation and question answering. In: ICCV (2015)
4. Xu, H., Saenko, K.: Ask, attend and answer: exploring question-guided spatial attention for visual question answering. arXiv preprint arXiv:1511.05234 (2015)
5. Zhou, B., Tian, Y., Sukhbaatar, S., Szlam, A., Fergus, R.: Simple baseline for visual question answering. arXiv preprint arXiv:1512.02167 (2015)
6. Gao, H., Mao, J., Zhou, J., Huang, Z., Wang, L., Xu, W.: Are you talking to a machine? Dataset and methods for multilingual image question answering. arXiv preprint arXiv:1505.05612 (2015)
7. Shih, K.J., Singh, S., Hoiem, D.: Where to look: focus regions for visual question answering (2016)
8. Andreas, J., Rohrbach, M., Darrell, T., Klein, D.: Deep compositional question answering with neural module networks. CoRR abs/1511.02799 (2015)
9. Fukui, A., Park, D.H., Yang, D., Rohrbach, A., Darrell, T., Rohrbach, M.: Multimodal compact bilinear pooling for visual question answering and visual grounding. CoRR abs/1606.01847 (2016)
10. Jabri, A., Joulin, A., van der Maaten, L.: Revisiting visual question answering baselines. CoRR abs/1606.08390 (2016)
11. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A.C., Fei-Fei, L.: ImageNet large scale visual recognition challenge. IJCV **115**(3), 211–252 (2015)
12. Zhou, B., Lapedriza, A., Xiao, J., Torralba, A., Oliva, A.: Learning deep features for scene recognition using places database. In: NIPS (2014)
13. Levi, G., Hassner, T.: Age and gender classification using convolutional neural networks. In: CVPR Workshop (2015)
14. Levi, G., Hassner, T.: Emotion recognition in the wild via convolutional neural networks and mapped binary patterns. In: ACM ICMI (2015)
15. Maji, S., Bourdev, L., Malik, J.: Action recognition from a distributed representation of pose and appearance. In: CVPR (2011)
16. Chao, Y.W., Wang, Z., He, Y., Wang, J., Deng, J.: Hico: A benchmark for recognizing human-object interactions in images. In: ICCV (2015)
17. Pishchulin, L., Andriluka, M., Schiele, B.: Fine-grained activity recognition with holistic and pose based features. In: Jiang, X., Hornegger, J., Koch, R. (eds.) GCPR 2014. LNCS, vol. 8753, pp. 678–689. Springer, Heidelberg (2014). doi:10.1007/978-3-319-11752-2_56
18. Everingham, M., Van Gool, L., Williams, C.K., Winn, J., Zisserman, A.: The Pascal Visual Object Classes (VOC) challenge. IJCV **88**, 303–338 (2010)
19. Yao, B., Jiang, X., Khosla, A., Lin, A.L., Guibas, L., Fei-Fei, L.: Human action recognition by learning bases of action attributes and parts. In: ICCV (2011)

20. Ronchi, M.R., Perona, P.: Describing common human visual actions in images. In: BMVC (2015)
21. Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft COCO: common objects in context. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8693, pp. 740–755. Springer, Heidelberg (2014). doi:10.1007/978-3-319-10602-1_48
22. Maron, O., Lozano-Pérez, T.: A framework for multiple-instance learning. In: NIPS (1998)
23. Zhang, C., Platt, J.C., Viola, P.A.: Multiple instance boosting for object detection. In: NIPS (2005)
24. Hoffman, J., Pathak, D., Darrell, T., Saenko, K.: Detector discovery in the wild: joint multiple instance and representation learning. In: CVPR (2015)
25. Vezhnevets, A., Buhmann, J.M.: Towards weakly supervised semantic segmentation by means of multiple instance and multitask learning. In: CVPR (2010)
26. Pinheiro, P.O., Collobert, R.: From image-level to pixel-level labeling with convolutional networks. In: CVPR (2015)
27. Gkioxari, G., Girshick, R., Malik, J.: Contextual action recognition with R* CNN. In: ICCV (2015)
28. Bell, S., Zitnick, C.L., Bala, K., Girshick, R.B.: Inside-outside net: detecting objects in context with skip pooling and recurrent neural networks. CoRR abs/1512.04143 (2015)
29. Mostajabi, M., Yadollahpour, P., Shakhnarovich, G.: Feedforward semantic segmentation with zoom-out features. In: CVPR (2015)
30. Girshick, R.: Fast R-CNN. In: ICCV (2015)
31. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. CoRR abs/1409.1556 (2014)
32. Ren, S., He, K., Girshick, R., Sun, J.: Faster R-CNN: towards real-time object detection with region proposal networks. In: NIPS (2015)
33. Girshick, R., Donahue, J., Darrell, T., Malik, J.: Rich feature hierarchies for accurate object detection and semantic segmentation. In: CVPR (2014)
34. Agrawal, P., Girshick, R., Malik, J.: Analyzing the performance of multilayer neural networks for object recognition. In: Fleet, D., Pajdla, T., Schiele, B., Tuytelaars, T. (eds.) ECCV 2014. LNCS, vol. 8695, pp. 329–344. Springer, Heidelberg (2014). doi:10.1007/978-3-319-10584-0_22
35. Krizhevsky, A., Sutskever, I., Hinton, G.E.: ImageNet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems, pp. 1097–1105 (2012)
36. Gong, Y., Ke, Q., Isard, M., Lazebnik, S.: A multi-view embedding space for modeling internet images, tags, and their semantics. IJCV **106**, 210–233 (2014)
37. Mikolov, T., Sutskever, I., Chen, K., Corrado, G.S., Dean, J.: Distributed representations of words and phrases and their compositionality. In: NIPS (2013)
38. Google: Word2vec trained model. https://code.google.com/archive/p/word2vec/. Accessed 8 Mar 2016
39. Klein, B., Lev, G., Sadeh, G., Wolf, L.: Associating neural word embeddings with deep image representations using fisher vectors. In: CVPR (2015)