Learning Object Localization and 6D Pose Estimation from Simulation and Weakly Labeled Real Images

Jean-Philippe Mercier[1], **Chaitanya Mitash**[2], **Philippe Giguère**[1], **Abdeslam Boularias**[2]
[1]Laval University
[2]Rutgers University

Abstract: This work proposes a process for efficiently training a point-wise object detector that enables localizing objects and computing their 6D poses in cluttered and occluded scenes. Accurate pose estimation is typically a requirement for robust robotic grasping and manipulation of objects placed in cluttered, tight environments, such as a shelf with multiple objects. To minimize the human labor required for annotation, the proposed object detector is first trained in simulation by using automatically annotated synthetic images. We then show that the performance of the detector can be substantially improved by using a small set of *weakly annotated* real images, where a human provides only a list of objects present in each image without indicating the location of the objects. To close the gap between real and synthetic images, we adopt a domain adaptation approach through *adversarial training*. The detector resulting from this training process can be used to localize objects by using its per-object activation maps. In this work, we use the activation maps to guide the search of 6D poses of objects. Our proposed approach is evaluated on several publicly available datasets for pose estimation. We also evaluated our model on classification and localization in unsupervised and semi-supervised settings. The results clearly indicate that this approach could provide an efficient way toward fully automating the training process of computer vision models used in robotics.

Keywords: Object Detection, Pose Estimation, Domain Adaptation, Weakly-Supervised Learning

1 Introduction

Robotic manipulators are increasingly deployed in challenging situations that include significant occlusion and clutter. Prime examples are warehouse automation and logistics, where such manipulators are tasked with picking up specific items from dense piles of a large variety of objects, as illustrated in Figure 1. The challenging nature of this task was highlighted during the recent Amazon Robotics Challenges [3]. These robotic manipulation systems are typically endowed with a perception pipeline that starts with object recognition, followed by the object's six degrees-of-freedom (6D) pose estimation. It is known to to be a computationally challenging problem [1, 4, 5, 6, 7, 8], largely due to the combinatorial nature of the corresponding global search problem. A typical pose estimation algorithm consists in generating a large number of candidate 6D poses for each object in the scene, rendering the CAD models of the objects according to the candidate poses, and comparing the rendered depth image with the observed one for selecting the best candidate poses. The computational efficiency of this search problem is directly affected by the number of candidate poses. Therefore, an accurate point-wise object detector is necessary for reducing the number of candidate poses by focusing the search on relevant parts of the image.

On the other hand, accurate point-wise object detectors can be relatively easily obtained using Convolutional Neural Network (CNN) for object recognition and segmentation [9, 10, 11]. However, CNNs typically require large amounts of annotated images to achieve a good performance. While such large datasets are publicly available for general-purpose computer vision, specialized datasets in certain areas such as robotics and medical image analysis tend to be significantly scarcer and costly to obtain. In a warehouse context, new items are routinely added to inventories. It is thus

2nd Conference on Robot Learning (CoRL 2018), Zrich, Switzerland.

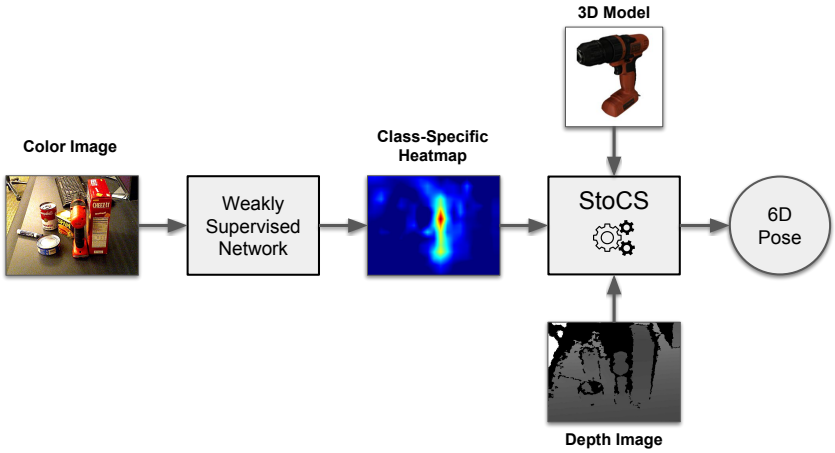


Figure 1: Overview of our approach for 6D pose estimation. This figure shows the pipeline for the drill object of the YCB-video dataset [1]. A deep learning model is trained with weakly annotated images. We use the extracted class-specific heatmaps along with 3D models and depth images to estimate 6D object poses with the Stochastic Congruent Sets (StoCS) method [2]. Further details of the network are available in Section 3.

impractical to collect and manually annotate a new dataset every time an inventory gets updated, particularly if it must covers all possible conditions that a robot may encounter during deployment. One possible solution to alleviate this annotated dataset problem is to employ synthetic images, as they can be automatically annotated. However, the visual features difference between real and synthetic images can large to the point of leading to poor performance on real objects. The problem of learning from data sampled from non-identical distributions is known as *domain adaptation*.

Domain adaptation has been increasingly seen as a solution to bridge between domains, as per recent surveys [12, 13]. Roughly speaking, domain adaptation tries to generalize the learning from a *source domain* to a *target domain*, or in our case, from synthetic to real images. Since labeled data in the target domain is unavailable or partially available, the standard way of doing domain adaptation is to train on labeled source data while trying to minimize the distribution discrepancy between source and target domains. One successful approach is called Domain-Adversarial Training of Neural Networks (DANN) [14]. While being good to learn domain-invariant features, it has a detrimental tendency to align the whole feature distribution together, instead of in a class-specific way. This in turns decreases the discriminative power of the approach. To reduce feature misalignments between classes, the Multi-Adversarial Domain Adaptation (MADA) [15] approach proposes to use a domain discriminator for each class, and weight their input features by their associated class probability outputted by a classification module.

Another approach, called *domain randomization*, circumvents the whole issue by generating so many variant of domains that the true one can be perceived as yet another instance of domain. This approach has been leveraged in the context of object detection by [16]. Interestingly, they show that deep networks can perform as well or even better on unrealistic synthetic data than on realistic. They also show the power of fine-tuning with labeled real images, even on a small set, which we will also explore in our work.

While having a small labeled dataset on a target domain allows to boost performances, it may still require significant human effort for this annotation. This is particularly true for tasks such as object detection, segmentation and pose estimation for which bounding boxes, pixelwise annotations and 6D poses are required. Weakly supervised learning methods significantly decrease this annotation effort, albeit with reduced performance compared to fully-annotated. Some [17, 18] have been shown to be able to retrieve a high level representation of the input data (such as object localization) while only being trained for object classification, without any localization information. To the best of our knowledge, this promising kind of approach has not yet been applied within a robotic manipulation context.

In this paper, we propose to leverage weakly-annotated images and semi-supervised learning in the context of 6D pose estimation. More precisely, our approach consists first in training a classifier through domain adaptation using weakly labeled synthetic and real color images, which allows to retrieve class-specific heatmaps that are used to localize objects. These localization heatmaps are then subsequently refined with an independent 6D pose estimation method called StoCS. Our complete method achieves competitive results on the YCB-video object dataset [1] and Occluded Linemod [7] while using only synthetic images and a few weakly labeled real images per object in training. We also empirically demonstrate that for our test case, using domain adaptation in semi-supervised settings is preferable than training in unsupervised settings and fine-tuning on available weakly labeled real images.

2 Related Works

Learning in Simulation Training with synthetic data has recently gained significant traction, as shown by the multiple synthetic datasets that became available [19, 20, 21, 22, 23, 24]. Some of the related works using these datasets aim at optimizing the realism of the generated images. While high fidelity simulation can decrease to a certain degree the gap between real and synthetic images, it somehow defeats the purpose of using simulation as a cheap way to gather data. To circumvents this issue, [25, 26] proposed to create images using segmented object instances copied on real images. This type of approach is however limited to the number of object views that are available and their respective illumination. Recently, [27, 16] trained object detectors with 3D models rendered in simulation with randomized parameters, such as lighting, number of objects, object poses, and backgrounds, and showed promising results. While [27] only uses synthetic images in training, [16] demonstrated the benefits of fine-tuning on a limited labeled set of real images. [16] also showed that using photorealistic synthetic images does not necessarily improve object detection compared to training on a less realistic synthetic dataset generated with randomized parameters.

Domain Adaptation Domain adaptation techniques [12, 13] can be a useful tool to decrease the distribution discrepancy between different domains. DANN [14] is one of the most popular domain adaptation methods in the last few years. It uses a classifier for a certain task trained on labeled data from a source domain and a domain classifier that classifies whether the input data is from the source or target domain. Both classifiers share the first part of the network, which acts as a feature extractor. The network is trained in an adversarial way: domain classifier parameters are optimized to minimize the domain classification loss and shared parameters are optimized to maximize the domain classification loss. It is possible to achieve this minimax optimization in a single step by using a gradient reversal layer that reverses the sign of the gradient between shared and non-shared parameters of the domain classifier. To the best of our knowledge, the present work is the first use of DANNs for point-wise object localization, which is an important problem in robotic manipulation.

Weakly Supervised Learning We are interested in weakly supervised learning with inexact supervision, for which only coarse-grained labels are available [28]. In [17], a network was trained with weak image-level labels only (classes that are present in images) and max-pooling was used to retrieve approximate location of objects. In [18] the proposed *WILDCAT* model performs classification and weakly supervised point-wise detection and segmentation. This architecture learns multiple localized features for each class and uses a spatial pooling strategy that generalizes to different pooling strategies (max pooling, global average pooling and negative evidence). In the present work, we push the paradigm of minimum human supervision further and propose to train WILDCAT with synthetic images in addition to weakly supervised real ones and use DANN for domain adaptation.

6D Pose Estimation Recent literature in pose estimation focuses on learning to predict 6D poses using deep learning techniques. [1] predicts separately the object center in images for translation and regresses over the quaternion representation for predicting the rotation. [8, 29] predict 3D object coordinates, followed by a RANSAC-based scheme to predict the object's pose. [29] also uses geometric consistency to refine the predictions from the learnt model. These methods need access to several images that are manually labeled with object poses, which is costly to acquire. Some other approaches make use of the object segmentation output to guide a global search process for estimating object poses in the scene [2, 30, 31]. Although the search process could compensate

for errors in prediction when the segmentation module is trained with synthetic data, the domain gap could be large, and a computationally expensive search process may be needed to bridge that gap.

3 Proposed Approach

We present here our approach to object localization and 6D pose estimation using a mix of synthetic images, rendered in *OpenGL*, and weakly-annotated real training images.

3.1 Overview

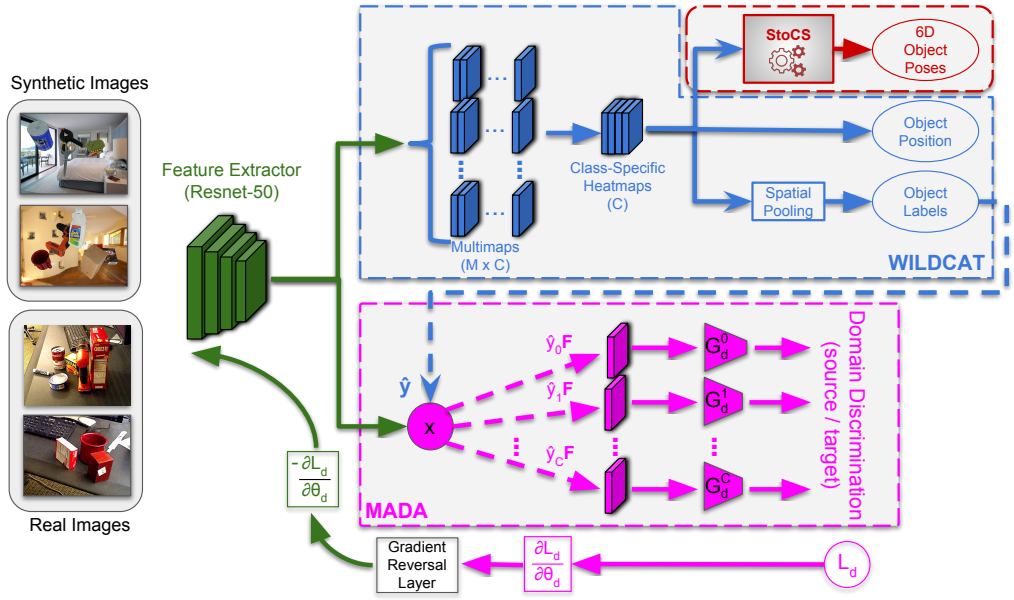


Figure 2: Overview of the proposed approach for object localization and 6D pose estimation with domain adaptation, using a mix of synthetic images and weakly labeled real images.

Figure 2 shows an overview of our proposed system. It comprises *i*) a *ResNet-50* model pre-trained on *ImageNet* as a feature extractor (green), *ii*) a weak classifier inspired from the WILDCAT model [18] (blue), *iii*) Stochastic Congruent Sets (*StoCS*) for 6D pose estimation (red) [2], and *iv*) a Multi-Adversarial Domain Adaptation network (*MADA*) [15] with a gradient reversal layer to learn domain-independent features. The different parts of the system and the training process are explained in the following sections. ResNet extracts features from the images that are useful for object classification, which could also be used for point-wise localization by WILDCAT. To force ResNet to extract similar features for both synthetic and real images, MADA is trained to classify any input image according to its domain $\{\text{real}, \text{synthetic}\}$, and the weights of ResNet are updated such that this classification's error increases. Thus, ResNet is trained to extract features that make it hard for MADA to distinguish between synthetic and real images. MADA's role ends once the training of ResNet and WILDCAT is over. Given a real test image, ResNet and WILDCAT return a heatmap that indicates the label distribution of each pixel. This probability distribution is then fed to StoCS, a robust pose estimation algorithm that is specifically designed to deal with noisy localization.

3.2 Synthetic Data Generation

For synthetic data generation, we use a modified version of the SIXD toolkit[1]. This toolkit generates color and depth images of 3D object models rendered on black backgrounds for which virtual camera viewpoints are sampled on spheres of different radii, following the approach described in [32]. We extended the toolkit by adding the functionality of rendering more than one object per image and also

[1] https://github.com/thodan/sixd_toolkit

used random backgrounds taken from the LSUN dataset [33] instead of generic black backgrounds. Similarly to recent *domain randomization* techniques [34], we observed from our experiments that these simple modifications help transferring from simulation to real environments where there are multiple objects of interest, occlusions and diverse backgrounds. Figure 2 displays some examples of the generated synthetic images that we used for training the pair-wise object detector.

3.3 Weakly Supervised Learning with WILDCAT

The images used for training our system are weakly labeled. Only a list of object classes is provided for the objects present in an image. In order to recover localization from weak labels, we leverage the WILDCAT architecture [18]. Despite being only trained for classification, WILDCAT implicitly recovers localization information for each object through responses in specific feature maps. As a feature extractor, we employ a ResNet-50 pretrained on ImageNet for which the last layers (global average pooling and fully connected layers) are removed. The WILDCAT architecture added on top of this ResNet-50 comprises three main modules: *multimap transfer layer*, *class pooling* and *spatial pooling*. The *multimap transfer layer* consists of 1×1 convolutions that extracts M class-specific modalities per class C. We used $M = 8$, as it achieved the best performance. The *class pooling* module is an average pooling layer that reduces the number of feature maps from MC to C. Then, the *spatial pooling* module selects k regions with maximum/minimum activations to calculate scores for each class. We normalize the scores by standard deviation when the output is passed to our domain adaptation module, MADA, explained in the next section.

3.4 Multi-Adversarial Domain Adaptation with MADA

We use the *Multi-Adversarial Domain Adaptation* (MADA) approach [15] to bridge the "reality gap". MADA extends the *Domain Adversarial Networks* (DANN) approach [14] by using one domain discriminator per class instead of a single global discriminator to help align class-specific features between domains. The loss L_d for the K domain discriminators and input $\mathbf{x_i}$ is defined as

$$L_d = \frac{1}{n} \sum_{k=1}^{K} \sum_{\mathbf{x_i} \in D_s \cup D_t} L_d^k \Big(G_d^k \Big(\hat{y}_i^k G_f(\mathbf{x_i}) \Big), d_i \Big), \tag{1}$$

wherein $i \in \{1, \ldots, n\}$, and $n = n_s + n_t$ is the total number of training images in source domain D_s (synthetic images) and the target domain D_t (real images). G_f is the feature extractor (the same for both domains), $\hat{y}_i^k$ is the probability of label k for image $\mathbf{x_i}$. The probability $\hat{y}_i^k$ is the output of the weak classifier WILDCAT. G_d^k is the k-th domain discriminator and L_d^k is its cross-entropy loss, given the ground truth domain $d_i \in \{\text{synthetic}, \text{real}\}$ of image $\mathbf{x_i}$. Our global objective function is

$$C = \frac{1}{n_s} \sum_{\mathbf{x_i} \in D_s} L_y \Big(G_y \Big(G_f(\mathbf{x_i}) \Big), y_i \Big) - \frac{\lambda}{n} \sum_{k=1}^{K} \sum_{\mathbf{x_i} \in D} L_d^k \Big(G_d^k \Big(\hat{y}_i^k G_f(\mathbf{x_i}) \Big), d_i \Big). \tag{2}$$

The heat-map probability distribution extracted from WILDCAT is used to guide the StoCS algorithm in its search for 6D poses, as explained in the next section.

3.5 Pose Estimation with Stochastic Congruent Sets (StoCS)

The StoCS method [2] is a robust pose estimator that takes as input an object probability heat maps, and outputs their 6D poses. The heat maps provide the probability π of an object O_k being located at a given pixel p_i and is generated by normalizing over an intermediate output of the WILDCAT network with activation w_{p_i},

$$\pi_{p_i \to O_k} = \Big(\frac{w_{p_i} + w_{min}}{w_{max}} \Big)^2. \tag{3}$$

The algorithm then follows the paradigm of a randomized alignment technique and iteratively samples a set of four points, called a base B, on the point cloud S and finds corresponding set of points on the object model M. Each corresponding set of four points defines a rigid transformation T, for which an alignment score is computed between the transformed model cloud and the heatmap

for that object. The optimization criteria is defined as

$$T_{opt} = arg\,max_T \sum_{m_i \in M_k} f(m_i, T, S_k), \tag{4}$$

$$\text{with } f(m_i, T, S_k) = \pi_k(s*), if \mid T(m_i) - s* \mid < \delta_s. \tag{5}$$

The base sampling process in this algorithm considers the joint probability of all four points belonging to the object in question, given as

$$Pr(B \rightarrow O_k) = \frac{1}{Z} \prod_{i=1}^{4} \{\phi_{node}(b_i) \prod_{j=1}^{j<i} \phi_{edge}(b_i, b_j)\}. \tag{6}$$

where ϕ_{node} is obtained from the heatmap and ϕ_{edge} is computed based on the point-pair features of the pre-processed object model. Thus, the method combines the output of the network with the geometric model of objects to obtain base samples which belong to the object with high probability.

4 Weakly Supervised Learning Experiments

In this section, we describe the evaluation of variations of our approach for classification and point-wise localization, in order to assess its various components. We first evaluated our approach with unsupervised learning (no human annotations at all). Then, we evaluated it in weakly semi-supervised settings when a different number of weakly labeled real images are available. We performed these evaluations on the YCB-video dataset [1]. This dataset contains 21 objects with available 3D models and has full annotations for detection and pose estimation on 113,198 training images and 20,531 test images. A subset of 2,949 test images (keyframes) is also available. Our results are reported for this subset, since most images in the bigger test set are video frames that are too similar.

4.1 Unsupervised Domain Adaptation

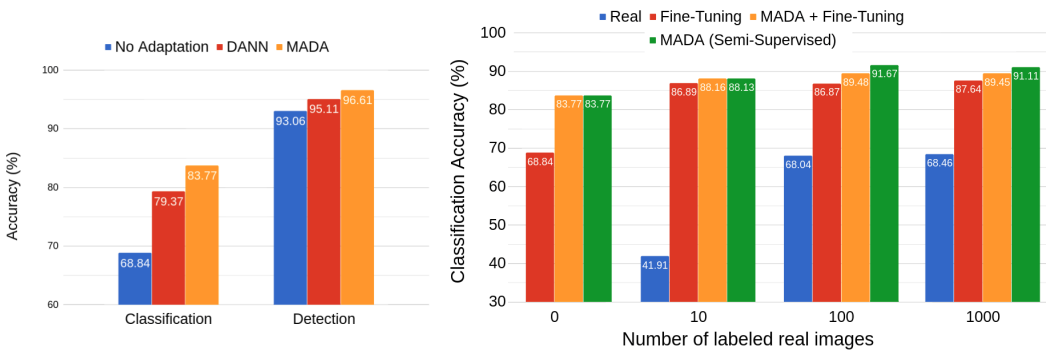
For this experiment, we trained our model with weakly labeled synthetic images and unlabeled real images from the training set of YCB-video dataset [1]. We tested three architecture configurations of domain adaptation: 1) without any domain adaptation module (trained WILDCAT model with synthetic images only), 2) with DANN and 3) with MADA (our proposed pipeline). We evaluated each of these configurations for both classification and detection. For classification, we used the accuracy metric to evaluate our model's capacity to discriminate which objects are in the image. We used a threshold of 0.5 on classification scores to predict the presence or absence of an object. For detection, we employed the point-wise localization metric [17], which is a standard metric to evaluate the ability of weakly supervised networks to localize objects. For each object in the image, the maximum value in their class-specific heatmap was used to retrieve the corresponding pixel in the original image. If that pixel is located inside the bounding box of the object of interest, that is counted as a good detection. Since the class-specific heatmap is a reduced scale of the input image due to pooling, a tolerance equal to the scale factor was added to the bounding box. In our case, a location in the class-specific heatmaps corresponds to a region of 32 pixels in the original image.

We trained the networks with a batch size of 8 (4 per domain), 500 iterations per epoch and 20 epochs. In Figure 3a, we report the average scores of the last 5 epochs over 3 independent random runs for each network variation. These results confirm the importance of having one domain discriminator G_d^k for each of the X objects in the YCB database, instead of a single one for DANN.

4.2 Semi-Supervised Domain Adaptation

As mentioned earlier, a significant challenge for agile deployment of robots in industrial environments is that they ideally should be trained with limited annotated data. We thus evaluated the performance of four different strategies as a function of the number of weakly-labeled real images:

1. Without domain adaptation:
   (a) Real Only: Trained only on weakly labeled real images,
   (b) Fine-Tuning: Trained on synthetic images and then fine-tuned on weakly labeled real images,



(a) (b)

Figure 3: Performance analysis. In (a), we compare classification accuracy and point-wise detection when no label on real images are available. In (b), we compare the performance of different training processes when different numbers of real images are weakly labeled.

2. With domain adaptation:
   (a) Fine-Tuning: Trained on synthetic images and then fine-tuned on weakly labeled real images,
   (b) Semi-Supervised: Trained with synthetic images and weakly labeled real images at the same time.

For 1.a and 1.b, we validate that using fine-tuning on a network pre-trained with synthetic data is preferable to training directly on real images. For 2.a and 2.b, we compare the performance of our approach trained with fine-tuning and in a semi-supervised way using images from both domains at the same time. We are particularly interested in comparing the two approaches, since [35] achieved the lowest error rate compared to any other semi-supervised approach by only using fine-tuning.

Our results are summarized in Figure 3b. From the results, it can be concluded that training with synthetic images improves performance drastically, especially when few labels are available. Also, our approach performs slightly better when trained with available labels (semi-supervised settings) than with fine-tuning where the model is trained in unsupervised settings and fine-tuned on labeled real images.

5 Pose Estimation Experiments

In this section, we evaluate our full approach for 6D pose estimation on YCB-video [1] and Occluded Linemod [7] datasets.

5.1 YCB-Video Dataset

This dataset comprises several frames from 92 video sequences of cluttered scenes created with 21 YCB objects. The training for competing methods [1, 6, 36] is performed using 113,199 frames from 80 video sequences with semantic and pose labels. For our proposed approach, we used only 10 randomly sampled weakly annotated (class labels only) real images per object class combined with synthetic images. The evaluation metric proposed in the dataset benchmark [1] was used to report the accuracy. It uses the average distance (ADD) metric to plot the accuracy-threshold curve, and the area under the curve (AUC) is reported. The ADD metric measures the average distance between model points transformed by the ground truth and the predicted pose.

Results are reported in Table 1. Our proposed method achieves **86.5%** accuracy. It outperforms competing approaches, with the exception of PoseCNN+ICP. However, our approach has a large computational advantage with an average runtime of **0.6 seconds** per object as opposed to approximately **10 second** per object for the modified-ICP refinement for PoseCNN.

Method	Modality	Supervision	Real-images	Accuracy
PoseCNN [1]	RGB	6D pose labels	113,199	75.9
PoseCNN+ICP [1]	RGBD	6D pose labels	113,199	93.0
DeepHeatmaps [6]	RGB	6D pose labels	113,199	66.1
FCN + Drost et. al. [36]	RGBD	Pixelwise class labels	113,199	83.9
OURS	RGBD	Object class labels	210	86.5

Table 1: Pose estimation on YCB-Video dataset

5.2 Occluded Linemod Dataset

This dataset contains 1215 frames from a single video sequence with pose labels for 9 objects from the LINEMOD dataset with high level of occlusion. The training is performed using real, pose labeled images extracted from the LINEMOD dataset (around 1200 images for each object sequence) and using data augmentation techniques. Again, our proposed method simply uses 10 images per object with just class labels for the training and achieves a score of **68.8%** for the ADD evaluation metric. Whereas it outperforms all approaches using RGB only data, its performance is slightly inferior to some competing methods that use RGBD. We suspect the sensor noise leads to error prone surface normal computation and the fact that the StoCS approach uses surface normals extensively for base sampling and in the optimization cost.

Method	Modality	Supervision	Real-images	Accuracy
DeepHeatmaps [6]	RGB	6D pose labels	LINEMOD	28.7
PoseCNN [1]	RGB	6D pose labels	LINEMOD	24.9
PoseCNN+ICP [1]	RGBD	6D pose labels	LINEMOD	78.0
Brachmann et al [8]	RGBD	6D pose labels	LINEMOD	56.6
Michel et. al. [29]	RGBD	6D pose labels	LINEMOD	76.7
OURS	RGBD	Object class labels	10 labels/object	68.8

Table 2: Pose estimation on Occluded-LINEMOD dataset

6 Conclusion

In this paper, we explored the problem of object detection, classification and pose estimation in the context of limited annotated training datasets. To this effect, we proposed a novel deep neural network architecture that merged two state-of-the-art and orthogonal approaches. The first one, the WILDCAT architecture, was used to leverage weakly-labeled real data, enabling the object localization without pose annotation. The second one, a domain adaptation technique called MADA, allowed the use of a mixture of synthetic and real, unannotated data. We experimentally justified the use of MADA over the vanilla version of DANN in Section 4.1. We then used the output of our network to initialize a 6D pose-search algorithm called StoCS. Pose estimation experiments on the YCB-Video and the Occluded Linemod datasets showed that our approach is competitive with recent approaches such as PoseCNN despite requiring significantly less real annotated images. Moreover, we circumvented the need to employ time-consuming algorithms such as ICP.

References

[1] Y. Xiang, T. Schmidt, V. Narayanan, and D. Fox. Posecnn: A convolutional neural network for 6d object pose estimation in cluttered scenes. arXiv preprint arXiv:1711.00199, 2017.

[2] C. Mitash, A. Boularias, and K. Bekris. Robust 6d object pose estimation with stochastic congruent sets. arXiv preprint arXiv:1805.06324, 2018.

[3] N. Correll, K. E. Bekris, D. Berenson, O. Brock, A. Causo, K. Hauser, K. Osada, A. Rodriguez, J. Romano, and P. Wurman. Analysis and Observations From the First Amazon Picking Challenge. IEEE Trans. on Automation Science and Engineering (T-ASE), 2016.

[4] S. Hinterstoisser, V. Lepetit, N. Rajkumar, and K. Konolige. Going further with point pair features. In European Conference on Computer Vision, pages 834–848. Springer, 2016.

[5] Y. Li, G. Wang, X. Ji, Y. Xiang, and D. Fox. Deepim: Deep iterative matching for 6d pose estimation. arXiv preprint arXiv:1804.00175, 2018.

[6] M. Oberweger, M. Rad, and V. Lepetit. Making deep heatmaps robust to partial occlusions for 3d object pose estimation. arXiv preprint arXiv:1804.03959, 2018.

[7] A. Krull, E. Brachmann, F. Michel, M. Ying Yang, S. Gumhold, and C. Rother. Learning analysis-by-synthesis for 6d pose estimation in rgb-d images. In Proceedings of the IEEE International Conference on Computer Vision, pages 954–962, 2015.

[8] E. Brachmann, A. Krull, F. Michel, S. Gumhold, J. Shotton, and C. Rother. Learning 6d object pose estimation using 3d object coordinates. In European conference on computer vision, pages 536–551. Springer, 2014.

[9] A. Zeng, K.-T. Yu, S. Song, D. Suo, E. Walker Jr, A. Rodriguez, and J. Xiao. Multi-view self-supervised deep learning for 6d pose estimation in the amazon picking challenge. In IEEE International Conference on Robotics and Automation (ICRA), 2017.

[10] C. Hernandez, M. Bharatheesha, W. Ko, H. Gaiser, J. Tan, K. van Deurzen, M. de Vries, B. Van Mil, J. van Egmond, R. Burger, et al. Team delft's robot winner of the amazon picking challenge 2016. In Robot World Cup, pages 613–624. Springer, 2016.

[11] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In IEEE Conf. on Computer Vision and Pattern Recognition, pages 3431–3440, 2015.

[12] M. Wang and W. Deng. Deep visual domain adaptation: A survey. arXiv preprint arXiv:1802.03601, 2018.

[13] G. Csurka. Domain adaptation for visual applications: A comprehensive survey. arXiv preprint arXiv:1702.05374, 2017.

[14] Y. Ganin, E. Ustinova, H. Ajakan, P. Germain, H. Larochelle, F. Laviolette, M. Marchand, and V. Lempitsky. Domain-adversarial training of neural networks. The Journal of Machine Learning Research, 17(1):2096–2030, 2016.

[15] Z. Pei, Z. Cao, M. Long, and J. Wang. Multi-adversarial domain adaptation. In AAAI Conference on Artificial Intelligence, 2018.

[16] J. Tremblay, A. Prakash, D. Acuna, M. Brophy, V. Jampani, C. Anil, T. To, E. Cameracci, S. Boochoon, and S. Birchfield. Training deep networks with synthetic data: Bridging the reality gap by domain randomization. arXiv preprint arXiv:1804.06516, 2018.

[17] M. Oquab, L. Bottou, I. Laptev, and J. Sivic. Is object localization for free?-weakly-supervised learning with convolutional neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 685–694, 2015.

[18] T. Durand, T. Mordan, N. Thome, and M. Cord. Wildcat: Weakly supervised learning of deep convnets for image classification, pointwise localization and segmentation. In IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2017). IEEE, 2017.

[19] A. Gaidon, Q. Wang, Y. Cabon, and E. Vig. Virtual worlds as proxy for multi-object tracking analysis. arXiv preprint arXiv:1605.06457, 2016.

[20] N. Mayer, E. Ilg, P. Hausser, P. Fischer, D. Cremers, A. Dosovitskiy, and T. Brox. A large dataset to train convolutional networks for disparity, optical flow, and scene flow estimation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 4040–4048, 2016.

[21] W. Qiu and A. Yuille. Unrealcv: Connecting computer vision to unreal engine. In European Conference on Computer Vision, pages 909–916. Springer, 2016.

[22] G. Ros, L. Sellart, J. Materzynska, D. Vazquez, and A. M. Lopez. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 3234–3243, 2016.

[23] M. Johnson-Roberson, C. Barto, R. Mehta, S. N. Sridhar, K. Rosaen, and R. Vasudevan. Driving in the matrix: Can virtual worlds replace human-generated annotations for real world tasks? In Robotics and Automation (ICRA), 2017 IEEE International Conference on, pages 746–753. IEEE, 2017.

[24] S. R. Richter, Z. Hayder, and V. Koltun. Playing for benchmarks. In International Conference on Computer Vision (ICCV), 2017.

[25] D. Dwibedi, I. Misra, and M. Hebert. Cut, paste and learn: Surprisingly easy synthesis for instance detection. ArXiv, 1(2):3, 2017.

[26] G. Georgakis, A. Mousavian, A. C. Berg, and J. Kosecka. Synthesizing training data for object detection in indoor scenes. arXiv preprint arXiv:1702.07836, 2017.

[27] S. Hinterstoisser, V. Lepetit, P. Wohlhart, and K. Konolige. On pre-trained image features and synthetic images for deep learning. arXiv preprint arXiv:1710.10710, 2017.

[28] Z.-H. Zhou. A brief introduction to weakly supervised learning. National Science Review, 2017.

[29] F. Michel, A. Kirillov, E. Brachmann, A. Krull, S. Gumhold, B. Savchynskyy, and C. Rother. Global hypothesis generation for 6d object pose estimation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 462–471, 2017.

[30] V. Narayanan and M. Likhachev. Discriminatively-guided deliberative perception for pose estimation of multiple 3d object instances. In Robotics: Science and Systems, 2016.

[31] C. Mitash, A. Boularias, and K. E. Bekris. Improving 6d pose estimation of objects in clutter via physics-aware monte carlo tree search. arXiv preprint arXiv:1710.08577, 2017.

[32] S. Hinterstoisser, S. Benhimane, V. Lepetit, P. Fua, and N. Navab. Simultaneous recognition and homography extraction of local patches with a simple linear classifier. In BMVC, pages 1–10, 2008.

[33] F. Yu, Y. Zhang, S. Song, A. Seff, and J. Xiao. LSUN: construction of a large-scale image dataset using deep learning with humans in the loop. CoRR, abs/1506.03365, 2015.

[34] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel. Domain randomization for transferring deep neural networks from simulation to the real world. In IROS, pages 23–30. IEEE, 2017.

[35] A. Oliver, A. Odena, C. Raffel, E. D. Cubuk, and I. J. Goodfellow. Realistic evaluation of deep semi-supervised learning algorithms. arXiv preprint arXiv:1804.09170, 2018.

[36] B. Drost, M. Ulrich, N. Navab, and S. Ilic. Model globally, match locally: Efficient and robust 3d object recognition. In Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on, pages 998–1005. Ieee, 2010.