Improving Out-of-distribution Distractor Handling through Data Augmentation

Lukas Flatz[1], Stefan Thalhammer[2], Timothy Patten[2] and Markus Vincze[2]

***Abstract*—Object detection is a necessary vision task for understanding hand-object interaction. It is therefore important that object detectors are robust to the occlusion induced by hands. A common approach to improve occlusion handling is random image augmentation. We substantiate the use of existing and available data sources as a cheap and efficient data augmentation method for handling the specific occlusion of human hands. Augmenting training images with available hand masks leads to a relative improvement of up to 13% compared to popular procedural methods and up to 19% compared to baseline data.**

I. Introduction

Robust object detection is a crucial requirement for various machine vision tasks. Visual object tracking systems require consistent and confident predictions, even if the object is partially occluded. Object detectors perform especially poorly when objects of interest are occluded by unknown objects, i.e., background objects that are not visible in the training set. We refer to these images as *out-of-distribution* samples and to the unknown objects as *distractors*.

It is desired to train on available datasets to avoid manual data creation and annotation. However, these datasets often do not cover use case related challenges, i.e, *out-of-distribution distractors*. Applying random image augmentations results in improved occlusion and illumination handling by increasing the training data variation [1], [2]. These procedural methods are easy to use yet might lack the sophistication to properly handle task-specific occluders. For the scenario of detecting the object during hand-object interaction, we therefore augment the training data with available samples of the expected distractors, i.e., human hands. Fig 1 shows the proposed data augmentation method. We show that augmenting images with inexpensive available hand distractors improves the out-of-distribution handling for two CNN-based detectors [3], [4] during hand-object interaction.

II. Hand Augmentation

This section discusses details of our proposed data augmentation method. We use RGB images as training data and refer to it as *original* data. The available augmentation sources are referred to as *augmentations* and the set of query images, featuring *out-of-distribution distractors*, is referred to as *target* data. Occluder-specific information is taken from the *augmentations* and are applied to *original* images to handle the expected distractors in the *target* data.

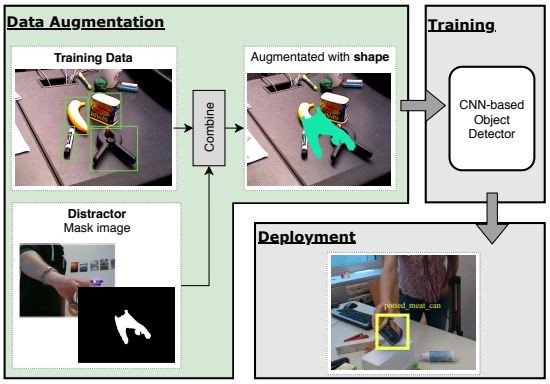

Fig. 1. Inexpensive *out-of-distribution distractor* handling by leveraging on available training data.

The only required information of the expected distractors is a segmentation mask, which comes at little cost. Existing datasets with known hand poses are exploited: Masks are obtained by directly thresholding depth images, as done in [5], or mesh models are used to render hands in varying hand poses [6]. The image locations to apply the *augmentations* are randomly sampled per image and the masks are "pasted" onto the *original* RGB images. The pixels of each mask in the *original* image are filled uniformly with a randomly sampled RGB-triplet. Finally, the annotations of the object (i.e., bounding boxes and visibility ratios) are adjusted accordingly. The augmented data is thus prepared to train object detectors for the specific use case.

III. Experiments

The proposed method is tested on samples of human-object interaction. A dataset not featuring human-object interaction is used for training, i.e., *original* data. This data is augmented with the expected distractors (*augmentations*) in order to handle variations in the *target* data. Results are compared using the F1 score averaged over all classes. For the experiments, we refer to our proposed augmentation as *shape*. *Random erase* augmentation as proposed by Zhong et al. [2] is used as a representative baseline for occlusion handling. Additionally we also provide results when directly applying hand shapes and appearance from the real-world training images (*real*) to the *original* data. Lastly, filling our hand shapes with the strategy of [2] is referred to as *random*.

[1]Lukas Flatz is with TU Wien, Vienna, Austria e1618873@student.tuwien.ac.at

[2]All authors are with the Faculty of Electrical Engineering and Information Technology, TU Wien, 1040 Vienna, Austria {thalhammer, patten, vincze}@acin.tuwien.ac.at

This work has been supported by the Austrian Research Promotion Agency in the program Production of the Future funded project MMAssist_II (FFG No. 858623), the Austrian Ministry for Transport, Innovation and Technology (bmvit) and the Austrian Science Fund (FWF) under grant agreement No. I3969-N30 (InDex)

A. Training Data

The YCB-video dataset [7] is used as the *original* training dataset. It features 92 videos with 133,827 frames of cluttered scenes with 21 YCB-objects [8] and annotated occlusion statistics. As *augmentations*, segmentation masks are taken from the HO3D [6] training dataset. These masks are used directly without rescaling. The HO3D dataset features 66,034 training images and 11,524 test images of YCB-objects during in-hand manipulation. The test set consists of 13 sequences of which 10 do not contain hand models featured in the training set. Consequently, these 10 sequences are used as *target* data in order to reason about *out-of-distribution distractor* handling. Experiments on YCB-video only consider the 9 YCB-objects featured in HO3D's test set.

B. Object Detectors

We employ RetinaNet [4] and YOLOv3 [3] as object detectors. RetinaNet uses Feature Pyramid Networks [9] on top of Resnet50 [10]. RetinaNet starts with 9 bounding box priors for each anchor location and scale. The upper left and lower right corners are predicted conditioned on the priors' width and height. YOLOv3 predicts only a single bounding box per grid cell and scale. The box offset from the upper left image corner and box width and height are predicted conditioned on the priors' width and height. Both detectors apply non-maximum suppression to their detections. YOLOv3 is faster but has less detection hypothesis per image compared to RetinaNet. Both detectors are pre-trained on the Imagenet dataset [11] and use a learning rate of 1e-5.

C. Object Detection Performance

For evaluation we set the Intersection over Union threshold for true positives to 0.5 and consider all detections with an object score above 0.5 as valid detections. The F1 score averaged over all 9 YCB-video objects is computed to compare the performance of different augmentation strategies.

Table I shows the results after training YOLOv3 on 192,000 images and RetinaNet on 286,000 images. The last row indicates detection performance when training on the official training data of HO3D. For *out-of-distribution distractor* handling (last column) using YOLOv3 (upper table) our method shows a relative performance increase of 8.3% compared to the *original* data and 3.7% compared to the baseline *Random erase* [2]. For RetinaNet (lower table) the relative performance increase is 18.9% and 13.2%, respectively. Interestingly, *shape* augmentation also outperforms *real* augmentations. Even though *real* is closer to real-world data. It has to be mentioned that a considerable number of hand masks from HO3D contain artifacts from YCB-v objects, which may cause confusion.

TABLE I
OVERALL F1 SCORE COMPARISON FOR DIFFERENT AUGMENTATIONS.

F1 Score (YOLOv3, 9 classes)	**Validation Data**		
	YCB-V	YCB-V + real	HO3D
Training Data			
YCB-V (original)	**0.904**	0.831	0.590
YCB-V + real	0.885	0.863	0.616
YCB-V + random	0.893	0.862	0.637
YCB-V + shape (ours)	0.900	**0.864**	**0.639**
YCB-V + Random erase [2]	0.864	0.806	0.616
HO3D	0.287	0.256	0.939

F1 Score (RetinaNet, 9 classes)	**Validation Data**		
	YCB-V	YCB-V + real	HO3D
Training Data			
YCB-V (original)	0.733	0.656	0.570
YCB-V + real	0.735	0.700	0.639
YCB-V + random	0.752	**0.781**	0.664
YCB-V + shape (ours)	0.758	0.666	**0.678**
YCB-V + Random erase [2]	**0.770**	0.717	0.599
HO3D	0.367	0.312	0.724

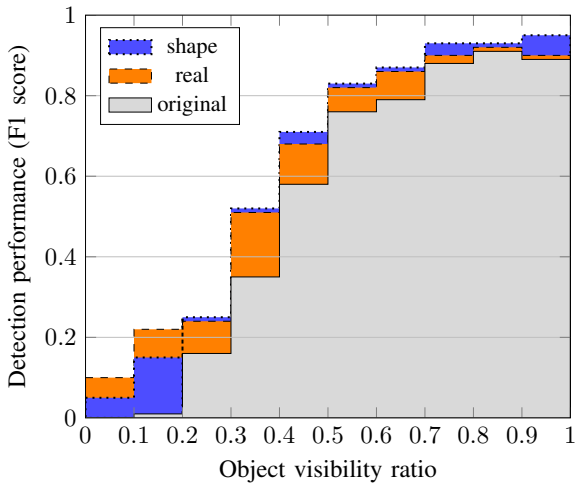

Fig. 2. Occlusion handling on the hand-augmented YCB-video test set using YOLOv3. The histogram shows the averaged F1 scores along 10% object visibility steps.

D. Hand Occlusion Handling

In the second experiment, we compare the occlusion handling capability of our proposed augmentation method compared to using real-world textures for training (*real*). HO3D's evaluation set does not provide hand segmentation masks thus cannot be used to analyze occlusion handling. Consequently, we use the augmentation of the YCB-video test images with hands from HO3D's training data set. Fig. 2 shows the results for YOLOv3. Compared to *original*, both augmentation methods show increased robustness against occlusion, especially for visibility ratios below 50%. The *shape* augmentation shows a slightly more robust occlusion handling compared to the hand augmentation for higher visibility ratios [1]. For general performance comparison on our augmented test data set refer to Table I (middle column).

IV. CONCLUSION

We proposed an effective and inexpensive data augmentation method for improving the performance of two common object detectors in order to deal with the occlusion of the human hand in hand-object interaction. Future work will consider finding effective methods to generate more realistic augmentation data, e.g., augmenting each image with realistic hand poses.

[1]Despite *real* performing better below 30% visibility, the results are not meaningful because of the low number of samples.

REFERENCES

[1] F. Mueller, D. Mehta, O. Sotnychenko, S. Sridhar, D. Casas, and C. Theobalt, "Real-time hand tracking under occlusion from an egocentric rgb-d sensor," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2017, pp. 1284–1293.

[2] Z. Zhong, L. Zheng, G. Kang, S. Li, and Y. Yang, "Random erasing data augmentation," *arXiv preprint arXiv:1708.04896*, 2017.

[3] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi, "You only look once: Unified, real-time obj/ect detection," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 779–788.

[4] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 2980–2988.

[5] S. R. Malireddi, F. Mueller, M. Oberweger, A. K. Bojja, V. Lepetit, C. Theobalt, and A. Tagliasacchi, "Handseg: A dataset for hand segmentation from depth images," *CoRR*, vol. abs/1711.05944, 2017. [Online]. Available: http://arxiv.org/abs/1711.05944

[6] S. Hampali, M. Rad, M. Oberweger, and V. Lepetit, "Honnotate: A method for 3d annotation of hand and object poses," 2020.

[7] 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.

[8] B. Calli, A. Walsman, A. Singh, S. Srinivasa, P. Abbeel, and A. M. Dollar, "Benchmarking in manipulation research: Using the yale-cmu-berkeley object and model set," *IEEE Robotics Automation Magazine*, vol. 22, no. 3, pp. 36–52, 2015.

[9] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie, "Feature pyramid networks for object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2117–2125.

[10] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[11] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A Large-Scale Hierarchical Image Database," in *CVPR09*, 2009.