Deep Learning Performance in the Presence of Significant Occlusions - An Intelligent Household Refrigerator Case

Gregor Koporec
Gorenje, d. o. o.
Partizanska 12, SI-3320 Velenje, Slovenia
gregor.koporec@gorenje.com

Janez Perš
Faculty of Electrical Engineering,
University of Ljubljana, Slovenia
janez.pers@fe.uni-lj.si

Abstract

Real-world environments, inhabited by people, still pose significant challenges to deep learning methods. Object occlusion is one of such problems. Humans deal with the occlusion in a complex way, by changing the viewpoint and using hands to manipulate the scene. However, not all robotic systems can do that due to cost or design constraints. The question we address in this paper is, how well modern object detection methods work on a model case of an intelligent household refrigerator, where numerous occlusions occur. To motivate our research, we actually performed a worldwide survey of refrigerator occupancy to realistically judge the extent of the problem, but the results could be generalized to any unstructured storage environment where people are in charge. The survey results enabled us to generate a dataset of photo-realistic renderings of a typical refrigerator interior, where the object identity, location, and the degree of the refrigerator occupancy are all readily available. Our results are represented as the Average Precision depending on a refrigerator occupancy for two well known deep models.

1. Introduction

Object detection is an important field in computer vision. The task is to detect objects which appear in individual images or image sequences. It is an important part of various robotic applications, such as autonomous driving (detecting obstacles or pedestrians), assistive technologies [19], and in many tasks that involve manipulating objects in the real world.

Recently, deep learning models have drastically improved the performance of the object detection [19]. Despite the success of deep learning, the methods still have problems with occluded objects. Occlusion is inevitable in real-world scenarios. In robotics, for instance, there is the Amazon Picking Challenge (APC) [6], where we want to automatically pick or stow objects in unstructured environments. In the APC picking task, a robot has to put objects from a partially filled shelf into a tote. The stowing task is the opposite of the picking task. The objects are stored mixed and partially occluding each other which makes it difficult to detect them [6].

Figure 1: Two examples of human-designed "order". Left: contents of the actual household refrigerator, supplying a 4-member family. Right: a marina, with the row of boats, neatly moored to a pier. Image is taken from the publicly available Marine Obstacle Detection Dataset (MODD) [17]. Heavy occlusion is visible in both images.

1.1. Human environment

The problem of occlusions is not as severe if the access to the environment is restricted to robots only – then, by necessity, all or most positions of objects in the environment are the consequence of robot picking, placing and stowing operations, which can be documented. However, the challenges arise, when robots need to operate *alongside humans*.

The human idea of *order* and *organization* is very different from the organization that would be best for an exclusively robotic environment. Human abilities to adapt their viewpoint, recognize objects, pick, manipulate and stow them are yet unsurpassed. Last but not least, the human

ability to adapt to different environments is staggering. Unfortunately, future robots will have to work in conjunction with humans, and it will be expected of them to work in an environment that is organized in a way that humans prefer.

This leads to the question, how effective are state-of-the-art methods in the environment, that is created with humans, not robots, in mind. Two such environments that are very different, but share important common characteristics, are shown in Figure 1.

1.2. Household refrigerator as the model of human-influenced environment

As can be seen from Figure 1, there exist environments, which have been constructed with order and organization in mind, but nevertheless, pose an extremely difficult challenge to computer vision algorithms. One of those challenging cases of object detection and recognition is occlusion.

We chose a household refrigerator ("fridge") as a model for the human-influenced environment for multiple reasons. For one, "intelligent" refrigerators or refrigerator add-ons are already being marketed, and sometimes the ability to recognize the contents inside is being promised. Secondly, a refrigerator is still not human environment as such, it is subject to technology and engineering constraints, but people still have (albeit limited) influence on its organization. Engineering constraints make the inside standardized to a certain degree, which makes our problem easier. Third, refrigerators are a common household item and household robotization will definitely have to include them in an active or passive form, that is, either as "intelligent refrigerator" or the object, the robot has to mechanically interact with to fetch certain food item.

The remaining questions are: is the refrigerator an appropriate model for this task? Is the refrigerator content from Figure 1 an exception or a rule? At which occupancy level are methods allowed to degrade? To answer that, we carried out a simple user study. 100 subjects from five English-speaking countries were shown photo-realistic images (Figure 2, top) of a shelf in the refrigerator at different degrees of occupancy. Participants were asked to select an image that resembles the appearance of their refrigerator shelves most closely. As one can see, 70 % filled refrigerator appears basically full. Note that occupancy was determined from the geometry of 3D CAD models of a refrigerator and 3D models of objects. Results are shown in the bottom part of Figure 2. We can see that for most real-world fridges out there, occlusion is a serious problem. Further details of the study can be found in the experimental design section.

1.3. Object detection in a refrigerator

It is obvious that object detection inside of a household refrigerator is one of the more challenging detection tasks. First, we have many intra-class and inter-class occlusions.

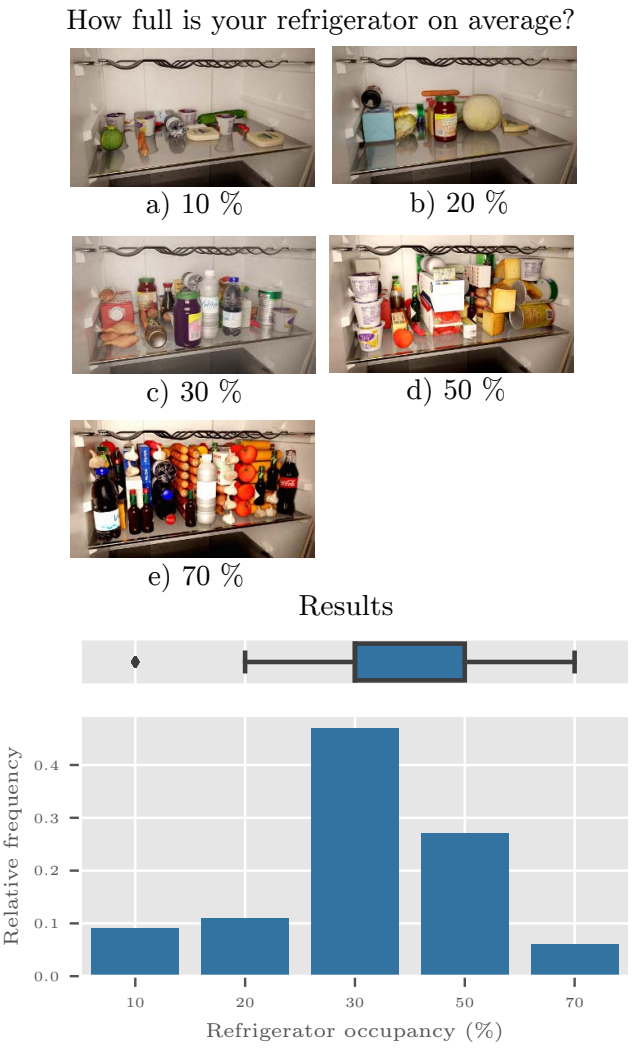

Figure 2: Simple user study to determine how full are fridges on average. Top: Human subjects were shown those images and they were asked to select the one that represents the average occupancy of their refrigerator. Bottom: results show the distribution of answers (bars and the box plot shown). The calculated average refrigerator occupancy was 30 %.

Terms were defined by Wang *et al*. [39], where objects are occluded by the objects from the same or different categories. Second, objects at the back are normally heavily occluded and almost impossible to detect. Third, objects in use (groceries) are non-rigid and therefore susceptible to deformations when fridges are filled up.

In such a challenging environment we assume the performance of the state-of-the-art detection algorithms will drop, and the amount of drop will depend on the occupancy of the refrigerator. The more full the refrigerator is, the bigger will be the performance drop, due to occlusions. So the main question which we want to address in this study (Fig-

ure 3) is *how well modern object detection methods work in unstructured storage environments?*

To perform that study using real-world images on state-of-the-art algorithms, it would require to manually fill the refrigerator hundreds or thousands of times, and somehow assess its occupancy and occlusion rate, in addition to labeling. Due to a large number of images needed by deep learning algorithms, this is an impossible task. To tackle this problem we generated synthetic images from CAD models, where we can precisely asses the occupancy and occlusion rate for each generated synthetic image.

There already exists some tests based on the rate of the occlusion [28, 16, 41, 42, 32, 27, 10], but they do not assess the performance of the latest state-of-the-art detection algorithms. In this work, we will test the performance of 2 state-of-the-art deep learning detection algorithms with COCO evaluation metrics [20].

Our contributions in this work are:

- The worldwide survey and statistical analysis of the importance of an AI (robotics, computer vision) problem that involves both algorithms and *human behaviour*.
- New large scale dataset for evaluation of occlusion handling. Dataset consists of synthetic images where object identity, location, and the degree of the refrigerator occupancy are all readily available.
- Evaluation of occlusion handling performance of state-of-the-art deep learning detection algorithms.

2. Related work

2.1. Training with synthetic images

Using synthetic images for training deep learning algorithms can be very attractive because eliminates expensive and time consuming manual annotation of images. But synthetic images are unable to fully reproduce the statistics of real-world data [14]. This is mainly due to the fact that some physics cannot be captured by rendering engines [36]. Because of that, we can assume algorithms trained on such images will have difficulties getting good results on real data. Nevertheless, some authors [26, 33] showed that one can still get decent results when using synthetic images that utilize domain-specific image statistics. They generated images by adding random background and textures from real-world images.

Authors [23, 30] proposed using *photo-realistic rendering*, where we reproduce the statistics of real-world data by controlling the lighting and camera properties. With such approach models trained on rendered data were good as those trained on real data. Movshovitz *et al*. [23] pointed out that models trained on rendered data could be even better as they need to adapt to the real domain which brings an additional level of difficulty. By supplementing real-world training data with photo-realistic images [23, 30] showed models outperformed those trained only in one domain.

However, photo-realistic rendering is hard and slow, which makes it an expensive task [37]. Dwibedi *et al*. [8] therefore proposed *a patch level realism*. They cut object instances and pasted them on random backgrounds. Their approach eliminated the requirement of scene geometry estimation but pixel artifacts between object instances and background changed object features. The problem was solved by forcing algorithms to ignore those artifacts.

Better approach than [8] was proposed by Tremblay *et al*. [37] and Tobin *et al*. [36]. They used a technique called *a domain randomization* where rendering parameters (lightning, textures) are randomized in non-realistic ways. The hypothesis is that enough variability in rendered data will help models generalize to the real world. The real world then appears to the model as just another variation [36].

Hinterstoisser *et al*. [14] argued that the domain randomization works only on simple objects and scenarios and is therefore not so useful. They proposed new effective training where we use the models pre-trained on real images and train only the last layers with synthetic images.

2.2. Occlusion datasets

There already exists a number of datasets tackling the occlusion problem [15, 16, 24, 13, 3, 38] but they are mainly too small and cannot be used to successfully train deep learning models.

CMU Grocery dataset (CMU10_3D) [15] has 620 images of 10 grocery items in a natural kitchen environment. The Items and the environment are very similar to our dataset. It is well suited for testing the domain adaptation of selected models but cannot be used for sufficient training.

Hsiao and Hebert [16] generated CMU Kitchen Occlusion Dataset (CMU_KO8). Dataset consists of 1600 images with 8 texture-less household items. Items have severe occlusions and are positioned in a kitchen environment.

A similar environment was done by Walas and Leonardis [38] in UoB Highly Occluded Object Challenge (UoB-HOOC). Challenge was used for detection and scale and pose estimation of objects in RGB-D scenes with 20 objects.

Pose estimation was also the main problem in ICCV2015 Occluded Object Challenge [13, 3]. 8 objects were positioned in a realistic setting of heavy occlusion.

CUHK Occlusion Dataset [24] is specifically used only for pedestrian occlusions. It was obtained by selecting the images from popular pedestrian datasets and contains 1063 images with occluded pedestrians.

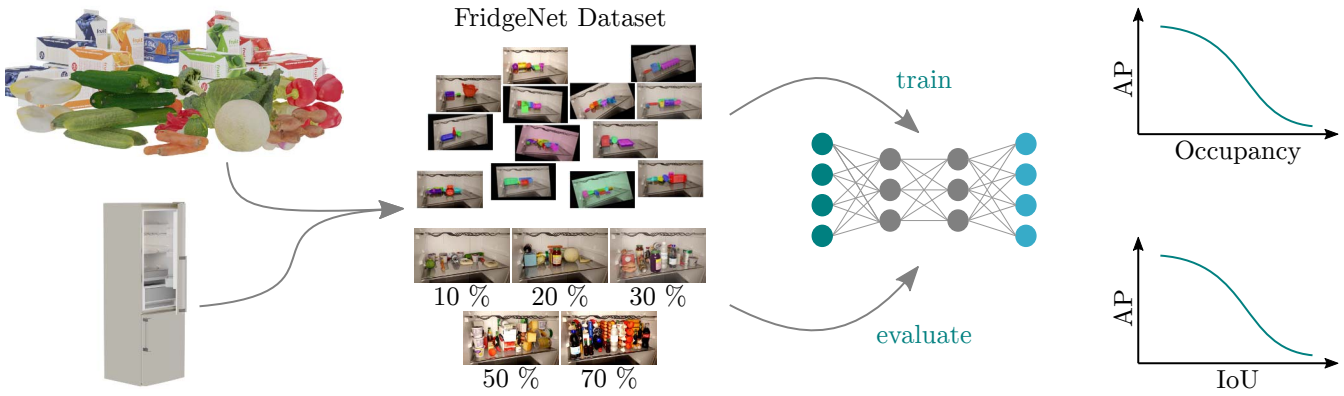

Figure 3: We build a new large scale FridgeNet dataset. The dataset is a COCO style [20] dataset and contains 95 000 synthetic images. Images contain 36 different categories of common grocery items on a glass shelf in a standard free-standing refrigerator. Two state-of-the-art deep learning detection algorithms were studied: MASK_RCNN [11] and YOLO9000 [29]. Models were pre-trained on COCO [20] in ImageNet [7] dataset respectively and fine-tuned on FridgeNet dataset. PASCAL VOC metric AP@50 and strict metric AP@75 [20] used as performance metrics. Metrics were analyzed on the refrigerator occupancy and IoU object occlusion.

2.3. Occlusion handling

Occlusion handling was extensively researched mainly in pedestrian detection where occlusion remains one of the hardest challenges [40]. A common approach to tackle the problem is to train part-based detectors [24, 22, 25, 35, 41, 42], where each model detects only part of a human body. Detections are then combined to localize occluded pedestrians.

In many cases, a detector is often confused since pedestrians have similar appearances [39]. Therefore Zhang *et al*. [40] and Wang *et al*. [39] proposed new loss functions to reduce false detections of overlapping pedestrians.

Lin Chu and Krzyżak [21] tested a hypothesis that deep belief network architectures perform better than convolutional neural networks when recognizing occluded objects. The hypothesis comes from the fact that DBNs can partially reconstruct the image which can aid in classification. They found that the architecture and training algorithm does not contribute to better recognition.

Researchers in [32] investigated the difference between recurrent and feedforward networks when recognizing partial occluded objects. They introduced two recognition tasks digit clutter and digit debris. In former task, multiple target digits occlude one another and in latter target, digits are occluded by digit fragments. They showed that recurrent neural networks outperform feedforward networks.

Models for detection occlusion patterns were proposed by [28]. They showed that occlusion patterns can be mined and can aid object detection.

Hsiao and Hebert [16] showed that a model of 3D interaction of objects can be used to represent an occlusion. With such a model additional training data is not needed.

Occlusions in semantic segmentation were researched by Chen *et al*. [4]. They proposed a top-down approach with an energy minimization problem to handle occluded regions. Occluded regions are fed into classifiers to obtain categories and likelihood maps. Meanwhile, examples are used in the shape predictor to obtain better shape estimation. All outputs are then used in an energy minimization problem to get better segmentation.

Occlusion handling performance was more systematically studied in [28, 22, 16, 41, 42]. Pepik *et al*. [28] studied the recall of deformable part models for 5 occlusion levels in intervals of 20 %. In [22] mean recall and miss rate were assessed on occlusion levels from 0 % to 50 %. Models in [16] were tested for occlusion handling in low 35 % and high $<$ 35 % occlusion levels. Authors in [41, 42] tested pedestrian detection miss rate on three occlusion levels: reasonable (1 %), partial (1 %–35 %), and heavy (36 %–80 %).

Pepik *et al*. [27] found out that AlexNet [18], GoogleNet [34] and VGG16 [31] are not invariant to appearance factors such as rotation, size, occlusion and truncation. For occlusion, adding training data didn't improve the results. They suggested architectural changes are needed to obtain improvements.

3D object recognition was evaluated in [10]. Authors implemented a special ConvNet for 3D object recognition and studied its performance over a 3D CAD model dataset. They simulated occlusions and noise with respect to RGB-D sensors and showed that the important factor for occlusion robustness are volumetric representations of 3D models.

3. FridgeNet Dataset

Current datasets are mainly too small to successfully train deep learning models. One could argue for the use of pre-trained models to tackle the problem, but still, those models need around 2000 images per category to fine tune them appropriately [1]. For example, the CMU_KO8 dataset [16] contains roughly 200 images per category, which is 10 times smaller than the recommended value. Therefore, we built a large scale FridgeNet dataset from synthetic images that can be used for deep learning algorithms. The dataset is currently not available to the general public, but we plan to release it at a later time.

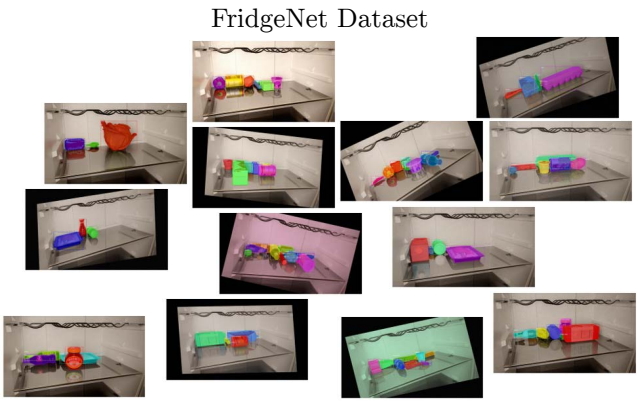

Figure 4: Sample images of the large scale FridgeNet dataset. An object identity, location as segmentation, and the degree of the refrigerator occupancy are readily available.

FridgeNet is a COCO style dataset [20] and contains 72 000 training samples, 17 000 validation samples, and 6000 test images. Object identity, pixelwise segmentation, bounding boxes and the degree of the refrigerator occupancy for each image are also included. Sample images are shown in Figure 4. There are 36 categories of common grocery items that can be found in a refrigerator. All samples are 1024×576 photo-realistic images rendered from 3D CAD models by Cycles renderer [2]. All images represent realistic occupancy of a glass shelf in a standard free-standing refrigerator with dimensions $60 \times 185 \times 64\,\mathrm{cm}$ (W×H×D).

3D CAD models except refrigerator were bought from online repositories. Refrigerator model has been kindly supplied by the home appliance manufacturer. It is a professional engineering CAD model used for production purposes. 36 *diverse* grocery models were carefully selected based on the following criteria. A category of a model must be a common grocery item. Common items are the ones that can be identified by most of the people like milk, cucumber or water bottle. Models had to have physically based rendering (PBR) materials and be compatible with Blender format. With PBR materials we assured best photo-realistic results.

Images were generated by placing grocery items on a glass shelf. To simplify the procedure, CAD models were considered to be rigid cuboids. Without this simplification, the algorithm to automatically place the objects in the refrigerator would be too complex and too slow for this task.

Also, we didn't vary the camera pose. Varying camera pose would simulate how humans help themselves in recognizing objects by changing the viewpoint. This would possibly also help the algorithms and therefore, evaluation of the algorithms' occlusion invariance would not be realistic. Algorithms would fail on datasets without additional assisting data.

We decided to put the objects on the single shelf, as the variation between the different shelves is too small to warrant the increase in complexity of the experiment, and on the other hand, adds very little to realism regarding occlusions.

Object rotations were constrained to prevent visual artifacts or unnatural positions such as bottles oriented upside down. Physics simulation of gravity was not used to save processing time. Other physics properties were still used to prevent collisions between the objects. To assure that the poses were diverse we randomly put objects on non-occupied surfaces of glass shelf and already positioned object cuboids to fill the free space in the refrigerator.

Additional occupancy parameter was used to indirectly influence the occlusion rate. All training and validation samples were rendered with occupancy of $10 \pm 1\,\%$. For testing data we used occupancies $10 \pm 1\,\%$, $20 \pm 1\,\%$, $30 \pm 1\,\%$, $50 \pm 1\,\%$, and $70 \pm 1\,\%$.

Rendered images were augmented to add an additional layer of realism and to randomize the domain. The training set was augmented by Gaussian blur, affine transformations (scale, rotation, and shear), changing brightness, contrast normalization, and additive Gaussian noise. Affine transformations were not used on testing images.

The reason why we use synthetic images and not real-world data is mainly because of the complexity of the dataset. With real-world images, one immediately runs into the problem of how to objectively measure the occupancy of the shelf. Enrolling a large group of human labelers could perhaps be used to group the images by occupancy, but would still provide a rather subjective result. Obtaining tens of thousands of images that show *realistic refrigerator interior* is another problem. Using web-crawling runs into a problem where there are far too few images that represent realistic occupancy of the refrigerator. Images that are available are mainly promotional images from home appliance makers, where image aesthetics, not realism, is the main objective. A better approach would be to photograph our own refrigerator and randomly fill it with groceries but the procedure would be impossibly time-consuming, should

we target the same number of images we have now, that is 90.000 total images.

As pointed out in Sec. 2.1 there are four main approaches to generate appropriate synthetic images (adding domain-specific image statistics, photo-realistic rendering, patch level realism, and domain randomization). Given our problem, the best solution is the use of photo-realistic rendering to get clean images, and then to augment them to additionally randomize the domain.

4. Experimental design

First, we performed a worldwide survey of refrigerator occupancy to estimate the degree of a problem the object detection methods face. Human subjects were recruited from UK, Ireland, USA, Canada, and Australia to minimize the effect of the language barriers and were shown photo-realistic images (Figure 2) of a shelf in the refrigerator, each with different occupancy (10 %, 20 %, 30 %, 50 % and 70 %). Subjects were asked to select an image that represents the normal occupancy of their refrigerator. The user study was done on 100 subjects. The one-sample Wilcoxon signed rank test [9] was used for statistical analysis of the study.

Then we tested the following deep learning detection algorithms: MASK_RCNN with ResNet-50-FPN backbone [12] and YOLO9000 [29]. Detection algorithms can be roughly sorted into two groups: region-based and single shot methods. In region based methods object proposals are made and then sent to classification stage. Single shot methods eliminate proposal generation by incorporating proposals and classification into a single network. MASK_RCNN is a state-of-the-art representative of the former group and YOLO9000 of the later.

To avoid any unexpected problems caused by different domains, we used pre-trained models and only fine-tuned them. The process was proposed by [14]. MASK_RCNN was pre-trained on the COCO dataset [20] and then fine-tuned the network heads on our dataset for 40 epochs. MASK_RCNN backbone wasn't additionally trained. YOLO9000 was pre-trained on ImageNet dataset [7]. The whole network was then fine-tuned on our dataset for 18 000 iterations or roughly 6 epochs. Note that YOLO9000 was trained on slightly adjusted FridgeNet dataset. Dataset format for YOLO9000 implementation from [1] is different from the COCO dataset format [20], therefore we generated a modified FridgeNet dataset specifically for YOLO framework. Modified FridgeNet contains roughly 6000 training samples per category (total of 175 622 train samples and 43 915 validation samples). The number of test samples stayed the same. Because the modified dataset was generated from the same population of rendered images, dataset statistics are the same as from FridgeNet with COCO format.

To evaluate the performance of modern object detection methods in unstructured storage environments, we used two different metrics. First, we determined PASCAL VOC metric AP@50 and strict metric AP@75 (defined in [20]) over selected occupancies. Occupancies for each image were determined from the rendering process by calculating volumes of models' cuboids. Second, the same AP metrics were calculated over the occlusion rate. The occlusion rate was determined as the standard Intersection over Union (IoU) of objects' bounding boxes. The standard IoU of a ground truth (gt) and detected object (dt) is defined by (1).

$$IoU(gt, dt) = \frac{area(intersect(gt, dt))}{area(union(gt, dt))} \tag{1}$$

In evaluating occlusion invariance of the algorithms average performance drop rate (ADR) at different IoU thresholds was used. ADR is defined by (2), where y is AP and x refrigerator occupancy or occlusion.

$$ADR = \frac{1}{N} \sum_{i=1}^{N} \frac{y_i - y_{i-1}}{x_i - x_{i-1}} \tag{2}$$

5. Results

Regarding the user study, the power analysis of 100 user answers resulted in 99.99 % confidence for selected high effect size 0.5 suggested by [5], and $\alpha = 0.05$. Answers are presented as the bar and box plot in the bottom part of Figure 2. Results indicate that refrigerator occupancy is mainly in 30 %–50 % range. To further investigate the occupancy we analyzed the null hypothesis that the average occupancy of a household refrigerator is 50 %. For the alternative, the average occupancy is less than 50 %. Hypotheses were tested by the one-sample Wilcoxon signed rank test for 0.01 significance level. With the results of $p = 3.318e-12$, we are confident that the average household refrigerator occupancy is 30 %.

The performance of selected deep models based on the refrigerator occupancy is shown in Figure 5. IoU threshold used is appended to the model name. The bold vertical line in Figure 5 represents the average refrigerator occupancy that was determined from the user study. This line represents an occupancy threshold for occlusion invariance.

One of the questions that have arisen in the analysis of the results is the maximum achievable rate, as objects inevitably become fully occluded and therefore invisible, as the occupancy increases.

AP for the occupancy threshold and below can be ideally 100 %. Algorithm performance should aim for this as well. Beyond the threshold, the degradation line was naively determined as a linear line passing through points $P1 = (30, 1)$ and $P2 = (100, 0)$. Point $P1$ represents ideal AP at occupancy threshold. Point $P2$ represents 0 precision

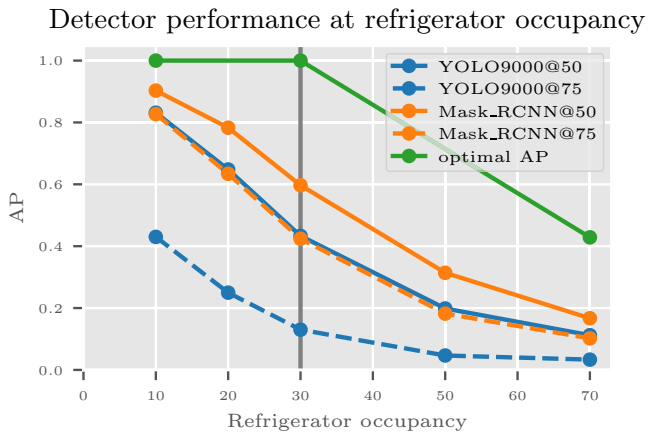

Figure 5: Detector performance at refrigerator occupancy. IoU threshold used is appended to the model name. The Vertical line represents average refrigerator occupancy. The green line represents the desired AP score. MASK_RCNN is more successful regarding the AP metrics. But its ADR is high.

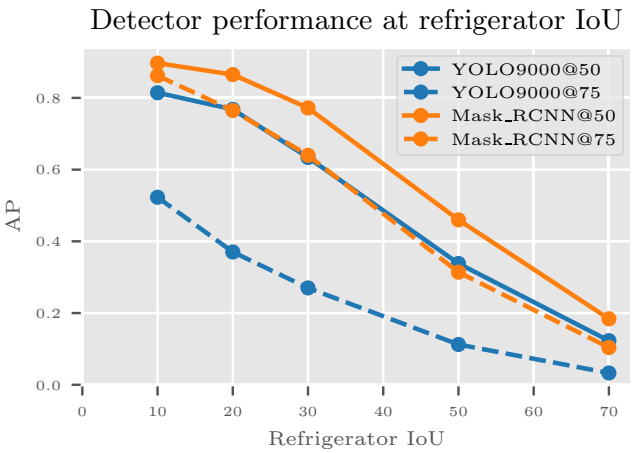

Figure 6: Detector perfomance at refrigerator occlusions. MASK_RCNN outperformed YOLO9000 detector in AP metrics.

at 100 % refrigerator occupancy. This calculation is shown in Figure 5 as a green line. Models should be invariant to occlusion for occupancies less than 30 % and allowed to degrade beyond that.

YOLO9000's performace for AP@50 metric dropped from 83.15 % to 43.33 % when refrigerator occupancy increased from 10 % to 30 %. This is nearly 40 % drop for 20 % of increased occupancy. Average drop rate at 50 % IoU threshold (ADR@50) was estimated to 1.40 and ADR@75 was 0.87.

Similarly MASK_RCNN's perfomance dropped from 90.3 % at 10 % occupancy to 59.7 % at 30 % occupancy (31 % drop). ADR@50 was 1.30 and ADR@75 was 1.41.

MASK_RCNN has proved to be more successful in AP. Results are better than those of YOLO9000 for every metric and occupancy. Moreover, we have lower performance drops depending on increasing the IoU threshold. Nevertheless, occupancy invariance for MASK_RCNN is quite low for both IoU thresholds. Best occupancy invariance was achieved by YOLO9000@75 (ADR@75 = 0.87).

Comparing the results depending on the refrigerator occupancy to the results depending on the actual occlusion (Figure 6), it can be seen that drops in performance are smaller for the same nominal change in occupancy/occlusion. This, of course, indicates that occlusion increases quickly (non linearly) with increasing occupancy. For AP@50 metric the performance of YOLO9000 dropped from 81.44 % at 10 % occlusion to 63.39 % at 30 % occlusion. Performance drop was around 18 %. ADR@50 was 1.09 and ADR@75 was 0.93.

For MASK_RCNN the perfomance dropped from 89.70 % at 10 % occlusion to 77.20 % at 30 % occlusion (12.5 % performance drop). ADR@50 was 1.05 and ADR@75 was 1.23. By average precision metrics, MASK_RCNN still outperformed YOLO9000 detector.

6. Conclusions

Object occlusion still poses significant challenges to deep learning methods. To test how well modern object detection methods work in unstructured storage environments we worked on a model case of an intelligent household refrigerator. The selected environment is one of the more challenging tasks in detection because of many intra-class and inter-class occlusions, heavily occluded objects at the back of the refrigerator and usage of deformable objects.

The survey of the refrigerator occupancy in human households was used to judge the extent of the problem in such an environment and motivate our research. Statistical analysis showed that the average household refrigerator occupancy is approximately 30 %, which already causes significant occlusion. Surprisingly, the average refrigerator occupancy was lower than expected. We anticipated higher values, almost 70 %, which is approximate amount shown in Figure 1. Refrigerator content from Figure 1 is therefore more of an exception than a rule. Average household refrigerator occupancy level was then selected as an occupancy threshold. To avoid frustrating the majority of the user, any models used in the real world should be invariant to occlusion at least for the occupancies below that threshold, unless the hardware mechanism to adjust the view is in place (e.g. moving robotic head).

To perform experiments, we build a new large scale "FridgeNet" dataset with 95 000 synthetic images, containing 36 different categories of common grocery items on a glass shelf in a standard free-standing refrigerator. To add variability, randomized augmentation was added to clean

rendered images. CAD models were considered to be rigid cuboids. Without this, the algorithm for object placement would be too complex. Unfortunately, this simplification didn't work well for rounded items as it produced some minor visual artifacts, where items were floating over the rounded objects. This could be overcome by also using the gravity in physics simulations. Another solution would be to use other simplified shapes besides cuboids such as spheres, capsules, cylinders, and cones. We can consider these visual artifacts as position offsets as they don't influence the method's performance. From rendering point of view images still result as photo-realistic.

Two state-of-the-art deep learning detection algorithms MASK_RCNN and YOLO9000 were pre-trained on COCO and ImageNet dataset respectively and fine-tuned on FridgeNet dataset. PASCAL VOC metric AP@50 and strict metric AP@75 were used as performance metrics. Metrics were obtained for different levels of refrigerator occupancy and IoU object occlusion.

Results confirmed our claims on performance drop when increasing the occupancy of a refrigerator. None of the evaluated models got near maximum achievable AP score. Nevertheless, the models did not show up any signs of occlusion invariance below occupancy threshold $30\,\%$. Minimum performance drop in lower occupancy area was as high as $31\,\%$ which is $11\,\%$ worse than a linear drop of $20\,\%$. ADR metrics for refrigerator occupancy were also indicating the models cannot handle occlusions well. Nevertheless, MASK_RCNN outperformed YOLO9000 in AP metrics by around $10\,\%$. This was somehow expected as region-based methods normally have better detection performances over single shot methods.

Similar results were obtained when evaluating the actual occlusion of items in the refrigerator. Drops in performance were smaller for both methods, but still they didn't achieve any occlusion invariance. MASK_RCNN similarly outperformed YOLO9000 which only further proves that MASK_RCNN can better cope with the problem of occlusion.

We conclude that using these algorithms in their present form in the real-world household fridges is almost certain to frustrate the majority of a refrigerator owners, and by extension, performance in other non-structured environments (e.g. human living quarters, unless they are tidied and organized in a robot-friendly manner) is questionable as well. Object detectors are also not ready to deal with human-designed highly-cluttered environments. Using additional data from different viewpoints could increase the performance as this simulates how people are helping themselves when recognizing items in such environments. But frequently this cannot be done and also this would not systematically solve the occlusion invariance problem. As other researchers pointed out, architectural changes are needed to overcome the problem.

Our future work will include improving this aspect of state-of-the-art detection algorithms.

References

[1] Alexey. Yolo-v3 and Yolo-v2 for Windows and Linux. https://github.com/AlexeyAB/darknet. Accessed: 2019-02-16, 2016. 5, 6

[2] Blender Online Community. Blender 2.79 Reference Manual, 2018. 5

[3] E. Brachmann, A. Krull, F. Michel, S. Gumhold, J. Shotton, and C. Rother. Learning 6d object pose estimation using 3d object coordinates. In *Computer Vision - ECCV 2014*, pages 536–551, 2014. 3

[4] Y.-T. Chen, X. Liu, and M.-H. Yang. Multi-instance Object Segmentation with Occlusion Handling. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3470–3478, 2015. 4

[5] J. Cohen. *Statistical Power Analysis for the Behavioral Sciences*. Lawrence Erlbaum Associates, Hillsdale, NJ, 2nd editio edition, 1988. 6

[6] C. H. Corbato, M. Bharatheesha, J. Van Egmond, J. Ju, and M. Wisse. Integrating Different Levels of Automation: Lessons From Winning the Amazon Robotics Challenge 2016. *IEEE Transactions on Industrial Informatics*, 14(11):4916–4926, 2018. 1

[7] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 248–255, 2009. 4, 6

[8] D. Dwibedi, I. Misra, and M. Hebert. Cut, Paste and Learn: Surprisingly Easy Synthesis for Instance Detection. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 1301–1310, 2017. 3

[9] F. Faul, E. Erdfelder, A.-G. Lang, and A. Buchner. G* Power 3: A flexible statistical power analysis program for the social, behavioral, and biomedical sciences. *Behavior Research Methods*, 39(2):175–191, 2007. 6

[10] A. Garcia-Garcia, J. Garcia-Rodriguez, S. Orts-Escolano, S. Oprea, F. Gomez-Donoso, and M. Cazorla. A study of the effect of noise and occlusion on the accuracy of convolutional neural networks applied to 3d object recognition. *Computer Vision and Image Understanding*, 164:124 – 134, 2017. Deep Learning for Computer Vision. 3, 4

[11] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask R-CNN. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 2961–2969, 2017. 4

[12] W. He, J. Wei, X. Chen, N. Carlini, and D. Song. Adversarial Example Defenses: Ensembles of Weak Defenses are not Strong. In *11th USENIX Workshop on Offensive Technologies (WOOT 17)*, 2017. 6

[13] S. Hinterstoisser, V. Lepetit, S. Ilic, S. Holzer, K. Konolige, G. Bradski, and N. Navab. Technical Demonstration on Model Based Training, Detection and Pose Estimation of Texture-Less 3D Objects in Heavily Cluttered Scenes. In *Computer Vision – ECCV 2012 Workshops*, pages 593–596, 2012. 3

[14] S. Hinterstoisser, V. Lepetit, P. Wohlhart, and K. Konolige. On Pre-Trained Image Features and Synthetic Images for Deep Learning. In *Computer Vision – ECCV 2018 Workshop*, pages 682—-697, 2018. 3, 6

[15] E. Hsiao, A. Collet, and M. Hebert. Making specific features less discriminative to improve point-based 3D object recognition. In *2010 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2653 – 2660, 2010. 3

[16] E. Hsiao and M. Hebert. Occlusion Reasoning for Object Detection under Arbitrary Viewpoint. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(9):1803–1815, 2014. 3, 4, 5

[17] M. Kristan, V. Sulić, S. Kovačič, and J. Perš. Fast Image-Based Obstacle Detection From Unmanned Surface Vehicles. *IEEE Transactions on Cybernetics*, 46(3):641–654, 2016. 1

[18] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. *Advances in neural information processing systems*, pages 1097–1105, 2012. 4

[19] M. Leo, A. Furnari, G. G. Medioni, M. Trivedi, and G. M. Farinella. Deep learning for assistive computer vision. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 0–0, 2018. 1

[20] T.-Y. Lin, M. Maire, S. Belongie, L. Bourdev, R. Girshick, J. Hays, P. Perona, D. Ramanan, C. L. Zitnick, and P. Dolí. Microsoft COCO: Common Objects in Context. In *Computer Vision – ECCV 2014*, pages 740–755, 2014. 3, 4, 5, 6

[21] J. Lin Chu and A. Krzyżak. The recognition of partially occluded objects with support vector machines, convolutional neural networks and deep belief networks. *Journal of Artificial Intelligence and Soft Computing Research*, 4(1):5–19, 2014. 4

[22] M. Mathias, R. Benenson, R. Timofte, and L. V. Gool. Handling Occlusions with Franken-classifiers. In *2013 IEEE International Conference on Computer Vision (ICCV)*, pages 1505–1512, 2013. 4

[23] Y. Movshovitz-Attias, T. Kanade, and Y. Sheikh. How useful is photo-realistic rendering for visual learning? In *Computer Vision – ECCV 2016 Workshops*, pages 202–217, 2016. 3

[24] W. Ouyang and X. Wang. A Discriminative Deep Model for Pedestrian Detection with Occlusion Handling. In *2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3258–3265, 2012. 3, 4

[25] W. Ouyang and X. Wang. Joint deep learning for pedestrian detection. In *2013 IEEE International Conference on Computer Vision (ICCV)*, pages 2056–2063, 2013. 4

[26] X. Peng, B. Sun, K. Ali, and K. Saenko. Learning Deep Object Detectors from 3D Models. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1278–1286, 2015. 3

[27] B. Pepik, R. Benenson, T. Ritschel, and B. Schiele. What is Holding Back Convnets for Detection? In *German Conference on Pattern Recognition*, pages 517–528, 2015. 3, 4

[28] B. Pepik, M. Stark, P. Gehler, and B. Schiele. Occlusion Patterns for Object Class Detection. In *2013 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3286–3293, 2013. 3, 4

[29] J. Redmon and A. Farhadi. YOLO9000: Better, Faster, Stronger. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7263–7271, 2017. 4, 6

[30] S. R. Richter, V. Vineet, S. Roth, and V. Koltun. Playing for Data: Ground Truth from Computer Games. In *Computer Vision – ECCV 2016*, pages 102–118, 2016. 3

[31] K. Simonyan and A. Zisserman. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *3rd International Conference on Learning Representations (ICLR)*, 2015. 4

[32] C. J. Spoerer, P. McClure, and N. Kriegeskorte. Recurrent convolutional neural networks: A better model of biological object recognition. *Frontiers in Psychology*, 8:1551, 2017. 3, 4

[33] B. Sun and K. Saenko. From Virtual to Reality: Fast Adaptation of Virtual Object Detectors to Real Domains. In *Proceedings of the British Machine Vision Conference*, pages 1–12, 2014. 3

[34] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus. Intriguing properties of neural networks. In *2nd International Conference on Learning Representations (ICLR)*, 2014. 4

[35] Y. Tian, P. Luo, X. Wang, and X. Tang. Deep Learning Strong Parts for Pedestrian Detection. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1904–1912, 2015. 4

[36] 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 *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 23–30, 2017. 3

[37] 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. In *2018 IEEE Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 1082–1090, 2018. 3

[38] K. Walas and A. Leonardis. UoB Highly Occluded Object Challenge (UoB-HOOC). http://www.cs.bham.ac.uk/%7ewalask/uob_hooc/. Accessed: 2019-02-17, 2016. 3

[39] X. Wang, T. Xiao, Y. Jiang, S. Shao, J. Sun, and C. Shen. Repulsion Loss: Detecting Pedestrians in a Crowd. In *2018 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7774–7783, 2018. 2, 4

[40] S. Zhang, L. Wen, X. Bian, Z. Lei, and S. Z. Li. Occlusion-aware R-CNN: Detecting Pedestrians in a Crowd. In *Computer Vision - ECCV 2018*, pages 637–653, 2018. 4

[41] C. Zhou and J. Yuan. Learning to Integrate Occlusion-specific Detectors for Heavily Occluded Pedestrian Detection. In *Asian Conference on Computer Vision*, pages 305–320, 2016. 3, 4

[42] C. Zhou and J. Yuan. Multi-label Learning of Part Detectors for Heavily Occluded Pedestrian Detection. Technical report, 2017. 3, 4