Occlusion-Robustness of Convolutional Neural Networks via Inverted Cutout

Matthias Körschens
Computer Vision Group
Friedrich Schiller University Jena
07737 Jena, Germany
Email: matthias.koerschens@uni-jena.de

Paul Bodesheim
Computer Vision Group
Friedrich Schiller University Jena
07737 Jena, Germany
Email: paul.bodesheim@uni-jena.de

Joachim Denzler
Computer Vision Group
Friedrich Schiller University Jena
07737 Jena, Germany
Email: joachim.denzler@uni-jena.de

***Abstract*—Convolutional Neural Networks (CNNs) are able to reliably classify objects in images if they are clearly visible and only slightly affected by small occlusions. However, heavy occlusions can strongly deteriorate the performance of CNNs, which is critical for tasks where correct identification is paramount. For many real-world applications, images are taken in unconstrained environments under suboptimal conditions, where occluded objects are inevitable. We propose a novel data augmentation method called Inverted Cutout, which can be used for training a CNN by showing only small patches of the images. Together with this augmentation method, we present several ways of making the network robust against occlusion. On the one hand, we utilize a spatial aggregation module without modifying the base network and on the other hand, we achieve occlusion-robustness with appropriate fine-tuning in conjunction with Inverted Cutout. In our experiments, we compare two different aggregation modules and two loss functions on the Occluded-Vehicles and Occluded-COCO-Vehicles datasets, showing that our approach outperforms existing state-of-the-art methods for object categorization under varying levels of occlusion.**

I. Introduction

Image classification plays a considerable role in many applications today and presents the basis of many other more complex tasks like segmentation and detection. While Convolutional Neural Networks (CNNs) can nowadays reliably categorize objects in images, there still exist special cases which have yet to be solved. One such case is classification under occlusion. This task is essential in several fields, such as automotive technology, where correct object recognition even under unfavorable conditions is crucial. Moreover, when applying classification algorithms in the wild, the objects in the images often are only partially visible due to circumstantial occlusion. Such cases show the ubiquity of occlusion and, therefore, the importance of occlusion-robustness of convolutional neural networks, which are still the most prevalent approach in object recognition.

Kortylewski *et al.* [1] showed that CNNs are not robust to occlusion and presented several approaches [2], [1] to tackle this problem. Their primary approach is a generative one, using clustering methods to detect object parts and using this information to generate a possible classification with Compositional Models. While effective, the approach of Kortylewski *et al.* relies on rather complex techniques requiring more intricate strategies to train. Moreover, extending the aforementioned approach to new problems is also hard due to this complexity.

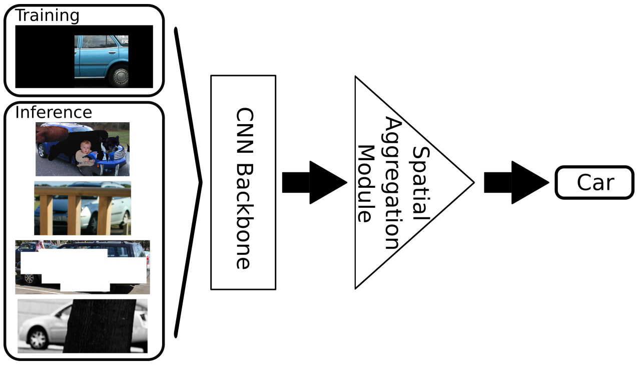

Fig. 1: The basic processing concept of our approach: a network backbone in conjunction with a spatial aggregation module is trained using only cut-out parts of the image to improve occlusion-robustness in the network.

In this work, we investigate a novel data augmentation method called Inverted Cutout, which enables training a network in standard discriminative fashion to achieve occlusion-robustness. The intuition behind this augmentation method is that the network merely sees a small part of the image at a time and relates this information to the class of the complete image. In contrast to the original Cutout [3] method, with which small parts of the image are masked out, Inverted Cutout leads to better generalization due to the stronger focus on smaller areas and their semantic meaning. We investigate this data augmentation method regarding two aspects: making the network robust against occlusion without the need of fine-tuning the base network and making it robust by including fine-tuning. Especially for the former, we investigate two feature aggregation methods. The first one utilizes global average pooling and is thus location-unaware; the second method, in contrast, is a convolutional layer aggregating features in a location-aware manner. The basic processing concept of our method is also shown in Figure 1. Our approach is evaluated on two datasets containing images of 12 classes of vehicles and furniture under partial occlusion.

Contributions: Our contributions in this work are as follows: we introduce a novel data augmentation method that is able to make networks robust against occlusion by standard discriminative classification training. We analyze this method in detail and shed light on possible extensions of the base network, i.e., aggregation modules, which enable occlusion-robustness without fine-tuning the network.

II. Related Work

Occlusion in Classification: Kortylewski *et al*. [2] developed a method, which utilizes a compositional model in combination with CNN features to predict the object class in occluded images. This approach has later been improved to be differentiable and, thus, end-to-end trainable [1]. Compositional models are generative models and have to be trained using multiple losses in conjunction with maximum-likelihood estimation. In contrast to their method, our approach is purely discriminative, much simpler to implement, directly combinable with standard CNNs, and does not require any additional loss terms other than a classification loss for training.

Moreover, several data augmentation techniques were proposed in the last years [3], [4], which introduced artificial occlusion in the training process. However, Kortylewski *et al*. [1] show that these augmentations alone only add limited robustness against partial occlusion to the network. We argue that this happens due to the network focusing more on the occluded areas and how to interpret them instead of focusing on the visible object parts and utilizing them to generate a conclusive classification decision.

Xiao *et al*. [5] presented a convolutional network architecture with an attention mechanism, which masks out occluded features in the network to improve its robustness against occlusion. However, Kortylewski *et al*. [1] also showed that this network does not perform well in the case of real occlusions.

In addition, there are approaches that investigate dropping out parts of feature maps or images based on high activations [6] or attention [7], [8], which work similar to Cutout by masking out specifically selected image regions. In contrast to the latter two, our data augmentation approach is model-free making it much cheaper to compute. Moreover, with Inverted Cutout we investigate paying attention only to a small patch of the image instead of the complete image besides a small cropped out patch as in the three aforementioned approaches and Cutout [3].

Occlusion in Segmentation: In addition to occlusion in classification, there also exist several approaches concerned with occlusion cases in instance segmentation, also referred to as amodal instance segmentation. Several recent approaches in this area do not require any amodal segmentation data for model training, but utilize standard instance segmentation annotations. Zhan *et al*. [9] apply several regularizing losses to two U-Net-like architectures [10], one of which receives an instance segmentation mask as input and outputs the respective amodal mask. The second network utilizes this mask to complete the partially occluded object. Ling *et al*. [11] propose a similar framework. However, they apply a variational autoencoder to generate a selection of possible amodal masks.

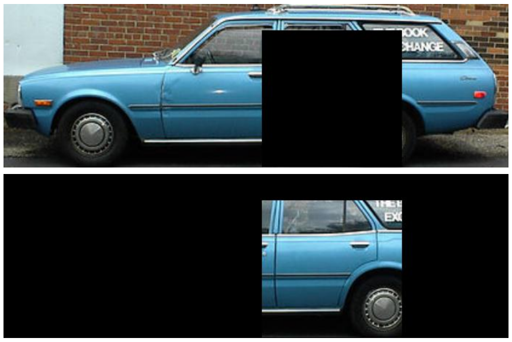

Fig. 2: An example of the difference between Cutout [3] (top) augmentation and our Inverted Cutout augmentation (bottom).

While the underlying task, i.e., object recognition under occlusion, is similar, the abovementioned methods necessarily require segmentation annotations as training data and are not easily extendable to classification. As segmentation annotations are not available in the case of occlusion during classification, we require another approach to tackle classification of occluded objects. Our proposed approach fills this gap, while being easier to implement and significantly simpler to train compared to other methods.

III. Our Approach

In this section, we introduce our approach, which comprises two parts. The first part is the Inverted Cutout augmentation, which can be used to train a CNN towards occlusion-robustness. The second part is the aggregation module, which can be used on top of the network backbone and can be used to improve a network's occlusion-robustness without fine-tuning the original network itself.

A. Inverted Cutout Augmentation

For training our model, we developed a novel data augmentation we refer to as Inverted Cutout or IC in short. As the name suggests, it is based on the popular Cutout [3] data augmentation method, which has been shown to improve generalization of CNNs. While with standard Cutout, a patch is being erased from the image, with Inverted Cutout, we do the opposite: we cut out a part of the image, which we keep for training while erasing the rest of the image, as seen in Figure 2. The likely reason for this is that, when using Cutout or training on the image as a whole, the model learns co-occurring object components in the remaining image parts and the classification decision depends on these co-occuring components during inference. This is problematic for components being spatially far away from each other, e.g., on the left and right border of the object, because real occlusions during inference are then likely to cover only one of these components making the learned co-occurence useless for classifying the object. With IC, the network is forced to predict the object class using only the small image patch that is left, leading to the network focusing primarily on the few isolated object parts being visible in the image to identify the class, which leads to a better generalization of the network. Regarding co-occuring object components, the network can then only exploit constellations of components that are nearby with respect to their spatial position in the image and which have a higher likelihood of being jointly visible in case of occlusion.

While training with Inverted Cutout, we use differing sizes of square patches that are cut out from the images. The size of the patches is sampled in a range of a predetermined minimum and maximum value for each image in the batch.

It should also be noted that our Inverted Cutout approach significantly differs from cropping and resizing the images, as with IC the scale and location of the cut out image parts are preserved. This is not the case for resized crops, which would, hence, drastically deteriorate network performance. Moreover, while not done in this work, Inverted Cutout can potentially be applied multiple times on the same image to generate multiple cutouts in a single training step. This would essentially create multiple "windows" showing small image regions in the same training step, which is not possible by cropping.

B. Spatial Aggregation Modules

As mentioned above, we investigate the usage of two different but simple spatial aggregation modules, whose purpose is to summarize the features extracted by a backbone in a single feature vector that is used for classification. The basic processing steps are shown in Figure 3. Both modules follow the depicted scheme and use an initial 1×1 convolutional layer with ReLU-nonlinearity, which we refer to as the transform layer. This transform layer is applied directly to the features extracted by the backbone and improves network performance especially when training only the aggregation module by increasing its modeling capacity.

The first aggregation module, later on referred to as $\mathrm{A}_{\mathrm{GAP}}$, utilizes the common global average pooling (GAP) method as used, for example, in the ResNet50 architecture [12], to aggregate the features from the transform layer into a single vector, which results in a location-unaware feature aggregation. This feature vector is then processed by a fully-connected layer followed by the classification layer.

In the second aggregation module, referred to as A_{FC}, the transform layer is followed by a single large convolutional layer, which has a kernel size equal to the size of its input feature map. Therefore, this layer can be seen as equivalent to applying a fully-connected layer on a flattened feature map. This type of layer results in a location-aware feature aggregation, in which the location of each input feature still plays a role in contrast to global average pooling. For the latter, each location is weighted equally, while for the large convolutional layer a different weighting can be learned at each input location. The output of this convolutional layer is also a single feature vector, which is used as an input to the subsequent classifier. We aim at comparing both modules in order to observe the differences between location-unaware and location-aware aggregation when applied to occluded objects.

IV. Experimental Results

Our experiments were done on the same two datasets utilized in [1]: the Occluded-Vehicles dataset based on Pascal3D+ [13] and the Occluded-COCO-Vehicles dataset, which was introduced in [1] and is based on MS-COCO [14]. In the following, we give brief introductions to both datasets, describe our experimental setup, and then report our results by comparing our approach with state-of-the-art methods.

A. Datasets

Pascal3D+: The Pascal3D+ Occluded-Vehicles dataset was introduced in [15] and later extended in [2]. The dataset features images of occluded objects from 12 classes with four differing occlusion levels and four kinds of artificial occlusion. The occlusion levels and their respective approximate occlusion percentages are L0 (0%), L1 (20-40%), L2 (40-60%), and L3 (60-80%). The types of occlusion featured in the dataset are: inserted objects (o), boxes of white color (w), boxes containing random noise (n), and boxes containing textures (t). Several example images are shown in Figure 4. It should be noted that the training set does not contain any occlusions, but contains all 12 classes, while the test set comprises merely 6 classes of vehicles under the aforementioned different types of occlusion.

MS-COCO: As the occlusions in the Pascal3D+-based Occluded-Vehicles dataset are added artificially, it is also necessary to evaluate the performance of occlusion-robustness models in a more realistic occlusion scenario. For this purpose, the Occluded-COCO-Vehicles dataset [1] contains images with natural occlusions, thus enabling an evaluation that is more related to a real-world application. The dataset comprises the same classes and occlusion levels as the previous one. Moreover, it contains 2036 training images without occlusion (L0), as well as 2036 test images without occlusion (L0), 768 with occlusion level L1, 306 with level L2, and 73 with level L3 occlusions. Two example images from this datasets can be found in Figure 5.

B. Setup

In our experiment, we utilize a VGG-16 network [16] pretrained on ImageNet [17] as a backbone similar to [1]. As also done in [1], features from this backbone are extracted by concatenating the outputs of the fourth and fifth convolutional block, referred to as block p4 and p5. We have chosen the same network architecture and the same feature representation as in [1] to allow for a fair comparison of the results.

During the training process, we initially train the aggregation module alone for a fixed number of epochs with the backbone being frozen. While we also evaluate the results from this training process, we afterwards fine-tune the complete network and evaluate the results of the networks a second time in order to observe the influence of fine-tuning the whole network. For training, we utilize the AdamW optimizer [18], [19] with a weight decay equal to 1e-4 as well as different learning rates and training durations based on the dataset and setup. The corresponding values are shown in Table I.

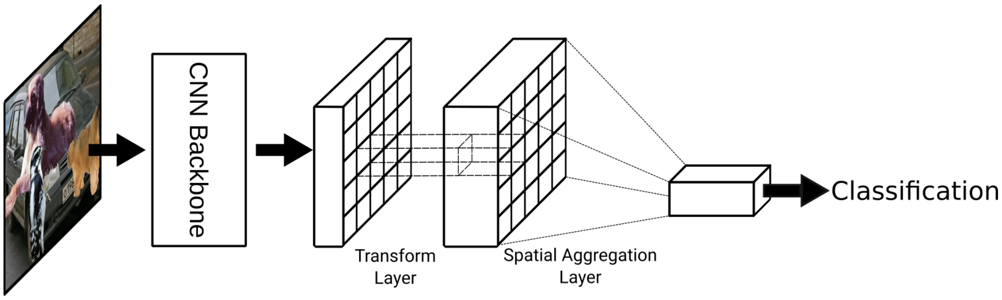

Fig. 3: An overview of our proposed architecture with an additional spatial aggregation module that is trained towards occlusion-robustness with our proposed Inverted Cutout data augmentation strategy. The aggregation module comprises two layers: a transform layer and an aggregation layer.

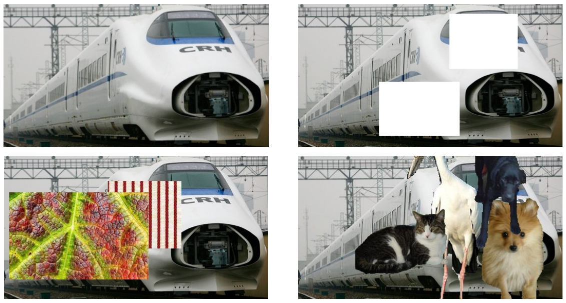

Fig. 4: Example images from the Occluded-Images Pascal3D+ dataset with several different levels and kinds of occlusion. From top-left to bottom-right: 0% occlusion (L0), 20-40% white box occlusion (L1), 40-60% texture occlusion (L2), 60-80% object occlusion (L3).

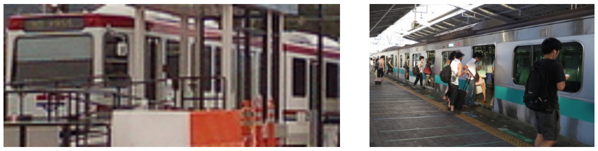

Fig. 5: Example images from the Occluded-COCO-Images dataset with different levels of occlusion. Left: 20-40% occlusion (L1); right: 40-60% occlusion (L2).

TABLE I: The hyperparameters used in the training process of the networks. * denotes hyperparameters used during fine-tuning of the network, A_{GAP} denotes the aggregation module with global average pooling, and A_{FC} the module with kernel size equal to the input feature map. CCE and BCE denote the Categorical (Softmax) and Binary Cross Entropy losses, respectively.

Dataset	Agg.	Loss	LR	EP	LR*	EP*
Pascal3D+	A_{GAP}	CCE	1e-3	90	1e-5	90
		BCE	1e-2	90	1e-4	90
	A_{FC}	CCE	1e-4	90	1e-4	90
		BCE	1e-4	90	1e-5	90
MSCOCO	A_{GAP}	CCE	1e-2	180	1e-4	90
		BCE	1e-2	180	1e-4	90
	A_{FC}	CCE	1e-4	180	1e-4	90
		BCE	1e-4	180	1e-4	90

For our Inverted Cutout augmentation, we first resize images to 224×224 pixels, aggregate them to batches of size 24, then sample the cutouts for each image as squares with side lengths of at least 16 and at most 128 pixels and apply them to each image in the batch. Independent of the setup, the learning rate is decreased by a factor of 10 after $\frac{2}{3}$ and $\frac{8}{9}$ of the total number of epochs, respectively. This is done during training of the aggregation modules as well as during fine-tuning.

The transform layer, which receives the output of the backbone as input, has a depth of 512, and the following aggregation layer generates 256 features. Aside from our IC augmentations, we only use horizontal flipping as data augmentation. All experiments below are averaged over four repetitions.

C. *Results*

In this section, we analyze the results of our experiments. As mentioned above, we evaluate the results after training our new module alone and after fine-tuning it in conjunction with the rest of the network on the respective dataset. We compare the performance of our IC method with the identical setups that do not employ this augmentation method. Moreover, we compare our results with those of previous methods as shown in [1], including the state-of-the-art method CompNet-Multi proposed by Kortylewski *et al.* [1]. We also investigate the difference between training with the standard Softmax Categorical Cross Entropy loss (CCE loss) and with the Binary Cross Entropy loss (BCE loss) utilizing the sigmoid activation, which is usually used in binary classification or multi-label classification. It should be noted that during training we only utilize images without occlusions, which are merely augmented with artificial occlusions when using IC. Hence, the network never sees any realistic occlusions during training.

Pascal3D+: The results for the Pascal3D+ Occluded Vehicles dataset can be seen in Table II. First and foremost, we can see a significant improvement in the accuracy when using Inverted Cutout compared to the ablations without the new augmentation. Depending on the setup, the results improve by at least 10.5% up to 21.3%. This shows that our novel augmentation method can be highly beneficial for classifying occluded objects. Moreover, we see that the fully-

TABLE II: The accuracies (in %) of our method in comparison with previously introduced methods on the Pascal3D+ Occluded-Vehicles dataset. The values of methods used for comparison have been taken from [1]. * marks the results received after fine-tuning the whole network. The occlusion types are: w - white box occlusion, n - noise box occlusion, t - texture occlusion, o - occlusion by segmented objects. CCE and BCE denote the Categorical (Softmax) and Binary Cross Entropy loss, respectively. A_{GAP} and A_{FC} denote the aggregation module with global average pooling and the one utilizing a convolutional layer, respectively.

Occ. Area	**L0: 0%**	**L1: 20-40%**				**L2: 40-60%**				**L3: 60-80%**				Mean
Occ. Type	-	w	n	t	o	w	n	t	o	w	n	t	o	
VGG [16]	99.2	96.9	97.0	96.5	93.8	92.0	90.3	89.9	79.6	67.9	62.1	59.5	62.2	83.6
CoD [2]	92.1	92.7	92.3	91.7	92.3	87.4	89.5	88.7	90.6	70.2	80.3	76.9	87.1	87.1
VGG+CoD [2]	98.3	96.8	95.9	96.2	94.4	91.2	91.8	91.3	91.4	71.6	80.7	77.3	87.2	89.5
TDAPNet [5]	99.3	98.4	98.6	98.5	97.4	96.1	97.5	96.6	91.6	82.1	88.1	82.7	79.8	92.8
CompNet-Multi [1]	99.3	98.6	98.6	98.8	97.9	98.4	98.4	97.8	94.6	91.7	90.7	**86.7**	88.4	95.4
A_{GAP}+CCE	99.5	97.5	97.4	97.1	92.9	90.3	88.2	88.3	68.7	58.2	47.2	46.6	47.1	78.4
A_{GAP}+CCE*	99.5	97.6	97.5	97.3	93.1	90.5	88.9	88.9	69.2	59.3	48.5	47.3	47.4	78.9
A_{FC}+CCE	99.7	97.9	98.0	98.0	95.2	93.2	92.7	91.4	80.4	61.8	56.2	51.9	60.8	82.9
A_{FC}+CCE*	99.7	98.1	98.2	98.1	95.3	93.3	93.1	91.6	80.4	62.0	56.8	52.4	61.1	83.1
A_{GAP}+BCE	98.5	93.7	92.7	92.6	84.9	83.2	80.4	79.0	59.9	49.3	43.4	41.2	41.8	72.4
A_{GAP}+BCE*	99.2	96.8	97.1	97.0	94.5	90.0	91.0	90.6	83.2	62.9	64.6	61.4	68.4	84.3
A_{FC}+BCE	**99.8**	98.7	98.5	98.7	97.0	95.4	94.7	94.1	83.6	66.4	60.9	59.5	61.4	85.3
A_{FC}+BCE*	99.7	98.7	98.6	98.7	97.0	95.3	94.8	94.1	83.8	66.3	61.1	59.3	61.8	85.3
A_{GAP}+CCE+IC	99.5	99.2	99.1	98.9	96.8	99.0	97.7	96.3	90.8	95.2	86.6	74.4	78.0	93.2
A_{GAP}+CCE+IC*	99.7	99.4	99.3	99.0	97.8	99.2	98.3	96.9	92.8	96.4	87.0	75.4	82.3	94.1
A_{FC}+CCE+IC	99.5	99.5	99.2	99.3	97.9	99.2	98.2	97.5	93.7	96.8	87.8	78.9	85.1	94.8
A_{FC}+CCE+IC*	99.6	99.6	99.2	**99.4**	98.6	99.3	97.9	97.5	95.7	95.9	83.0	80.4	88.9	95.0
A_{GAP}+BCE+IC	99.1	98.9	98.3	98.3	97.4	98.0	96.4	96.7	93.1	92.8	83.8	80.5	84.8	93.7
A_{GAP}+BCE+IC*	99.7	99.6	**99.4**	**99.4**	**98.9**	99.2	98.4	98.1	**96.9**	95.8	87.4	85.4	**92.1**	96.2
A_{FC}+BCE+IC	99.6	99.6	99.3	99.3	98.7	99.4	98.3	98.1	95.4	97.0	90.5	82.8	87.4	95.8
A_{FC}+BCE+IC*	99.7	**99.7**	**99.4**	**99.4**	**98.9**	**99.6**	**98.8**	**98.5**	96.2	**97.9**	**90.8**	83.9	89.4	**96.3**

convolutional aggregation module (A_{FC}) performs better in all cases. The reason for this is that the network is also able to consider the location of the features, which can also be helpful for identification. This is especially true for occlusion cases, in which multiple parts of the objects are occluded and the network can nevertheless use the visible features and their spatial positions to assign the correct object class to the image. However, this is not possible in the case of global average pooling, as the information of the location is lost due to the pooling process. This advantage of the convolutional aggregation module can be seen in the results with and without Inverted Cutout, and it therefore seems to be advantageous on every occasion. Interestingly, while using the Softmax Categorical Cross Entropy loss works well, we found that the Binary Cross Entropy loss actually works better in most cases and improves the results by several percentage points even in scenarios with more complex occlusions.

When looking at the difference between the runs during which we only train the network head and the ones with a fully fine-tuned network, we note that the improvement is greater for the aggregation module with global average pooling than for the fully-convolutional one. This also suggests that the information about the feature locations plays a vital role in identification under occlusion. The spatial information can only be included by training with the fully-convolutional aggregation block or by fine-tuning the complete network. This, in turn, results in a generally better performance of the fully-convolutional aggregation block even during training of the block alone with only minor improvements in case of additional network fine-tuning. At the same time, the pooling module can only utilize spatial information by fine-tuning the backbone, resulting in a more significant jump in performance afterward.

In comparison to previous methods, we observe that independent of the aggregation module, all approaches using our proposed IC augmentation outperform the previous approaches except for CompNet-Multi [1]. The top results received by training the network using Softmax Cross Entropy, i.e., an average accuracy of 94.8% and 95.0% before and after fine-tuning the network with fully-convolutional module, perform comparably to CompNet-Multi [1]. However, when using Binary Cross Entropy loss during training, the same setup outperforms CompNet-Multi by 0.4% and 0.9% for a fixed backbone and a fine-tuned backbone, respectively. It should also be noted that the fine-tuned network with GAP also outperforms CompNet-Multi by 0.8%.

MS-COCO: The results of our experiments on the Occluded-COCO-Vehicles dataset are shown in Table III. Overall, on most occasions, the results are similar to those on the previous dataset. This includes better results using the Binary Cross Entropy loss compared to the Softmax Categorical Cross Entropy loss, as well as the advantage of the fully-convolutional aggregation block over the one with GAP. However, A_{FC} outperforms A_{GAP} after training only the aggregation blocks, whereas the situation is vice versa after fine-tuning the complete network. This is likely caused by a

TABLE III: Results on the Occluded-COCO-Vehicles dataset, comparing our approaches with previously introduced methods based on classification accuracy (in %). The values of methods used for comparison have been taken from [1]. * marks the results received after fine-tuning the network. CCE and BCE denote the Categorical (Softmax) and Binary Cross Entropy losses, respectively. A_{GAP} and A_{FC} denote the aggregation module with global average pooling and with a convolutional layer, respectively.

Train Data	**MS-COCO**				
Occ. Area	**L0**	**L1**	**L2**	**L3**	**Avg**
VGG [16]	99.1	88.7	78.8	63.0	82.4
VGG [16] + Cutout	99.3	90.9	87.5	75.3	88.3
TDAPNet [5]	99.4	88.8	87.9	69.9	86.5
CompNet-Multi [1]	99.4	95.3	90.9	86.3	93.0
A_{GAP}+CCE	99.2	85.3	80.0	68.8	83.3
A_{GAP}+CCE*	99.3	85.3	80.2	68.8	83.4
A_{FC}+CCE	99.6	89.9	83.7	70.2	85.9
A_{FC}+CCE*	99.6	90.2	84.2	72.3	86.6
A_{GAP}+BCE	99.3	86.3	78.3	69.5	83.4
A_{GAP}+BCE*	99.3	86.4	78.3	69.5	83.4
A_{FC}+BCE	**99.7**	90.6	84.9	74.0	87.3
A_{FC}+BCE*	**99.7**	90.7	85.5	74.3	87.5
A_{GAP}+CCE+IC	99.1	90.6	87.2	87.0	91.0
A_{GAP}+CCE+IC*	99.5	94.5	90.8	89.4	93.5
A_{FC}+CCE+IC	99.4	94.2	90.0	82.9	91.6
A_{FC}+CCE+IC*	99.4	94.7	90.8	84.2	92.3
A_{GAP}+BCE+IC	99.1	89.4	88.0	86.0	90.6
A_{GAP}+BCE+IC*	99.4	94.6	91.6	**92.8**	**94.6**
A_{FC}+BCE+IC	99.4	94.1	92.0	86.3	93.0
A_{FC}+BCE+IC*	99.5	**95.4**	**92.1**	88.7	93.9

different distribution of objects in the images, as the Occluded-COCO-Vehicles dataset contains objects in natural occlusion scenarios. Hence, the objects might more often be off-center. At the same time, the fully-convolutional aggregation block in our case favors the same object parts being at the same locations in the image and this property is observed less common in this dataset. Therefore, the greater focus on the locations might be detrimental in this scenario. Nevertheless, CompNet-Multi, as well as all other previous methods, are outperformed by the fine-tuned networks trained with our IC augmentation method, and comparable performances are achieved without fine-tuning. Furthermore, it should be noted that we also include the experiments done with the VGG architecture [16] and standard Cutout [3] as performed by Kortylewski *et al.* [1]. However, we find that the performance gain with standard Cutout is far behind our new IC method.

To summarize, our experiments show that we can also utilize a rather simple data augmentation technique, namely Inverted Cutout, to make pre-trained networks robust against occlusion and achieve new state-of-the-art results on both evaluation datasets. It also drastically outperforms the classical Cutout [3] method, which is shown in an ablation study in the supplementary material. Moreover, ablations for different IC parameters as well as ones for changes in the backbone architecture can be found in the supplementary material as well.

D. Technical Comparison with State-of-the-Art

As we mentioned above, the state-of-the-art approach for classifying occluded objects, called CompNet-Multi by Kortylewski *et al.*[1], is a generative approach utilizing Compositional Neural Networks. As such, it has to be trained using a maximum-likelihood estimation scheme with a four-component loss function, as shown in [1]. In addition to the weights for the individual loss terms, the method also introduces further hyperparameters (like the number of mixture components), which have to be optimized. All these aspects make the method quite complex and possibly hard to apply to new problems beyond the scope of its original application. In contrast, our proposed method only utilizes a new augmentation function, called Inverted Cutout, and simple aggregation modules that can easily be applied to any network. Hence, aside from the improvements regarding the quantitative evaluation, our method is also more straightforward and can easily be applied in new domains. Furthermore, by investigating an aggregation block that summarizes features in a fully-convolutional way without the usage of global average pooling, as is usually done in segmentation tasks, we assume that our method can also directly be applied in segmentation-like tasks such as amodal segmentation [20] without extensive modifications.

V. Conclusion

In this paper, we have introduced a novel data augmentation method called Inverted Cutout (IC), which can be used to make a pre-trained network robust against occlusion. In our experiments, we have shown that the knowledge learned by using IC can be applied in scenarios with different occlusion types, including simple white block occlusion, texture and noise occlusion, occlusion by inserted objects, as well as realistic occlusion. Moreover, we have compared two different feature aggregation blocks and have shown that it is not even necessary to fine-tune the entire network to receive good performance for classifying occluded objects, but the right choice of the feature aggregation method can already introduce high levels of occlusion-robustness into the network. While a limitation of our approach is the training on tiny objects, as they are likely often not contained at all in the IC window, our experiments show that IC works well on normal-sized to full-frame objects as contained in the datasets used. By applying such aggregation blocks that are optimized by training with our proposed IC augmentation method which optionally also allows for fine-tuning the entire network, our methods performed comparably or better than previous state-of-the-art approaches, while at the same time being much more straightforward, easier-to-implement and easier-to-train compared to the competing approaches.

Acknowledgment

Matthias Körschens thanks the Carl Zeiss Foundation for the financial support.

References

[1] A. Kortylewski, J. He, Q. Liu, and A. L. Yuille, "Compositional convolutional neural networks: A deep architecture with innate robustness to partial occlusion," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 8940–8949.

[2] A. Kortylewski, Q. Liu, H. Wang, Z. Zhang, and A. Yuille, "Combining compositional models and deep networks for robust object classification under occlusion," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2020, pp. 1333–1341.

[3] T. DeVries and G. W. Taylor, "Improved regularization of convolutional neural networks with cutout," *arXiv preprint arXiv:1708.04552*, 2017.

[4] S. Yun, D. Han, S. J. Oh, S. Chun, J. Choe, and Y. Yoo, "Cutmix: Regularization strategy to train strong classifiers with localizable features," in *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2019, pp. 6023–6032.

[5] M. Xiao, A. Kortylewski, R. Wu, S. Qiao, W. Shen, and A. Yuille, "Tdapnet: Prototype network with recurrent top-down attention for robust object classification under partial occlusion," *arXiv preprint arXiv:1909.03879*, 2019.

[6] S. Park and N. Kwak, "Analysis on the dropout effect in convolutional neural networks," in *Asian conference on computer vision*. Springer, 2016, pp. 189–204.

[7] M. Ghorbel, S. Ammar, Y. Kessentini, and M. Jmaiel, "Masking for better discovery: Weakly supervised complementary body regions mining for person re-identification," *Expert Systems with Applications*, vol. 197, p. 116636, 2022.

[8] T. Hu, H. Qi, Q. Huang, and Y. Lu, "See better before looking closer: Weakly supervised data augmentation network for fine-grained visual classification," *arXiv preprint arXiv:1901.09891*, 2019.

[9] X. Zhan, X. Pan, B. Dai, Z. Liu, D. Lin, and C. C. Loy, "Self-supervised scene de-occlusion," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020, pp. 3784–3792.

[10] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical Image Computing and Computer-assisted Intervention*. Springer, 2015, pp. 234–241.

[11] H. Ling, D. Acuna, K. Kreis, S. W. Kim, and S. Fidler, "Variational amodal object completion," *Advances in Neural Information Processing Systems*, vol. 33, 2020.

[12] 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 (CVPR)*, 2016, pp. 770–778.

[13] Y. Xiang, R. Mottaghi, and S. Savarese, "Beyond pascal: A benchmark for 3d object detection in the wild," in *IEEE winter conference on applications of computer vision*. IEEE, 2014, pp. 75–82.

[14] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *European conference on computer vision*. Springer, 2014, pp. 740–755.

[15] J. Wang, Z. Zhang, C. Xie, V. Premachandran, and A. Yuille, "Unsupervised learning of object semantic parts from internal states of cnns by population encoding," *arXiv preprint arXiv:1511.06855*, 2015.

[16] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[17] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein *et al.*, "Imagenet large scale visual recognition challenge," *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[18] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," *arXiv preprint arXiv:1711.05101*, 2017.

[19] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[20] K. Li and J. Malik, "Amodal instance segmentation," in *Proceedings of the European Conference on Computer Vision (ECCV)*. Springer, 2016, pp. 677–693.