Not Using the Car to See the Sidewalk — Quantifying and Controlling the Effects of Context in Classification and Segmentation

Rakshith Shetty[1] Bernt Schiele[1] Mario Fritz[2]
[1]Max Planck Institute for Informatics [2]CISPA Helmholtz Center for Information Security
Saarland Informatics Campus, Germany
[1]firstname.lastname@mpi-inf.mpg.de [2]lastname@cispa.saarland

Abstract

Importance of visual context in scene understanding tasks is well recognized in the computer vision community. However, to what extent the computer vision models are dependent on the context to make their predictions is unclear. A model overly relying on context will fail when encountering objects in different contexts than in training data and hence it is important to identify these dependencies before we can deploy the models in the real-world. We propose a method to quantify the sensitivity of black-box vision models to visual context by editing images to remove selected objects and measuring the response of the target models. We apply this methodology on two tasks, image classification and semantic segmentation, and discover undesirable dependency between objects and context, for example that "sidewalk" segmentation is very sensitive to the presence of "cars" in the image. We propose an object removal based data augmentation solution to mitigate this dependency and increase the robustness of classification and segmentation models to contextual variations. Our experiments show that the proposed data augmentation helps these models improve the performance in out-of-context scenarios, while preserving the performance on regular data.

1. Introduction

Visual context of an object in an image is an important source of information for scene understanding tasks in both human and computer vision [22, 15]. Contextual cues such as presence of frequently co-occurring objects can help resolve ambiguities between visually similar classes and improve performance in various vision tasks including object detection [13, 3] and segmentation [25]. However, objects can also appear in previously unseen context or be absent from a very typical context. For example, we might find a keyboard on a desk without a monitor (object-without-context), or find a monitor without a keyboard (context-without-object). While humans can handle both these atypical scenarios gracefully, computer vision models often fail by ignoring the visual evidence for the object in object-without-context case or hallucinating objects which are not actually present in the image in context-without-object case. For example, in our experiments we find that *keyboard* is often not recognized without a nearby *monitor*, and semantic segmentation of roads suffers without *cars* (see Figure 1). While context can be an important cue, this kind of too heavy or even pathological dependency on contextual signals is undesirable, and it is important to systematically identify and ideally fix such cases. In this work, we analyze and quantify the effect contextual information on two tasks, multi-label classification and semantic segmentation.

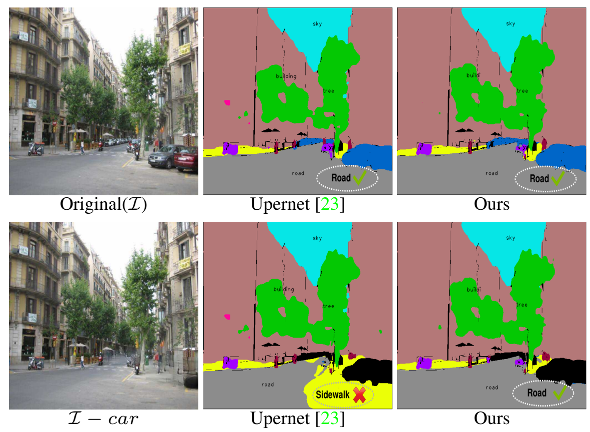

Figure 1: An example of the sensitivity of *road* and *sidewalk* segmentation to the context object *car*. Removing *car* from the image (second row) causes segmentation errors in the baseline model which hallucinates a sidewalk (yellow) when there is none. Our model trained with proposed data-augmentation is more robust to these context changes.

Context generally refers to different kinds of information including co-occurring objects, scene type and lighting. For our analysis, we limit context to only the set of co-occurring objects in the image. While this might seem restrictive, we find in our analysis that image classification and segmentation models learn many interesting and undesirable dependencies between an object and other co-occurring objects (context) in the image. We use object removal as the main methodology to understand and quantify the role of context in downstream vision models. Specifically, we compare the output of the target models on the original input image and an edited version of this image with one object removed from it. If the model heavily uses the contextual relationship between removed object and the objects present in the image, removal will have an adverse effect on the model output. Measuring this helps us quantify the contextual dependencies learnt by the target model.

Ideally we want models which can utilize contextual cues when available, but are robust to variations in context and can detect and segment objects even when they appear out of context. However, machine learning based vision models are biased to the data seen frequently in training and tend to perform poorly on less frequent situations, for example the object-without-context and context-without-object scenarios. We address this by proposing a data augmentation scheme to expose the image classification and segmentation models to different contexts during training, and thus improving the robustness of the models to context. This is done by removing selected objects from images and training the models on the edited images to recognize and segment other objects in the image, even with contextual objects removed. Our experiments show that the classification and segmentation models trained with this data augmentation scheme are less sensitive to context changes and perform better on real out-of-context datasets, while preserving the baseline performance on the regular data splits.

To summarize, the main contributions of this paper are as follows: a) We propose an object removal based method to understand and quantify sensitivity of vision models to context, b) We apply this to analyze image classification and segmentation models and find some interesting and undesirable dependencies learnt by the models between classes and contextual objects and c) We propose a data augmentation scheme based on object removal to make the models more robust to contextual variation and show that it helps improve performance in out-of-context scenarios.

2. Related work

The importance of semantic context in visual recognition is a well established with studies showing context can help humans recognize objects faster e.g. when dealing with difficult low resolution images [15, 1]. In computer vision, incorporating context information has been shown to improve performance in various tasks including object recognition [12, 22, 18] and action recognition [9], object detection [3] and segmentation [25]. Early approaches built explicit context models by incorporating co-occurrences [18] and spatial location statistics [6]. Recently, explicit context modeling has been replaced by deep convolutional neural network (CNN) encoders which summarize the whole image into compact features. Classification and segmentation models, built on top of these deep features, can exploit information about object and context to achieve good performance [11, 7, 14]. Approaches to improve the use of context in CNNs have been explored including using spatial pyramids [26], atrous convolutions [4] and learning context encoding with a separate neural network [25]. While this implicit context encoding with deep CNNs improves performance, it is less interpretable and is hard to know if the model decisions are based on object or contextual evidence.

Methods have been proposed to inspect neural networks by visualizing salient regions for classification decisions [19, 24], and quantifying interpretability of individual units [2]. While these works focus on interpreting the internal representations of the network, we look at quantifying the context sensitivity of black-box models from the input data perspective. By manipulating the input image to remove objects and observing the network output, we quantify the sensitivity of classification and segmentation models to context and discover some interesting and undesirable dependency between classes. Related work [16] proposes erasing randomly sampled pixels to visualize important regions for a black-box models decision. Despite some similarities in the methodology, we focus on measuring the effect of entire context objects on model predictions. Data augmentation by adding objects into new contexts was proposed in [8], to improve the performance of object detection models. By adding out-of-context objects into images [20] shows that object detection networks are brittle to the presence of out-of-context objects. In contrast, we quantify the contextual dependencies between object classes in segmentation and classification models and improve their context robustness with removal based data augmentation.

3. Quantifying the role of context

We use object removal to quantify the contextual dependence of image classification and segmentation models, by designing metrics which measure the change in the model output between the original and the edited images with context objects removed. Now, we will discuss our removal model, present the robustness metrics and the data augmentation strategies to reduce the contextual dependence and improve performance in out-of-context setting.

3.1. Object removal

To create edited images with context objects removed, we need a fully automatic object removal model. For this, we utilize ground-truth object masks to remove the desired object and use an in-painting network to fill in the removed region. We base our in-painting network on the model proposed in [21], since this inpainter is directly optimized for removal, and can better handle irregular masks used in removal [21]. More details about the network architecture can be found in the supplementary material. The above removal method works well for medium sized objects, but struggles for large objects since then the in-painter needs to synthesize most of the image. Hence, we impose size restrictions on the objects we choose to remove to be less than 30% of the image. In the classification scenario on the COCO dataset, we consider all 80 object categories for removal. In the segmentation setting on the ADE20k dataset, we consider only the non-stuff categories (90 categories) for removal and measure the effects of removing these objects on the segmentation of all 140 categories. The stuff categories include objects like road, sky and field which are typically very large and hard to inpaint and hence are excluded from removal. An important point to note here is that the in-painter is not aware of the downstream models and is not optimized to fool/change their decisions. The effects of the in-painter are local and only around the removed object. Qualitative examples in Figures 2 and 3 show that the in-painting works well in the object removal setting.

3.2. Measuring context dependency

To understand the effect of contextual cues on image-classification and segmentation models, we test them on edited images where a context object has been removed. Precisely, given an original image I containing a set of objects $C = \{c_1, \cdots c_n\}$, we create a set of edited images $\mathcal{I}_e = \{I - c_i | c_i \in C \text{ and removable}(c_i)\}$. Then, we test the target model on I and $\mathcal{I}_e$ and check if its output is consistent with the performed removal as described below.

Image-level classification. Given a trained classifier S_{c_i} for class c_i, we will now characterize how robust it is to changes in context of c_i. We first obtain classifier scores for the original image I, edited image $I - c_i$ with object c_i removed and for the edited set $\mathcal{I}_{\text{owc}} = \{I - c_j : c_j \in I, j \neq i\}$, all of which contain the object c_i but have one context object removed. Ideally, if the classifier S_{c_i} is robust to context changes it should score all the images in $\mathcal{I}_{\text{owc}}$ higher than the image $I - c_i$, since $I - c_i$ does not contain the object c_i and the images in $\mathcal{I}_{\text{owc}}$ do. Precisely, a classifier robust to context should satisfy the below in-equality:

$$S_{c_i}(I_{\text{owc}}) \geq S_{c_i}(I - c_i), \forall I_{\text{owc}} \in \mathcal{I}_{\text{owc}} \tag{1}$$

We can count the number of times this condition is violated to quantitatively measure the robustness of the classifier.

$$V^{\min}(c_i) = \frac{\sum_I \mathbb{1}\left[(\min_{I_{\text{owc}}} S_{c_i}(I_{\text{owc}})) < S_{c_i}(I - c_i)\right]}{\sum_I \mathbb{1}[c_i \in I]} \tag{2}$$

$$V^{\text{mean}}(c_i) = \frac{\sum_I \mathbb{1}\left[\mathbb{E}_{I_{\text{owc}}}[S_{c_i}(I_{\text{owc}})] < S_{c_i}(I - c_i)\right]}{\sum_I \mathbb{1}[c_i \in I]} \tag{3}$$

where $\mathbb{1}$ is the indicator variable. $V^{\min}(c_i)$ is a strict metric counting instances classifier scores $I - c_i$ higher than any of the edited images, whereas $V^{\min}(c_i)$ is a softer metric counting instances where $I - c_i$ is scored higher than the average score assigned to the edited images.

Semantic segmentation. To understand the role context plays in this pixel-level labeling task, we analyze the behaviour of a trained segmentation model by removing one object at a time from the original image. Specifically, we measure how the segmentation correctness of the rest of the image changes (as compared to segmentation of the original image) when we remove an object from the original image. Given a segmentation model P, we compute the intersection-over-union (IoU) for a class c_i (w.r.t. ground-truth) on the original image I and edited image $I - c_j$. If the IoU value changes more than threshold α, we consider the segmentation prediction for class c_i to be affected by removal of c_j. Counting these violations we get,

$$AR(c_i, c_j) = \frac{\sum_I \mathbb{1}\left[\left|\Delta\text{IoU}_{c_i c_j}\right| \geq \alpha\right]}{\sum_I \mathbb{1}\left[c_i, c_j \in I\right]} \tag{4}$$

where $\Delta\text{IoU}_{c_i c_j}$ is the change in IoU of class c_i with removal of object c_j and α is the change threshold. The matrix $AR(c_i, c_j)$ represents the fraction of images where removing the object c_j, affects the segmentation of the object c_i with high values of $AR(c_i, c_j)$ indicating that the segmentation model depends heavily on the presence of the context object c_j to segment c_i.

3.3. Data augmentation with object removal

We now present our data augmentation solution to reduce the sensitivity of classification and segmentation models to context distribution. The main idea is to expose these models to training images of object-without-context and context-without-object scenarios. This will help the models deal with the lack of contextual information and hence get more robust to context changes. For this, we perform object removal to create edited images with some objects removed and add these edited images to the training batch. Specific details of how to pick objects for removal and how to use them in training for the two tasks are discussed below.

Classification. We experiment with two strategies to use the edited images in the classifier training. In the first approach *Data-aug-rand*, a uniformly randomly sampled with uniform probability and the classifier is trained with simple binary cross-entropy loss using both original and edited images. Edited image is assigned the same labels as the same as the original image excluding the removed object class. In the second approach *Data-aug-const*, we explicitly optimize for robustness by including the in-equality in (1) in the loss function. To do this, for randomly selected images in the training batch, we create the full edited image set $\{I - c_i : c_i \in I\}$. Then we can incorporate the robustness constraint as a hinge loss L_h with final loss being a weighted sum of the cross-entropy and the hinge losses.

$$\mathcal{L}_h(I) = \sum_{c_i \in I} \max\left[0, S_{c_i}(I - c_i) - \min_{c_j, j \neq i} S_{c_i}(I - c_j)\right] \tag{5}$$

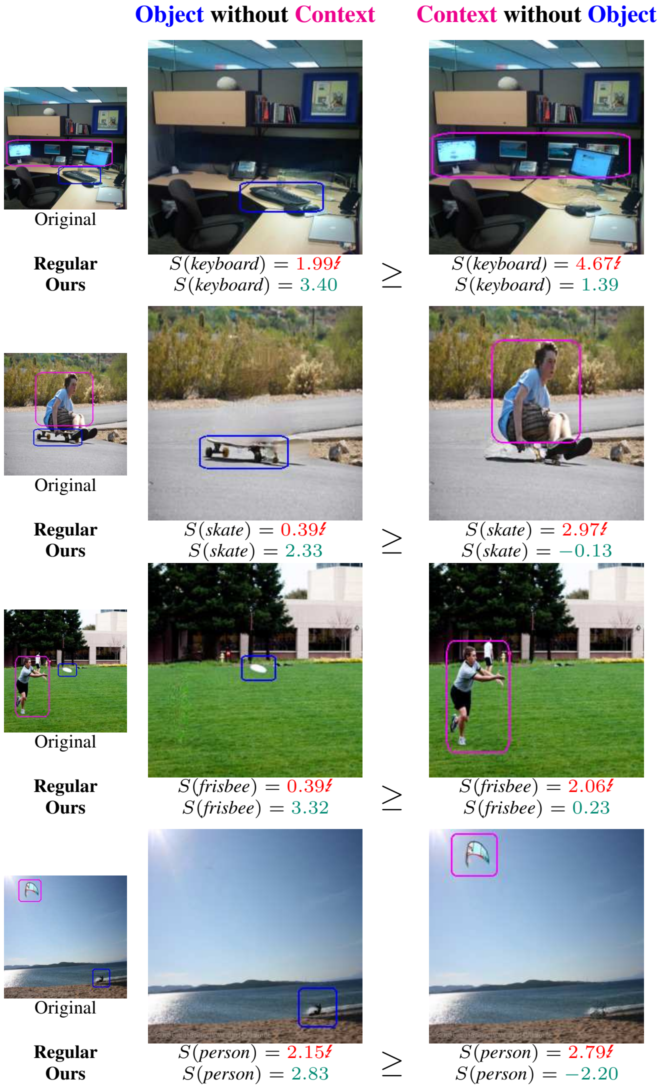

Figure 2: Context violations by image-level classifier. The primary object is marked with blue box and the context object is marked with magenta. The first column shows the original image, middle shows the image with only object and the third with only the context. We see that the baseline classifier depends heavily on the context and always scores the context only images (last column) higher than the image with only the primary object (middle column). The data augmented model does better and gets the ordering right.

Segmentation. We also perform data augmentation on the segmentation task by creating edited images with selected objects removed. The edited images can be used in training the segmentation model in two ways. First we can ignore the removed pixels and train the model to predict the original ground-truth labels on the rest of the image (*Ignore*). This helps the model learn that the labeling of a pixel should not be affected by the removal of a context object. Alternatively, we can explicitly tell the model that the removed object is not present by minimizing the likelihood assigned to the removed class at the edited pixels (*Negative loss*).

We explore three strategies to select objects to remove. The first strategy, *Random*, selects one random object to remove from the objects present in the image with uniform probability. However, sometimes the *Random* strategy can select very large object for removal, which can harm the quality of the edited image. To address this the *Size-based* strategy selects objects based on their relative sizes in the image, assigning higher probability to smaller objects. The probability for picking an object is computed as $p(c_i, I) \propto \left[\frac{\sum_{c_i \in I} a(I,c_i)}{a(I,c_i)}\right]$ where $a(I, c_i)$ is the area of the class c_i in image I.We also explore a hard negative mining based strategy, where we create harder training examples for the segmentation model by removing easy classes. This allows the model to focus on segmenting the harder classes while also becoming robust to context. Concretely, in *Hard-Negative* strategy we monitor the average cross-entropy loss $l_{\text{avg}}(c_i)$ for an object class c_i and calculate the probability of removal of c_i as inversely proportional to $l_{\text{avg}}(c_i)$.

4. Experiments and Results

This section presents the results of our analysis of how much the contextual information influences the performance of image classification and segmentation models. Using the robustness metrics defined in Section 3.2, we discover that the classification predictions on many well-performing classes are sensitive to context, and perform poorly on object-without-context and context-without-object images. Similar results are also found in the segmentation setting with the model depending heavily on context objects to correctly segment classes like *road*, *sidewalk*, *grass*. We also present results from our data-augmentation strategies, which help reduce this context dependence and improve robustness, without sacrificing performance.

4.1. Image level classification

4.1.1 Experimental setup for classification

Training data. We run our classification experiments on the COCO dataset [10], which contains 80 labeled object classes in their natural contexts. The dataset also has bounding box and segmentation annotation for each object. We use image-level labels to train the classifiers and use the object segmentation masks to test them with object removal.

Out-of-context testing. Apart from testing the classifier models on regular COCO data we conduct additional experiments to quantify the performance in out-of-context scenarios with natural images. We divide the COCO images into two splits: the first split *Co-occur* with images having at least two objects in them and the second split *Single* with images containing a single object. The *Full* split is all images combining *Co-occur* and *Single*. The idea behind this splitting of the dataset is to separate out images

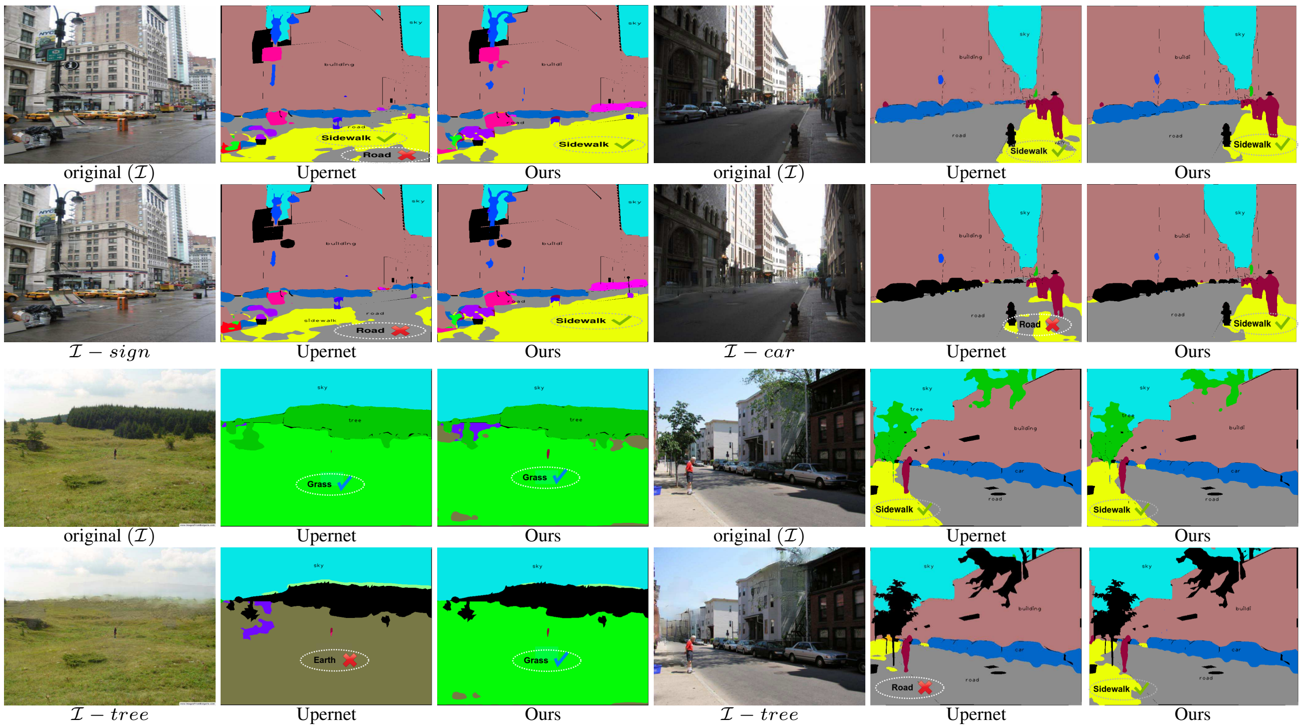

Figure 3: Examples of segmentation failures due to removal of a single context object. We see the segmentation of road, sidewalk and grass affected significantly when context objects like signboard, car and tree is removed (comparing odd and even rows). Model trained with proposed data-augmentation is more robust to these changes.

where objects occur in their context (*Co-occur*) and images where object occur alone without the usual co-occurring context objects *Single*. Now we can train our models on the *Co-occur* split and test it on the *Single* split to measure, using only real images, how a classifier trained with only co-occurring objects performs when objects appear without the context seen in training. Additionally we also test our COCO trained models on the *UnRel* dataset [17] which contains natural images with objects occurring in unusual contexts and relationships. We keep the classes which map to one of the 80 object classes in COCO, leaving 29 classes and 1071 images in the UnRel dataset.

Baseline classifier. The image-level classification model we test is based on the architecture proposed in [14]. It consists of a Imagenet [5] pre-trained VGG-19 network for feature extraction network followed by two convolution layers, global max-pooling layer and a linear classification layer with sigmoid activations. The model is trained with binary cross-entropy loss. We train and test the model at single scale at 256x256 resolution, to simplify the analysis. Our classifier achieves similar mAP on real COCO data as reported in [14], with our mAP slightly lower (0.600 vs 0.628 in [14]) due to single scale training and testing.

4.1.2 Analyzing classifier robustness to context

To measure the robustness of the trained classifier to context, we test it on real images and edited images and compute the robustness scores V^{min} and V^{mean} as described in Section 3.2. Table 1 shows the robustness scores averaged over all classes computed on the COCO test along with the standard performance metric mean average precision (mAP) for the baseline classifier (first row). We can see that, despite achieving good mAP (0.6), the baseline classifier trained on full data performs poorly in-terms of robustness metrics. In about 34% of cases the model violates the context consistency requirement of (1). This means in 34% cases, the classifier scores images without the target object higher than an image where object is present but a context object has been removed. Comparing the per-class robustness score, $V^{\text{min}}(c_i)$ and the per-class average precision (AP) (see supplementary for visualization), we see that good performance in AP does not mean the classifier is robust to context. Many classes like mouse, keyboard, sink, tennis racket etc, which are performing well in AP ($\geq$ 0.8), but have poor robustness to changes in context ($V_o^{\text{min}} \geq 50\%$). In extreme case, the *mouse* classifier violates the consistency in more than 90% of cases, despite having very good AP (0.88). This indicates that the classifiers are relying too much on contextual evidence to detect the objects but perform poorly when tested on images where the context distribution is different from training.

We visualize the violations in Figure 2. In the first row we can see that the keyboard classifier scores the image with

Model	Training Data	COCO test set			Robustness Metrics		UnRel
		Full ↑	Co-occur ↑	Single ↑	V^{min} ↓	V^{mean} ↓	dataset ↑
Baseline	Full (39k)	0.60	0.57	0.62	34%	24%	0.50
Data-aug-rand	Full (39k)	**0.61**	**0.58**	**0.65**	32%	22%	**0.54**
Data-aug-const	Full (39k)	0.60	**0.58**	0.63	**25%**	**14%**	0.52
Baseline	Co-occur (30k)	0.56	0.55	0.58	34%	24%	0.46
Data-aug-rand	Co-occur (30k)	**0.58**	**0.57**	**0.60**	31%	21%	0.49
Data-aug-const	Co-occur (30k)	**0.58**	**0.57**	**0.60**	**27%**	**15%**	**0.51**

Table 1: Effect of data augmentation on classification model

Model	all (407 images)		with car (258)		without car (149)	
	Road	Sidewalk	Road	Sidewalk	Road	Sidewalk
Upernet	0.81	0.59	**0.86**	**0.67**	0.68	0.40
DataAug	**0.82**	**0.60**	**0.86**	0.65	**0.72**	**0.46**

Table 2: Comparing the performance of road and sidewalk segmentation on natural images with and without cars.

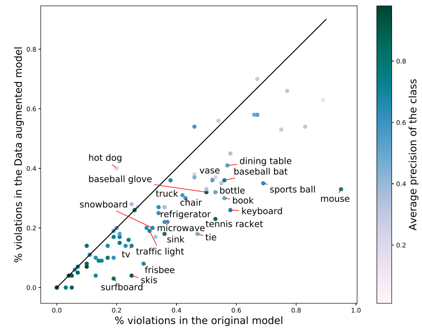

Figure 4: Comparing the % of violations in different classes with and without data augmentation. Points below the diagonal line show improvement with data-augmentation and the ones above degrade. The colors denote the average precision.

the keyboard removed higher (4.67) than the image with the keyboard but with the monitors removed (1.99). Similarly, we see the skateboard and the frisbee classifiers relying on person to hallucinate the respective objects. The violations shown in the first three rows of Figure 2 occur in objects with high co-occurrence dependence with other classes. However, context violations also occur in classes like *person* which appear in diverse contexts as seen in the last row of Figure 2. Here, the violation occurs in a difficult image where the *person* is small, but a more distinct class with co-occurrence dependence on person is clearly visible (*kite*). The classifier uses the *kite* context to hallucinate that there is a *person*, even when the *person* has been removed.

4.1.3 Data augmentation to improve robustness

We train two variants of the data-augmented image classification models as described in Section 3.3. The first *Data-aug-rand* learns with standard cross-entropy loss on the edited images with a random object removed and the second *Data-aug-const* which is optimized directly for robustness using a set of edited images and hinge loss.

Quantitative results. We present the evaluation of the data-augmented and the baseline models in Table 1. On models trained with *Full* training data, the data-augmented model *Data-aug-rand* provides a small improvement in overall mAP on the COCO test set (0.61 vs 0.60). However measuring the performance on the two splits *Co-occur* and *Single* reveals that the improvement is significant on the *Single* split (0.65 vs 0.62), indicating that the data augmentation helps the classifier better deal with out of context objects. This is also seen when comparing the performance of the two models on the UnRel dataset, where *data-aug-rand* significantly improves over the baseline model (0.54 vs 0.50). This improved robustness of the data augmented classifier to context changes is also measured by our robustness metrics V^{min} and V^{mean}. *Data-aug-rand* classifier makes overall 2% less violations under both worst-case (V^{min}) and average-case (V^{mean}) context changes. Directly optimizing the robustness constraints allows the model *Data-aug-const* to significantly improve upon the baseline model in robustness metrics, while still obtaining improvement in the performance metrics. It exhibits much less worst-case (25% vs 34% for baseline) and average-case violations (14% vs 24% for baseline), while improving the performance in the UnRel dataset (0.52 mAP vs 0.50 for baseline). The benefit of optimizing for robustness is clearly seen when we constrain the training data to the *Co-occur* set, where the classifier never sees objects alone. Baseline model trained on the *Co-occur* set drops in performance on the *Single* (0.58 from 0.62 on when trained on *Full*) and the UnRel test sets (0.46 vs 0.50 with *Full*) . However, with data augmentation and enforcing robustness constraints, we can recover some of this performance. On the *Single* test set *Data-aug-const* model trained on *Co-occur* set gets 0.58 mAP compared to 0.60 by baseline model trained on full data and even surpass it on the UnRel test set with 0.51 mAP. This shows that the data augmented model is able to overcome the contextual bias in the training set and perform well in unseen contexts.

When we compare the per-class robustness metrics between regular and data augmented models (data-aug-const), as shown in the Figure 4, we see that data-augmentation significantly reduces the worst case violations (V^{min}) on well-performing classes. For example, V^{min} drops from 95% to less 36% for the mouse class and from 58% to 28% for the keyboard class. The effect of this increased robustness is seen in qualitative examples in Figure 2. In the first row, the baseline keyboard classifier gives too much weight to evidence from *monitor* and scores the image with only *monitor* higher than the image with only keyboard. However, the data augmented model correctly orders the two images.

4.2. Semantic segmentation

So far, we have seen that multi-label classification models suffer from sensitivity to context, with classifiers often mixing up contextual and visual evidence. Next we will measure the context sensitivity of models in a more local and strongly supervised task of semantic segmentation.

4.2.1 Experimental setup for segmentation

Training and test data. We conduct our semantic segmentation experiments primarily on the ADE20k dataset [27] containing 140 categories of labeled objects, in different settings. Some of the 140 classes are typical background classes like *sky*, *sea* and *wall* and are large and difficult to in-paint and are hence excluded from removal.

Out-of-context testing. Following the process in image-level classification, we also measure the performance of the segmentation models on real out-of-context data. This in done in two ways. First, we train the segmentation model in a restricted setting with only three classes *car*, *road* and *sidewalk*. Now, we can again make two splits of the training and testing images into the *Co-occur* split of images with at-least two objects (3317 images) and the *single* split with only a single object (1693 images). Then we train the segmentation models on *co-occur* split and test on *single* split to see how well it can perform segmentation without context. Additionally we also test the models trained with ADE20k data on the Pascal-context dataset [13] in order to measure the performance under a different context distribution. This is done by manually mapping the 59 labels in the pascal-context to ADE20k labels and restricting the segmentation model to produce only the mapped labels.

Baseline segmentation model. We use the recent UperNet [23] model, with good results on the ADE20k, as our baseline segmentation model. We train the variant with the Resnet-50 encoder and a Upernet decoder with batch size of 6 images (maximum that fit in GPU) and with the default hyper-parameters suggested by the authors. This model achieves mean intersection-over-union (mIoU) of 0.377 and accuracy of 78.19% with single scale testing.

4.2.2 Context in semantic segmentation

We analyze robustness of the segmentation models to context by removing objects and computing the matrix $AR(c_i, c_j)$ presented in Section 3.2, which measures the % of images where removal of object c_j significantly affects segmentation of object c_i. The matrix $AR(c_i, c_j)$ we obtain for the Upernet model in ADE20k dataset is a sparse matrix with sharp peaks (see supplementary for a visualization). This indicates that the classes depend on specific context objects and are significantly affected by their removal. The sparsity also indicates that the effects on the segmentation are due the class being removed and not in-painting artifacts (otherwise the segmentation would be affected by all removal). Some of dependencies we discover in $AR(c_i, c_j)$ are reasonable and harmless, for example between *pot* and *plant* ($AR = 50\%$). Once you remove the *plant*, *pot* looks more like a *trash can* and the segmentation model often flips the label to *trash can*. However other dependencies are spurious and not desirable. For example, we notice that often the segmentation model uses presence of *car* to differentiate between *road* and *sidewalk*. Removing *car* affects the IoU of the *road* and *sidewalk* in 21% and 22% of cases respectively. This dependence is undesirable, and can be catastrophic in applications like self-driving cars.

We show qualitative examples where removal affects segmentation of Upernet model in Figure 3. The first two rows show the cases where removal of an object negatively impacts the segmentation of other objects. This include cases where removal of *street sign* and *car* severely affects segmentation of *road* and *sidewalk*, and a case where removal of *trees* affects segmentation of *grass*. We can see from these examples that while edit on the image is small and local, the effects of this removal on segmentation prediction is not local. Removal of a small objects can have drastic effects on segmentation in a far-away region.

Model	Removed pixels	mIoU	Acc
Upernet[23]	-	0.377	78.31
DA (random)	Ignore	0.320	75.2
DA (sizebased)	Ignore	0.379	78.31
DA (hard negative)	Ignore	0.375	77.8
DA (sizebased)	Negative	0.377	78.25
DA (hard negative)	Negative	**0.385**	**78.47**

Table 3: Data augmentation results on ADE20k dataset

4.2.3 Data augmentation for segmentation

Next we will look at the results of using data-augmentation for segmentation models. For this purpose we train the Upernet [23] based data-augmented models on the ADE-20k dataset with on three different strategies for selecting the object to remove as discussed in Section 3.3.

Quantitative results. Table 3, shows the results comparing the data-augmented models with the baseline Upernet model. We can see that random sampling strategy, which worked well in image classification, fails here leading to drop in performance. This is because, many object categories in ADE20k dataset are large and difficult to remove like bed, sofa and mountain and random strategy suffers by picking these. Instead when we switch to size-based and hard-negative based sampling, we see that the performance improves and the the size-based sampling model achieves the best mIoU of the three models (0.379). Applying negative likelihood loss on the removed object class gets further improvement when combined with hard negative sampling. This model also improves upon the Upernet baseline (achieving 0.385 IoU vs 0.377 by Upernet), despite the fact that the removal based data-augmentation is designed to make the model more robust to contextual variations.

To understand how data-augmentation impacts sensitivity to context, Figure 5 visualizes the maximum sensitivity of a class to removal of other classes, $\max_{c_j} AR(c_i, c_j)$ for different classes with and without data-augmentation. We see that for majority of classes robustness to context improves with data augmentation. For example *pillow* class is only affected 32% of the time with context changes, compared to 53% before data augmentaion. Similary, *road* and *sidewalk* classes are only affected 9% and 14% of the time respectively, compared to 21% and 22% before. This improved robustness translates into better generalization to real out-of-context data. We can see this in Table 2 where the performance of the *road* and *sidewalk* segmentation is measured on the validation set on images with and without cars. On the full set and on the split with cars, we see that the performance of the baseline Upernet and our augmented model (DA hard negative with negative loss) is equivalent. However, when we look at only images without car, the Upernet model performs significantly worse in both road (0.68 vs 0.72 for ours) and sidewalk (0.40 vs 0.46 for ours) segmentation. This quantitatively shows that the baseline model struggles to distinguish between *road* and *sidewalk* without *car* in the image, whereas our data augmentation is more robust and performs well even without context (car).

We also see the benefit of data augmentation in experiments on restricted *Co-occur* training set and on the Pascal-context dataset. Our data augmented model outperforms the Upernet model (both trained on the ADE20k dataset) when tested on the Pascal-context dataset in both mIoU and pixel accuracy. While the Upernet model achieves mIoU of 0.284 and pixel accuracy of 61.3% our data augmented model achieves 0.293 and 62.10% respectively, indicating that it is able to generalize better when tested on a dataset with different context distribution than one seen during training. Table 4 presents the experiments with the *Co-occur* training set in the three class setting. First we can see that when we switch from training on *Full* training data to *Co-occur* split (containing only images with atleast two objects), the performance of the Upernet greatly drops on the *Single* test split (from 0.67 to 0.52). This is indicates that the model overfits to the context it sees, and is not able to segment objects when it seeing them out of context. However, with data-augmentation we generate images of objects without context, and can recover most of this performance loss (0.646). Surprisingly, data-augmented model trained on smaller *co-occur* data also outperforms the baseline trained with *Full* data when tested on the *co-occur* split. Further quantification of robustness for different network architectures are included in the supplementary material.

Qualitative examples in Figure 3 also show the effect of increased robustness to context. While the baseline Upernet model is affected by context object removal causing drastic changes in predictions of other regions, our data augmented model is more stable. For example the removal of *signboard*, *car* or *tree* does not effect the segmentation of the *road* or *sidewalk* by our model.

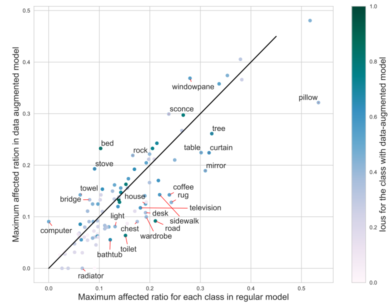

Figure 5: Comparing the context sensitivity of different classes with and without data augmentation with $\max_{c_j} AR(c_i, c_j)$ metric. Points below the diagonal improve with data-augmentation. The color denotes the mIoU.

Model	Training Data	Full	Only Cooccur	Only Single
Upernet	Full (5k)	0.774	0.797	0.670
Data Aug	Full (5k)	0.742	0.754	**0.675**
Upernet	Co-occur (3.3k)	0.680	0.713	0.520
Data Aug	Co-occur (3.3k)	**0.82**	**0.86**	0.646

Table 4: Experiments in three class setting on ADE20k

5. Conclusions

We have presented a methodology to analyze and quantify the context sensitivity of image classification and segmentation models, based on editing images to remove objects and measuring the effect on the target model output. Our analysis shows that despite good performance in-terms on mAP, classifiers for certain classes like keyboard, mouse, skateboard are very sensitive to context objects and perform poorly when seen out of context. In semantic segmentation setting, our analysis shows similar dependency between classes. For example we discover that the model depends on the presence of car to segment roads and sidewalk and fails drastically when the car is not present in the image. We present a data augmentation scheme based on object removal to mitigate this and make the classification and segmentation models more robust to context changes. Our experiments show that the proposed data augmentation scheme can help models generalize to out of context scenarios without losing performance in standard setting, indicating that the data augmented models better balance contextual and visual information.

Acknowledgments

This research was supported in part by the German Research Foundation (DFG CRC 1223).

References

[1] E. Barenholtz. Quantifying the role of context in visual object recognition. *Visual Cognition*, 22(1), 2014. 2

[2] D. Bau, B. Zhou, A. Khosla, A. Oliva, and A. Torralba. Network dissection: Quantifying interpretability of deep visual representations. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2

[3] S. Bell, C. Lawrence Zitnick, K. Bala, and R. Girshick. Inside-outside net: Detecting objects in context with skip pooling and recurrent neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 1, 2

[4] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, and A. L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 40, 2018. 2

[5] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. ImageNet: A Large-Scale Hierarchical Image Database. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009. 5

[6] C. Desai, D. Ramanan, and C. C. Fowlkes. Discriminative models for multi-class object layout. *International Journal of Computer Vision (IJCV)*, 95(1), 2011. 2

[7] T. Durand, N. Thome, and M. Cord. Weldon: Weakly supervised learning of deep convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2

[8] N. Dvornik, J. Mairal, and C. Schmid. Modeling visual context is key to augmenting object detection datasets. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018. 2

[9] M. Jain, J. C. van Gemert, and C. G. Snoek. What do 15,000 object categories tell us about classifying and localizing actions? In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 2

[10] 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 *Proceedings of the European Conference on Computer Vision (ECCV)*, 2014. 4

[11] J. Long, E. Shelhamer, and T. Darrell. Fully convolutional networks for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 2

[12] M. Marszalek and C. Schmid. Semantic hierarchies for visual object recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2007. 2

[13] R. Mottaghi, X. Chen, X. Liu, N.-G. Cho, S.-W. Lee, S. Fidler, R. Urtasun, and A. Yuille. The role of context for object detection and semantic segmentation in the wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014. 1, 7

[14] M. Oquab, L. Bottou, I. Laptev, and J. Sivic. Is object localization for free?-weakly-supervised learning with convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015. 2, 5

[15] D. Parikh, C. L. Zitnick, and T. Chen. Exploring tiny images: The roles of appearance and contextual information for machine and human object recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 34(10), 2012. 1, 2

[16] V. Petsiuk, A. Das, and K. Saenko. Rise: Randomized input sampling for explanation of black-box models. In *Proceedings of the British Machine Vision Conference (BMVC)*, 2018. 2

[17] J. Peyre, I. Laptev, C. Schmid, and J. Sivic. Weakly-supervised learning of visual relations. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2017. 5

[18] A. Rabinovich, A. Vedaldi, C. Galleguillos, E. Wiewiora, and S. Belongie. Objects in context. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, 2007. 2

[19] M. T. Ribeiro, S. Singh, and C. Guestrin. Why should i trust you?: Explaining the predictions of any classifier. In *Proceedings of the ACM SIGKDD international conference on Knowledge discovery and data mining*, 2016. 2

[20] A. Rosenfeld, R. Zemel, and J. K. Tsotsos. The elephant in the room. *arXiv preprint arXiv:1808.03305*, 2018. 2

[21] R. Shetty, M. Fritz, and B. Schiele. Adversarial scene editing: Automatic object removal from weak supervision. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2018. 2

[22] A. Torralba, K. P. Murphy, and W. T. Freeman. Using the forest to see the trees: exploiting context for visual object detection and localization. *Communications of the ACM*, 53(3), 2010. 1, 2

[23] T. Xiao, Y. Liu, B. Zhou, Y. Jiang, and J. Sun. Unified perceptual parsing for scene understanding. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2018. 1, 7

[24] M. D. Zeiler and R. Fergus. Visualizing and understanding convolutional networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2014. 2

[25] H. Zhang, K. Dana, J. Shi, Z. Zhang, X. Wang, A. Tyagi, and A. Agrawal. Context encoding for semantic segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 1, 2

[26] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia. Pyramid scene parsing network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 2

[27] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, and A. Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 7