Training Convolutional Neural Networks to be Background Invariant

Ricardo Cruz
ricardo.p.cruz@inesctec.pt
Jaime S. Cardoso
jaime.cardoso@inesctec.pt

INESC TEC
Faculty of Engineering, University of Porto
Portugal

Abstract

Convolutional neural networks have been shown to be vulnerable to changes in the background. For example, a CNN trained using objects on top of a blue background often performs terribly when evaluated using a green background. The proposed method is an end-to-end method that augments the training set by introducing new backgrounds during the training process. The novelty is that these backgrounds are created on-the-fly using a generative network that is trained as an adversary to the model. The adversary dynamics ensures that the model has seen a wide range of backgrounds. The method is experimented with using MNIST and Fashion-MNIST as test cases.

1 Introduction

Possibly due to the fact that neural networks learn from static images, and so do not have to deal with depth as us humans, they have a brittle understanding of what an object is and are vulnerable to changes in the background – for example, when there is a mismatch in the background between the training and test sets, performance degrades terribly, as exemplified by Figure 2. In that case, the classifier is trained with digits in a clean, white background (a trivial task) and then evaluated with digits inserted in diverse backgrounds.

These disparities in background between training and testing set have not been studied in detail. There is one work that uses an attention mechanism but only avoids some artifacts, such as irregular borders [2].

Generative adversarial networks (GAN) generate realistic images through a min-max problem whereby two models (generator vs discriminator) try to optimize a given loss function in the opposite direction [1]. This work is loosely inspired by this dynamic. A generator is proposed to augment the training set by producing backgrounds that purposefully have an adverse effect on the performance of the target model, making the target model more robust as a result. To introduce the new background, the object must first be segmented therefore a third neural network that is trained in an unsupervised manner.

2 Related Work

Literature exists in predicting classifier confidence for dataset shifts so that changes in the background could be detected. However, making the classifier itself robust to changes in the background seems to have been the subject of little study. One work proposes an attention mechanism to avoid artifacts, particularly irregular borders, from influencing the classifier [2]. Two classifiers are used: a *global* CNN, G, and a *local* CNN, L. The proposed method works by having G find the bounding box of the relevant object in order to create a cropped version of the image, and then use L to classify the cropped version.

That attention mechanism works in two phases. Firstly, G is trained to classify the entire image x. Then, a truncated version G^T is used to obtain activation maps and find a bounding box around the object so that a function f produces a cropped image x'. Finally, L is then trained using

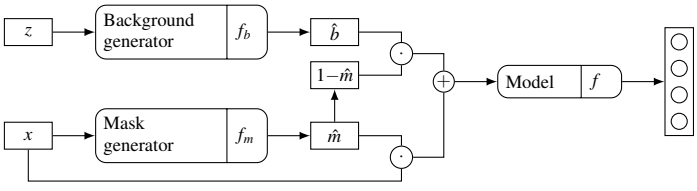
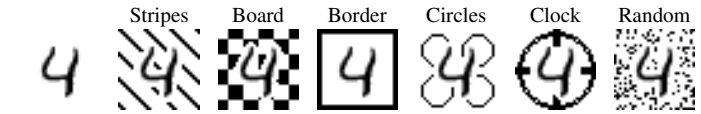


Figure 1: Proposed adversarial background augmentation during training.



	Stripes	Board	Border	Circles	Clock	Random
Traditional	38.0	24.3	61.4	32.9	19.7	11.2
Proposal	92.3	76.8	93.1	93.7	70.8	86.2

The model is a CNN with VGG blocks as detailed in Section 4, trained for MNIST. Accuracy values for the entire testing set when different backgrounds are used.

Figure 2: Background change can produce wild disparate accuracies (%).

the smaller image x' [2]. To then predict the class y of the image x, this chained process can then output the class $\hat{y} = L\big(f(G^T(x),x)\big)$.

Two disadvantages are immediate: (i) L operates on a rectangular cropped version of the image and therefore is still influenced by artifacts that remain inside that rectangle, and (ii) model G is still influenced by artifacts because it did not have the benefit of being trained against the artifacts. While such artifacts are not presented in the training set, they could be generated in a controlled fashion, as the method next proposed.

3 Method

The goal is to (during training) be able to place the object in a multitude of contexts (backgrounds), facilitating the learning of robust representations, focused on "what" the object is rather than "where" the object is. We propose to adopt adversarially generated backgrounds to promote the learning of strong representations. However, the insertion of adversarial backgrounds in the image cannot be allowed to destroy the concept (class) one is trying to learn. Since the spatial delineation of the object is unknown, we propose to learn, simultaneously with the recognition, the segmentation mask. This mask is used to inject the adversarial background only in the non-object pixels.

Model: A model f is optimized to minimize a loss $\mathcal{L}(y, f(x))$ using an image x as input with label y as the ground-truth. This image is subject to data augmentation through the process illustrated in Figure 1.

The framework is agnostic of the task and other losses could be used for different tasks: regression, semantic segmentation, reinforcement learning, etc. In these experiments, classification was performed using cross-entropy,

$$\mathcal{L}(y,\hat{y}) = \sum_{i=1}^{N} y_i \log \hat{y}. \tag{1}$$

Mask generator: Firstly, a model f_m is trained to produce a mask $\hat{m}$ using a sigmoid activation function to ensure $\hat{m} \in [0,1]$ so that it can be used to segment the image through a element-wise product, $x' = x \odot \hat{m}$. The model f_m can be optimized in an unsupervised fashion by finding the best mask that minimizes the previous loss, $\mathcal{L}(f(x \odot f_m(x)), y)$. To help prevent the mask from including background, a term $\mathcal{L}_A$ is used to constraint its size

$$\mathcal{L}_A(\hat{m}) = \max(0, A(\hat{m}) - a), \tag{2}$$

where A approximates the percentage of the area of the mask by computing $A(\hat{m}) = \frac{1}{wh}\sum_{x,y}\hat{m}_{x,y}$, and a is the average area for the object given as an hyperparameter ($a = 0.2$ is used in all experiments). For better performance, after model f_m has been trained, a non-differentiable transformation T can henceforth be applied to further improve the segmentation. For example, a threshold t, $T(\hat{m}) = \mathbb{1}_{x,y}(\hat{m}_{x,y} \geq t)$, can be chosen using Otsu's method or the a-quantile such that $A(\hat{m} \geq t) = a$.

Notice that the mask generator being background invariant is unimportant since it is only used during training and on training images. A typical architecture for the mask generator would be a U-Net [5]. For better results, the mask could be provided by the user through manual segmentation.

Background generator: Secondly, the background is generated by a neural network f_g that transforms noise z into a background $\hat{b}$ image. Unlike the others, this model is trained to *maximize* the loss $\mathcal{L}$. The generator focuses on producing backgrounds or artifacts that could potentially adversely affect the output of the model.

In the case of MNIST and Fashion-MNIST which are monochrome, the background generator could "cheat" by producing a background with the same color as the object, thus obfuscating the object. In these cases, a constrain was added in the form of the additional regularization $\mathcal{L}_{B_A}(\hat{b})$ term which is added to disallow the background from filling over half the pixels, $\mathcal{L}_{B_A}(\hat{b}) = \max\left(\frac{1}{N}\sum_{i=1}^{N}\hat{b}_i - 0.5, 0\right)$.

Overall dynamic: All in all, the min-max optimization problem can be summarized as

$$\min_{f,f_m}\max_{f_b}\sum_{i=1}^{N}\mathcal{L}\left(f\left(\hat{m}_i \odot x_i + \hat{b}_j \odot (1-\hat{m}_i)\right), y_i\right) + \mathcal{L}_A(\hat{m}_i) + \mathcal{L}_{B_A}(\hat{b}). \quad (3)$$

Notice that, while the optimization problem was inspired by GANs, this is not a GAN framework, there is no discriminator used. Also, while this problem could be optimized end-to-end, we have performed this optimization in three stages: (i) train model f, (ii) train mask generator f_m, (iii) train both model f and its adversarial background generator f_b. Training in stages is useful for debugging and fine-tuning, but also it allows applying non-differentiable transformations on top of f_m such as thresholds to help produce more realistic masks.

4 Experiments

MNIST [3] and Fashion-MNIST [6] are artificially enhanced by introducing backgrounds as illustrated in Figure 3. This enhanced versions are used only for *testing* purposes, while the original unmodified dataset is used for *training*. The idea is to see how well the model performs when background textures are introduced.

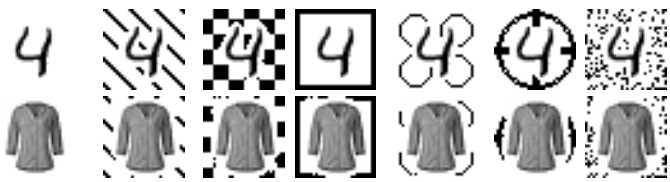
Table 1 summarizes the results showing the proposed method (proposal) to have become background invariant. Interestingly the attention mechanism results only negligibly improve on the baseline classifier. This mechanism works by cropping the image and, not surprisingly, it was found to perform best in the border case (with over 50% accuracy); still, this result was worse than the proposal.

To better understand the impact of changes in the background, let us vary the rate of the random parameter from the previous Figure 3 (g). In Table 2, a Bernoulli distribution is used for the background with varying parameter values, as illustrated in the images. While the baseline naturally produces better results for the unchanged image, as the rate is increased, the drop in baseline's performance is fathomed while the proposal drops more smoothly.

Furthermore, to better understand what could be improved on the framework, the mask generator is changed so that a manual segmentation is used instead of a neural network, and also the background generator is changed to produce noise instead of trained adversarially (see Table 3). Two conclusions are apparent: (a) the fact we train the mask generator in an unsupervised fashion means that the masks are imperfect which greatly influence performance, (b) using noise as the background is not sufficient to avoid the network being fooled by more intricate patterns as those in the testing set (Figure 3).

5 Conclusion

This work was fomented by previous work where the goal was to train a drone using a dataset that was easier to acquire indoors (inside a studio) rather than outdoors where it was going to be used, because it dealt with electricity insulators [4].



(a) Original (b) Stripes (c) Board (d) Border (e) Circles (f) Clock (g) Random

Figure 3: Backgrounds introduced for MNIST and Fashion-MNIST.

Table 1: General Results (Validation Accuracy in %)

Method	Stripes	Board	Border	Circles	Clock	Random	Avg
	MNIST						
Traditional	38.0	24.3	61.4	32.9	19.7	11.2	31.2
Attention [2]	28.1	26.8	57.3	40.1	29.3	25.1	34.5
Proposal	92.3	76.8	93.1	93.7	70.8	86.2	**85.5**
	Fashion-MNIST						
Traditional	21.3	24.6	36.9	28.5	29.6	16.8	26.7
Attention [2]	18.2	20.1	51.8	26.0	31.8	36.2	30.7
Proposal	62.9	61.5	66.5	60.9	60.8	45.9	**59.8**

Table 2: Effect of varying the random noise rate in terms of Accuracy (%).

	0.0	0.01	0.05	0.1	0.2	0.3	0.4	0.5
Baseline	90.1	33.3	13.2	11.2	10.5	10.2	10.2	10.6
Proposal	70.7	70.1	61.6	56.7	45.5	43.2	39.6	35.0

Table 3: Average accuracy for Fashion-MNIST when using different mask or background generators.

	Proposal	True mask	Noise background
Accuracy (%)	59.8	72.8	10.0

For that purpose, an adversarially trained model is proposed that is invariant to the background. During training, the target model tries to minimize its loss, but a generator counteracts it by injecting new backgrounds by optimizing for backgrounds that maximize the loss, thus making the target model robust to background changes. The method is evaluated using a synthetic dataset.

While the proposed method was evaluated for the task of classification, it could potentially be used for other tasks involving a CNN, such as regression problems, segmentation, or reinforcement learning tasks.

Acknowledgments

This work is financed by the ERDF – European Regional Development Fund through the Operational Programme for Competitiveness and Internationalisation – COMPETE 2020 Programme and by National Funds through the Portuguese funding agency, FCT - Fundação para a Ciência e a Tecnologia within project "POCI-01-0145-FEDER-028857". Furthermore, Ricardo Cruz was supported by Ph.D. grant SFRH/BD/122248/2016, also provided by FCT.

References

[1] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014.

[2] Qingji Guan, Yaping Huang, Zhun Zhong, Zhedong Zheng, Liang Zheng, and Yi Yang. Diagnose like a radiologist: Attention guided convolutional neural network for thorax disease classification. *arXiv*, 2018.

[3] Yann LeCun, Corinna Cortes, and CJ Burges. MNIST handwritten digit database. *AT&T Labs [Online]. Available: http://yann.lecun.com/exdb/mnist*, 2010.

[4] Ricardo M. Prates, Ricardo Cruz, AndrÃl' P. Marotta, Rodrigo P. Ramos, Eduardo F. Simas Filho, and Jaime S. Cardoso. Insulator visual non-conformity detection in overhead power distribution lines using deep learning. *Computer and Electrical Engineering*, 2019. doi: 10.1016/j.compeleceng.2019.08.001.

[5] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.

[6] Han Xiao, Kashif Rasul, and Roland Vollgraf. Fashion-MNIST: a novel image dataset for benchmarking machine learning algorithms. *CoRR*, 2017.