Model Vulnerability to Distributional Shifts over Image Transformation Sets

Riccardo Volpi*, Vittorio Murino*,†

Abstract. We provide a summary of the pre-print *'Model Vulnerability to Distributional Shifts over Image Transformation Sets'* (arxiv.org/abs/1903.11900) [22].

1. Introduction

When we devise and deploy a machine learning system, a generally desirable property is its ability to generalize to unknown scenarios. For example, we would like a vision module (*e.g.*, for a self-driving car) to perform well under a broad variety of visual conditions and environments. Albeit, modern learning systems are well known to be vulnerable to the dataset bias issue [6, 2, 1, 19]: when trained on data from some distribution, they will typically learn the peculiar statistics of the training data, and in result will perform more poorly in different settings.

In light of this, two fundamental research directions are (i) training more robust models against the distributional shifts that they might encounter after deployment – domain adaptation [6, 2, 16, 4, 20, 18, 13, 21] and domain generalization [9, 14, 15, 17, 12, 23, 10, 11, 23] are possible directions to face this problem – and (ii) developing tools to understand the vulnerability regions of a model before its deployment. In this work, we propose methods related to the second branch, and further exploit them to devise algorithms associated with the first one.

Focusing on computer vision models, we start from the assumption that, given a set of standard, content-preserving image transformations, we can generate a huge set of possible distributional shifts by concatenating them and applying them to the datapoints at our disposal. We propose a combinatorial optimization problem aimed at detecting the concatenations of different transformations that a given (black-box) model is most vulnerable to, and face it through random search and evolution-based search.

Endowed of these tools, we propose a training procedure where, over iterations, harmful image transformations in the given set are searched, and used as data augmentation rules throughout the training procedure. Models trained with our method, not only are more resistant against the transformations from the provided set, but also better generalize to unseen scenarios. For example, we propose results associated with a semantic segmentation module trained on CamVid [3], showing that it better generalizes to foggy scenarios even though it has never encountered them during training.

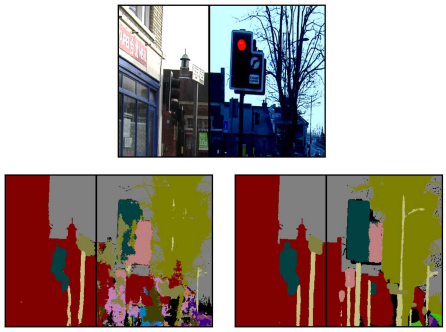

Figure 1. Image transformations (*e.g.*, the rightmost part of the *top* image) can cause distributional shifts that models are not able to handle *(bottom-left)*. Models trained with the methods proposed in this paper are more robust against a variety of image transformations *(bottom-right)*

2. Problem formulation

Let $\mathbb{M}(.)$ be a model that takes in input images and provides an output according to the given task. Let $D = \{(x^{(i)}, y^{(i)})\}_{i=0}^{m} \sim P(X, Y)$ be a set of datapoints with their labeling, drawn from some data distribution. Finally, let $\mathbb{T} = \{(\tau^{(i)}, l_i^{(j)}), i = 1 \dots N_T, j = 1 \dots N_i\}$ be a finite set, where each object $t = (\tau^{(i)}, l_i^{(j)})$ is a data transformation τ with a related magnitude l. The transformations give in output images in the same format as the input ones. We define a concatenation of different transformations as a transformation tuple; $\mathbb{T}_N$ is the set of all the possible transformation tuples that one can obtain by combining objects in $\mathbb{T}$. We define $\mathbb{T}$ with the following transformations: *autocontrast (20), sharpness (20), brightness (20), color (20), contrast (20), grayscale conversion (1), R-channel enhancer (30), G-channel enhancer (30), B-channel enhancer (30), solarize (20)*, where the numbers in parenthesis indicate the number of different magnitude levels (Table 2 in [22]). Armed with this set, we propose the following combinatorial optimization problem

$$\min_{T^* \in \mathbb{T}_N} f(\mathbb{M}, T^*, D) \tag{1}$$

where f is a function that measures the accuracy of a model $\mathbb{M}$ provided with set of labelled datapoints D, modified through a transformation tuple T^*. The $N-$tuples that

*Istituto Italiano di Tecnologia, Pattern Analysis and Computer Vision
†Università di Verona, Computer Science Department

Algorithm 1 Robust Training

1: **Input:** $D = \{(x^{(i)}, y^{(i)})\}_{i=1}^{n}$, init. weights θ_0, $N-$tuple set $\mathbb{T}_N$, init. data augmentation set $\mathbb{T}_{tr}$, learning rate α, loss ℓ.
2: **Output:** learned weights θ
3: **Initialize:** $\theta \leftarrow \theta_0$
4: **for** $j = 1, ..., J$ **do**
5: Minimize ℓ through k SGD steps using data sampled from D modified with transformation tuples sampled from $\mathbb{T}_{\mathbb{N}}$
6: Find $T^* \in \mathbb{T}_N$ by running RS or ES on a subset of D and append it to $\mathbb{T}_{tr}$

induce lower f values are the ones that a model $\mathbb{M}$ is more vulnerable to, with respect to the chosen metric.

Search algorithms. In order to approach the combinatorial optimization problem 1, we rely on Random Search (RS) and Evolution-based Search (ES). In RS, we test an arbitrary number of transformation tuples from the set $\mathbb{T}_{\mathbb{N}}$ and choose the one that leads to the lowest f value. In ES, we use standard operations from the genetic algorithm literature (*selection*, *mutation* and *cross-over*) to find harmful transformations more efficiently; for the details, we refer to the original work [22].

3. Training more robust models

We define a training procedure to learn models (ConvNets [8]) robust against image transformations from an arbitrary set as follows: (a) we initialize a transformation set to sample from during training (the "data augmentation set" $\mathbb{T}_{tr}$) with the *identity* transformation; (b) we train the network via gradient descent updates, augmenting samples via data augmentation procedures sampled from $\mathbb{T}_{tr}$; (c) we run RS or ES, using appropriate fitness function f and tuple set $\mathbb{T}_N$, and append the so-found transformation to $\mathbb{T}_{tr}$. We alternate between steps (b) and (c) for the desired number of times. See Algorithm 1 for a detailed view.

4. Experiments

In our original work [22], we have tested our search methods and training procedure on a variety of tasks. We report here the results associated with a semantic scene segmentation task (FC-DenseNet [7] with 103 layers trained on CamVid [3]), and refer to [22] for the other results (as well as for the description of the hyper-parameters). In Algorithm 1, we set the number of transformations concatenated as $N = 5$ and use the cross-entropy function between the output of the model and the ground truth labels as loss ℓ.

Figure 2 shows the output of a model trained via standard Empirical Risk Minimization (middle row) and the output of our model (last row), when the original input (first column) is perturbed by different image transformations

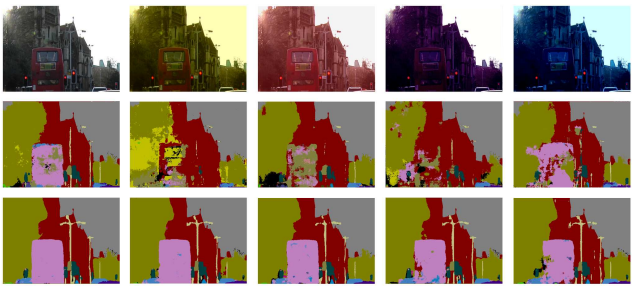

Figure 2. A sample from CamVid *(column 1, row 1)* modified with image transformations found via RS and ES *(columns 2 – 5, row 1). Rows 2 and 3* report the output of a model trained via standard ERM and a model train through Algorithm 1 with ES, respectively.

Performance of CamVid models

	Testing			
Method	**Original**	**RS**	**ES**	**Fog [5]**
ERM	$.862 \pm .007$	$.458 \pm .027$	$.311 \pm .013$	$.726 \pm .017$
Ours	$.851 \pm .002$	$.820 \pm .007$	$.822 \pm .008$	$.744 \pm .006$

Table 1. Pixel accuracy of CamVid models trained with standard ERM *(first row)* and with Algorithm 1 *(second row)*, and tested in different conditions *(columns)*. Results obtained by averaging over 3 different runs.

found via ES (first row). These results qualitatively show that models trained through Algorithm 1 are more resistant against image transformations; furthermore, they show that the transformations in $\mathbb{T}_N$ are reasonable approximations of possible visual conditions that a computer vision model might face after deployment. For example, images in the first row, second and third column, can be interpreted as simulations of the light that one could face during dawn or sunset—and in which the baseline model we compare against performs poorly.

Table 1 reports the pixel accuracy values. The columns *RS* and *ES* indicate the lowest values obtained by running the two search procedures using models trained via standard Empirical Risk Minimization (ERM) and with our method (first and second row, respectively) They confirm the higher level of robustness of models trained through Algorithm 1, using ES as search algorithm (line 6). The last column shows results obtained by manipulating the images with artificial fog [5]; also in this case, our models show better generalization properties.

5. Conclusions

We propose a combinatorial optimization problem to find harmful distributional shifts for a given model, defined in terms of concatenations of image transformations from an arbitrary set. We show that random search and, in particular, evolution-based search are effective approaches to face this problem. Further, we show that the these search algorithms can be embedded in a training procedure, where harmful transformations are searched and used as data augmentation rules throughout training.

References

[1] S. Ben-David, J. Blitzer, K. Crammer, and F. Pereira. Analysis of representations for domain adaptation. In B. Schölkopf, J. C. Platt, and T. Hoffman, editors, *Advances in Neural Information Processing Systems 19*, pages 137–144. MIT Press, 2007. 1

[2] J. Blitzer, R. McDonald, and F. Pereira. Domain adaptation with structural correspondence learning. In *Proceedings of the 2006 Conference on Empirical Methods in Natural Language Processing*, EMNLP '06, pages 120–128, Stroudsburg, PA, USA, 2006. Association for Computational Linguistics. 1

[3] G. J. Brostow, J. Shotton, J. Fauqueur, and R. Cipolla. Segmentation and recognition using structure from motion point clouds. In *ECCV (1)*, pages 44–57, 2008. 1, 2

[4] Y. Ganin and V. S. Lempitsky. Unsupervised domain adaptation by backpropagation. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, pages 1180–1189, 2015. 1

[5] D. Hendrycks and T. Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations*, 2019. 2

[6] H. D. III and D. Marcu. Domain adaptation for statistical classifiers. *CoRR*, abs/1109.6341, 2011. 1

[7] S. Jegou, M. Drozdzal, D. Vazquez, A. Romero, and Y. Bengio. The one hundred layers tiramisu: Fully convolutional densenets for semantic segmentation. *arXiv e-prints*, abs/1611.09326, 2016. 2

[8] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation applied to handwritten zip code recognition. *Neural Comput.*, 1(4):541–551, Dec. 1989. 2

[9] D. Li, Y. Yang, Y. Song, and T. M. Hospedales. Deeper, broader and artier domain generalization. *The IEEE International Conference on Computer Vision (ICCV)*, 2017. 1

[10] D. Li, J. Zhang, Y. Yang, C. Liu, Y. Song, and T. M. Hospedales. Episodic training for domain generalization. *CoRR*, abs/1902.00113, 2019. 1

[11] Y. Li, Y. Yang, W. Zhou, and T. M. Hospedales. Feature-critic networks for heterogeneous domain generalization. *CoRR*, abs/1901.11448, 2019. 1

[12] M. Mancini, S. R. Bul, B. Caputo, and E. Ricci. Robust place categorization with deep domain generalization. *IEEE Robotics and Automation Letters*, 3(3):2093–2100, July 2018. 1

[13] P. Morerio, J. Cavazza, and V. Murino. Minimal-entropy correlation alignment for unsupervised deep domain adaptation. *International Conference on Learning Representations*, 2018. 1

[14] S. Motiian, M. Piccirilli, D. A. Adjeroh, and G. Doretto. Unified deep supervised domain adaptation and generalization. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 1

[15] K. Muandet, D. Balduzzi, and B. Schlkopf. Domain generalization via invariant feature representation. In S. Dasgupta and D. McAllester, editors, *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pages 10–18, Atlanta, Georgia, USA, 17–19 Jun 2013. PMLR. 1

[16] K. Saenko, B. Kulis, M. Fritz, and T. Darrell. Adapting visual category models to new domains. In *Proceedings of the 11th European Conference on Computer Vision: Part IV*, ECCV'10, pages 213–226, Berlin, Heidelberg, 2010. Springer-Verlag. 1

[17] S. Shankar, V. Piratla, S. Chakrabarti, S. Chaudhuri, P. Jyothi, and S. Sarawagi. Generalizing across domains via cross-gradient training. In *International Conference on Learning Representations*, 2018. 1

[18] B. Sun and K. Saenko. Deep CORAL: correlation alignment for deep domain adaptation. In *Computer Vision - ECCV 2016 Workshops - Amsterdam, The Netherlands, October 8-10 and 15-16, 2016, Proceedings, Part III*, pages 443–450, 2016. 1

[19] A. Torralba and A. A. Efros. Unbiased look at dataset bias. In *Proceedings of the 2011 IEEE Conference on Computer Vision and Pattern Recognition*, CVPR '11, pages 1521–1528, Washington, DC, USA, 2011. IEEE Computer Society. 1

[20] E. Tzeng, J. Hoffman, K. Saenko, and T. Darrell. Adversarial discriminative domain adaptation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 1

[21] R. Volpi, P. Morerio, S. Savarese, and V. Murino. Adversarial feature augmentation for unsupervised domain adaptation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 1

[22] R. Volpi and V. Murino. Model vulnerability to distributional shifts over image transformation sets. *arXiv:1903.11900*, 2019. 1, 2

[23] R. Volpi*, H. Namkoong*, O. Sener, J. Duchi, V. Murino, and S. Savarese. Generalizing to unseen domains via adversarial data augmentation. In *Advanced in Neural Information Processing Systems (NeurIPS) 32*, December 2018. 1