https://doi.org/10.2352/EI.2022.34.10.IPAS-392
© 2022, Society for Imaging Science and Technology

Expert Training: Enhancing AI Resilience to Image Coding Artifacts

Alban Marie, Karol Desnos, Luce Morin and Lu Zhang
Univ Rennes, INSA Rennes, CNRS, IETR - UMR6164; Rennes, France

Abstract

In the Machine-to-Machine (M2M) transmission context, there is a great need to reduce the amount of transmitted information using lossy compression. However, commonly used image compression methods are designed for human perception, not for Artificial Intelligence (AI) algorithms performances. It is known that these compression distortions affect many deep learning based architectures on several computer vision tasks. In this paper, we focus on the classification task and propose a new approach, named expert training, to enhance Convolutional Neural Networks (CNNs) resilience to compression distortions. We validated our approach using MnasNet and ResNet50 architectures, against image compression distortions introduced by three commonly used methods (JPEG, J2K and BPG), on the ImageNet dataset. The results showed a better robustness of these two architectures against the tested coding artifacts using the proposed expert training approach. Our code is publicly available at `https://github.com/albmarie/expert_training`.

Introduction

Machine-to-Machine (M2M) connections are already part of our daily life, used in a wide spectrum of applications such as consumer electronics, autonomous vehicles, public utilities, telemedicine and manufacturing. In the span of 5 years, starting from 2017, the number of M2M connections is expected to increase fourfold [11]. Such substantial growth comes at the expense of the consumed bandwidth, which will be a major bottleneck for the Internet of Things. Face to the continuous increase of transmitted data and limited communication bandwidth, the data is commonly compressed lossyly. While 80% of the total bandwidth is used for video content only [8], lossy image/video compression is particularly important.

The usual goal in the image/video coding research field is to achieve the best trade off between the quantity of transmitted data and the perceptual quality. However, in the M2M context, this data is transmitted neither to be seen by a human observer, nor to be stored, but only to be used as the input of AI algorithms. Since camera side computing resources are scarce in many M2M transmissions, the use of deep learning based computer vision tasks is impossible without outsourcing the computations. Thus, the image/video is encoded to be transmitted with minimal bandwidth to an external server, where the computer vision task will be performed. In such context, the goal is to preserve the vision task accuracy under compression artifacts, while minimizing the transmitted information. Figure 1 shows an example of the described M2M scheme. Note that preserving CNNs performance is of greater importance than preserving perceptual quality here.

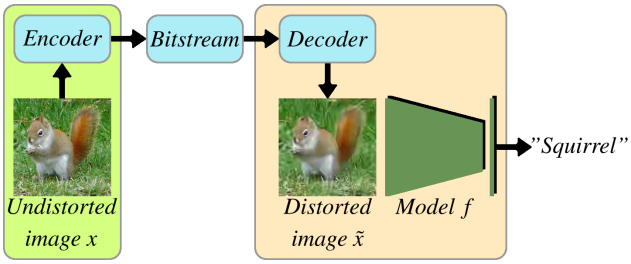


Figure 1. *The proposed Machine-to-Machine (M2M) scheme at inference. The goal is to outsource model f computation, while both minimizing quantity of data in transmitted bitstream and error rate of model f. We explicitly distinguish the camera side (left) and the vision task side (right) with colored frames.*

Artifacts in compressed data is the main source of CNNs accuracy drop at lower rates [6, 22, 4], and tackling this lack of resilience is of major importance for M2M transmissions. To this end, we propose a novel technique, namely expert training, that helps to enhance the resilience of CNNs to image coding artifacts. In a nutshell, expert training takes advantage of a regularization term added to the original loss function to enhance model resilience to distorted images. This regularization term gives an incentive to the trained model to minimize the distance between predictions on distorted images and undistorted images. Expert training has multiple advantages: (i) it is straightforward to implement. (ii) it does not increase the complexity for inference. (iii) it consistently brings a performance increase at equivalent quality.

Our work is presented as follows. The section *Related work* makes a quick overview of the literature. Section *Proposed expert training* presents the proposed training procedure and loss function. Our approach is then validated through experiments in section *Experiments*, followed by a conclusion.

Related work

Multiple ways of addressing the M2M problem have been proposed in the literature. One proposal is to adapt the compression level according to the content in images and videos. Following this idea, Galteri et al. [12] and Choi and Bajic. [2] use a saliency map to harshly compress areas without objectness, while Kong and Dai [19] propose a mode-decision to skip macroblocks with unnecessary temporal fluctuations in the background of a static surveillance camera. While these approaches may appear promising, the added complexity at the camera side reduces the outsourcing advantages. It is not clear whether these approaches are complementary or redundant with well known compression standards for M2M transmission.

Compression standards can also be used to reduce transmit-

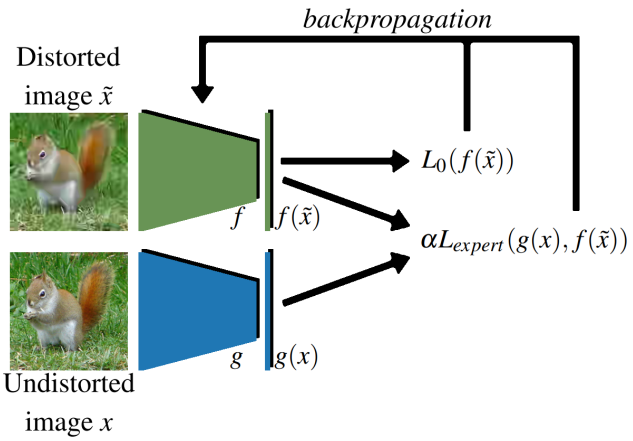


Figure 2. *Diagram of the proposed training procedure. Model f resilience to distorted images $\tilde{x}$ is enhanced through expert training using L_0 and L_{expert} loss functions. Model g weights are fixed, only model f weights are updated through backpropagation.*

ted information in the M2M context [7]. Cameras often integrate hardware implementation of such codecs, allowing a minimal increase in latency for real-time applications. One major drawback of using compression standards to reduce transmitted information in the M2M context is the impact on CNNs performances [9, 21]. Especially on unknown distortions at training time, CNNs performances are severely impacted when the quality of transmitted images is lowered by image compression artifacts [3, 6, 4, 27, 22, 10].

Many approaches have been proposed to overcome the lack of generalization of CNNs to image distortions. Zheng et al. [28] propose to use a stability training technique to strengthen CNNs robustness on small image perturbations. However, several papers [23, 16] put into question the robustness that stability training can bring with various experiments. In [23], authors add new nonlinear layers in existing CNNs architectures to increase their robustness to higher moment statistics shifts, such as skewness or kurtosis. Hendrycks et al. [16] perform rigorous benchmarks to evaluate ImageNet [5] classifiers to image corruptions and perturbations. As shown by the authors, from AlexNet [20] to ResNets [15], CNNs resilience to corruptions failed to improve, while accuracy on clean images went up. Techniques that successfully enhance CNNs robustness are also presented afterwards. Such techniques include the use of multiscale networks, histogram equalization, stylized ImageNet [13] and even an adversarial defense called Adversarial Logit Pairing (ALP) [17].

While improving CNNs generalization is a research field of major interest, existing methods still struggle to generalize to unseen distortions. Hence, we propose a method called expert training, that improves model accuracy on specific image coding artifacts.

Proposed expert training

Achieving high accuracies on a known distortion can be done by including artifacts from this distortion in the training set, at the cost of generalization ability [26]. Interest of such approach can be shown with the M2M context, where generalization could not be desired. Indeed, one could carefully choose a specific image quality that satisfies a bandwidth constraint, and train a CNN to be an expert of this known distortion.

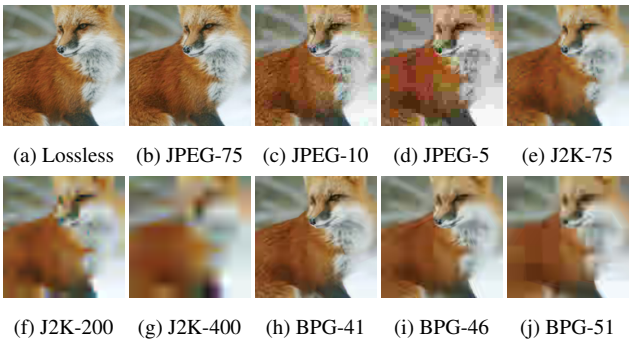


Figure 3. *Example of artifacts with considered distortions. Original image is from ImageNet [5] validation set.*

Our approach is inspired from the stability training technique [28]. In a nutshell, stability training wants CNNs to give consistent outputs for similar inputs. The purpose is to have more resilient CNNs not only to adversarial attacks [14], but also to acquisition or transmission noise. This is done by introducing a new stability term $L_{stability}$ to the loss function L_0, so that the trained model f must output similar results for original images x and slightly distorted images x'. The final loss function L defined in [28] is as follows:

$$L(x,x';\theta_f) = L_0(x;\theta_f) + \alpha L_{stability}(x,x';\theta_f) \tag{1}$$

$$L_{stability}(x,x';\theta_f) = D(f(x),f(x')) \tag{2}$$

where α controls the balance between the original loss L_0 and $L_{stability}$, θ_f represents weights of the trained model, and D is a distance function. Images x' are slightly distorted versions of images x, where Additive White Gaussian Noise (AWGN) was added. Stability training helps CNNs to be resilient to small image perturbations by ensuring that predictions on images x and x' are similar with the stability term $L_{stability}$. Stability training does not exactly address our problem, as we seek to train experts of a specific and harsh distortion. Additionally, stability training must keep good accuracy on undistorted images x, while a CNN that will handle only distorted images does not have this constraint.

In order to address the targeted issue, we propose a novel loss function, called expert training, defined as follows:

$$L(x,\tilde{x};\theta_f) = L_0(\tilde{x};\theta_f) + \alpha L_{expert}(x,\tilde{x};\theta_f) \tag{3}$$

$$L_{expert}(x,\tilde{x};\theta_f) = D(g(x),f(\tilde{x})) \tag{4}$$

where $\tilde{x}$ is distorted with the distortion we want to be robust against. Comparing with stability training, we have two distinct differences in the loss function definition itself. First, the original loss function L_0 is evaluated on $\tilde{x}$, not on x. Second, the model g is used in L_{expert} to get prediction on undistorted images x. The model g has the same architecture as f, but model weights θ_g are fixed. Both model weights θ_f and θ_g are initialized with a pretrained model on undistorted images. Since g is not trained, the model will stay specialized on undistorted images, while f will learn how to handle distortion in $\tilde{x}$ images through expert training.

Top-1 validation accuracy comparison at different level of distortions for 2 classifiers. Shown PSNR, SSIM [25] and rates values are an average on all 50000 images in ImageNet [5] validation set.

Codec	Q_{codec}	PSNR (dB)	SSIM	Rate/img	Pre-trained	Stab. training	Fine-tuning	Our	Gain
-	Lossless	$+\infty$	1.000	-	73.13 %	72.90 %	73.51 %	73.58 %	+0.07 %
JPEG	75	32.04	0.917	10899 B	70.05 %	68.40 %	72.70 %	72.71 %	+0.01 %
	10	25.27	0.738	3018 B	45.94 %	45.50 %	66.03 %	66.59 %	+0.56 %
	5	22.81	0.632	2170 B	21.64 %	23.71 %	59.80 %	60.59 %	+0.79 %
J2K	75	24.33	0.648	2003 B	31.02 %	27.86 %	61.80 %	62.58 %	+0.78 %
	200	21.19	0.499	761 B	9.62 %	8.30 %	47.21 %	47.85 %	+0.64 %
	400	18.59	0.399	390 B	1.77 %	1.61 %	26.56 %	27.01 %	+0.45 %
BPG	41	25.66	0.752	1795 B	50.77 %	51.70 %	66.50 %	67.00 %	+0.50 %
	46	24.32	0.679	870 B	35.33 %	36.14 %	60.99 %	61.39 %	+0.40 %
	51	22.96	0.597	400 B	17.58 %	18.70 %	51.73 %	52.35 %	+0.62 %

(a) MnasNet [24]

Codec	Q_{codec}	PSNR (dB)	SSIM	Rate/img	Pre-trained	Stab. training	Fine-tuning	Our	Gain
-	Lossless	$+\infty$	1.000	-	75.80 %	74.35 %	76.72 %	76.84 %	+0.12 %
JPEG	75	32.04	0.917	10899 B	73.74 %	70.21 %	75.79 %	75.86 %	+0.07 %
	10	25.27	0.738	3018 B	49.58 %	52.76 %	69.36 %	69.92 %	+0.56 %
	5	22.81	0.632	2170 B	18.48 %	30.60 %	62.44 %	63.13 %	+0.69 %
J2K	75	24.33	0.648	2003 B	39.33 %	36.11 %	65.00 %	65.79 %	+0.79 %
	200	21.19	0.499	761 B	14.45 %	13.10 %	48.33 %	48.84 %	+0.51 %
	400	18.59	0.399	390 B	2.17 %	2.12 %	25.75 %	26.27 %	+0.52 %
BPG	41	25.66	0.752	1795 B	57.89 %	52.93 %	69.94 %	70.59 %	+0.65 %
	46	24.32	0.679	870 B	42.98 %	40.80 %	64.20 %	64.83 %	+0.63 %
	51	22.96	0.597	400 B	21.33 %	24.29 %	53.56 %	54.31 %	+0.75 %

(b) ResNet50 [15]

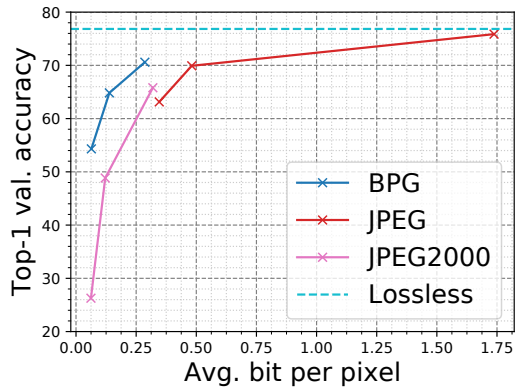


Figure 4. *Rate-distortion comparison for considered distortions on ResNet50 [15] classifier using expert training.*

The purpose of these two changes is to authorize the trained model f to forget how to handle full quality images, and to focus only on the distortion it must be robust against. A diagram of our proposed training procedure is presented in Figure 2 for the sake of clarity.

Note that our approach differs from fine-tuning because of the added expert term L_{expert}. While both approaches seek to have greater performances on a known distortion, experimental results show the superiority of CNNs trained with expert training in terms of accuracy.

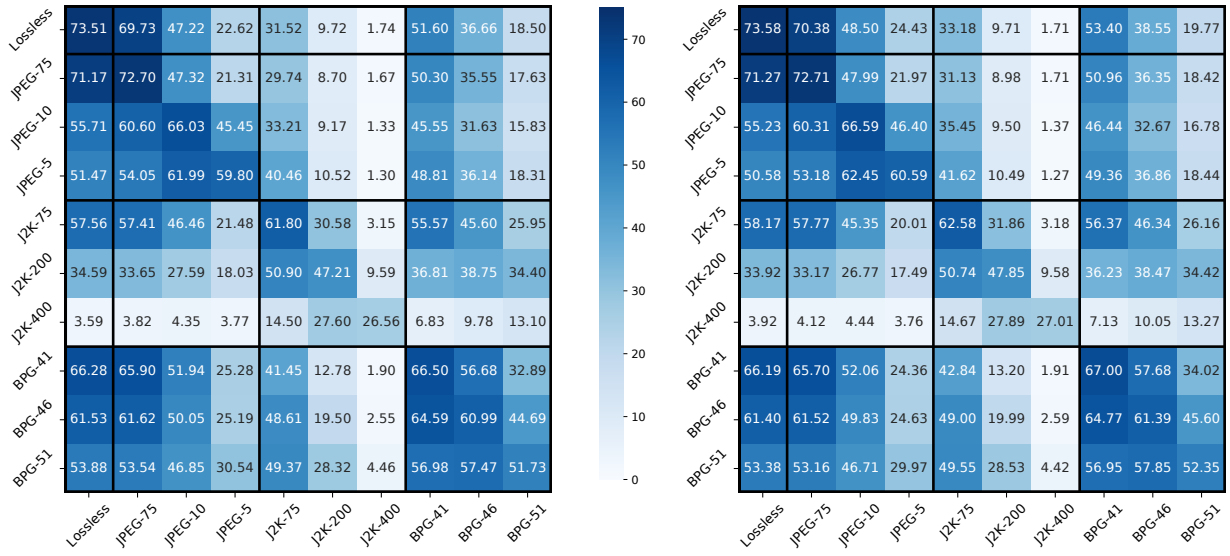
Another difference between stability training and expert training is the way the set of images x' and $\tilde{x}$ used for training are built. In stability training, Additive White Gaussian Noise (AWGN) is used as an unbiased distortion to create slightly distorted images x' out of x. In our case, $\tilde{x}$ is built by performing image compression with a compression algorithm and a specific compression strength, possibly resulting in strong distortions between $\tilde{x}$ and x. Note that $\tilde{x}$ may be obtained using any other distortion scheme, depending on the type of distortion we want to be robust against through expert training.

Experiments

Experimental setup

To assess the effectiveness of expert training, we consider the context of M2M image transmission for the classification task on ImageNet [5] dataset. For each considered distortion, we can compute images $\tilde{x}$ out of images x in ImageNet training and validation sets. This allows us to make a model more resilient to that specific distortion by using fine-tuning or expert training described in the previous section. To reduce the bandwidth in the context of M2M, the considered distortion algorithms are compression algorithms, such as JPEG, JPEG2000, and BPG [1]. For JPEG, qualities Q_{JPEG} of 75, 10, and 5 are used, where a quality of 100 represents the best quality and 1 denotes the lowest. For JPEG2000, a quality layer of 0 corresponds to lossless, and increasing this value lowers the quality of the distorted image. We choose to use quality layers Q_{J2K} of 75, 200 and 400. Finally, we use the quantization parameter Q_{BPG} of 41, 46 and 51 for BPG, where a bigger value corresponds to a lower quality. Some of the considered distortions artifacts are shown in Figure 3. While other parameters such as chroma subsampling or image downscaling could have a significant impact, we make the choice to not consider their impact in this study for clarity.

Images given as inputs to CNNs must match the architecture input size. It is important to note that undistorted images



(a) Fine-tuning (b) Expert training

Figure 5. *MnasNet [24] top-1 validation accuracy comparison among considered distortions. Each row refers to the considered distortion at training time to generated images $\tilde{x}$. Each column refer to the distortion on which the trained model is evaluated.*

x are already preprocessed with cropping and resizing, and that the compression to obtain distorted images $\tilde{x}$ is applied afterwards. Indeed, performing compression before cropping and resizing would be incoherent in a M2M context, because parts of the transmitted information would not be used at the end. Thus, since compressed images $\tilde{x}$ in our study are globally of smaller size compared to other studies [6, 4, 27], accuracies obtained at a given quality for a given codec are not comparable. Since we use MnasNet [24] and ResNet50 [15] architectures which require an input size of 224×224, all images x, x' and $\tilde{x}$ in our experiments have a size of 224×224.

The ADAM optimizer [18] ($\beta_1 = 0.5, \beta_2 = 0.99$) and an α of 10^{-2} are used for 45 epochs. We use a learning rate of 10^{-5} for all trainings on MnasNet. For ResNet50, we report the best accuracy obtained among learning rates of 10^{-5}, 5×10^{-6} and 10^{-6}. For stability training, we used for MnasNet a learning rate of 10^{-6}, $\alpha = 5 \times 10^{-3}$, $\sigma = 4 \times 10^{-2}$ for the AWGN standard deviation, while we used for ResNet50 a learning rate of 10^{-7}, $\alpha = 5 \times 10^{-3}$, and $\sigma = 5 \times 10^{-4}$. We also use a learning rate decay by dividing learning rate by 5 when the validation loss fails to improve for 3 consecutive epochs. Similarly to stability training, we use the KL-divergence as the distance function D in our expert term L_{expert} and the cross-entropy for L_0 since we are performing classification. For θ_f and θ_g initialization of MnasNet and ResNet50 architecture, we use the *Pytorch* implementation with a reported top-1 validation accuracy of 73.45% and 76.13%, respectively.

As shown by Geirhos et al. [13], convolution kernels are biased towards textural information, which is not pertinent on harshly compressed images where high-frequencies are generally discarded first. Thus, there is a need to re-train convolution layers. Therefore, all weights θ_f are updated with fine-tuning and expert training, while stability training updates only weights in fully connected layers.

Results

In table 1, we compare the proposed expert training against stability training and fine-tuning with respect to top-1 validation accuracy. For each distortion used, we show the average PSNR and SSIM [25] values for all images in the ImageNet validation set. We provide these values to give the reader an idea of applied distortions strength. When there are no distortions or only minor artifacts, expert training has no significant impact on model accuracy. This is reasonable since L_{expert} is almost 0. However, whenever the applied compression decrease the baseline model accuracy, our proposed expert training consistently outperforms the classic fine-tuning approach to regain some of the lost accuracy. We obtain a 0.79% at best and an average of 0.61% accuracy improvement over fine-tuning on lossy distortions, excluding JPEG-75. It can also be observed that, at equivalent quality, fine-tuning and expert training allow greater accuracy gains over pre-trained with JPEG compression. Models that are trained on ImageNet, a dataset only composed of JPEG images like most datasets, tend to be more resilient to JPEG artifacts, as opposed to artifacts from other codecs. Note that JPEG-75 almost reaches Lossless accuracy, since JPEG images in ImageNet must contain artifacts of a similar strength.

Figure 4 shows the rate-accuracy curve for the considered codecs. Note that training a different model on each distortion removes some bias in this experiment. On one hand, using pre-trained models [6, 10] allows to measure robustness to statistical shift between original and distorted dataset distributions. On the other hand, training a different model on each distortion allows to measure how well a given architecture can fit the distorted dataset distribution. Since weights θ_f are initialized with a pre-trained model on ImageNet through expert training, this experiment is in favor of JPEG. Nevertheless, we can still observe the superiority of codecs with better rate-distortion optimization such as BPG, which is using HEVC intra mode with a minimal header for images. Therefore, a better trade-off between model accuracy and

rate can be achieved at the cost of higher coding complexity, even when the observer is a machine and not a human.

We provide comparison with stability training [28], since expert training is inspired from it. It is unclear that stability training is beneficial to enhance a model resilience to unseen distortions, with experiments performed in this paper. As mentioned in Section *Related work*, this is coherent with other experiments performed on stability training [16, 23]. While accuracy gains using stability training can be observed on low JPEG and BPG qualities in table 1, reached accuracies are still far behind fine-tuning and expert training approaches. As explained in Section *Proposed expert training*, stability training strives for more generalization by not using $\tilde{x}$ in the training. Thus it may not be adapted to the context of M2M, where we have a priori knowledge on the distortion we want to be robust against.

Knowing the exact distortion during the training stage is not possible for some applications. Thus we also show the inter-codec performances using fine-tuning and expert training in Figure 5. We can observe that for both fine-tuning and expert training, model accuracy does not drop if there is a small quality change from Q to $\hat{Q}$ between training and evaluation, as long as the codec is the same. If training is performed on a set of images $\tilde{x}$, gains can be observed as long as the distortion used at evaluation is not too different from the one used at training. This intra-codec generalization can be observed between BPG-46 and BPG-51, or between J2K-75 and J2K-200 in Figure 5. More surprisingly, there is also an inter-codec generalization, for similar levels of distortion. This can be seen with BPG-46 and J2K-75 that generate samples $\tilde{x}$ of similar PSNR, where training on one codec and evaluating on the other one brings gains over not using any distortion. Inter-codec generalization with JPEG is less notable, probably because it is the only codec among the considered ones to introduce very strong blocking artifacts at lower rates. Figure 3 illustrates well similarity between BPG-51 and J2K-200, and also the different nature of JPEG artifacts at very low rates.

Overall, these experiments show that expert training consistently improves accuracy over fine-tuning for a large range of distortion strength, from lossless to extreme levels of distortion. Since using even higher level of distortion than the considered ones is not realistic, expert training brings improvements, regardless of the distortion strength.

Conclusion

In this paper, we proposed a general, easy-to-implement and lightweight novel technique, namely expert training, which focuses on improving the robustness of CNNs against known distortions, which may be valuable for the M2M transmission. The effectiveness of expert training has been validated here using three image codecs and two CNNs architectures on the classification task. In the future, the expert training will be further explored in other computer vision tasks.

References

[1] Fabrice Bellard. Better Portable Graphics, 2014.

[2] H. Choi and I. V. Bajic. High Efficiency Compression for Object Detection. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1792–1796, 2018.

[3] Nilaksh Das, Madhuri Shanbhogue, Shang-Tse Chen, Fred Hohman, Siwei Li, Li Chen, Michael E. Kounavis, and Duen Horng Chau. SHIELD: Fast, Practical Defense and Vaccination for Deep Learning Using JPEG Compression. In *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '18, pages 196–204, New York, NY, USA, 2018. Association for Computing Machinery. event-place: London, United Kingdom.

[4] Mathieu Dejean-Servières, Karol Desnos, Kamel Abdelouahab, Wassim Hamidouche, Luce Morin, and Maxime Pelcat. *Study of the impact of standard image compression techniques on performance of image classification with a convolutional neural network*. PhD Thesis, INSA Rennes; Univ Rennes; IETR; Institut Pascal, 2017.

[5] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.

[6] S. Dodge and L. Karam. Understanding how image quality affects deep neural networks. In *2016 Eighth International Conference on Quality of Multimedia Experience (QoMEX)*, pages 1–6, 2016.

[7] Lingyu Duan, Jiaying Liu, Wenhan Yang, Tiejun Huang, and Wen Gao. Video Coding for Machines: A Paradigm of Collaborative Compression and Intelligent Analytics. *IEEE Transactions on Image Processing*, 29:8680–8695, 2020.

[8] Maxime Efoui-Hess. Climate crisis: The unsustainable use of online video. *The Shift Project: Paris, France*, 2019.

[9] K. Fischer, C. Herglotz, and A. Kaup. On Intra Video Coding And In-Loop Filtering For Neural Object Detection Networks. In *2020 IEEE International Conference on Image Processing (ICIP)*, pages 1147–1151, 2020.

[10] Kristian Fischer, Fabian Brand, Christian Herglotz, and André Kaup. Video Coding for Machines with Feature-Based Rate-Distortion Optimization. In *2020 IEEE 22nd International Workshop on Multimedia Signal Processing (MMSP)*, pages 1–6, 2020.

[11] GMDT Forecast. Cisco visual networking index: global mobile data traffic forecast update, 2017–2022. *Update*, 2017:2022, 2019.

[12] L. Galteri, M. Bertini, L. Seidenari, and A. Del Bimbo. Video Compression for Object Detection Algorithms. In *2018 24th International Conference on Pattern Recognition (ICPR)*, pages 3007–3012, 2018.

[13] Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A Wichmann, and Wieland Brendel. ImageNet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness. *arXiv preprint arXiv:1811.12231*, 2018.

[14] Ian Goodfellow, Jonathon Shlens, and Christian Szegedy. Explaining and Harnessing Adversarial Examples. In *International Conference on Learning Representations*, 2015.

[15] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016.

[16] Dan Hendrycks and Thomas G. Dietterich. Benchmarking Neural Network Robustness to Common Corruptions and Perturbations. *CoRR*, abs/1903.12261, 2019. _eprint: 1903.12261.

[17] Harini Kannan, Alexey Kurakin, and Ian J. Goodfellow. Adversarial Logit Pairing. *CoRR*, abs/1803.06373, 2018. _eprint: 1803.06373.

[18] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceed-*

ings, 2015.

[19] L. Kong and R. Dai. Object-Detection-Based Video Compression for Wireless Surveillance Systems. *IEEE MultiMedia*, 24(2):76–85, 2017.

[20] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. ImageNet Classification with Deep Convolutional Neural Networks. In *Proceedings of the 25th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'12, pages 1097–1105, Red Hook, NY, USA, 2012. Curran Associates Inc. event-place: Lake Tahoe, Nevada.

[21] J. Löhdefink, A. Bär, N. M. Schmidt, F. Hüger, P. Schlicht, and T. Fingscheidt. On Low-Bitrate Image Compression for Distributed Automotive Perception: Higher Peak SNR Does Not Mean Better Semantic Segmentation. In *2019 IEEE Intelligent Vehicles Symposium (IV)*, pages 424–431, 2019.

[22] Prasun Roy, Subhankar Ghosh, Saumik Bhattacharya, and Umapada Pal. Effects of Degradations on Deep Neural Network Architectures. *CoRR*, abs/1807.10108, 2018. _eprint: 1807.10108.

[23] Z. Sun, M. Ozay, Y. Zhang, X. Liu, and T. Okatani. Feature Quantization for Defending Against Distortion of Images. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7957–7966, 2018.

[24] M. Tan, B. Chen, R. Pang, V. Vasudevan, M. Sandler, A. Howard, and Q. V. Le. MnasNet: Platform-Aware Neural Architecture Search for Mobile. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2815–2823, 2019.

[25] Zhou Wang, A.C. Bovik, H.R. Sheikh, and E.P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004.

[26] D. H. Wolpert and W. G. Macready. No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1(1):67–82, 1997.

[27] Farhad Ghazvinian Zanjani, Svitlana Zinger, Bastian Piepers, Saeed Mahmoudpour, Peter Schelkens, and others. Impact of JPEG 2000 compression on deep convolutional neural networks for metastatic cancer detection in histopathological images. *Journal of Medical Imaging*, 6(2):027501, 2019. Publisher: International Society for Optics and Photonics.

[28] S. Zheng, Y. Song, T. Leung, and I. Goodfellow. Improving the Robustness of Deep Neural Networks via Stability Training. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4480–4488, 2016.

Author Biography

Alban Marie is a Ph.D. student in the Institute of Electronics and Telecommunications of Rennes (IETR) laboratory of Rennes. He received his M.S. degree from Rennes Superior School of Engineering (ESIR) at Rennes 1 University in 2020. Prior to his thesis, he published his first conference paper at the International Conference on Acoustics, Speech and Signal Processing (ICASSP), where he proposed a motion estimation algorithm for on-the-sphere compression of omnidirectional videos.

Karol Desnos is an associate professor at the National Institute of Applied Sciences (INSA) of Rennes and a researcher at the IETR in the VAADER team. His research interests focus on dataflow models of computation and associated implementation techniques for the rapid prototyping of applications running on heterogeneous MPSoCs. He is leading the development of the open-source GEGELATI library, a lightweight framework for training and executing artificial intelligences based on tangled program graphs.

Luce Morin is Full-Professor at the Electrical and Computer Engineering Department in the National Institute of Applied Sciences (INSA) and a researcher at the Institute of Electronics and Telecommunications of Rennes (IETR). She leads the VAADER team in the IETR laboratory. She has published more than 70 scientific papers in international journals and conferences. Her research activities deal with computer vision, 3D reconstruction, image and video compression, and representations for 3D videos and multiview videos.

Lu Zhang is an associate professor at INSA Rennes in France. She received the B.S degree from Southeast University and the M.S. degree from Shanghai Jiaotong University in China in 2004 and 2007, respectively. From 2009 to 2012, she was a PhD student of the LISA and CNRS IRCCyN labs. She is a board member of VQEG (Video Quality Experts Group). She works on human perception understanding, image analysis and coding, and image quality assessment.