Data Augmentation for JPEG Steganalysis

Tomer Itzhaki
Department of ECE
Binghamton University
titzhak1@binghamton.edu

Yassine Yousfi
Department of ECE
Binghamton University
yyousfi1@binghamton.edu

Jessica Fridrich
Department of ECE
Binghamton University
fridrich@binghamton.edu

***Abstract*—Deep Convolutional Neural Networks (CNNs) have performed remarkably well in JPEG steganalysis. However, they heavily rely on large datasets to avoid overfitting. Data augmentation is a popular technique to inflate the datasets available without collecting new images. For JPEG steganalysis, the augmentations predominantly used by researchers are limited to rotations and flips (D4 augmentations). This is due to the fact that the stego signal is erased by most augmentations used in computer vision. In this paper, we systematically survey a large number of other augmentation techniques and assess their benefit in JPEG steganalysis.**

***Index Terms*—Steganography, steganalysis, convolutional neural network, data augmentation**

I. Introduction

Convolutional Neural Networks (CNNs) are the superior detectors of steganography today [1]. They replace the so-called rich media models, which are high-dimensional feature representations hand-designed for specific purposes in steganalysis as a well as the related field of digital forensics. In contrast to rich models, CNNs learn the best (internal) image representation as well as the detector itself via a training process, which is usually a form of a Stochastic Gradient Descent (SGD).

Data augmentation is a way to increase the training set size by including in training transformed versions of the images. Typical augmentations used in computer vision are rotations, resizing, cropping, channel shuffle, dropout, and in general any transformation that fundamentally preserves the label assigned to the image. Larger training sets usually lead to better detectors / classifiers because they are exposed to more diverse content. Augmentations can be domain-specific, depending on what task the CNN is trained on. For steganalysis, the signal of interest is rather fragile, formed by slight perturbations of cover image pixels or quantized DCT coefficients (for JPEG images). Thus, augmentations that remove or suppress this signal are undesirable and cannot improve the detection performance. Steganalysts typically employ the so-called dihedral D4 augmentation, which consists of rotations by integer multiples of 90 degrees and mirrorrings. Indeed, such transformations do not disturb the stego signal while exposing the network to a more diverse dataset. On the other hand, resizing and rotations by non-integer multiples of 90 degrees are not desirable as the resampling that is inherently part of these transformations disturbs the stego signal to a large degree.

WIFS'2021, December 7-10, 2021, Montpellier, France. 978-1-7281-9930-6/20/$31.00

Previous work in steganalysis [2], [3] addressed the need for an increased dataset size by acquiring more images using similar devices or using other datasets that are close in development to the test dataset. Not only this solution does not follow our definition of data augmenting (i.e. not requiring acquisition of new images), but it is unclear how one would replicate a cover source as noted by the winners of the BOSS competition who failed to duplicate the test set's cover source even when knowing the camera model and the development script used [4]. Other steganalysis augmentation techniques have been introduced, such as BitMix [5] (Section III-C) and Pixels-off [6]. The latter was not included in this study because it was not developed for the JPEG domain.

In this paper, we look beyond the usual random D4 augmentation group in search for new augmentations that can improve the detector performance for steganalysis of digital images. We use the Albumentations Library [7] as well as custom augmentations specifically designed for steganalysis. In particular, we take a look at various forms of drop out augmentations, which make good sense for computer vision tasks because they simulate "occlusions" that may naturally occur and thus robustify the classifier. We also study color channel shuffle, bitmix, convex combinations of cover and stego images (with soft labels), and multiple stego image sampling to expose the network to multiple versions of the stego image embedded with different stego keys.

Interestingly, the idea to use symmetries of natural images to robustify the detector is already present in rich models [8], [9]. Their "features" are formed by co-occurrences of adjacent quantized and truncated noise residuals obtained via pixel predictors. These features are typically "symmetrized" or robustified by leveraging directional and sign symmetries of natural images. In particular, co-occurrences computed from the original image as well as their versions rotated by integer multiples of 90 degrees and their mirrored forms were typically added to one, better populated co-occurrence matrix (feature). Because noise residuals exhibit symmetrical marginals centered around zero, additional co-occurrences can be added by flipping the signs of the noise residuals, a process that required some caution when applied to the so-called "min" and "max" non-linear residuals in the Spatial Rich Model (SRM) [8].

In Section II, we describe the setup of all our experiments,

the datasets, and performance measures to evaluate the effectiveness of various augmentations. All tested augmentations are explained in Section III. Section IV-A shows all experimental results in a graphic form together with a discussion. Finally, the paper is closed in Section V.

II. Experimental Setting

A. Datasets

We use the ALASKA II 256×256 dataset [10] which contains 75,000 different cover images compressed with quality factors 75 and 95. The covers were randomly divided into three sets with 66,000, 3,000, and 6,000 images, for training, validation, and testing, respectively. The images were embedded using J-UNIWARD [11], J-MiPOD [12], and F5 [13] with payloads 0.5 , 0.4, 0.3, 0.2, and 0.1 bpnzac. For J-UNIWARD, the payload was spread into the chrominance channels using Color Channels Merging (CCM), which concatenates the color cost maps before minimizing the additive distortion. For J-MiPOD and F5, we only embedded the payload in the luminance channel.

B. Detectors

We use the EfficientNet B3 [14] pre-trained on ImageNet [15] and refined for JPEG domain steganalysis [16], [17] with the training hyper-parameters described in Section 4.2 in [17]. No modifications were done to the architecture besides changing the Fully Connected (FC) layer.

We use the following three performance measures to compare detectors: $P_{\text{E}} = \min(P_{\text{D}}(P_{\text{FA}}) + P_{\text{FA}})$, wAUC [10], MD5 $= P_{\text{MD}}(P_{\text{FA}} = 0.05)$, and FA80 $= P_{\text{FA}}(P_{\text{MD}} = 0.8)$.

III. Augmentations

In this section, we describe all augmentation techniques surveyed in this paper.

A. Dropout augmentations

Dropout style augmentations simulate occlusions.

a) CoarseDropout: CoarseDropout is a dropout augmentation that randomly zeros out rectangular regions of the image. It evolved from the cutout augmentation [18], which drops a single square region. The location of the dropped regions (holes) is randomized, while their size is set to 8×8 and their number to 32 holes. Figure 1 shows an example of a CoarseDropout augmented image. Note that the holes can overlap as well as not completely fit in the image. They also do not respect the 8×8 grid of JPEG blocks.

b) GridDropout: GridDropout [19] is another dropout augmentation that drops out rectangular regions of an image in a grid fashion. The grid shape is a hyper-parameter of the augmentation. We set the grid to correspond to JPEG 8×8 squares, and vary the dropout ratio parameter which controls the number of dropped blocks, the number of dropped blocks was set to 36. Figure 2 shows an example of a GridDropout augmented image.

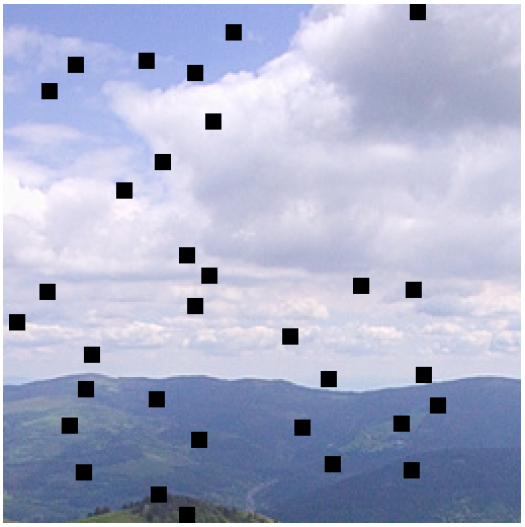

Figure 1. Image '06285.jpg' from ALASKA II augmented using CoarseDropout with 32 holes of 8×8 pixels.

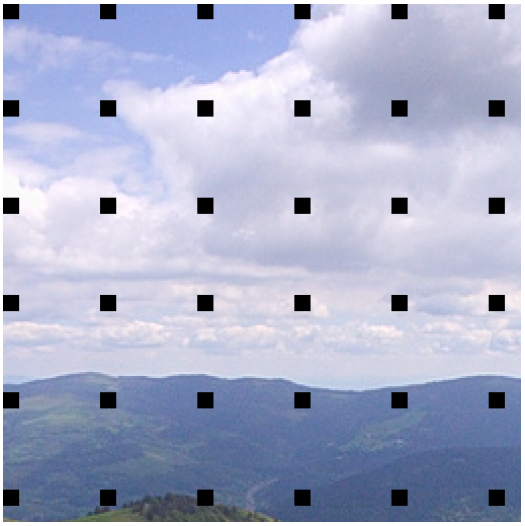

Figure 2. Image '06285.jpg' from ALASKA II augmented using GridDropout with 36 dropped holes in a grid.

c) RandomGridDropout: This augmentation combines the GridDropout and CoarseDropout. It drops a number of non overlapping 8×8 squares while respecting the 8×8 JPEG grid; the number of holes was also set to 32. Figure 3 shows an example of a RandomGridDropout augmented image.

B. Channel augmentations

This section describes augmentations operating on the channel dimension of the input image. Such augmentations are only useful for color steganography.

a) ChannelShuffle: This is a channel-style augmentation that randomizes the order of channels in a color image. For example, an RGB image could become a GBR image. Note that this augmentation is only used when training on RGB in-

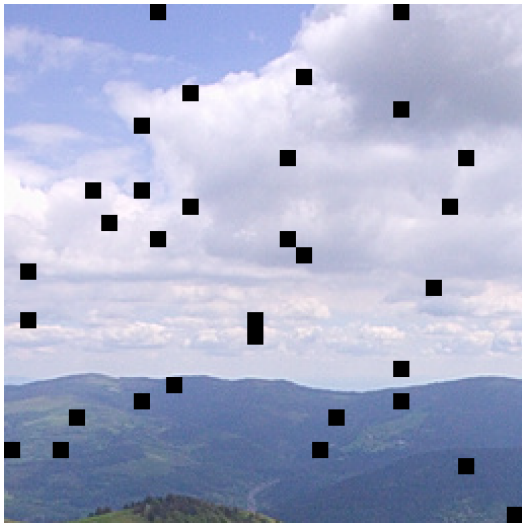

Figure 3. Image '06285.jpg' from ALASKA II augmented using RandomGridDropout with 32 dropped holes.

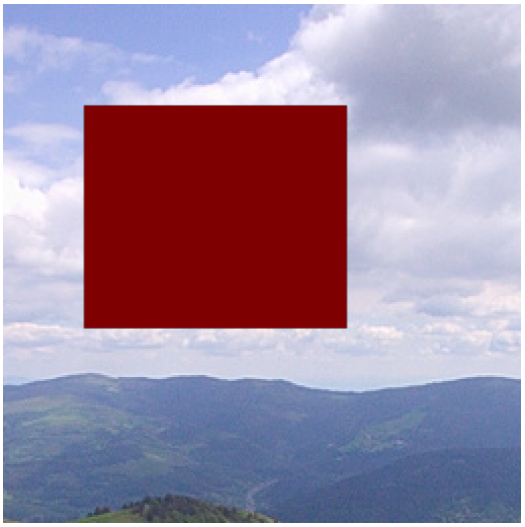

Figure 4. Image '06285.jpg' from ALASKA II augmented using BitMix. For visualization, the cover region is overlayed with red coloring while the rest is a stego region embedded with J-UNIWARD 0.4 bpnzac. The augmented image has a label $y = (0.1451, 0.8549)$.

puts since swapping the channels in the YC_bC_r representation is detrimental because of the heterogeneity of these channels.

b) ToGray: ToGray converts the sampled image to grayscale. This augmentation does not completely destroy the stego signal since a vast majority of the payload is typically in the luminance channel. This augmentation was also used with networks trained with RGB inputs.

C. Mixing augmentations

Next, we describe augmentations which mix two images, a cover image C and a stego image S, to create an augmented image X. Such augmentations evolved from the Mixup augmentation [20], which saw a great success in computer vision applications. These augmentations often require changing the label vector to reflect the amount of mixing between the classes. The loss used is the cross-entropy with soft targets.

a) BitMix: BitMix [5] takes a cover image and replaces a randomly sampled patch with the stego image and vice versa. This patch is chosen by randomly sampling a rectangular area whose maximum size is determined by a maximum mix ratio parameter. The patch is represented using a binary mask M. For simplicity, we assume the mask is applied to a cover image C but in practice it is applied to cover and stego images. The label vector y is changed using the system of Equations 1–3:

$$X = M \odot C + (1 - M) \odot S \quad (1)$$

$$\lambda = \frac{\|M \odot C - M \odot S\|_1}{\|C - S\|_1} \quad (2)$$

$$y_X = (\lambda, 1 - \lambda). \quad (3)$$

Figure 4 shows an example of a BitMix augmented image as well as its corresponding soft label.

b) ConvexMix: This augmentation forms a convex combination of a cover-stego image pair by sampling the mixing parameter $\lambda \sim U(0, 1)$:

$$X = \lambda C + (1 - \lambda)S \quad (4)$$

$$y_X = (\lambda, 1 - \lambda). \quad (5)$$

D. Sampling augmentations

a) StegoSampling: This augmentation is special to the steganalysis task. In fact, each stego image S is a random sample from the steganographic simulator, which simulates embedding changes operating on the rate–distortion bound. This enables sampling different stego images from the same cover and with the same payload while getting different stego samples at each iteration. In practice, we use the change rates β^+, β^- and the cover image to sample a stego image on the fly while training the network. This augmentation will be called StegoSampling.

IV. Results

A. Successful augmentations

Figure 5 shows the results for all tested augmentations, three embedding algorithms, two payloads, and two quality factors. The baseline detectors were trained with the standard D4 augmentation. For J-UNIWARD, every tested augmentation was successful in improving on the baseline. For larger payloads, the StegoSampling and dropout augmentations performed very well. This is clear for QF75 and even more so for QF95. For smaller payloads, StegoSampling and dropout augmentations are still very capable but this changes for the larger quality 95. Despite the drop in performance from QF75, there are still meaningful improvements for small payloads at QF95. The results suggest that detection of steganography in images

Table I
BASELINE PERFORMANCE AND COARSEDROPOUT FOR EFFICIENTNET B3 TRAINED ON 10,000 PAIRS OF COVER AND J-UNIWARD IMAGES, QF75 AT 0.4 BPNZAC.

Data Augmentation	Accuracy	MD5	FA80	wAUC
	QF75 J-UNIWARD 0.4 bpnzac			
Baseline	0.8881	0.1701	0.0335	0.9797
CoarseDropout, 16 blocks	0.9029	0.1488	0.0293	0.9812

with low QFs will benefit from the assistance of augmentations the most, whereas there is a limit on how helpful the augmentations can be for higher QFs images as the payload size decreases.

For J-MiPOD, the additional tested augmentations were less impactful. The one that stood out was ConvexMix, which helped detection of J-MiPOD much more than J-UNIWARD. Furthermore, at QF75 the RandGridDropout and GridDropout augmentations actually hurt the testing accuracy. Even when the augmentations provide improvements, the effectiveness of additional augmentations falls of much more quickly in comparison to J-UNIWARD. As the payload size decreases and the QF increases, the benefits of augmentations in detecting images embedded with J-MiPOD become less pronounced.

For the F5 algorithm, the 0.1 bpnzac payload was different compared to the lowest payloads for J-MiPOD and J-UNIWARD. Every augmentation except for the RandGridDropout augmentation resulted in worse accuracies for both QFs. For the 0.3 bpnzac payload, every augmentation was able to provide a slight improvement upon the baseline. Moving from QF75 to QF95 seemed to produce nearly equivalent results. For F5, the payload influenced the amount of improvement the most.

B. Low data regime

Another experiment that we ran was taking an augmentation and testing it against a smaller dataset. This was done using the CoarseDropout augmentation with the results shown in Table I. The settings were identical to those described in Section II except for the training data set size, which was reduced from 66,000 to 10,000. The original number of 32 dropout blocks used for CoarseDropout as described in Section III-A did not work with this smaller dataset. However, when the number of blocks was halved there was a substantial gain in accuracy and MD5. This suggests that smaller datasets are more delicate but still benefit from toned down version of the augmentations used in our experiments.

C. Unsuccessful augmentations

Here, we report on the augmentations that failed to improve upon the baseline. The channel augmentations for color images (ChannelShuffle and ToGray) failed to give better results than the baseline as shown in Table II.

Additionally, we tried to combine all augmentations that produced a gain to see if their combined effect would provide further benefit. Surprisingly, this was not the case as shown in Table III. The augmentations were combined using a "OneOf" strategy: for each sample, a randomly sampled augmentation technique was applied.

Table II
CHANNELSHUFFLE AND TOGRAY PERFORMANCE FOR J-UNIWARD QF75 AND 95 AT 0.4 BPNZAC.

Data Augmentation	Accuracy	MD5	FA80	wAUC
	QF75 J-UNIWARD 0.4 bpnzac			
Baseline	0.9571	0.0308	0.0012	0.9961
ChannelShuffle	0.9509	0.0297	0.0018	0.9961
ToGray	0.9515	0.0272	0.0015	0.9962
	QF95 J-UNIWARD 0.4 bpnzac			
Baseline	0.8308	0.3264	0.1300	0.9498
ChannelShuffle	0.8194	0.3571	0.1538	0.9459
ToGray	0.8292	0.3212	0.1366	0.9491

Table III
BASELINE, BEST SINGLE AUGMENTATION, AND ALL AUGMENTATIONS PERFORMANCE FOR J-UNIWARD AND J-MIPOD AT QF75 AND 95.

Data Augmentation	Accuracy	MD5	FA80	wAUC
	QF75 J-UNIWARD 0.4 bpnzac			
Baseline	0.9571	0.0308	0.0012	0.9961
GridDropout	0.9669	0.0173	0.0012	0.9974
All	0.9603	0.0155	0.0020	0.9973
	QF95 J-UNIWARD 0.4 bpnzac			
Baseline	0.8309	0.3264	0.1300	0.9498
GridDropout	0.8490	0.2935	0.1044	0.9562
All	0.8398	0.3128	0.1200	0.9531
	QF75 J-MiPOD 0.5 bpnzac			
Baseline	0.9128	0.1243	0.0166	0.9854
ConvexMix	0.9180	0.1215	0.0178	0.9853
All	0.9193	0.1156	0.0173	0.9863
	QF95 J-MiPOD 0.5 bpnzac			
Baseline	0.7132	0.5946	0.3706	0.8662
CoarseDropout	0.7186	0.5879	0.3724	0.8679
All	0.7164	0.5917	0.3641	0.8680

V. CONCLUSIONS AND FUTURE WORK

Our study of augmentations shows that there are ways to successfully augment data for steganographic deep learning applications beyond the standard D4 augmentations. With some care, the correct selection of additional augmentations can result in a substantial boost in performance (up to 3% in accuracy and 5% in MD5). We observed that smaller data sets are more likely to benefit from using the proposed data augmentations than large datasets because the augmentations effectively increase the size of the training set and make deep learning models more robust.

For possible future directions, we recommend further investigation of the effect of the cover or stego source on the gains of each augmentation. For example, one could study a more diverse stego source comprised of different stego schemes with different payloads. Additionally, it is probably worth looking at how much the augmentations boost deeper CNN architectures, such as the EfficientNet B7.

ACKNOWLEDGMENT

This work was supported by NSF grant No. 2028119.

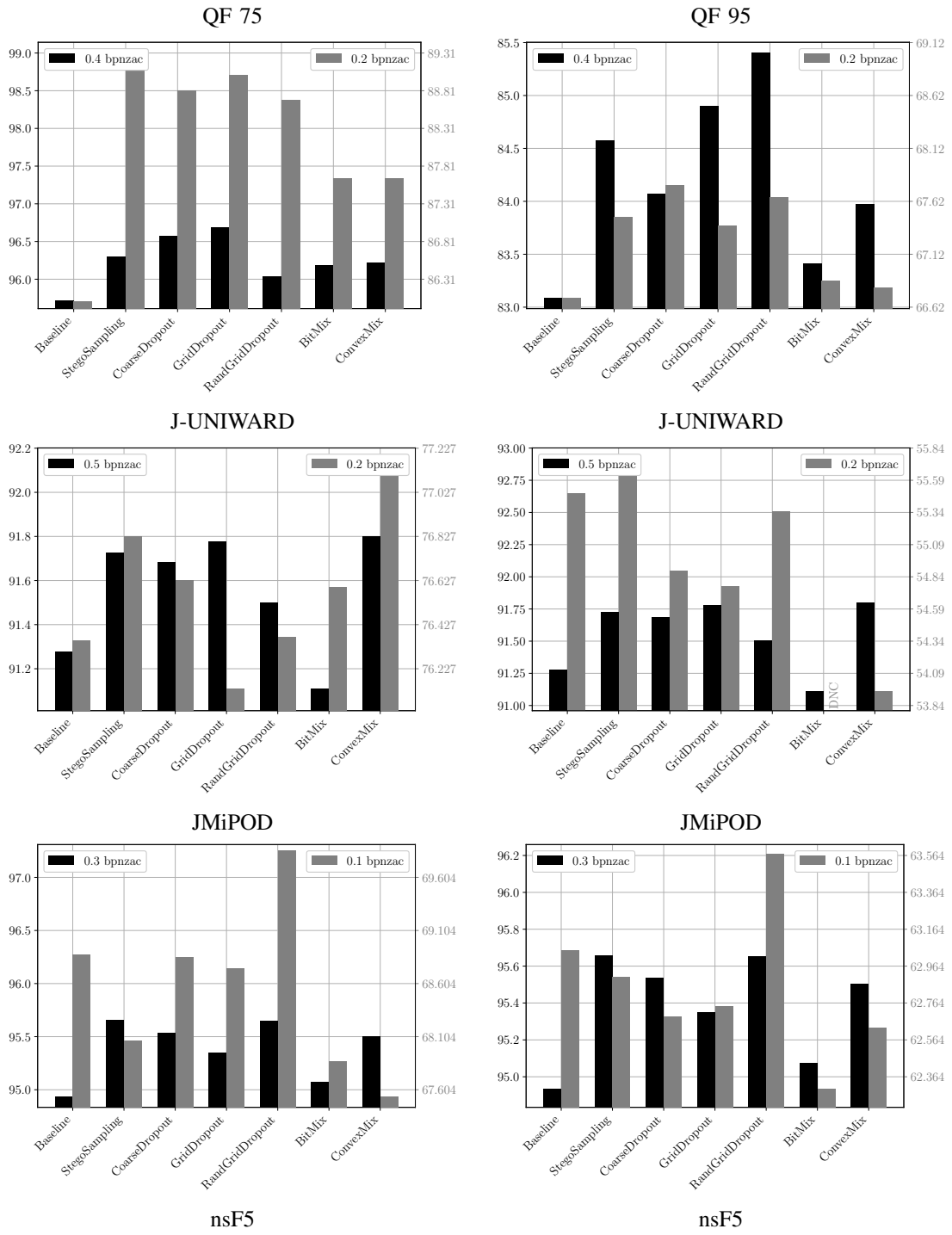

Figure 5. Accuracy of EfficientNet B3 trained with various augmentations vs. baseline for J-UNIWARD, J-MiPOD, and nsF5 for two payloads (left and right y-axis) and two quality factors. DNC stands for Did Not Converge.

REFERENCES

[1] M. Chaumont, "Deep learning in steganography and steganalysis," in *Digital Media Steganography: Principles, Algorithms, Advances* (M. Hassaballah, ed.), ch. 14, pp. 321–349, Elsevier, 2020.

[2] J. Ye, J. Ni, and Y. Yi, "Deep learning hierarchical representations for image steganalysis," *IEEE Transactions on Information Forensics and Security*, vol. 12, pp. 2545–2557, November 2017.

[3] M. Yedroudj, M. Chaumont, and F. Comby, "How to augment a small learning set for improving the performances of a CNN-based steganalyzer?," in *Proceedings IS&T, Electronic Imaging, Media Watermarking, Security, and Forensics 2018* (A. Alattar and N. D. Memon, eds.), (San Francisco, CA), January 29–February 1, 2018.

[4] J. Fridrich, J. Kodovský, M. Goljan, and V. Holub, "Breaking HUGO – the process discovery," in *Information Hiding, 13th International Conference* (T. Filler, T. Pevný, A. Ker, and S. Craver, eds.), Lecture Notes in Computer Science, (Prague, Czech Republic), pp. 85–101, May 18–20, 2011.

[5] I.-J. Yu, W. Ahn, S.-H. Nam, and H. Lee, "Bitmix: data augmentation for image steganalysis," *Electronics Letters*, vol. 56, pp. 1311–1314, November 2020.

[6] M. Yedroudj, M. Chaumont, F. Comby, A. Oulad Amara, and P. Bas, "Pixels-off: Data-augmentation complementary solution for deep-learning steganalysis," ACM Press, 2020.

[7] A. Buslaev, V. I. Iglovikov, E. Khvedchenya, A. Parinov, M. Druzhinin, and A. A. Kalinin, "Albumentations: Fast and flexible image augmentations," *Information*, vol. 11, no. 2, 2020.

[8] J. Fridrich and J. Kodovský, "Rich models for steganalysis of digital images," *IEEE Transactions on Information Forensics and Security*, vol. 7, pp. 868–882, June 2011.

[9] J. Kodovský and J. Fridrich, "Steganalysis of JPEG images using rich models," in *Proceedings SPIE, Electronic Imaging, Media Watermarking, Security, and Forensics 2012* (A. Alattar, N. D. Memon, and E. J. Delp, eds.), vol. 8303, (San Francisco, CA), pp. 0A 1–13, January 23–26, 2012.

[10] R. Cogranne, Q. Giboulot, and P. Bas, "ALASKA#2: Challenging academic research on steganalysis with realistic images," in *IEEE International Workshop on Information Forensics and Security*, (Held virtually), December 6–11, 2020.

[11] V. Holub, J. Fridrich, and T. Denemark, "Universal distortion design for steganography in an arbitrary domain," *EURASIP Journal on Information Security, Special Issue on Revised Selected Papers of the 1st ACM IH and MMS Workshop*, vol. 2014:1, 2014.

[12] R. Cogranne, Q. Giboulot, and P. Bas, "Steganography by minimizing statistical detectability: The cases of JPEG and color images," in *The 8th ACM Workshop on Information Hiding and Multimedia Security* (C. Riess and F. Schirrmacher, eds.), (Held virtually), ACM Press, 2020.

[13] J. Fridrich, T. Pevný, and J. Kodovský, "Statistically undetectable JPEG steganography: Dead ends, challenges, and opportunities," in *Proceedings of the 9th ACM Multimedia & Security Workshop* (J. Dittmann and J. Fridrich, eds.), (Dallas, TX), pp. 3–14, September 20–21, 2007.

[14] T. Mingxing and V. L. Quoc, "EfficientNet: Rethinking model scaling for convolutional neural networks," in *Proceedings of the 36th International Conference on Machine Learning*, vol. 97, pp. 6105–6114, June 9–15, 2019.

[15] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "ImageNet: A large-scale hierarchical image database," in *2009 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 248–255, June 20–25, 2009.

[16] Y. Yousfi, J. Butora, E. Khvedchenya, and J. Fridrich, "ImageNet pre-trained CNNs for JPEG steganalysis," in *IEEE International Workshop on Information Forensics and Security*, (Held virtually), December 6–11, 2020.

[17] Y. Yousfi, J. Butora, J. Fridrich, and C. Fuji Tsang, "Improving efficient-net for JPEG steganalysis," in *The 9th ACM Workshop on Information Hiding and Multimedia Security* (D. Borghys and P. Bas, eds.), (Held virtually), ACM Press, June 22–25, 2021.

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

[19] P. Chen, S. Liu, H. Zhao, and J. Jia, "Gridmask data augmentation," *arXiv preprint arXiv:2001.04086*, 2020.

[20] H. Zhang, M. Cisse, Y. N. Dauphin, and D. Lopez-Paz, "Mixup: Beyond empirical risk minimization," in *International Conference on Learning Representations*, 2018.