JPEG Steganalysis Detectors Scalable With Respect to Compression Quality

Yassine Yousfi and Jessica Fridrich, Department of ECE, SUNY Binghamton, NY, USA {yyousfi1, fridrich}@binghamton.edu

Abstract

Practical steganalysis inevitably involves the necessity to deal with a diverse cover source. In the JPEG domain, one key element of the diversification is the JPEG quality factor, or, more generally, the JPEG quantization table used for compression. This paper investigates experimentally the scalability of various steganalysis detectors w.r.t. JPEG quality. In particular, we report that CNN detectors as well as older feature-based detectors have the capacity to contain the complexity of multiple JPEG quality factors within a single model when the quality factors are properly grouped based on their quantization tables. Detectors trained on multiple JPEG qualities show no loss of detection accuracy when compared with dedicated detectors trained for a specific JPEG quality factor. We also demonstrate that CNNs (but not so much feature-based classifiers) trained on multiple qualities can generalize to unseen custom quantization tables compared to detectors trained for specific JPEG qualities. Their ability to generalize to very different quantization tables, however, remains a challenging task. A semi-metric comparing quantization tables is introduced and used to interpret our results.

Introduction

The ALASKA stegnalysis challenge [9] revealed how time and resource demanding it is to train deep learning steganalysis detectors for the "real world." A large part of this complexity is due to training detectors for each JPEG quality factor as done by the winners of the challenge [24]. This approach is not only fastidious but also not scalable to cover a potentially large number of custom quantization tables. Since deep learning architectures have shown markedly better performance than classifiers trained on hand crafted feature sets, in this paper we explore the topic of building steganalysis detectors that would cover a wider range of quantization tables to alleviate the computational and complexity burden associated with having to train a separate detector for each quantization table.

This paper starts by laying out preliminary definitions and notation, discussing relevant prior art and describing datasets used, steganographic schemes employed, and steganalysis tools evaluated. In Section "Scalability w.r.t. JPEG quality," we provide experimental evidence that CNN detectors as well as older feature-based detectors are scalable w.r.t. JPEG quality; quantitative comparison with dedicated detectors is given. In section "Robustness w.r.t. custom quantization tables," we look at the problem of mismatched JPEG quantization tables and show that CNN detectors trained on a range of quality factors do generalize to slightly different custom tables within the same range when measured with a semi-metric that we introduce for this purpose. In contrast, feature-based classifiers trained on the same range of qualities appear to experience a much larger loss. Generalizing to markedly different tables remains a challenge for both types of detectors. The paper is concluded in the last section.

Preliminaries

JPEG stands for the Joint Photographic Experts Group that published a format standard in 1992. A detailed description of the format can be found in [21]. In this paper, we only consider grayscale images represented with the luminance component Y quantized by rounding the DCT coefficients divided by quantization steps to the nearest integer.[1]

During JPEG compression, quantization of DCT coefficients is performed on 8×8 blocks using 8×8 quantization tables $\mathbf{q}$. Because the JPEG standard allows arbitrary quantization tables to be used, as long as they are stored in the header of the JPEG file, engineers and camera makers are free to create their own. The JPEG standard recommends a set of quantization matrices parametrized by a quality factor $Q \in \{1,2,...,100\}$:

$$\mathbf{q}(Q) = \begin{cases} \max\{\mathbf{1}, \text{round}(2(1-Q/100)\cdot\mathbf{q}(50))\} & Q > 50 \\ \min\{255\cdot\mathbf{1}, \text{round}((50/Q)\cdot\mathbf{q}(50))\} & Q \leq 50 \end{cases}, \tag{1}$$

where $\mathbf{q}(50)$ is the 50% quality standard quantization table :

$$\mathbf{q}(50) = \begin{bmatrix} 16 & 11 & 10 & 16 & 24 & 40 & 51 & 61 \\ 12 & 12 & 14 & 19 & 26 & 58 & 60 & 55 \\ 14 & 13 & 16 & 24 & 40 & 57 & 69 & 56 \\ 14 & 17 & 22 & 29 & 51 & 87 & 80 & 62 \\ 18 & 22 & 37 & 56 & 68 & 109 & 103 & 77 \\ 24 & 35 & 55 & 64 & 81 & 104 & 113 & 92 \\ 49 & 64 & 78 & 87 & 103 & 121 & 120 & 101 \\ 72 & 92 & 95 & 98 & 112 & 100 & 103 & 99 \end{bmatrix}. \tag{2}$$

A standard quantization table for quality Q is denoted $\mathbf{q}(Q)$. Custom quantization tables (or when no distinction is needed) are denoted by $\mathbf{q}$ or $\mathbf{p}$ not necessarily indexed by any quality factor.

Relevant prior art

The problem of mismatched JPEG quantization tables has been addressed in [18], where the authors used the 548-dimensional CC-PEV feature vector and the 22,510-dimensional CC-JRM rich model to steganalyze nsF5 [10] in different JPEG sources. The authors proposed a semi-metric comparing quantization tables, and showed that both training on a mixture of JPEG

[1] As described in [1], some cameras and phones use the operation of truncation (rounding towards zero) instead of rounding.

qualities as well as using the semi-metric to find the best detector from a bank of pre-trained detectors can be used in practice to steganalyze custom quantization tables without training dedicated detectors. This work does not show how those detectors trained on a mixture of qualities compare to dedicated detectors.

In [26], the authors use a kernel based feature transformation to adapt CC-PEV and CC-JRM to mismatched JPEG quantization tables. However, it is unclear how to adapt this transformation to deep learning detectors where the feature representation is learned.

The mismatch of JPEG quantization tables between training and testing sets is a special case of what is recognized as the cover source mismatch problem [20, 11]. In [19], it is shown that, for a fixed feature set (CC-300), simple classifiers, such as the FLD-ensemble, are more robust to the cover source mismatch. This begs a question of how the mismatch affects modern detectors built using deep learning, which jointly optimize the feature representation and the classifier and are thus highly non-linear.

In [5], the authors showed that the SRNet (trained as a multi-class detector) is able to contain the complexity of a diversified stego source. The findings of this paper were used by the winners of the ALASKA challenge [24] to build detectors for a more diverse stego source. Even though the ability to generalize to unseen steganographic methods is still a challenge, these results indicate that properly trained CNNs do have the capacity to deal with diverse sources.

Datasets

Most experiments in this paper were executed on images prepared from BOSSbase 1.01 [2] and BOWS2 [3] each with 10,000 grayscale images resized to 256×256 using the 'imresize' function in Matlab with default parameters. The dataset was randomly divided into three sets with 14,000 (BOSSbase+BOWS2) / 1,000 (BOSSbase) / 5,000 images (BOSSbase) for training, validating and testing respectively. The splits were made to be compatible with the datasets used in [4, 6].

In Section "ALASKA," the ALASKA v1 [9] dataset has been used with development and processing scripts adapted to produce only 256×256 crops. The ALASKA dataset was randomly divided into three sets with 42,500 / 3,500 / 3,500 for training, validating and testing, respectively. The splits were made to be compatible with the datasets used in [24].

When training detectors based on hand-crafted feature sets, the validation set is merged with the training set and a k-fold cross validation or any other prediction performance estimate can be used to determine the optimal hyper-parameters.

The steganographic algorithms used in this paper are: J-UNIWARD [15], UED-JC [12], EBS [22], and nsF5 [10], embedded with 0.4 and 0.3 bpnzac (BOSS+BOWS2) or adaptive payload based on the image processing history, with priors $\pi_i = 0.4$, 0.3, 0.15, and 0.15, respectively (ALASKA v1). In ALASKA v1, color steganography is done by spreading the payload between between the image components (Y, C_r, and C_b) as described in Section *Payload repartition among color channels* in [9].

Steganalysis feature sets and CNN architecture

Feature based steganalysis

The steganalysis community has come up with numerous feature sets built in different domains and for different stego algorithms. In this paper, we work with the popular feature set called DCTR [14].

The DCTR features are histograms of absolute values of undecimated DCT coefficients quantized by

$$q_Q = 8 \times \left(2 - \frac{Q}{50}\right). \tag{3}$$

The undecimated DCT coefficients are defined as a set of 64 convolutions indexed by (k,l) with the DCT bases $\mathbf{B}^{(k,l)}_{m,n} = \frac{1}{4}\alpha(k)\alpha(l)\cos\left[\frac{(2m+1)k\pi}{16}\right]\cos\left[\frac{(2n+1)l\pi}{16}\right]$, where $\alpha(0) = \frac{1}{\sqrt{2}}$ and $\alpha(k) = 1$ for $k > 0$.

In this paper, we extend (3) to custom quantization tables $\mathbf{q}$ as

$$q = 8 \times \bar{\mathbf{r}}, \tag{4}$$

where $\mathbf{r} = \mathbf{q}./\mathbf{q}(50)$ is the elementwise division of both matrices, and $\bar{\mathbf{r}}$ corresponds to the average of all elements of matrix $\mathbf{r}$.

The FLD-ensemble [17] is then used with its default parameters for constructing the detector. When trained on multiple JPEG qualities, the training dataset corresponds to copies of the same set of images compressed with each quality.

CNN steganalysis

Recently, the community turned to deep learning for steganalysis in an attempt to improve detection accuracy by jointly optimizing the image representation (features sets) as well as the classifier. Deep learning architectures, such as [23, 25, 4], have been shown to outperform hand-crafted feature sets in the JPEG domain. A detailed survey on deep learning in steganalysis can be found in [8].

In this paper, we use the SRNet [4], a residual [13] CNN with 3×3 convolution kernels and ReLU activation functions. The first 8 layers of SRNet are un-pooled, and the next convolutional blocks are pooled using a 3×3 averaging layer with stride 2, as well as strided 1×1 convolutions in the skip connections. SRNet applies global average pooling in the last pooled layer to a 512 feature map, which is then Fully Connected (FC) to the classification logits. SRNet is trained with the Adamax optimizer [16] using various mini-batch sizes adapted to the diversity of the sources, as opposed to the mini-batch size of 32 initially proposed in [4]. At the time of publishing this work, SRNet achieved the best overall results for steganalysis in the JPEG domain.

When trained on multiple qualities, each batch is formed by repeatedly uniformly sampling a JPEG quality factor and selecting a cover-stego pair of that JPEG quality.

Scalability w.r.t. JPEG quality

In this section, we investigate whether feature-based and CNN steganalysis can contain the complexity of multiple JPEG quality factors within a single model. Note that quality factors 99 and 100 are not studied in this section because a very reliable JPEG compatibility attack is available for these qualities [7]. We study the scalability of this attack w.r.t. JPEG quality in .

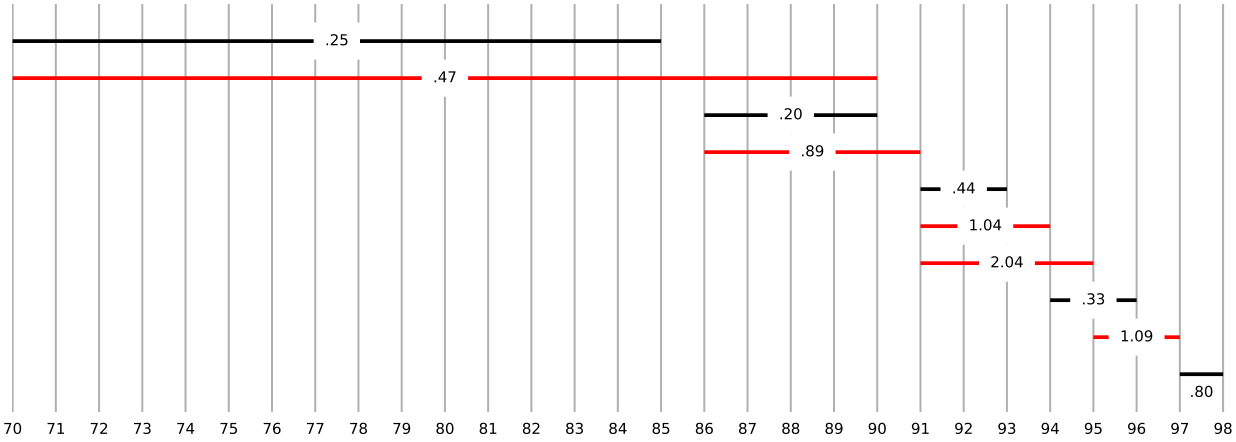


Figure 1: Maximum loss of accuracy of multi-quality SRNets w.r.t. dedicated detectors in multiple JPEG quality ranges for J-UNIWARD (0.4 bpnzac). Black lines correspond to selected ranges, widening them (red lines) leads to an increase of this loss.

Starting with small ranges, we compute the maximum loss of accuracy of the detector trained on the range w.r.t. dedicated detectors trained on individual qualities. The range is then expanded until we start observing high losses. Figure 1 shows the results of these experiments.

The selected ranges of quality factors in Figure 1 show an interesting general rule of thumb. A range of JPEG quality factors $[Q_{\min}, Q_{\max}]$ can be grouped in a single detector as long as :

$$\mathbf{q}_{kl}(Q_{\min})/\mathbf{q}_{kl}(Q_{\max}) \lesssim 2, \forall\, 0 \leq k,l \leq 7. \tag{5}$$

Note that when training SRNet on the range [70, 85] we use mini-batch size 128 due to the increased diversity introduced by mixing many JPEG qualities. All other ranges are trained using mini-batch size 64, no scaling of learning rates or the number of training iterations has been performed.

BOSS+BOWS2

Figure 2 shows the minimum total error probability P_{E} under equal priors for J-UNIWARD (0.4 bpnzac) and UED (0.3 bpnzac) for detectors dedicated to a specific JPEG quality (crosses) and detectors trained on each bin (range). Both DCTR+FLD-ensemble and SRNet seem to scale to multiple quality factors with no substantial loss in performance in both stego sources.

SRNet, however, has a significantly better detection accuracy that does not come at the expense of capacity of scaling to multiple qualities. The "ripples" in performance are explained in [6] and are due to the rounding and maxing in quantization matrices 1.

ALASKA

In order to move the experiments to a more realistic setting, we now use the ALASKA v1 dataset to test the proposed JPEG qualities grouping strategy. We show that the proposed grouping scales to more diverse cover and stego sources as well. Figure 3 shows the minimum error probability P_{E} and MD5[2] of YC_rC_b-SRNet tile detectors (c.f., *Channel separation* in [24]), trained as multi-class and used as binary detectors as executed by the authors. The grouping strategy does not affect the detectors' performance using either performance measure. For all QF ranges, we use mini-batch size 128 due to the increased diversity of the ALASKA dataset.

Note that, unlike the figures in [24], where the authors reported on a single test set comprising the stego mixture, measures in Figure 3 are computed using the following characterization of the ROC curve. If we denote the soft-output of a detector as $\hat{y}$ and the true label as y, then:

$$\begin{aligned} P_{\text{MD}}(T) &= P(\hat{y} \leq T \mid y > 0) \\ &= \sum_{i=1}^{4} P(\hat{y} \leq T \mid y = i) P(y = i \mid y > 0) \\ P_{\text{MD}}(T) &= \sum_{i=1}^{4} \pi_i P(\hat{y} \leq T \mid y = i). \end{aligned} \tag{6}$$

$$\begin{aligned} P_{\text{FA}}(T) &= P(\hat{y} \geq T \mid y = 0) \\ P_{\text{FA}}(T) &= 1 - P(\hat{y} \leq T \mid y = 0). \end{aligned} \tag{7}$$

This gives a more robust estimation of performance measures as P_{MD} is computed over stego versions of all available TST covers instead of only a portion of TST covers for each stego scheme.

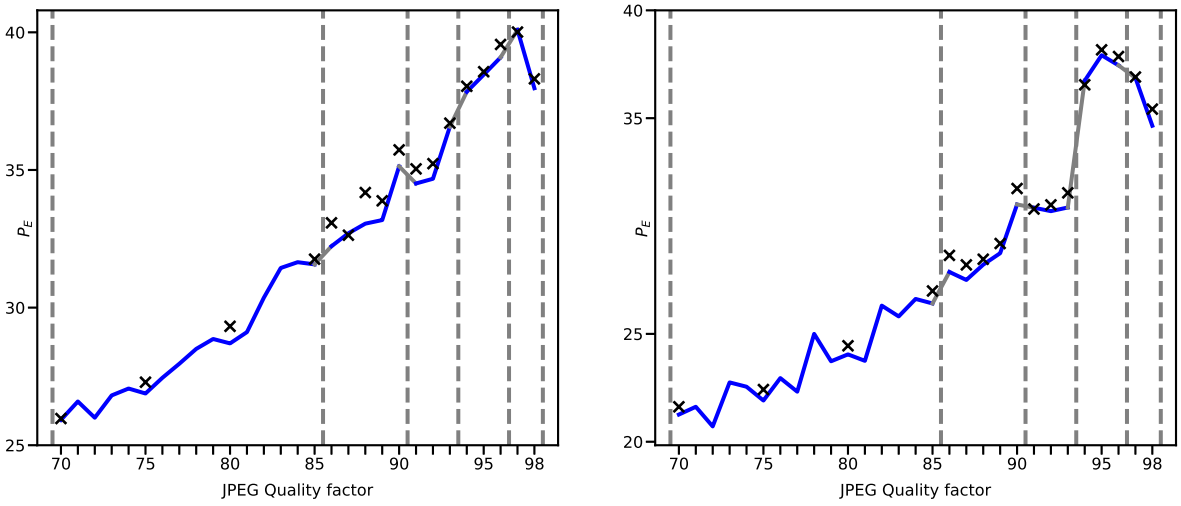
Reverse JPEG compatibility attack

The reverse JPEG compatibility attack is a powerful universal steganalysis attack for quality factors 99 and 100. In [7], the authors explain that for JPEG QF99 and QF100, the best detectors are built by training on the rounding errors of decompressed images instead of the images themselves. In particular, they replace the inputs of SRNet with the rounding errors to get the best detectors. Table 1 shows that for these two qualities, SRNet trained on rounding errors is also scalable w.r.t. the diversification.

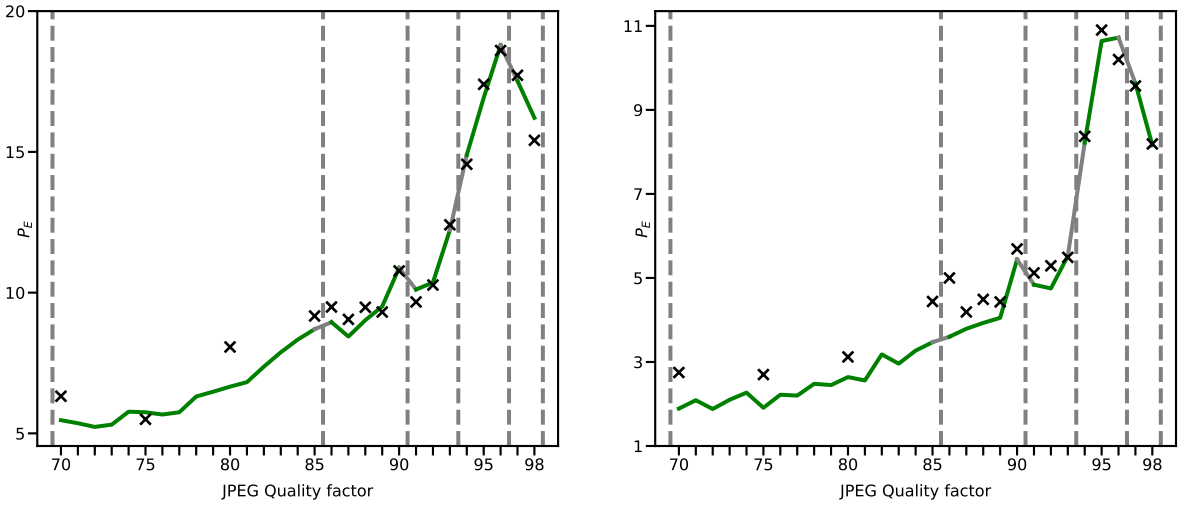
Robustness w.r.t. custom quantization tables

In this section, we selected 14 custom quantization tables from various camera models. The goal is to investigate the ability

[2] Missed detection rate at 5% false alarm.



(a) DCTR+FLD-ensemble



(b) SRNet

Figure 2: Minimum error probability P_{E} of multi-quality detectors for J-UNIWARD (0.4 bpnzac) (left) and UED (0.3 bpnzac) (right) detectors compared to dedicated detectors using DCTR+FLD-ensemble (a) and SRNet (b). Dashed grey lines represent the bins of JPEG qualities for multi-quality detectors.

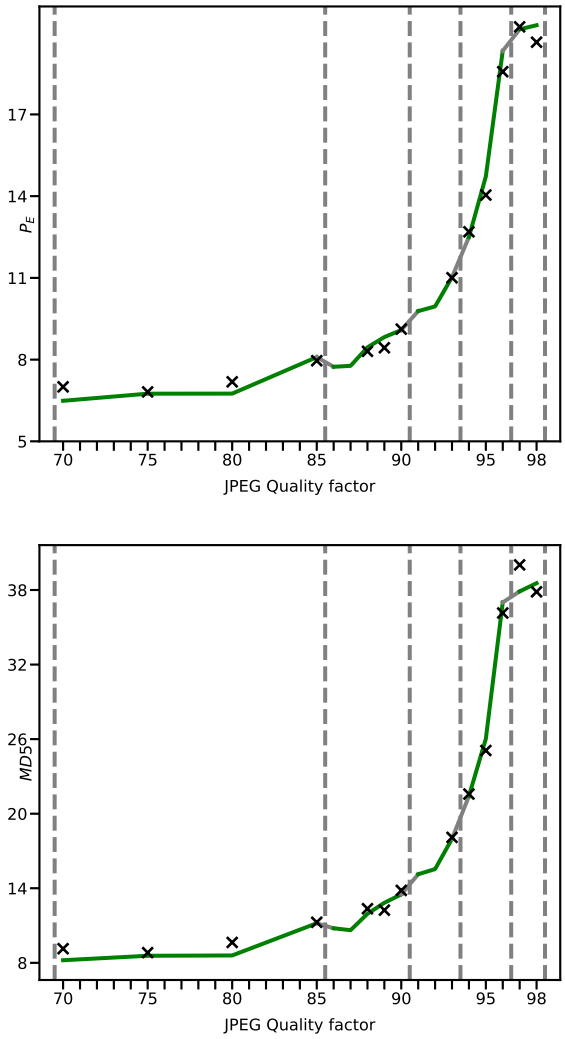


Figure 3: Minimum error probability P_{E} and missed detection rate at 5% false alarm MD5 of multi-quality detectors for ALASKA v1 compared to dedicated detectors trained during the competition. Dashed grey lines represent the bins of JPEG qualities for multi-quality detectors.

QF	100		99	
Payload	0.1	0.05	0.1	0.05
Dedicated	0.02	0.54	6.84	20.11
Trained on QF99–100	0.09	0.43	6.96	19.41

Table 1: Minimum error probability P_{E} of multi-quality detectors for J-UNIWARD compared to dedicated detectors trained using of the reverse JPEG compatibility attack.

of both detection paradigms to generalize to unseen custom JPEG quantization tables.

A semi-metric comparing quantization tables

We introduce the following semi-metric to compare two quantization tables $\mathbf{q}$ and $\mathbf{p}$:

$$d^2(\mathbf{q},\mathbf{p}) = \sum_{k,l} \frac{1}{(k+l)^2} \left(\frac{\mathbf{q}_{kl} - \mathbf{p}_{kl}}{\mathbf{q}_{kl} + \mathbf{p}_{kl}} \right)^2, \tag{8}$$

which is a weighted sum of the squares of relative differences between corresponding quantization coefficients. The weights are larger for low spatial frequencies (upper left of the table) and lower for high spatial frequencies (lower right of the table). We refer to it as the quantization table "dissimilarity" measure. It allows us to link each quantization table to the nearest JPEG quality: $\hat{Q}(\mathbf{q}) = \operatorname{argmin}_Q d(\mathbf{q},\mathbf{q}(Q))$. For a quantization table $\mathbf{q}$, we denote $\mathscr{B}(\mathbf{q})$ as the bin of JPEG qualities (used for training) to which $\hat{Q}(\mathbf{q})$ belongs. For notational simplicity, we denote the P_{E} obtained when training on $\mathbf{p}$ (one or multiple standard or custom quantization tables) and testing on $\mathbf{q}$ as $P_{\mathrm{E}}(\mathbf{p},\mathbf{q})$.

Figure 4 shows how the dissimilarity relates to SRNet's performance when the quantization tables are mismatched: the minimums are synchronized or sometimes relatively flat around the optimal values. This shows that the $\hat{Q}(\mathbf{q})$ computed using the proposed dissimilarity measure will usually be the best JPEG quality to steganalyze with, i.e., $\hat{Q}(\mathbf{q}) = \operatorname{argmin}_Q d(\mathbf{q},\mathbf{q}(Q)) = \operatorname{argmin}_Q P_E(\mathbf{q}(Q),\mathbf{q})$, which is a desirable property of the dissimilarity measure.

Table 3 shows the P_{E} of SRNet and DCT+FLD-ensemble in different settings. Each row corresponds to a custom quantization table $\mathbf{q}$. These results are visualized in Figure 5, which shows that SRNet is markedly more robust to mismatched custom quantization tables. Figure 5 also shows that training on multiple JPEG qualities does not seem to affect this robustness on average. Finally, Figure 5 points at some irregular behavior of the dissimilarity measure proposed. Even though, in most cases, a large dissimilarity value implies a larger loss in P_{E}, these losses are not consistently decreasing as a function of the dissimilarity.

Conclusions

This paper investigates the problem of detecting steganography in a diverse cover source of JPEG images. We are particularly interested in how steganalysis detectors scale to multiple JPEG qualities within a single model training, which we refer to as scalability w.r.t. JPEG quality factors. We show that both feature-based and CNN based detectors scale to multiple quality factors. We propose a set of detectors trained on a mixture of quality factors which, when compared with dedicated detectors trained for a specific JPEG quality factor, show no substantial loss in performance. The mixtures have been developed by gradually adding quality factors until a loss is observed when compared to dedicated detectors.

A set of 14 custom quantization tables with various dissimilarity measures to standard tables has been used to experimentally demonstrate that the scalability w.r.t. multiple JPEG qualities does not come at the expense of the detectors' robustness when facing mismatched custom quantization tables. CNN based ste-

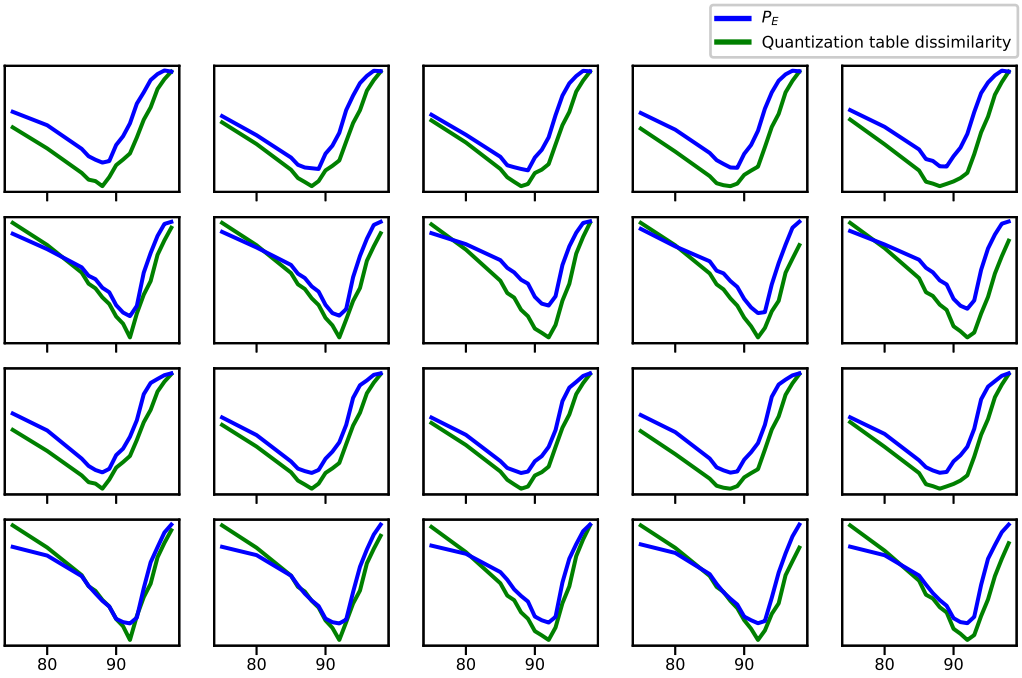
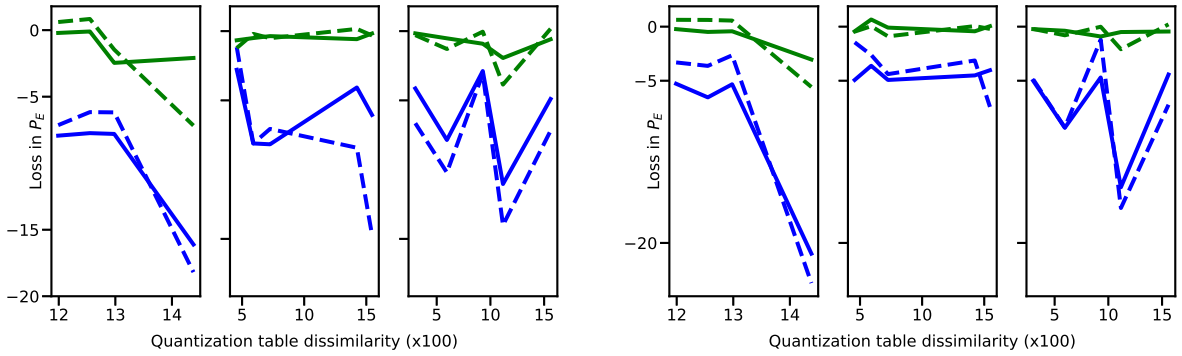


Figure 4: Minimum error probability of SRNet $P_{\mathrm{E}}(\mathbf{q}(Q),\mathbf{q})$ and $d(\mathbf{q}(Q),\mathbf{q})$ for different quality factors Q ranging between 75 and 98 and for 10 custom quantization tables $\mathbf{q}$. The first two rows correspond to J-UNIWARD (0.4 bpnzac), while the rest correspond to UED (0.3 bpnzac).



(a) J-UNIWARD (0.4 bpnzac)

(b) UED (0.3 bpnzac)

Figure 5: Loss in P_{E} for custom quantization tables: $P_{\mathrm{E}}(\mathbf{q},\mathbf{q}) - P_{\mathrm{E}}(\mathbf{q}(\hat{Q}),\mathbf{q})$ in solid, $P_{\mathrm{E}}(\mathbf{q},\mathbf{q}) - P_{\mathrm{E}}(\mathscr{B}(\mathbf{q}),\mathbf{q})$ in dashed for SRNet (green) and DCTR+FLD-ensemble (blue). Subplots refer to the three groups of quantization tables in Table 3.

ganalysis show a markedly better robustness compared to feature-based detectors.

This paper's general outcome is that we do not need to train a detector for each quality factor. This is very useful in practice, where one inevitably faces a diverse JPEG cover source, as it was the case, for example, in the ALASKA challenge.

Acknowledgements

This material is based on research sponsored by DARPA under agreement number FA8750-16-2-0173. The U.S. Government is authorized to reproduce and distribute reprints for Governmental purposes notwithstanding any copyright notation thereon. The views and conclusions contained herein are those of the authors and should not be interpreted as necessarily representing the official policies or endorsements, either expressed or implied, of DARPA or the U.S. Government.

References

[1] S. Agarwal and H. Farid. Photo forensics from rounding artifacts. In *IEEE Transactions on Information Forensics and Security*. IEEE, 2019. Under review.

[2] P. Bas, T. Filler, and T. Pevný. Break our steganographic system – the ins and outs of organizing BOSS. In T. Filler, T. Pevný, A. Ker, and S. Craver, editors, *Information Hiding, 13th International Conference*, volume 6958 of Lecture Notes in Computer Science, pages 59–70, Prague, Czech Republic, May 18–20, 2011.

[3] P. Bas and T. Furon. BOWS-2. http://bows2.ec-lille.fr, July 2007.

[4] M. Boroumand, M. Chen, and J. Fridrich. Deep residual network for steganalysis of digital images. *IEEE Transactions on Information Forensics and Security*, 14(5):1181–1193, May 2019.

[5] J. Butora and J. Fridrich. Detection of diversified stego sources using CNNs. In A. Alattar and N. D. Memon, editors, *Proceedings IS&T, Electronic Imaging, Media Watermarking, Security, and Forensics 2019*, San Francisco, CA, January 14–17, 2019.

[6] J. Butora and J. Fridrich. Effect of JPEG quality on steganographic security. In R. Cogranne and L. Verdoliva, editors, *The 7th ACM Workshop on Information Hiding and Multimedia Security*, Paris, France, July 3–5, 2019. ACM Press.

[7] J. Butora and J. Fridrich. Reverse JPEG compatibility attack. *IEEE Transactions on Information Forensics and Security*, 15:1444–1454, September 2019.

[8] M. Chaumont. Deep learning in steganography and steganalysis from 2015 to 2018. In *Digital Media Steganography: Principles, Algorithms, Advances*, volume abs/1904.01444. Elsevier, 2020.

[9] R. Cogranne, Q. Giboulot, and P. Bas. The ALASKA steganalysis challenge: A first step towards steganalysis "into the wild". In R. Cogranne and L. Verdoliva, editors, *The 7th ACM Workshop on Information Hiding and Multimedia Security*, Paris, France, July 3–5, 2019. ACM Press.

[10] J. Fridrich, T. Pevný, and J. Kodovský. Statistically undetectable JPEG steganography: Dead ends, challenges, and opportunities. In J. Dittmann and J. Fridrich, editors, *Proceedings of the 9th ACM Multimedia & Security Workshop*, pages 3–14, Dallas, TX, September 20–21, 2007.

[11] Q. Giboulot, R. Cogranne, and P. Bas. Steganalysis into the wild: How to define a source? January 29–February 1, 2018.

[12] L. Guo, J. Ni, and Y. Q. Shi. Uniform embedding for efficient JPEG steganography. *IEEE Transactions on Information Forensics and Security*, 9(5):814–825, May 2014.

[13] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, June 27–30, 2016.

[14] V. Holub and J. Fridrich. Low-complexity features for JPEG steganalysis using undecimated DCT. *IEEE Transactions on Information Forensics and Security*, 10(2):219–228, February 2015.

[15] 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*, 2014:1, 2014.

[16] D. P. Kingma and J. Ba. Adam: A method for stochastic optimization. *CoRR*, 2014. http://arxiv.org/abs/1412.6980.

[17] J. Kodovský, J. Fridrich, and V. Holub. Ensemble classifiers for steganalysis of digital media. *IEEE Transactions on Information Forensics and Security*, 7(2):432–444, April 2012.

[18] J. Kodovský, V. Sedighi, and J. Fridrich. Study of cover source mismatch in steganalysis and ways to mitigate its impact. In A. Alattar, N. D. Memon, and C. Heitzenrater, editors, *Proceedings SPIE, Electronic Imaging, Media Watermarking, Security, and Forensics 2014*, San Francisco, CA, February 3–5, 2014.

[19] I. Lubenko and A. D. Ker. Steganalysis with mismatched covers: Do simple classifiers help. In J. Dittmann, S. Katzenbeisser, and S. Craver, editors, *Proc. 13th ACM Workshop on Multimedia and Security*, pages 11–18, Coventry, UK, September 6–7, 2012.

[20] J. Pasquet, S. Bringay, and M. Chaumont. Steganalysis with cover-source mismatch and a small learning database. In *2014 22nd European Signal Processing Conference (EUSIPCO)*, pages 2425–2429. IEEE, 2014.

[21] W. Pennebaker and J. Mitchell. *JPEG: Still Image Data Compression Standard*. Van Nostrand Reinhold, New York, 1993.

[22] C. Wang and J. Ni. An efficient JPEG steganographic scheme based on the block–entropy of DCT coefficents. In *Proc. of IEEE ICASSP*, Kyoto, Japan, March 25–30, 2012.

[23] G. Xu. Deep convolutional neural network to detect J-UNIWARD. In M. Stamm, M. Kirchner, and S. Voloshynovskiy, editors, *The 5th ACM Workshop on Information Hiding and Multimedia Security*, Philadelphia, PA, June 20–22, 2017.

[24] Y. Yousfi, J. Fridrich, J. Butora, and Q. Giboulot. Breking ALASKA: Color separation for steganalysis in JPEG domain. In R. Cogranne and L. Verdoliva, editors, *The 7th ACM Workshop on Information Hiding and Multimedia Security*, Paris, France, July 3–5, 2019. ACM Press.

[25] J. Zeng, S. Tan, B. Li, and J. Huang. Large-scale JPEG image steganalysis using hybrid deep-learning framework. *IEEE Transactions on Information Forensics and Security*,

13(5):1200–1214, 2018.

[26] L. Zeng, X. Kong, M. Li, and Y. Guo. JPEG quantization table mismatched steganalysis via robust discriminative feature transformation. In A. Alattar and N. D. Memon, editors, *Proceedings SPIE, Electronic Imaging, Media Watermarking, Security, and Forensics 2015*, volume 9409, San Francisco, CA, February 8–12, 2015.

Author Biography

Yassine Yousfi is currently a PhD student in Electrical and Computer Engineering at Binghamton University. He received an MS in Machine Learning from Ecole Centrale de Lille in France in 2018. His research interests are in the field of media security and forensics, and, particularly steganography and steganalysis of digital images.

Jessica Fridrich is Distinguished Professor of Electrical and Computer Engineering at Binghamton University. She received her PhD in Systems Science from Binghamton University in 1995 and MS in Applied Mathematics from Czech Technical University in Prague in 1987. Her main interests are in steganography, steganalysis, and digital image forensics. Since 1995, she has received 20 research grants totaling over $12 mil that lead to more than 200 papers and 7 US patents.

Appendix

Custom Quantization Table								Standard Quantization Table								$\hat{Q}$	$d(\mathbf{q}, \mathbf{q}(\hat{Q}))$ ($\times 100$)
8	6	7	7	9	12	25	36	8	6	5	8	12	20	26	31		
6	6	7	9	11	18	32	46	6	6	7	10	13	29	30	28		
5	7	8	11	19	28	39	48	7	7	8	12	20	29	35	28		
8	10	12	15	28	32	44	49	7	9	11	15	26	44	40	31	75	11.99
12	13	20	26	34	41	52	56	9	11	19	28	34	55	52	39		
20	29	29	44	55	52	61	50	12	18	28	32	41	52	57	46		
26	30	35	40	52	57	60	52	25	32	39	44	52	61	60	51		
31	28	28	31	39	46	51	50	36	46	48	49	56	50	52	50		
8	8	8	8	8	17	25	25	8	6	5	8	12	20	26	31		
8	8	8	8	8	25	25	25	6	6	7	10	13	29	30	28		
8	8	8	8	17	25	25	25	7	7	8	12	20	29	35	28		
8	8	8	8	25	33	33	25	7	9	11	15	26	44	40	31	75	14.38
8	8	17	25	25	42	42	33	9	11	19	28	34	55	52	39		
8	17	25	25	33	42	50	42	12	18	28	32	41	52	57	46		
17	25	33	33	42	50	50	42	25	32	39	44	52	61	60	51		
33	42	42	42	50	42	42	42	36	46	48	49	56	50	52	50		
4	3	2	4	6	11	14	16	4	3	2	4	6	10	12	15		
3	3	3	5	7	16	16	15	3	3	3	5	6	14	14	13		
3	3	4	6	11	15	19	15	3	3	4	6	10	14	17	13		
3	4	6	8	14	24	22	17	3	4	5	7	12	21	19	15	88	4.57
4	6	10	15	18	30	28	21	4	5	9	13	16	26	25	18		
6	9	15	17	22	28	31	25	6	8	13	15	19	25	27	22		
13	17	21	24	28	33	33	27	12	15	19	21	25	29	29	24		
19	25	26	27	30	27	28	27	17	22	23	24	27	24	25	24		
3	3	3	3	8	5	13	13	4	3	2	4	6	10	12	15		
3	3	3	3	5	11	13	13	3	3	3	5	6	14	14	13		
3	3	5	5	5	11	16	16	3	3	4	6	10	14	17	13		
3	5	5	8	13	16	24	24	3	4	5	7	12	21	19	15	88	15.46
3	5	8	13	16	18	13	26	4	5	9	13	16	26	25	18		
11	8	13	16	18	24	18	24	6	8	13	15	19	25	27	22		
5	16	21	18	24	26	24	21	12	15	19	21	25	29	29	24		
16	18	21	21	26	26	24	24	17	22	23	24	27	24	25	24		
3	2	2	3	4	3	8	9	3	2	2	3	4	6	8	10		
2	2	2	3	4	9	9	8	2	2	2	3	4	9	10	9		
2	2	3	4	6	9	10	8	2	2	3	4	6	9	11	9		
2	3	4	5	8	13	12	9	2	3	4	5	8	14	13	10	92	3.01
3	4	6	8	10	16	15	11	3	4	6	9	11	17	16	12		
4	5	8	9	12	15	16	13	4	6	9	10	13	17	18	15		
7	9	11	13	15	18	17	15	8	10	12	14	16	19	19	16		
11	13	14	14	16	15	15	14	12	15	15	16	18	16	16	16		
2	2	2	2	6	4	9	9	3	2	2	3	4	6	8	10		
2	2	2	2	4	8	9	9	2	2	2	3	4	9	10	9		
2	2	4	4	4	8	13	11	2	2	3	4	6	9	11	9		
2	4	4	6	9	11	17	17	2	3	4	5	8	14	13	10	92	15.60
2	4	6	9	11	13	9	19	3	4	6	9	11	17	16	12		
8	6	9	11	13	17	13	17	4	6	9	10	13	17	18	15		
4	11	15	13	17	19	17	15	8	10	12	14	16	19	19	16		
11	13	15	15	19	19	17	17	12	15	15	16	18	16	16	16		

Table 2: Examples of custom quantization tables used and their closest standard counterparts.

$\hat{Q}$	$d(\mathbf{q},\mathbf{q}(\hat{Q}))$ (x100)	J-UNIWARD (0.4 bpnzac) $P_{\mathrm{E}}(\mathbf{q},\mathbf{q})$	$P_{\mathrm{E}}(\mathscr{B}(\mathbf{q}),\mathbf{q})$	$P_{\mathrm{E}}(\mathbf{q}(\hat{Q}),\mathbf{q})$	UED (0.3 bpnzac) $P_{\mathrm{E}}(\mathbf{q},\mathbf{q})$	$P_{\mathrm{E}}(\mathscr{B}(\mathbf{q}),\mathbf{q})$	$P_{\mathrm{E}}(\mathbf{q}(\hat{Q}),\mathbf{q})$
75	11.99	6.42	5.81	6.63	2.87	2.24	3.10
	12.55	6.83	5.99	6.93	3.14	2.52	3.63
	12.98	4.47	5.93	6.93	3.05	2.49	3.48
	14.38	6.91	14.08	9.00	1.35	6.93	4.39
88	4.57	10.58	11.80	11.24	4.66	5.12	5.09
	5.90	8.57	8.77	9.07	4.28	4.24	3.62
	7.25	8.39	8.90	8.75	3.41	4.31	3.50
	14.24	8.83	8.67	9.41	3.99	3.94	4.42
	15.46	9.66	9.97	9.83	4.09	4.30	4.05
92	3.01	9.84	10.13	10.00	5.10	5.32	5.33
	5.95	9.61	10.90	10.13	4.72	5.50	5.10
	9.29	13.10	13.14	14.01	6.63	6.64	7.54
	11.17	9.00	12.86	10.94	4.05	6.14	4.55
	15.60	12.28	12.11	12.87	5.59	5.38	6.04

(a) SRNet

$\hat{Q}$	$d(\mathbf{q},\mathbf{q}(\hat{Q}))$ (x100)	J-UNIWARD (0.4 bpnzac) $P_{\mathrm{E}}(\mathbf{q},\mathbf{q})$	$P_{\mathrm{E}}(\mathscr{B}(\mathbf{q}),\mathbf{q})$	$P_{\mathrm{E}}(\mathbf{q}(\hat{Q}),\mathbf{q})$	UED (0.3 bpnzac) $P_{\mathrm{E}}(\mathbf{q},\mathbf{q})$	$P_{\mathrm{E}}(\mathscr{B}(\mathbf{q}),\mathbf{q})$	$P_{\mathrm{E}}(\mathbf{q}(\hat{Q}),\mathbf{q})$
75	11.99	27.48	34.62	35.41	22.85	26.16	28.13
	12.55	28.2	34.36	35.94	24.51	28.14	31.05
	12.98	27.47	33.65	35.27	23.41	26.06	28.74
	14.38	23.73	41.92	39.84	15.79	39.5	36.78
88	4.57	35.82	36.98	38.61	30.43	31.88	35.36
	5.90	33.47	41.58	41.58	27.44	30.06	31.06
	7.25	33.18	40.24	41.35	27.15	31.54	32.08
	14.24	34.22	42.64	38.29	28.38	31.51	32.89
	15.46	34.02	48.74	40.07	27.24	34.66	31.27
92	3.01	35.24	41.89	39.38	30.64	35.52	35.65
	5.95	34.98	45.19	42.84	29.83	39.16	39.18
	9.29	37.88	40.96	40.76	34.35	35.55	39.05
	11.17	33.85	47.86	44.89	28.34	45.11	43.19
	15.60	37.07	44.26	41.97	31.92	39.14	36.35

(b) DCTR+FLD-ensemble

Table 3: Minimum total error probability P_{E} of various detectors: (i) dedicated $P_{\mathrm{E}}(\mathbf{q},\mathbf{q})$ (ii) trained on the corresponding bin $P_{\mathrm{E}}(\mathscr{B}(\mathbf{q}),\mathbf{q})$ (iii) trained on the closest JPEG quality $P_{\mathrm{E}}(\mathbf{q}(\hat{Q}),\mathbf{q})$, using SRNet (a) and DCTR+FLD-ensemble (b). Each row corresponds to a custom quantization table.