Bayesian Tools for Reliable Multimedia Forensics

(Invited Paper)

Anatol Maier
IT Security Infrastructures Lab
Univ. of Erlangen-Nürnberg
Erlangen, Germany
anatol.maier@fau.de

Benedikt Lorch
IT Security Infrastructures Lab
Univ. of Erlangen-Nürnberg
Erlangen, Germany
benedikt.lorch@fau.de

Christian Riess
IT Security Infrastructures Lab
Univ. of Erlangen-Nürnberg
Erlangen, Germany
christian.riess@fau.de

***Abstract*—The goal of multimedia forensics is to determine origin and authenticity of images and video. The currently most successful approaches use machine learning, and demonstrate excellent performance in lab settings. However, these methods are still challenged to generalize to images from the internet with potentially complex and partially unknown processing. The current best counter-measure is extensive training data augmentation, but this is extremely costly considering the many possible processing variants that must be covered.**

In this work, we review and consolidate our recent efforts on a different approach to cope with the challenge of images from unknown provenance. We propose to concentrate the training to the forensic task at hand, and to additionally include a measure for uncertainty to detect when a classifier is not confident on a given input. We believe that uncertainty-aware tools can complement existing efforts when data augmentation fails, and additionally provide valuable feedback to forensic analysts.

***Index Terms*—multimedia forensics, machine learning, Bayesian modeling, reliability**

I. Introduction

Multimedia forensics aims to provide clues to the authenticity and origin of an image. The image formation offers several complementary traces which can be forensically exploited. For example, classical methods use the scene geometry [1], sensor noise patterns [2], or distortions in the image or video compression [3, 4]. A more complete introduction to the breadth of clues in multimedia forensics can be found in textbooks [5, 6].

Many of the classical forensic traces are analytic, and are therefore linked to model assumptions that can be manually verified by a forensic analyst. However, analytic cues are inherently limited in the complexity of the image formation models that they use. Hence, these approaches have difficulties to operate on images or videos that have a complex processing history like many images on the internet. For example, the largest social media sites routinely downsample and recompress multimedia content prior to distribution, which complicates the use of analytic forensic cues.

Thus, research in multimedia forensics focused in recent years on learning-based methods. These approaches derive forensic cues directly from the data, which demonstrates considerable performance improvements on data that is difficult to model analytically. For example, impressive results have been reported for the characterization of image noise [7], for the detection of global image postprocessing [8], or for the detection of so-called DeepFakes, i.e., images of computer-generated faces [9].

However, the use of machine learning models introduces new practical challenges. To the forensic analyst, these models are typically opaque, such that predictions can not be validated. The expected performance of a machine learning model not only depends on the reported performance in laboratory settings, but also in its ability to generalize well to the single specific input that may be subject of the forensic investigation. Prior work showed that learning-based detectors are sensitive to test images that differ too much from the training data [10, 11]. As a consequence, it may easily occur that a model can not operate well on an input that slightly differs from the training data, e.g., in its noise or compression characteristics. Without the ability to investigate deeper into the implicit assumptions of the learned model, such failure cases are difficult to detect for a forensic analyst.

To address this challenge, we recently proposed machine learning classifiers that provide a built-in confidence score. These classifiers use a Bayesian framework to explicitly model the uncertainty that is associated with a classifier decision. The uncertainty can be further decomposed into aleatoric and epistemic uncertainty [12]. Aleatoric uncertainty indicates inherent ambiguities in the data, and is as such irreducible with the given model and data. Epistemic uncertainty indicates a lack of training data for the given input, and can be reduced by tailoring the training set to the input. A forensic analyst can use these uncertainties to derive a reliability in the classifier's decision, and also to reject decisions with marginal confidence.

In this work, we provide a gentle introduction to the Bayesian framework for uncertainty quantification, put our recent works [11, 13, 14] on classification with uncertainty in a joint perspective, and provide experiments for the application of JPEG forensics and resampling detection under different types of dataset shifts. We also provide a discussion on current limitations and open research challenges that hopefully benefit the overall progress in the field.

Work was supported by Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) as part of the Research and Training Group 2475 "Cybercrime and Forensic Computing" (grant #393541319/GRK2475/1-2019).

II. Related Work

There are currently two main strategies in multimedia forensics to mitigate performance degradation when the training data is not sufficiently representative for the test data.

First, several works propose to train specific detectors for images of different qualities [7, 15, 16]. Arguably, the most notable parameter that affects the detector performance on images is the JPEG compression quality. Hence, a common strategy is to train an individual detector on images that have been compressed with a specific JPEG quality factor. For evaluation, the best matching detector is determined based on the distance of the JPEG quantization matrices between the test image and the training images. If the compression parameters of the testing data are very similar to one of the trained models, this approach can lead to very strong results. However, it is also very costly to maintain a database with multiple copies of a model, each fully trained on one specific quality factor. Additionally, although this strategy improves the overall results, it may still fail for other types of unseen distortions or due to other subtle differences in the JPEG implementation [11].

The second widely used strategy to mitigate the training-test mismatch is to train a single classifier with extensive data augmentation [7, 17, 18]. This approach aims to anticipate a representative space of common image editing operations, and is also the best strategy in multimedia forensics competitions with completely unseen testing data. However, also this approach requires a significant effort in training resources while not necessarily guaranteeing that all possible processing variants are covered. Additionally, the performance of a single, broad classifier is oftentimes weaker than the performance of a specialized classifier, as noted in a recent study on classifier performance guarantees for image denoising [19].

In the broader machine learning literature, Bayesian learning techniques received increasing attention in the past years. Gal's Ph.D. thesis is an early work that covers Uncertainty in Deep Learning [20]. Blundell *et al.* presented a method to effectively train deep neural networks that follow the Bayesian framework [21]. Arguably the simplest approach to create uncertainties is via Monte-Carlo dropout at test time, which has been explored by Gal and Ghahramani [22]. Several later works explore different application fields of uncertainty estimates, for example in medical image analysis [23, 24].

A competing, fundamentally different approach to uncertainty is to use a conventional neural network, and to interpret the relative magnitude of the class activations in the output layer as confidence. Hendrycks and Gimpel examine the softmax statistics of the output layer to detect a training-test-mismatch [25]. However, these statistics are oftentimes skewed towards overly confident predictions. An alternative is to actively calibrate the statistics. To this end, Guo *et al.* propose Temperature Scaling [26], which inspired several follow-up works [27]. These approaches have the benefit that they can be applied on any commonly used neural network architecture. However, they lack the mathematical rigour of Bayesian approaches.

Ovadia *et al.* [28] recently presented a comparative study of several uncertainty metrics, which reports that deep Ensembles [29] perform quite well, i.e., an ensemble of neural networks with different training initializations. However, this approach requires that multiple copies of the same network are trained and evaluated, with the associated computational resource requirements.

III. Modeling Uncertainty in Multimedia Forensics

The approach to include an uncertainty metrics into the classifier is conceptually relatively straightforward: at least one scalar component of the classification system is replaced by a distribution, such that the predictions themselves becomes probabilistic. Examining the distribution of outputs enables drawing conclusions on the uncertainty. Narrow distributions, i.e., very similar predictions, are an indication that the model is very confident. Conversely, broad predictive distributions indicate that the model is not confident, i.e., that the input can lead to controversial decisions.

Mathematically, this behavior is achieved via Bayesian inference. We consider the predictive distribution

$$p(\mathbf{y}^*|\mathbf{x}^*, \mathbf{x}_\mathrm{t}, \mathbf{y}_\mathrm{t}) = \int p(\mathbf{y}^*|\mathbf{x}^*, \mathbf{w}) p(\mathbf{w}|\mathbf{x}_\mathrm{t}, \mathbf{y}_\mathrm{t}) \,\mathrm{d}\mathbf{w} \ , \quad (1)$$

where $\mathbf{y}^*$ denotes the label that shall be predicted given input $\mathbf{x}^*$ and the training data and labels $\mathbf{x}_\mathrm{t}$ and $\mathbf{y}_\mathrm{t}$. On the right hand side of the equation, the dependency on the classifier parameters is made explicit with $\mathbf{w}$. Some or all of the parameters form a distribution, which is calculated from the training data, indicated by the posterior distribution over the weights $p(\mathbf{w}|\mathbf{x}_\mathrm{t}, \mathbf{y}_\mathrm{t})$. The predictive distribution is obtained by marginalization over these weights via integration. The predictive distribution $p(\mathbf{y}^*|\mathbf{x}^*, \mathbf{x}_\mathrm{t}, \mathbf{y}_\mathrm{t})$ is categorical for classification tasks, and continuous for regression tasks.

The posterior distribution of weights $p(\mathbf{w}|\mathbf{x}_\mathrm{t}, \mathbf{y}_\mathrm{t})$ can be further decomposed via Bayes' theorem to

$$p(\mathbf{w}|\mathbf{x}_\mathrm{t}, \mathbf{y}_\mathrm{t}) = \frac{p(\mathbf{y}_\mathrm{t}|\mathbf{x}_\mathrm{t}) p(\mathbf{w})}{\int p(\mathbf{y}_\mathrm{t}|\mathbf{x}_\mathrm{t}, \mathbf{w}) p(\mathbf{w}) \,\mathrm{d}\mathbf{w}} \ , \quad (2)$$

where the numerator consists of the product of likelihood and prior, and the denominator of the marginal likelihood, which is also called evidence.

The integral in the evidence can in general not be solved analytically, and numerically integrating over the whole space of weights $\mathbf{w}$ is very expensive. An alternative is to use a variational approximation with the additional assumptions that the model parameters are independent, and that the posterior of each parameter follows a simple distribution. In that case, a distribution $q(\mathbf{w}|\boldsymbol{\theta})$ with parameters $\boldsymbol{\theta}$ can be calculated that approximates $p(\mathbf{w}|\mathbf{x}_\mathrm{t}, \mathbf{y}_\mathrm{t})$, such that the predictive distribution from Eqn. 1 becomes

$$p(\mathbf{y}^*|\mathbf{x}^*, \mathbf{x}_\mathrm{t}, \mathbf{y}_\mathrm{t}) \approx \int p(\mathbf{y}^*|\mathbf{x}^*, \mathbf{w}) q(\mathbf{w}|\boldsymbol{\theta}) \,\mathrm{d}\mathbf{w} \ . \quad (3)$$

The parameters $\boldsymbol{\theta}$ of $q(\mathbf{w}|\boldsymbol{\theta})$ are determined by minimizing the Kullback-Leibler (KL) divergence between $q(\mathbf{w}|\boldsymbol{\theta})$ and $p(\mathbf{w}|\mathbf{x}_\mathrm{t}\mathbf{y}_\mathrm{t})$,

$$\begin{aligned}\boldsymbol{\theta}^* &= \operatorname*{argmin}_{\boldsymbol{\theta}} \mathrm{KL}\left(q(\mathbf{w}|\boldsymbol{\theta})\|p(\mathbf{w}|\mathbf{x}_\mathrm{t},\mathbf{y}_\mathrm{t})\right) &(4)\\ &= \operatorname*{argmin}_{\boldsymbol{\theta}} \mathrm{KL}\left(q(\mathbf{w}|\boldsymbol{\theta})\|p(\mathbf{w})\right) &(5)\\ &\qquad - \mathbb{E}_{q(\mathbf{w}|\boldsymbol{\theta})}(\log p(\mathbf{y}_\mathrm{t}|\mathbf{x}_\mathrm{t},\mathbf{w})) \ .\end{aligned}$$

The full derivation of this identity is provided in Eqn. (2) of our earlier work [13]. The expression in Eqn. 5 is the negative evidence lower bound (ELBO). Hence, maximization of the ELBO minimizes the KL divergence between $q(\mathbf{w}|\boldsymbol{\theta})$ and $p(\mathbf{w}|\mathbf{x}_\mathrm{t},\mathbf{y}_\mathrm{t})$. The optimum is found iteratively via gradient descent on the negative ELBO. To this end, N random weights $\mathbf{w}_i$ are sampled from $q(\mathbf{w}|\boldsymbol{\theta})$ and evaluated with the approximation equation

$$\begin{aligned}&\mathrm{KL}\left(q(\mathbf{w}|\boldsymbol{\theta})\|p(\mathbf{w})\right) - \mathbb{E}_{q(\mathbf{w}|\boldsymbol{\theta})}(\log p(\mathbf{y}_\mathrm{t}|\mathbf{x}_\mathrm{t},\mathbf{w})) &(6)\\ &\approx \frac{1}{N}\sum_{i=1}^{N} \log q(\mathbf{w}_i|\boldsymbol{\theta}) - \log p(\mathbf{w}_i) - \log p(\mathbf{y}_\mathrm{t}|\mathbf{x}_\mathrm{t},\mathbf{w}_i) \ . &(7)\end{aligned}$$

With the trained classifier, a prediction is performed very similarly, by again sampling weights $\mathbf{w}_i$ and averaging the resulting predictions. The uncertainty is calculated from the variance of the predictions. In particular, the epistemic uncertainty, which expresses a mismatch between the training and testing data distribution, is calculated from the N draws of the test run as [12, 24]

$$\mathrm{Var}_\mathrm{ep}(\mathbf{y}^*) = \frac{1}{N}\sum_{i=1}^{N} \left(p(\mathbf{y}^*|\mathbf{x}^*,\mathbf{w}_i) - q(\mathbf{y}^*|\mathbf{x}^*)\right)^{\otimes 2} \quad , \qquad (8)$$

where $^{\otimes 2}$ denotes the outer vector product.

This framework requires two design decisions. In our experiments, we use a Gaussian weight prior $p(\mathbf{w})$. For the representation for $q(\mathbf{w}|\boldsymbol{\theta})$, we use a mean-field approximation. Furthermore, in practice, it is also important to introduce a weight in the loss of Eqn. 7, since the first two terms grow with the number of parameters, and the last term grows with the dataset size.

This framework is sufficiently general to cover different specific models. We explored Bayesian logistic regression [11], Gaussian processes [30], and Bayesian neural networks [13, 14] as specific realizations. For Bayesian neural networks, this framework can be effectively included into the standard backpropagation, as shown by Blundell *et al.* [21]. Simpler classifiers like Bayesian logistic regression enable a further simplification when used with a Gaussian weight prior, namely to estimate the variational posterior $q(\mathbf{w}|\boldsymbol{\theta})$ via Expectation-Maximization [11].

The derivation above makes the inclusion of a weight distribution into the classifier explicit. However, it is interesting to note that this is not the only design variant to obtain a distribution of the outputs. To notable alternatives are Monte-Carlo dropout [22] and deep Ensembles [29]. Monte-Carlo dropout randomly deactivates weights at test time to obtain a predictive distribution. Deep Ensembles form the predictive distribution from the outputs of multiple networks with identical architecture that were trained with different initializations. In both cases, the epistemic uncertainty can be calculated analogously as the variance of the predictions.

IV. Experiments

We illustrate the performance of this framework on selected experiments that complement earlier results using Bayesian Logistic Regression [11] and Bayesian Neural Networks [13].

A. Detection of Double JPEG Compression

We first investigate the application of double JPEG compression. This classical forensic problem is a binary classification task. We re-use the Bayesian Logistic Regression and the experimental setup from our earlier work [11], but study here the dataset shift when the quality factor of the secondary JPEG compression (QF2) is unknown. This scenario has been considered, e.g., by Amerini *et al.* [31], and may occur when a JPEG image is re-saved in a lossless format such as PNG, such that the JPEG header information is lost.

We use the RAISE1k dataset [32], with a 50/50 split into training and testing. For each image, we create a single-compressed version with fixed quality factor $\mathrm{QF2}_\mathrm{train}$ for training, and multiple double compressed versions with different primary quality factors $\mathrm{QF1} \in \{50, 55, \ldots, 90\}$ und again $\mathrm{QF2}_\mathrm{train}$ for the secondary compression. For the training, 500 of the 5000 double-compressed images are randomly selected, such that the identical number of single- and double-compressed images is used. This protocol is repeated for different secondary compression factors $\mathrm{QF2}_\mathrm{train} \in \{50, 55, \ldots, 90\}$. All compression steps are performed with `libjpeg`. We extract first-digit features from the first nine AC bands, and count the frequency of all nine possible first digits, which leads to a 81-dimensional feature vector, and use Bayesian Logistic Regression analogously to our previous work [11].

The evaluation investigates the performance of the method in the challenging case when the secondary quality factor differs between training and testing, i.e., $\mathrm{QF2}_\mathrm{train} \neq \mathrm{QF2}_\mathrm{test}$. Figure 1 shows the results. On top, the performance of detecting double compression is shown in terms of accuracy for all combinations of $\mathrm{QF2}_\mathrm{train}$ and $\mathrm{QF2}_\mathrm{test}$. As expected, the performance of the method is strongly reduced by a mismatch in the training and test distributions. Most combinations of secondary quality factors reduce the classifier even to guessing chance of 0.5. However, the second row shows the area under the curve (AUC) over the uncertainties from the Bayesian Logistic Regression: in almost all cases, the dataset shift can be perfectly detected with an AUC of 1. This information can be used as a feedback to the analyst to indicate that the classifier is applied to an out-of-distribution input.

The proposed framework also compares favorably to other methods. We investigate a k-nearest-neighbor (kNN) classifier with $k = 5$ to calculate the uncertainty from the distance

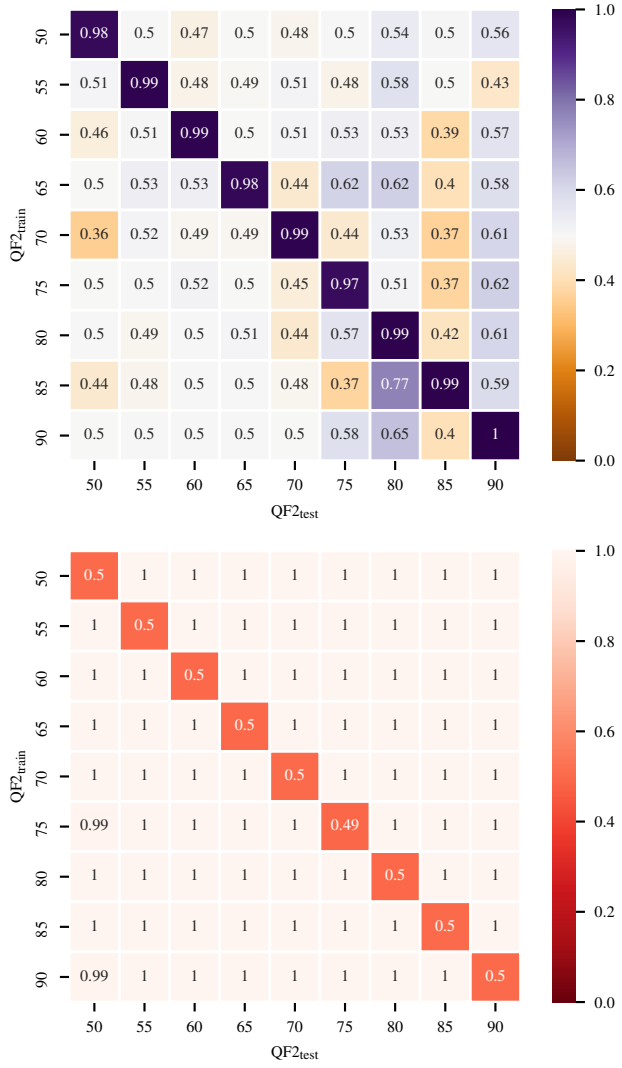

Fig. 1. Detection of double JPEG compression for the case when the QF2 factor between the training set and testing set differ. As expected, the accuracies (top) are strongly impacted by this dataset shift. However, the AUC over the uncertainties (bottom) indicates that these cases can be detected and reported to the forensic analyst.

of the test sample to the training data in the feature space. Additionally, we investigate the combined classification framework (CCF) by Wang *et al.* [33]. This framework consists of two one-class SVMs and one two-class SVMs for camera classification. The one-class SVMs are trained on each of the two classes, and use radial basis functions with $\nu = 0.001$ that is found via grid search. If both classes disagree, the two-class SVM decides between the classes. It uses a linear kernel with $C = 1.0$. The uncertainties are calculated from the binary decision whether a sample is recognized as inlier or outlier. Table I shows the result of this comparison. The proposed Bayesian Linear Regression (BLR) provides overall the best tradeoff. It outperforms the baselines both with respect to the detection accuracy for in-distribution samples, and is only matched by the k-NN classifier for the detection of out-of-distribution (OOD) samples.

TABLE I
DETECTION OF DOUBLE JPEG COMPRESSION UNDER DATASET SHIFT. COMPARISON OF THE PROPOSED BAYESIAN LINEAR REGRESSION (BLR) WITH THE K-NN CLASSIFIER AND THE COMBINED CLASSIFICATION FRAMEWORK (CCF) BY WANG *et al.* [33].

Method	In-dist. Acc.	OOD AUC
kNN	0.86	**1.00**
CCF	0.91	0.96
BLR	**0.99**	**1.00**

B. Detection of Resizing

We also investigate the application of resizing detection under dataset shift, in this case due to different types of postprocessing. The investigated classifiers are Bayesian Logistic Regression and Bayesian Neural Networks.

To evaluate the Bayesian Logistic Regression, we randomly extract 25 patches from the RAISE1k dataset, each with a size of 512×512 pixels. The patches are converted to gray. We create a copy of each patch, resized via bicubic interpolation by a factor of 1.3 and again cropped around the center to 512×512 pixels. The training and test sets are split 80/20, such that a source patch and its resized complement is either part of the training or set, but not in both. On each patch, we calculate the popular SPAM features with a quantization factor of 4.5 and cropping interval $T = 1$. This yields a 50-dimensional feature vector per patch. These feature vectors are used to train a Bayesian Logistic Regression Model analogously to our previous work [11].

For testing, we additionally apply Gaussian blur with blur kernels $\sigma_b \in \{0.3, 0.4, \ldots, 0.7\}$, additive Gaussian noise with a standard deviation of $\sigma_n \in \{0.2, 0.3, \ldots, 0.6\}$, and JPEG compression with quality factors $\mathrm{qf} \in \{100, 90, \ldots, 40\}$. These distortions are unseen during training, to simulate the situation of a dataset shift on input from unknown sources.

Figure 2 shows the evaluation results. On top, the detection accuracy is shown. As expected, the detection accuracy decreases with increasing amount of unseen postprocessing. On bottom, the AUC for detecting out-of-distribution inputs from the uncertainty measure is shown. Gaussian blur and JPEG compression can be reliably detected. The OOD detection for additive Gaussian noise is more difficult, the AUC only approaches 0.9 for the highest noise level. Nevertheless, the vast majority of OOD cases can be reliably detected for all three types of postprocessing.

We perform a very similar experiment with a Bayesian Neural Network (BNN), using the architecture and following the protocol of our earlier work [13]. To this end, we split the images of the RAISE1k dataset 80/10/10 into training, validation and testing data, convert the images to grayscale, and create rescaled copies of each image with resizing factors between 1 and 1.5 in steps of 0.05. Fig. 3 looks at the same experiment with a BNN. We then draw from the original image and one of its resized copies $N = 50$ non-overlapping patches of size 256×256 pixels. The detection network is a direct BNN adaptation of the CNN architecture by Bayar and Stamm [34]

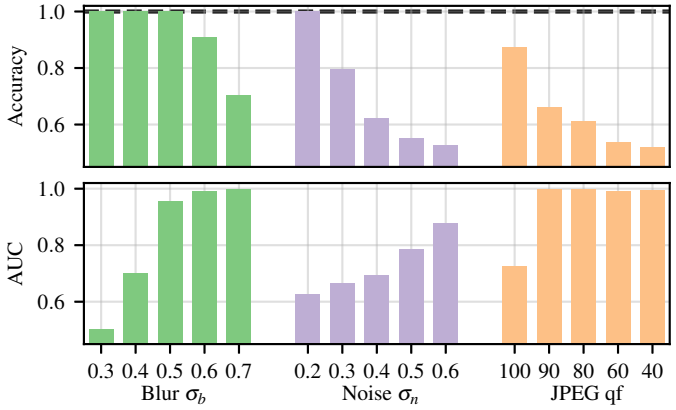

Fig. 2. Resizing detection via Bayesian Logistic Regression on SPAM features. The resized images are subject to various types of postprocessing.

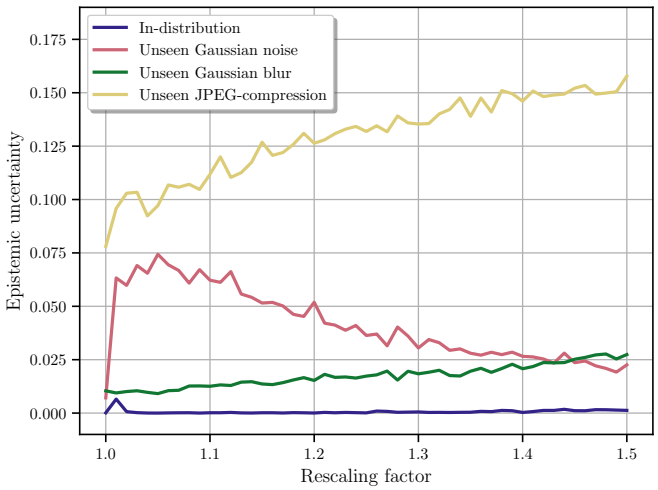

Fig. 3. Epistemic uncertainties of the Bayesian Neural Network for resampling detection under various postprocessing steps.

that we also used in our earlier work [13].

For testing, we additionally apply the unseen distortions of additive Gaussian noise with a standard deviation of 0.20, Gaussian blur with a kernel size of 5×5 and a JPEG compression quality of 85. The resulting epistemic uncertainties for each resizing factor are shown in Fig. 3. The epistemic uncertainty for indistribution data is consistently low across all rescaling factors. Conversely, the epistemic uncertainties of all distortions are notably larger than 0, which can be used to detect a mismatch between the training and test data. It is interesting to observe that different types of distortions lead to different magnitudes of the epistemic uncertainties. JPEG compression causes the highest uncertainties, while Gaussian blur leads to the lowest uncertainties. Furthermore, it is at first glance somewhat counter-intuitive to observe that the uncertainty of Gaussian blur increases with increasing rescaling factor, while the uncertainty of additive Gaussian noise decreases. We hypothesize that Gaussian blur removes interpolation traces that are more notable for higher rescaling factors, while Gaussian noise emulates a high-frequency variation of the image content, which the BNN interprets as a more natural image.

V. Open Questions and Future Research

There are still a number of open questions that need to be addressed for the proposed framework. So far, there is to our knowledge no systematic examination of the relationship between the magnitude of the uncertainty and the distortion of the data. A simple relationship would in the best case allow to introduce thresholds on the uncertainty measure. Moreover, it could be used to further fine-tune a practical tradeoff, namely whether to use a classifier with a slight performance penalty on data with only slight distortions.

Explainability is another important consideration in multimedia forensics, particularly when it is necessary to substantiate the decision of a forensic algorithm in court. In this respect, analytic features and simple classifiers have a clear advantage over the currently quite opaque neural network architectures. Fortunately, the Bayesian framework is applicable to both types of algorithms, as shown in the example of Bayesian Logistic Regression with analytic first digit features and Bayesian Neural Networks that are entirely data-driven. Besides explainability, another advantage of Bayesian Logistic Regression is its relative simplicity in training. Conversely, the training of Bayesian Neural Network is somewhat more brittle, also compared to standard CNNs: since each weight is a distribution, two parameters are at least required per weight to model the mean and the standard deviation. Hence, a BNN has twice the parameters of a CNN of identical architecture.

Another open question are the limits of the additional robustness of Bayesian classifiers. The experiments show that there is excellent potential in the detection of various types of out-of-distribution data. However, there is so far no in-depth study of the smallest distortions that can be detected, or distortion types that are indetectable, e.g., because they occur in the feature space at locations where also in-distribution samples are located. It would furthermore also be interesting to investigate the robustness of Bayesian classifiers to targeted attacks, in particular via adversarial examples.

VI. Conclusions

Machine learning classifiers are challenged by images with unseen processing, and reliable generalization of these classifiers is still an open issue. In this work, we present a recently proposed Bayesian framework for the detection of out-of-distribution samples in multimedia forensics. The framework can be flexibly used on simple classifiers such as Bayesian Logistic Expression and on the considerably more complex Bayesian Neural Networks. We demonstrate on two classical forensic tasks the suitability of the proposed framework, namely for the detection of double JPEG compression and resampling. In both cases, unseen postprocessing can be reliably detected. We hope that the

provided open questions and suggestions for future research further spur the development of reliable forensic classification systems.

REFERENCES

[1] M. K. Johnson and H. Farid, "Exposing digital forgeries by detecting inconsistencies in lighting," in *Proceedings of the 7th workshop on Multimedia and security*, 2005, pp. 1–10.

[2] J. Lukáš, J. Fridrich, and M. Goljan, "Digital Camera Identification From Sensor Pattern Noise," *IEEE Transactions on Information Forensics and Security*, vol. 1, no. 2, pp. 205–214, Jun. 2006.

[3] J. He, Z. Lin, L. Wang, and X. Tang, "Detecting doctored jpeg images via dct coefficient analysis," in *European conference on computer vision*. Springer, 2006, pp. 423–435.

[4] W. Wang and H. Farid, "Exposing digital forgeries in interlaced and deinterlaced video," *IEEE Transactions on Information Forensics and Security*, vol. 2, no. 3, pp. 438–449, 2007.

[5] H. T. Sencar and N. Memon, *Digital Image Forensics*. Springer, 2013.

[6] H. Farid, *Photo forensics*. MIT press, 2016.

[7] D. Cozzolino and L. Verdoliva, "Noiseprint: A CNN-Based Camera Model Fingerprint," *IEEE Transactions on Information Forensics and Security*, vol. 15, pp. 144–159, 2020.

[8] B. Bayar and M. C. Stamm, "Constrained convolutional neural networks: A new approach towards general purpose image manipulation detection," *IEEE Trans. on Inf. Forensics and Security*, vol. 13, no. 11, pp. 2691–2706, 2018.

[9] J. Frank, T. Eisenhofer, L. Schönherr, A. Fischer, D. Kolossa, and T. Holz, "Leveraging Frequency Analysis for Deep Fake Image Recognition," in *International Conference on Machine Learning*, 2020, pp. 3247–3258.

[10] S. Mandelli, N. Bonettini, P. Bestagini, and S. Tubaro, "Training CNNs in presence of JPEG compression: Multimedia forensics vs computer vision," in *IEEE Int. Conf. on Inf. Forensics and Security*, 2020.

[11] B. Lorch, A. Maier, and C. Riess, "Reliable JPEG forensics via model uncertainty," in *IEEE Int. Workshop on Inf. Forensics and Security*, 2020.

[12] A. Kendall and Y. Gal, "What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision?" in *International Conference on Neural Information Processing Systems*, 2017, pp. 5580–5590.

[13] A. Maier, B. Lorch, and C. Riess, "Toward reliable models for authenticating multimedia content: Detecting resampling artifacts with Bayesian neural networks," in *IEEE Int. Conf. on Image Processing*, 2020, pp. 1251–1255.

[14] J. Pan, A. Maier, B. Lorch, and C. Riess, "Reliable Camera Model Identification Through Uncertainty Estimation," in *IEEE International Workshop on Biometrics and Forensics*, May 2021.

[15] M. Boroumand and J. Fridrich, "Deep learning for detecting processing history of images," in *Electronic Imaging, Media Watermarking, Security, and Forensics*, Jan. 2018, pp. 213–1–213–9.

[16] M. Barni, A. Costanzo, E. Nowroozi, and B. Tondi, "CNN-Based Detection of Generic Contrast Adjustment with JPEG Post-Processing," in *IEEE International Conference on Image Processing*, 2018, pp. 3803–3807.

[17] M. Huh, A. Liu, A. Owens, and A. A. Efros, "Fighting fake news: Image splice detection via learned self-consistency," in *European Conference on Computer Vision*, 2018, pp. 101–117.

[18] A. Rössler, D. Cozzolino, L. Verdoliva, C. Riess, J. Thies, and M. Niessner, "FaceForensics++: Learning to Detect Manipulated Facial Images," in *IEEE/CVF International Conference on Computer Vision*, Oct. 2019.

[19] A. Gnanasambandam and S. Chan, "One Size Fits All: Can We Train One Denoiser for All Noise Levels?" in *37th International Conference on Machine Learning*, vol. PMLR 119, 2020, pp. 3576–3586.

[20] Y. Gal, "Uncertainty in deep learning," Ph.D. dissertation, University of Cambridge, 2016.

[21] C. Blundell, J. Cornebise, K. Kavukcuoglu, and D. Wierstra, "Weight Uncertainty in Neural Network," in *International Conference on Machine Learning*, 2015, pp. 1613–1622.

[22] Y. Gal and Z. Ghahramani, "Dropout as a Bayesian Approximation: Representing Model Uncertainty in Deep Learning," in *International Conference on Machine Learning*, 2016, pp. 1050–1059.

[23] R. D. Soberanis-Mukul, N. Navab, and S. Albarqouni, "Uncertainty-based Graph Convolutional Networks for Organ Segmentation Refinement," in *Medical Imaging with Deep Learning*, 2020.

[24] Y. Kwon, J.-H. Won, B. J. Kim, and M. C. Paik, "Uncertainty quantification using Bayesian neural networks in classification: Application to biomedical image segmentation," *Computational Statistics & Data Analysis*, vol. 142, 2020.

[25] D. Hendrycks and K. Gimpel, "A Baseline for Detecting Misclassified and Out-of-Distribution Examples in Neural Networks," in *International Conference on Learning Representations*, 2017.

[26] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On Calibration of Modern Neural Networks," in *International Conference on Machine Learning*, 2017, pp. 1321–1330.

[27] B. K. T. Y.-J. H. Jize Zhang, "Mix-n-Match: Ensemble and Compositional Methods for Uncertainty Calibration in Deep Learning," arXiv preprint, Tech. Rep., 2020, https://arxiv.org/pdf/2002.09437.pdf.

[28] Y. Ovadia, E. Fertig, J. Ren, Z. Nado, D. Sculley, S. Nowozin, J. Dillon, B. Lakshminarayanan, and J. Snoek, "Can You Trust Your Model's Uncertainty? Evaluating Predictive Uncertainty Under Dataset Shift," in *International Conference on Neural Information Processing Systems*, 2019, pp. 13 991–14 002.

[29] B. Lakshminarayanan, A. Pritzel, and C. Blundell, "Simple and scalable predictive uncertainty estimation using deep ensembles," in *International Conference on Neural Information Processing Systems*, 2017, pp. 6402–6413.

[30] B. Lorch, F. Schirrmacher, A. Maier, and C. Riess, "Reliable Camera Model Identification Using Sparse Gaussian Processes," *IEEE Signal Processing Letters*, 2021, *early access*. DOI: 10.1109/LSP.2021.3070206.

[31] I. Amerini, R. Becarelli, R. Caldelli, and A. D. Mastio, "Splicing Forgeries Localization through the Use of First Digit Features," in *IEEE International Workshop on Information Forensics and Security*, 2014, pp. 143–148.

[32] D.-T. Dang-Nguyen, C. Pasquini, V. Conotter, and G. Boato, "RAISE: A Raw Images Dataset for Digital Image Forensics," in *ACM Multimedia Systems Conference*, 2015, pp. 219–224.

[33] B. Wang, X. Kong, and X. You, "Source camera identification using support vector machines," in *Advances in Digital Forensics V*, 2009, pp. 107–118.

[34] B. Bayar and M. Stamm, "On the Robustness of Constrained Convolutional Neural Networks to JPEG Post-compression for Image Resampling Detection," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2017, pp. 2152–2156.