Enhancing Synthetic Generated-Images Detection through Post-Hoc Calibration

Enhancing Synthetic Generated-Images Detection through Post-Hoc Calibration

Giovanna Maria Dimitri
DIISM, Universitá di Siena
Via Roma 56, 53100, Siena (Italy)
giovanna.dimitri@unisi.it

Benedetta Tondi
DIISM, Universitá di Siena
Via Roma 56, 53100, Siena (Italy)
benedetta.tondi@unisi.it

Mauro Barni
DIISM, Universitá di Siena
Via Roma 56, 53100, Siena (Italy)
mauro.barni@unisi.it

Abstract

Post-hoc calibration is an important methodology for improving the reliability of confidence estimates in deep neural networks (DNNs). While modern DNNs achieve state-of-the-art performance across various domains, the output they provide often fails to align with the true likelihood of their predictions, a phenomenon known as miscalibration. This misalignment poses challenges for tasks like uncertainty quantification, score level fusion, and threshold selection in the case of binary detection systems. Furthermore post-hoc calibration methods, such as temperature scaling, Platt scaling, and isotonic regression [17] provide practical solutions to alleviate miscalibration, without retraining. In this work, we propose the use of post-hoc calibration for multimedia forensics applications, by focusing on the detection of synthetically AI generated images. In particular, we show how the application of well known post-hoc calibration methodologies can help to improve the interpretability of AI generated images in terms of likelihood ratios, and can also help to adjust the detection threshold in the presence of different AI generators, considered in the training and calibration sets.

1. Introduction

The rapid advancements of image generation technologies, based on artificial intelligence (AI) techniques like generative adversarial networks (GANs) and diffusion models [7, 12, 18, 21] have increased the rate of development of several image forensics tools capable of distinguishing between pristine and synthetic (AI-generated) images [1–3, 11]. In most cases, image forensic tools are designed to maximize their accuracy, without any effort to enable a truthful interpretation of the predictions they provide, i.e. without any effort to *calibrate* them. In machine learning, calibration refers to the alignment of predicted probabilities with actual outcomes, ensuring that a model's confidence reflects real-world likelihoods [13, 16]. As such, in fact, calibration is a crucial step to enable better decision-making, trust, and interpretability in several applications like risk assessment, classification, anomaly detection and many others [13, 20]. Calibrating neural networks, moreover, can become essential when integrating machine learning models into real-world decision-making systems, as the confidence in the decisions is just as crucial as the decisions themselves [17]. In particular, classification networks should not only be accurate in the predictions, but should also be able to indicate when they are most likely to be wrong. For instance in healthcare domain or in self-driving autonomous systems we would not only ideally have accurate predictions, but for instance in the healthcare sector we would like to aim to have a system in which the confidence of the disease diagnosis is also presented [8, 9, 17].

In the context of forensics applications, calibration has received limited attention. There are in fact, so far, only rare and scattered works highlighting the importance of calibration [19]. For instance in [6] an analysis concerning many different CNN-based methods for the detection of diffusion model detection is carried out. Results show the need of calibration for detectors to work with different generators, and to improve fusion approaches. Similarly in [19] experiments have shown how large AUC values can ensure only that the two distributions can be well separated, but the need for selecting a proper classification threshold still holds.

In this work, we performed preliminary experiments, showing how applying classical calibration methods improves the interpretability of synthetic generated image detection in terms of likelihood ratios. Additionally, we show how calibration can help adjusting the detection thresh-

olds to account for mismatches between training and test datasets. Preliminary experiments, in fact, suggested how calibration can enhance the accuracy of image forensic techniques, with a specific focus on the detection of synthetically AI generated images (GAN or latent diffusion models). That is applying post-hoc calibration to the scores, could help in defining a proper threshold and therefore improve accuracy. The paper is structured as follows. In Section 2 we describe the methodologies used in our experiments. In particular, we describe the detection deep neural network which was subjected to our calibration experiment, together with the calibration methodologies we tested (Section 2.2, and the metric used to measure the effectiveness of calibration (Section 2.3). In Section 3, we describe the datasets used in our experiments. In Section 4, we describe the results we performed and eventually in Section 6 we draw our conclusions and sketch some directions for future works.

2. Methods

Detection of synthetic generated images is a crucial task in forensics. In our work we employed a workflow in which we first trained a ResNet based architecture to predict pristine vs AI generated images. We describe the specific architecture used in Section 2.1. We subsequently applied post-hoc calibration methods on the final prediction scores and evaluated them using the Calibrated Log-Likelihood Ratio score which will be described in Section 2.3. An overall pipeline of our work is reported in Figure 1.

2.1. SRNET Architecture

For the detection of synthetic vs pristine images, we implemented a ResNet50 based model. Since its introduction in 2015, ResNet models have proved to be extremely powerful in performing multiple detection and computer vision related tasks [14]. In our work we applied a modification to the standard ResNet50 model [4] using the architecture proposed in [10]. In particular, for the original architecture, the first convolutional layer is presented with a stride of dimension 2 and therefore performs downsampling. Instead, in our case, and following the model proposed in [4], we set the stride to 1 and therefore the features dimension after the first convolutional block does not change. Overall this architecture, named SRNet, was carefully designed and aimed at minimizing heuristic design elements. It consists of three main parts: a front section for noise residual extraction, a middle section for feature map compactification, and a final classification segment using a fully connected layer with a softmax activation. The architecture uses random initialization for filters and optimizes them end-to-end, making it adaptable across spatial and JPEG embedding domains [4, 10]. For further details please refer to the original paper [4].

2.2. Post-Hoc Calibration Methods

Calibration can be performed using different approaches, considering post-hoc or embedded methods. Post-hoc calibration methods have the goal to calibrate a model after it has been trained and acting therefore directly on the predicted scores. The reason why we decided to follow a post-hoc calibration method, relied on the fact that we assumed the scenario in which a forensic synthetic AI detector is actually trained on a certain training dataset (made of generated and pristine images) and we further want to calibrate it, considering different synthetically AI generated scenarios. Moreover it was shown how post-hoc calibration can perform quite well without a huge amount of data and therefore suffer less from training complexity [17]. This is the reason why in a complex forensic scenario where several generators might want to be detected, considering such approach might be more convenient.

Logistic Regression

Logistic regression can be used as a post-hoc calibration processing technique used to adjust predicted probabilities of a model. The calibration process relies on logistic regression model, where the method adjusts a set of input probabilities, $\hat{p}_i$, to produce calibrated probabilities, $\tilde{p}_i$, using the equations:

$$\tilde{p}_i = \frac{1}{1 + \exp(-(w \cdot \hat{p}_i + b))},$$

where w and b are the weight and bias parameters learned during the calibration phase by minimizing a loss function, typically the negative log-likelihood over a separate calibration dataset.

During training, the parameters w and b are found to ensure that the predicted probabilities align with the observed frequencies of the true labels in the calibration dataset. A well-calibrated model satisfies the condition that for any predicted probability $\tilde{p}_i$, the proportion of positive labels, among all instances with similar probabilities, is approximately $\tilde{p}_i$. Logistic regression calibration is simple yet effective, as it can correct common issues like overconfidence or under–confidence in a model's predictions, enhancing its reliability in probabilistic tasks.

Beta Scaling

Beta scaling is a calibration technique used to adjust the predicted probabilities of a classification model by introducing additional flexibility compared to simpler methods like logistic regression calibration. It extends the basic sigmoid-based (logistic) scaling by incorporating separate parameters for scaling predictions towards both classes. The calibrated probability $\tilde{p}_i$ for a given uncalibrated prediction $\hat{p}_i$

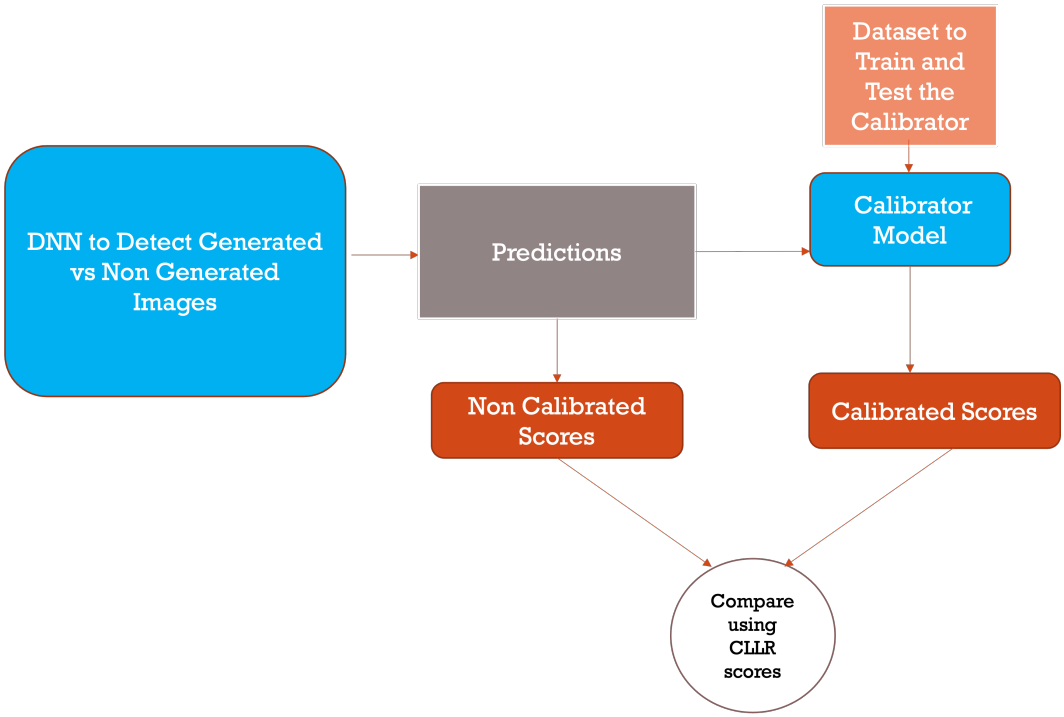

Figure 1. Figure showing the Calibration Pipeline we implemented in our paper

are computed as:

$$\tilde{p}_i = \frac{\hat{p}_i^{\beta}}{\hat{p}_i^{\beta} + (1 - \hat{p}_i)^{\gamma}},$$

where β and γ are scaling parameters learned from a calibration dataset. These parameters adjust the steepness of the curve for each class, allowing for asymmetric corrections to over- or under-confident predictions. The training of β and γ is typically performed by minimizing a loss function such as the negative log-likelihood over a validation or calibration dataset. This method is particularly useful in scenarios where the base model's predicted probabilities are biased or exhibit different levels of miscalibration for each class.
Beta calibration was originally proposed in [15], as an improvement over the more classic approach of Logistic Regression.
Among its main characteristics, it was shown to be suitable also for smaller datasets (being a parametric method) [15]. In the standard logistic regression, in fact, we assume that the ratio of the score distributions, for both binary classes, are similar to the ratio of two Gaussians distributions with the same variances, while in the beta regression case, it is assumed that they can be approximated as the ratio of two beta distributions [15].

Isotonic Regression

Isotonic regression is a well known method used in the context of post-hoc calibration. To solve this, one common approach is to use the Pool Adjacent Violator method (PAV) which is a non-parametric technique used for calibrating probabilities, based on the isotonic regression principle. Unlike parametric methods such as logistic regression, PAV calibration does not assume any specific functional form for the mapping between predicted probabilities and true probabilities.
Given a sequence of predicted probabilities $\hat{p}_i$ for each $i-th$ observation and their corresponding true binary labels, the PAV algorithm adjusts the probabilities to ensure they are monotonic while minimizing a loss function, such as the mean squared error. The output calibrated probabilities, $\tilde{p}_i$, are guaranteed to satisfy the isotonicity constraint. The PAV algorithm works by pooling adjacent probabilities that violate the monotonicity constraint. For each such violation, it replaces the non-monotonic segment with its weighted average, repeating this process until all probabilities are monotonic. This ensures that the calibrated probabilities align closely with the empirical likelihood of the observed labels.

2.3. CLLR Scores

The CLLR (Log-Likelihood Ratio Cost) score is a metric widely used to evaluate the performance of probabilistic binary classification systems, particularly in applications as

for instance speaker verification [5].
It measures the quality of the likelihood ratios (LLRs) output by a system, focusing on their alignment with the ground truth. Specifically, the CLLR score assesses both the discrimination and calibration properties of a classifier.

To evaluate the performance of a calibrated system we will use the Log Likelihood Ratio Cost Function (CLLR). The derivation of CLLR begins from the Cross Entropy (CE) formula:

$$CE = -\frac{1}{N}\sum_{k} \log p(c = c_k|x_k) \tag{1}$$

Weighting the average CE over the samples from each class by an artificial prior $p(c)$ for the class defines Weighted Cross Entropy (WCE):

$$\begin{aligned} WCE = &-\frac{p(c=0)}{N_0}\sum_{k|c_k=0} \log p(c=0|x_k) \\ &-\frac{p(c=1)}{N_1}\sum_{k|c_k=1} \log p(c=1|x_k) \end{aligned} \tag{2}$$

Dividing by the WCE of a perfectly calibrated dummy system and setting the prior to 0.5 defines the Log Likelihood Ratio Cost Function (CLLR):

$$CLLR = \frac{WCE}{-p(c=0)\log p(c=0) - p(c=1)\log p(c=1)} \tag{3}$$

Substituting $p(c=0) = p(c=1) = 0.5$:

$$CLLR = \frac{WCE}{-0.5\log(0.5) - 0.5\log(0.5)} \tag{4}$$

In the formula:

- c is the class of the probe, either 0 or 1 for Non-Target or Target, respectively.
- N_0 and N_1 are the number of Non-Target and Target probes, respectively.
- x_k is the Log Likelihood Ratio (LLR) of probe k.
- $\tilde{p}$ represents the following probabilities:
  - $\tilde{p}(c=0)$ is the probability of a class 0 trial (i.e., target trials).
  - $\tilde{p}(c=1)$ is the probability of a class 1 trial (i.e., non-target trials).
  - $\tilde{p}(c=0|x_k)$ is the probability of class 0 given LLR x_k.
  - $\tilde{p}(c=1|x_k)$ is the probability of class 1 given LLR x_k.

Type	Dataset	Number of Images
Pristine	CelebaHQ	10k
Pristine	FFHQ	10k
Pristine	Tanks	10k
Generated	LSGM (faces)	10k
Generated	StyleGAN3 (faces)	10k
Generated	StyleGAN2 (vehicle)	5k
Generated	Latent Diffusion (vehicles)	5k

Table 1. Description of the dataset used for training the Resnet50 based detection model

A perfect classifier achieves a CLLR score of 0, which indicates ideal discrimination and calibration. Conversely, higher values (closer to 1) suggest poor performance in generating well-calibrated likelihood ratios. The CLLR score can therefore be used successfully to evaluate how well the predicted probabilities or likelihood ratios represent true probabilities.

3. Datasets

In this work we used three datasets. A first dataset was used to train the ResNet based architecture described in 2, for detection of synthetically AI generated images. This first dataset was created collecting a balanced dataset composed by 30000 pristine images and 30000 synthetically AI generated images; details on its composition are presented in Table 1. The dataset was split with a proportion of 7:1:2 for training, validation and testing. To compose it, a series of publicly available dataset, as for instance CelebaHQ, were crawled. An example of face images taken from the CelebaHQ dataset is reported in Figure 2. We further collected a second dataset (which we will call for simplicity Dataset 2) for performing calibration. This second dataset was composed by images from the same AI generators used also in the dataset of 1, but coming from external sources, therefore with no ensurance that the same parameters for generating images were maintained. The details of the calibration dataset are described in Table 2. This was divided into a part which we will call calibration training dataset, and which was used to train the post-hoc calibrator models, and a part called calibration validation dataset used for validating calibrator. A third dataset was also collected, in which we considered synthetically AI generated images, where the generator was the software BING [1] never seen before by the detector (which we will name for simplicity Dataset BING). Also in this case such dataset was used to train the post-hoc calibration models, and a second validation dataset was used to evaluate the performance of the calibrator so trained. In this case both the BING training and validation datasets were composed by 240 faces-images

[1] https://www.bing.com/images

Figure 2. Example of Faces used in Training Dataset of Our Detection model, taken from the CelebaHQ dataset

generated by BING and 392 pristine faces.

4. Experiments and Discussion

4.1. Detection

For what concerns the overall settings of the pristine vs generated images ResNet based architecture, the input size of the images was set to $224 \times 224 \times 3$, batch size of 16 and learning rate of 0.0001. Moreover we used the Adam optimizer and cross-entropy loss. The model was trained for 30 epochs. During the training we performed scaling augmentation on the samples with a probability of 40% (and scaling parameters varying in a range between 0.8 and 1.3 (with a step of 0.1)). The model was chosen according to the one obtaining the best accuracy performances in the validation set. In particular, for the final chosen model, the accuracies performances obtained for the training, validation, and test set respectively were: 0.9673, 0.9624, and 0.9645.

4.2. Calibration on Dataset 2

As previously mentioned we decided to implement a post-hoc calibration approach. More specifically in our experimental setting we implemented three well-known post-hoc calibration techniques: logistic regression, beta calibration and isotonic regression [15, 16]. The pipeline of the calibration experiments worked as follows.
We first considered the outputs of the ResNet50-based detection network and we computed the relative predicted scores on the calibration datasets. Subsequently we trained the three post-hoc calibration methods on such predicted scores vectors, building respectively a logistic, a beta and a isotonic regression calibrator model. We did this step both for the second and third calibration datasets described in 3. The implementation was performed in python version 3.6.9, using the Netcal python library and the sklearn libraries. We therefore evaluated the accuracies and CLLR scores both for the calibration-training datasets and for the calibration-validation datasets. On these second ones, in fact, we considered the scores, applied the calibrator model (without retraining them) and reported their respective CLLR and balance accuracies. Since both calibration datasets suffered from unbalancing, we report always the balanced accuracy (leaving always the threshold to 0.5). The results obtained for the calibration training and validation dataset for Dataset 2, in terms of CLLR and balanced accuracies, are reported in Table 3. The AUC we obtain for the training and validation calibration dataset is in both cases around 0.94. Consider that being a monotonic transformation the ROC curve does not change between the uncalibrated and the calibrated case. Moreover, as an example in Figure 3 we show the behaviour of the predicted scores of the calibration validation dataset before and after calibration (in the Figure we show as an example case the beta calibration effect). As we can observe from Table 3 the CLLR decreases significantly between the uncalibrated and calibrated cases. Moreover the balance accuracy increase, for all of the post-hoc calibration method. This first results suggest how the use of post-hoc calibration can actually help in applying a transformation to the predicted scores, capable of shifting the distribution and therefore obtaining an increased accuracy, without having to change the threshold.

4.3. Testing and Calibrating Dataset BING

We further proved the use of calibration for different out-of-set generators not included in the original training set of the detection model. The newly analysed dataset, includes images generated by the BING AI text-to image generator as described in Section 3. An example of BING generated faces is shown in Figure 5.

More specifically, we performed similar steps as per Dataset 2.We first considered the predicted scores produced by our pre-trained ResNet model for the identification

Type	Dataset	Generator Type	Number of Images
Pristine Faces	Calibration Training	N/A	120
Pristine Vehicles	Calibration Training	N/A	240
Generated	Calibration Training	LSGM	120
Generated	Calibration Training	StyleGAN2	120
Generated	Calibration Training	StyleGAN3	120
Generated	Calibration Training	Latent Diffusion	240
Pristine Faces	Calibration Validation	N/A	120
Pristine Vehicles	Calibration Validation	N/A	120
Generated	Calibration Validation	LSGM	120
Generated	Calibration Validation	StyleGAN2	120
Generated	Calibration Validation	StyleGAN3	120
Generated	Calibration Validation	Latent Diffusion	240

Table 2. Description of the dataset used for training and validation of the calibrator model, that is the one defined Dataset 2

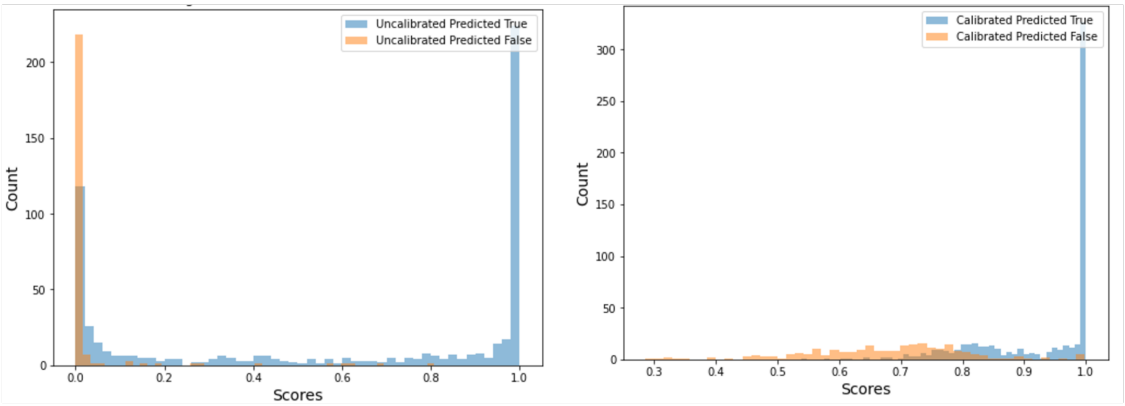

Figure 3. Histograms Distribution of the Scores before and after calibration for the Calibration DATASET 2 using the beta calibration method

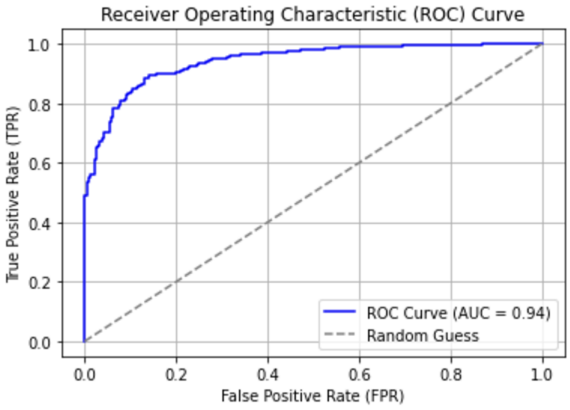

Figure 4. ROC curve for the calibration training dataset 2

of generated/non generated images using the calibration-training of the dataset BING (for the generated images) and a dataset of crawled faces from the web for the pristine cases (392 generated and 240 pristine images). We therefore trained a post-hoc calibration model using this dataset and tested it on the calibration validation dataset for BING.

Results are reported in Table 4. As we can see such results show an interesting behaviour. In the case of images coming from a different generator, in fact, we can see how the accuracy, as expected, drops in the case of the uncalibrated system. However, when we later perform calibration and use the calibration model in order to perform the experiments, we see an improvement in the case of the accuracy, which increases for the calibrated system. The accuracy in this case drops, while the AUC remains above 0.5, more specifically to 0.76.

5. Discussion

There are a few findings and points which we can highlight from our experiments. In terms of comparison between calibration methods, we can see that the lowest CLLR is obtained using post-hoc calibration : both when the calibration training and validation sets are from the same

Data	CLLRUnCal	CLLRLogi	CLLRBeta	CLLRPAV	BAccUncal	BAccCalLogi	BAccCalBeta	BAccCalISO
Cal Train Dataset 2	1.349	0.585	0.588	0.535	0.7311	0.843	0.845	0.844
Cal Validation Dataset 2	0.705	0.569	0.579	0.578	0.714	0.859	0.860	0.849

Table 3. Table reporting the results of the CLLR for the Uncalibrated (UnCal), Logistic Regression (Logi), Beta Calibration (Beta) and Isotonic Regression (ISO) Calibration Method as well as the balanced accuracies for the models. The results reported here are the ones related to experiments performed on the dataset described in Table 2.

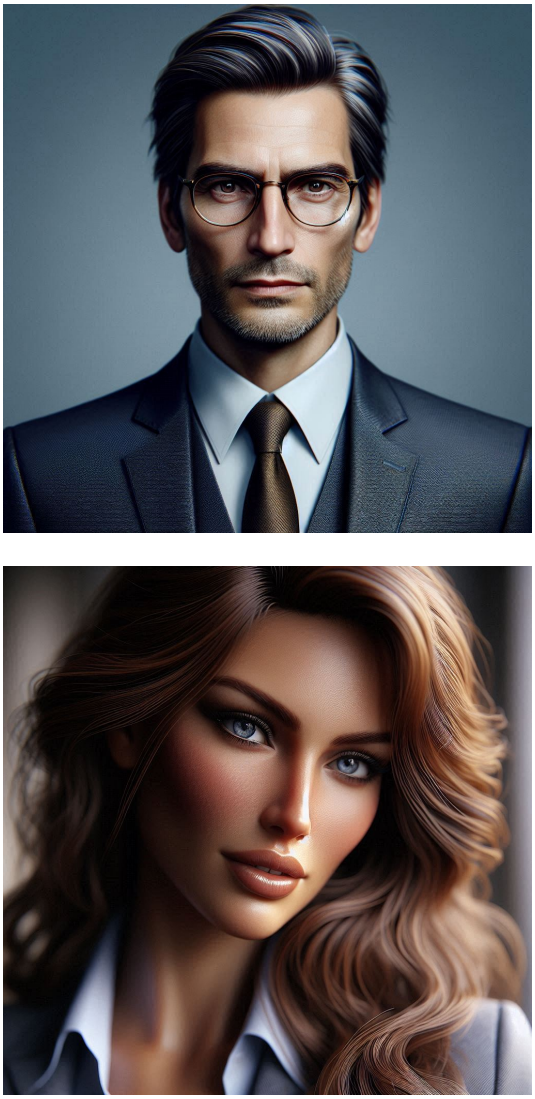

Figure 5. Examples of BING generated images

generators, and also when tested with an out of sample generator (as reported in Table 3 and Table 4). We can further notice another important aspect. Based on experiments,in fact, we can see how testing on an external generator or in general images not aligning with the training distribution, the accuracy improves after post-hoc calibration of the predicted scores. Therefore, our findings, even if limited to only these two preliminary test cases, suggest that it might not be necessary to re-train or fine-tune the detection tools on new data each time a new synthetic generated dataset is introduced, but it may be sufficient to calibrate the tool on a set of new data.

6. Conclusions

Our preliminary study shows the importance of post-hoc calibration in enhancing the reliability of deep learning-based forensic systems, specifically for detecting GAN-generated images. By employing post-hoc calibration methods such as Logistic Regression, Beta Calibration, and the Isotonic Regression algorithm, we observed an improvement the alignment of predictive probabilities with true event likelihoods.

Our experiments highlighted that calibration improved the system performance, as evidenced by reduced CLLR scores and improved AUC and accuracy metrics, even in challenging out-of-distribution scenarios. For instance, calibrated models showed resilience in identifying images from unseen generators like Bing, highlighting the utility of these methods in real-world applications where datasets change over time and new generators become available. A limitation of our present work is actually represented by the use of BING images only as proof of concept for showing the capability of calibration to help in such use case scenarios. However the use of additional AI generators images will be included in future work in order to strengthen the validity of our results and to also study possible biases due to the presence of certain characteristics in generators (as for instance the typical BING images presented in this study). Future work could explore the integration of additional calibration techniques and the impact of the dimensionality of the calibration dataset in the final performances. Additionally, further research into calibration robustness across diverse domains could provide deeper insights into optimizing AI based workflow for forensics applications as well as the comparison of post-hoc calibration with methods such as fine tuning or ensemble learning to strengthen and validate the results obtained in our study.

Data	CLLRUnCal	CLLRLogi	CLLRBeta	CLLRPAV	BAccUncal	BAccCalLogi	BAccCalBeta	BAccCalISO
Train Calibration Bing	3.980	0.900	0.899	0.840	0.603	0.717	0.712	0.744
Validation Calibration Bing	4.224	0.910	0.908	0.854	0.617	0.680	0.668	0.706

Table 4. Table reporting the results when the detection model is tested on a different out of scenario test set, in particular in the Dataset 3 made of images generated by BING. We can see therefore the effect of the calibration on the accuracy and AUC scores.

Acknowledgments

This work was partially supported by SERICS project (PE00000014) under the MUR National Recovery and Resilience Plan, funded by the European Union - NextGenerationEU.

References

[1] Lydia Abady, Giovanna Maria Dimitri, and Mauro Barni. A one-class classifier for the detection of gan manipulated multi-spectral satellite images. *Remote Sensing*, 16(5):781, 2024. 1

[2] Mauro Barni, Andrea Costanzo, Giovanna Maria Dimitri, and Benedetta Tondi. A gpu-accelerated algorithm for copy move detection in large-scale satellite images. In *Image and Signal Processing for Remote Sensing XXIX*, volume 12733, pages 220–234. SPIE, 2023. 1

[3] Jordan J Bird and Ahmad Lotfi. Cifake: Image classification and explainable identification of ai-generated synthetic images. *IEEE Access*, 2024. 1

[4] Mehdi Boroumand, Mo Chen, and Jessica Fridrich. Deep residual network for steganalysis of digital images. *IEEE Transactions on Information Forensics and Security*, 14(5):1181–1193, 2018. 2

[5] Niko Brummer and David A Van Leeuwen. On calibration of language recognition scores. In *2006 IEEE Odyssey-The Speaker and Language Recognition Workshop*, pages 1–8. IEEE, 2006. 4

[6] Riccardo Corvi, Davide Cozzolino, Giada Zingarini, Giovanni Poggi, Koki Nagano, and Luisa Verdoliva. On the detection of synthetic images generated by diffusion models. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE, 2023. 1

[7] Florinel-Alin Croitoru, Vlad Hondru, Radu Tudor Ionescu, and Mubarak Shah. Diffusion models in vision: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(9):10850–10869, 2023. 1

[8] Andre Esteva, Brett Kuprel, Roberto A Novoa, Justin Ko, Susan M Swetter, Helen M Blau, and Sebastian Thrun. Dermatologist-level classification of skin cancer with deep neural networks. *nature*, 542(7639):115–118, 2017. 1

[9] Tilmann Gneiting, Fadoua Balabdaoui, and Adrian E Raftery. Probabilistic forecasts, calibration and sharpness. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 69(2):243–268, 2007. 1

[10] Diego Gragnaniello, Davide Cozzolino, Francesco Marra, Giovanni Poggi, and Luisa Verdoliva. Are gan generated images easy to detect? a critical analysis of the state-of-the-art. In *2021 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6. IEEE, 2021. 2

[11] Diego Gragnaniello, Francesco Marra, and Luisa Verdoliva. Detection of ai-generated synthetic faces. In *Handbook of digital face manipulation and detection: From deepfakes to morphing attacks*, pages 191–212. Springer International Publishing Cham, 2022. 1

[12] Luca Guarnera, Oliver Giudice, Francesco Guarnera, Alessandro Ortis, Giovanni Puglisi, Antonino Paratore, Linh MQ Bui, Marco Fontani, Davide Alessandro Coccomini, Roberto Caldelli, et al. The face deepfake detection challenge. *Journal of Imaging*, 8(10):263, 2022. 1

[13] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q Weinberger. On calibration of modern neural networks. In *International conference on machine learning*, pages 1321–1330. PMLR, 2017. 1

[14] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 2

[15] Meelis Kull, Telmo M Silva Filho, and Peter Flach. Beyond sigmoids: How to obtain well-calibrated probabilities from binary classifiers with beta calibration. *Electronic Journal of Statistics*, 11(2):5052–5080, 2017. 3, 5

[16] Alexandru Niculescu-Mizil and Rich Caruana. Predicting good probabilities with supervised learning. In *Proceedings of the 22nd international conference on Machine learning*, pages 625–632, 2005. 1, 5

[17] Amir Rahimi, Kartik Gupta, Thalaiyasingam Ajanthan, Thomas Mensink, Cristian Sminchisescu, and Richard Hartley. Post-hoc calibration of neural networks. *arXiv preprint arXiv:2006.12807*, 2, 2020. 1, 2

[18] Walter J Scheirer, Lalit P Jain, and Terrance E Boult. Probability models for open set recognition. *IEEE transactions on pattern analysis and machine intelligence*, 36(11):2317–2324, 2014. 1

[19] Diangarti Tariang, Riccardo Corvi, Davide Cozzolino, Giovanni Poggi, Koki Nagano, and Luisa Verdoliva. Synthetic Image Verification in the Era of Generative Artificial Intelligence: What Works and What Isn't There yet . *IEEE Security & Privacy*, 22(03):37–49, May 2024. 1

[20] Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. Calibrate before use: Improving few-shot performance of language models. In *International conference on machine learning*, pages 12697–12706. PMLR, 2021. 1

[21] Tao Zhou, Qi Li, Huiling Lu, Qianru Cheng, and Xiangxiang Zhang. Gan review: Models and medical image fusion applications. *Information Fusion*, 91:134–148, 2023. 1