

Global Contrast Enhancement Detection via Deep Multi-Path Network

Cong Zhang, Dawei Du*, Lipeng Ke, Honggang Qi
School of Computer and Control Engineering
University of Chinese Academy of Sciences, Beijing, China
Email: zhangcong171@mails.ucas.ac.cn

Siwei Lyu
Computer Science Department
University at Albany, SUNY, Albany, USA
Email: slyu@albany.edu

Abstract—Identifying global contrast enhancement in an image is an important task in forensics estimation. Several previous methods analyze the “peak-gap” fingerprints in graylevel histograms. However, images in real scenarios are often stored in the JPEG format with middle/low compression quality, resulting in less obvious “peak-gap” effect and then unsatisfactory performance. In this paper, we propose a novel deep Multi-Path Network (MPNet) based approach to learn discriminative features from graylevel histograms. Specifically, given the histograms, their high-level peaks and gaps information can be exploited effectively after several shared convolutional layers in the network, even in middle/low quality compressed images. Moreover, the proposed multi-path module is able to focus on dealing with specific forensics operations for more robustness on image compression. The experiments on three challenging datasets (*i.e.*, Dresden, RAISE and UCID) demonstrate the effectiveness of the proposed method compared to existing methods.

I. INTRODUCTION

With the rapid development of digital imaging devices, it is important to verify the authenticity of digital images, as they can be easily manipulated using graphics editing software (*e.g.*, Adobe Photoshop). Detecting global contrast enhancement can provide important cues on the authenticity of a digital image. Contrast enhancement is a nonlinear function of pixel values that changes the overall distribution of the pixel intensities in a image, such as gamma correction, sigmoid stretching and histogram equalization. Global contrast enhancement operations may not be the direct result of tampering, but many image forgeries involve such operations to hide the traces of other tampering operations such as region splicing or cloning.

When a contrast enhancement operation is applied to an image, its pixel values and histogram bins undergo a nonlinear mapping that leaves a distinct “peak-gap” effect on the pixel histograms. These peaks and gaps, an example can be seen in Fig. 1(b) and (c), can be used as fingerprints to identify the contrast enhancement operation [1]–[4]. These methods work well for uncompressed images, but their performances are not satisfactory when the digital images are in the JPEG format [5] with middle/low compression quality. Particularly, low quality JPEG compression corresponds to a smoothing of pixel values, which weakens peak and gap bins in graylevel histograms and make them smooth again (see Fig. 1(d) and (f)), resulting

Dawei Du is the corresponding author.

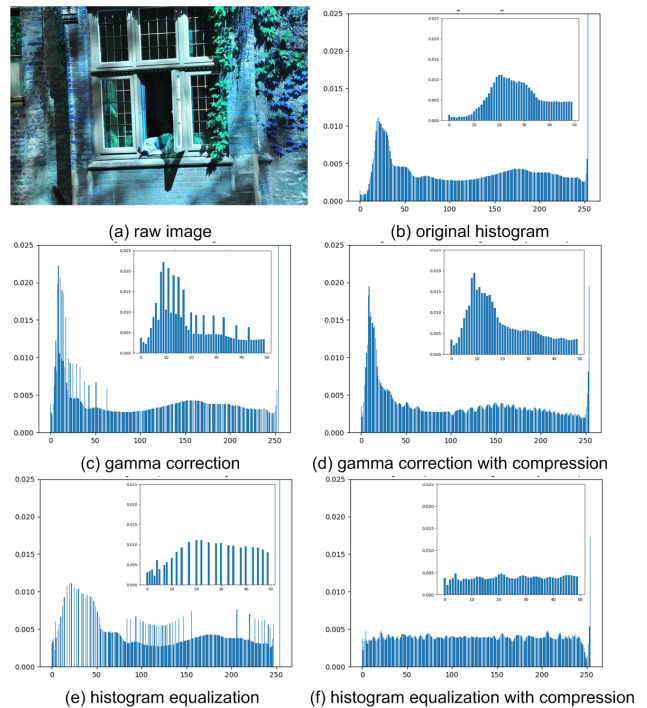


Fig. 1. Examples of global contrast enhancement operations.

in failures for existing global contrast enhancement detection methods. Furthermore, as seen in Fig. 1(c) and (e), the multiple source of forensics operations (*e.g.*, gamma correction and histogram equalization) usually has different kinds of peak-gap distributions. It is also difficult to detect them effectively with fixed rule based thresholds [3].

To solve these issues, in this paper, we propose a deep Multi-Path Network (MPNet) based method to detect global contrast enhancement operations. In contrast to existing methods, our network can exploit more discriminative features in consecutive bins of graylevel histograms to represent peaks and gaps, even when the original peak and gap features have been diminished by JPEG compression. Moreover, to detect different forensics operation effectively, we develop a multi-path module in the network, in which each path is to learn representations for a specific contrast enhancement operation. Ahead of these paths, we use several shared convolutional

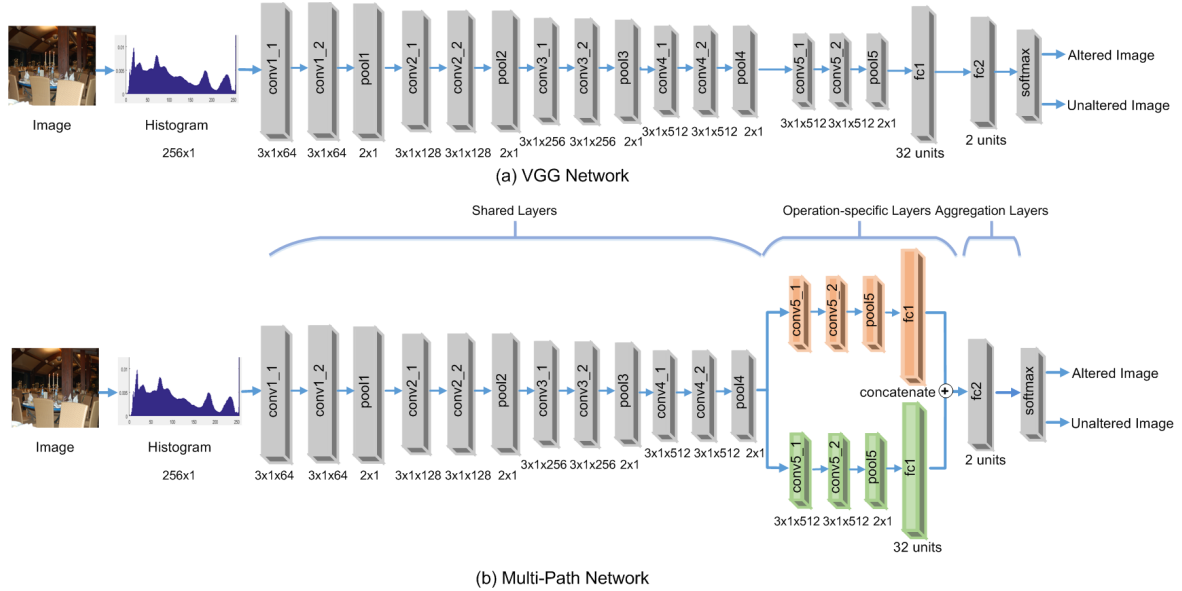


Fig. 2. Architecture of (a) VGG network and (b) the proposed multi-path network. The red and green path are used to learn features from gamma correction and histogram equalization operation, respectively.

layers to capture common low-level features of histograms. Notably, the back-propagation training algorithm does not distinguish these paths from each other, and thus cannot guarantee each path to learn specific representation for the corresponding forensics operation. Therefore we employ a multi-stage training method for the proposed network. First, we train the shared layers using VGG network; then we train the paths individually based on samples of corresponding operation types; finally we combine the multiple paths and output the label whether the image is altered. The contributions of this paper are summarized below:

- We propose a novel deep Multi-Path Network (MPNet) for global contrast enhancement detection, where each path is used to learn specific forensics representation.
- We develop a multi-stage training method to enable each path in the network to focus on one forensics operation.
- We perform extensive experiments on challenging datasets to evaluate our global contrast enhancement detection method against existing methods on JPEG images.

II. OUR METHOD

A. Multi-Path Network

The network structure we use is inspired from the VGG model [6]. As shown in Fig. 2(a), the VGG network has five feature extraction modules including two convolutional layers followed by the max-pooling layer. Then, two fully connected layers are added to the end of the network.

Different from the original VGG model that uses an image as input, the input to our network receives the histogram of the image with the size of 256×1 . Since global contrast enhancement affects through pixel histograms, the feature extracted from the histogram can generally represent the manipulation type. Moreover, it only requires fewer computation and storage

resources to process histograms than full images. As illustrated in Fig. 2(b), the proposed network consists of three parts:

- **Shared layers.** The shared layers contain eight convolutional layers ($conv1_1$ - $conv4_2$), and are used to capture shared low-level features of generic image contrast enhancement operation.
- **Operation-specific layers.** Operation-specific layers consists of several paths, each of which is constructed by two convolutional layers and one fully connected layer ($conv5_1$ - $conv5_2$, $fc1$). Different contrast enhancement operation denotes different path without shared weights.
- **Aggregation layers.** The aggregation layers learn to combine the outputs of preceding paths into a discriminative feature representation using a concatenation layer and a fully-connected layer ($fc2$). Then the binary classification between altered and unaltered images is conducted by a softmax loss. Specifically, we minimize the average loss E between the true class labels (*i.e.*, unaltered and altered). The network outputs using the following loss function is defined as

$$E = \frac{1}{N} \sum_{i=1}^N \sum_{k=1}^c \ell_i^k \log(y_i^k) + \frac{\lambda}{2} \|\omega\|_2, \quad (1)$$

where the first term is the cross entropy loss, and the second term is the L2 regularization to prevent over-fitting of aggregation layers. ℓ_i^k and y_i^k are the true label and the network output of the i -th image at the k -th class with N training images and c neurons in the fully-connected layer, respectively. Parameter λ is the balancing factor and ω is the vectored weights of the $fc2$ layer.

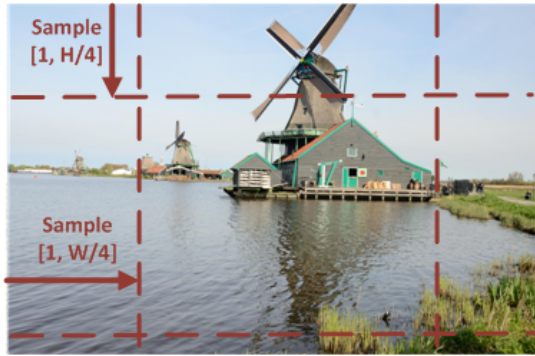


Fig. 3. Data augmentation.

B. Data Augmentation

We first briefly describe the way we generate training data. In this work, the positive samples are defined as the manipulated images, while the negative samples are the unaltered images. For data augmentation, we randomly select the content of each raw image to generate 5 cropped images as the positive examples. Let W be the width and H be the height of the raw image. As shown in Fig. 3, we first set (x, y) as the coordinate of the left top point of the cropped image, in which x is randomly chosen between 1 and $W/4$, and y is randomly chosen between 1 and $H/4$. We randomly choose the width of the cropped image w between 50 and $W - x$. To ensure the aspect ratios of the cropped images, we randomly choose the height of the cropped region between $0.7w$ and $1.3w$.

We generate negative examples based on positive examples. We first randomly choose the image enhancement operation type, in this work we consider the gamma correction and the histogram equalization. The histogram equalization has no parameters. The parameter of the gamma correction is randomly chosen between 0.4 and 2.1 with the step 0.1, *i.e.*, $[0.4, 0.5, \dots, 2.1]$. Then the processed images are stored in the JPEG format with compression quality 95. All the above parameters are set empirically.

C. Multi-stage Training

The back-propagation training algorithm in deep learning does not distinguish the operation-specific layers from each other, which is hard to learn specific representations for the corresponding forensics operation in each path. Therefore, we develop a three-stage training method to deal with this problem. The overall training process is summarized in Algorithm 1.

Stage 1: training shared layers. The first eight convolutional layers (*i.e.*, $conv1_1$ - $conv4_2$ in Fig. 2(b)) are used to capture shared information of histogram under different forensics operations. To determine the weights, we add the rest part of VGG model in Fig. 2(a) to construct a one-path network and train it using back-propagation strategy. After several epoches, the shared convolutional layers are determined.

Stage 2: training operation-specific layers. Once the shared layers are trained, we propose to train two paths individually.

Algorithm 1 Multi-stage training method.

Input: altered and unaltered samples

Output: multi-path network

- 1: Given all kinds of altered and unaltered samples, we train the VGG model in Fig. 2(a) and fix the bottom layers ($conv1_1$ - $conv4_2$);
 - 2: **for** each contrast enhancement operation **do**
 - 3: We remove the other paths of operation-specific layers and initialize the weights of the current path ($conv5_1$ - $conv5_2$, $fc1$) using MPNet in Fig. 2(b);
 - 4: Given each kind of altered and unaltered samples, we train the operation-specific layers using MPNet in Fig. 2(b);
 - 5: **end for**
 - 6: We enable all the operation-specific layers in MPNet in Fig. 2(b);
 - 7: Fixing the previous layers ($conv1_1$ - $fc1$), we train the aggregation layers ($fc2$) using MFNet in Fig. 2(b).
-

First, with fixed shared layers, we train each operation-specific layer by removing the other counterparts. Thus the multi-path network degenerates into one-path network, namely VGG network in Fig. 2(a). Then, the operation-specific layer is initialized with random weights. Finally, the network is updated based on a mini-batch that consists of the training samples from positive samples under each forensics and negative samples. The procedure is not finished until the network is converged.

Stage 3: training aggregation layers. After obtaining the operation-specific layers, we combine them to learn a discriminative feature representation. As shown in Fig. 2(b), we concatenate the operation-specific paths and learn their aggregation weights by fully-connected layer (*i.e.*, $fc2$). The last softmax layer is used for altered/unaltered image classification using the loss in (1).

III. EXPERIMENTS

To evaluate the effectiveness of the proposed method, extensive experiments are performed and we compare our method to state-of-the-art Stamm *et al.*'s method [2]¹. Besides, we train two SVM models [7] (including linear and RBF kernel, denoted as SVM_Linear and SVM_RBF) as baselines.

Datasets. The Dresden dataset² [8] is used to evaluate the above methods. It consists of nature images, dark/flatfield frames and JPEG scene captured in various indoor and outdoor scenes. We randomly select 16,000 nature images for training, 401 nature images for validation and 1,851 JPEG scene images for testing. We do not select dark/flatfield frames in the experiment because they include no semantic objects. To demonstrate the generalization ability of the method, we also use the RAISE dataset [9] and the Uncompressed Colour Image Database (UCID) [10] for testing. The RAISE dataset³ contains 5,999 uncompressed high-resolution images, which are guaranteed to be camera-native. The UCID dataset⁴ in-

¹We do not compare with Cao *et al.*'s method [3] because it is failed to detect gap numbers based on the altered image after JPEG compression.

²<http://forensics.inf.tu-dresden.de/ddimgdb/publications/ddimgdb>

³<http://mmlab.science.unitn.it/RAISE/download.html>

⁴<http://jasoncantarella.com/downloads/ucid.v2.tar.gz>

cludes 886 available uncompressed images on various topics such as natural scenes, man-made objects, indoors and outdoors. Using the raw images, we generate contrast enhanced images as described previously. The images are first transformed using contrast enhancement operations and then compressed with a quality factor QF .

Metric. To evaluate the global contrast enhancement detection methods, each test image is classified by determining if it is contrast-enhanced or not using a series of decision thresholds. We evaluate them by measuring the *true positive rate* and the *false positive rate*. The Receiver Operating Characteristic (ROC) curves are also generated to calculate the Area Under the ROC Curve (AUC) score for ranking them. Moreover, we calculate the probabilities of detection (P_d) and false alarm (P_{fa}) determined by thresholds (*i.e.*, 0.01, 0.05, 0.1) as the percentage of the enhanced images correctly classified and that of the unaltered images incorrectly classified, respectively.

Implementation Details. Similar to [2], the green channel of each testing image for training and testing. We compress images by applying the Python function *imwrite* in the *cv2* toolbox at different quality factors. When training our CNN, we set the batch size equal to 256 and optimize the parameters of the network with the Adadelta strategy [11]. We use Xavier algorithm [12] for weight initialization. We shuffle the training set between epochs, and use the early stopping strategy during the training process. The balancing factor in (1) is set as $\lambda = 0.01$. The learning rate is initially set as 0.4, and decays when the accuracy of the validation set is converged. We train shared layers in 20 epochs. The network is implemented using Tensorflow⁵ on a machine with a 3.50 GHz Intel i7 5930K processor and 48 GB memory and a NVIDIA GTX 1080 Graphical Card.

A. Performance Comparison

As shown in Fig. 4, we present the performance of different contrast enhancement (*i.e.*, histogram equalization and gamma correction) detection of our MPNet method and other compared methods on the Dresden-test, RAISE and UCID datasets. MPNet achieves the best AUC scores in identifying JPEG compression with $QF = 95$ compared to existing methods in three datasets. Moreover, our algorithm achieves higher detection rate $P_d > 0.6$ even under low $P_{fa} = 0.01$. This is attributed to the ability of our network to exploit relations among several consecutive bins in histogram. On the other hand, the SVM models with histogram input fail to consider such relations, leading to inferior performance. Besides, Stamm *et al.*'s method [2] detects contrast enhancement without training, so that it is hard to extract discriminative features from histograms in noisy situations.

B. Discussion

We further perform experiments to study the influence of various important factors of MPNet on the performance.

⁵We make the source codes of our method and the experimental results available on our website: <https://sites.google.com/site/daviddo0323/>.

TABLE I
COMPARISON BETWEEN HISTOGRAM-BASED AND IMAGE-BASED VGG MODELS IN UCID DATASET WITH $QF = 95$.

	AUC score	Speed	# of Param.	memory
VGG-Hist	93.50	2.03s	3.11M	0.70 MB
VGG-Image	94.14	2.60s	9.73M	59.27 MB

Effectiveness of histogram input. To confirm the assumption that the histogram can represent the significant information of the image, we separately use the histogram and the image samples to train the VGG network. In terms of image-based VGG, we crop the image samples with the size of 224×224 randomly based on the resized image with the size of 256×256 . According to the results in Table I, the two models achieve comparable performance, but histogram-based VGG obtains considerable improvement on time and memory expenses per sample.

Influence of JPEG quality. We explore the performance of the proposed method with different JPEG quality. In the experiment, we use the trained images with $QF = 95$ and test the images with $QF = 90, 70, 50, 30$. For comprehensive evaluation, the ROC curves on different forensics operations and QFs are presented in Fig. 5. With the compression degree deeper, the result still shows this method can be instructional. For the middle/low quality factor ($QF \leq 50$), our method keeps stable and outperforms other methods in a large margin. Stamm *et al.*'s method [2] achieves less than 50 AUC score because of failures in detecting peak-gap features when $QF \leq 50$. These results show MPNet performs well on the detection of the global contrast enhancement operation even when the image is stored in the JPEG format with middle/low quality.

Effectiveness of multi-path network. To demonstrate the effectiveness of multi-path network, we implement several VGG models with different output labels, namely VGG-2, VGG-2D, VGG-3 and VGG-3D. Specifically, VGG-2 and VGG-3 correspond to the network with the output label being altered/unaltered and gamma/histogram/unaltered, respectively. VGG-2D and VGG-3D indicate the number of weights from *conv5_1* to *fc1* is doubled to compare with the two paths in our method. As shown in Table II, the multi-path network improves the accuracy moderately. Moreover, our method provides higher detection rate with the same number of false alarms. Besides, the performance comparison between VGG-2 and VGG-3 indicates that it is hard to learn efficient discriminative representation of different forensics operation by just multi-label output configuration. Meanwhile, increasing the number of parameters of the network brings a little performance improvement. To sum up, the proposed multi-path scheme is able to accurately detect several types of single contrast enhancement operations using the same network architecture and number of parameters.

Effectiveness of aggregation layers. The aggregation layers are used to combine the multi-path representation. We remove

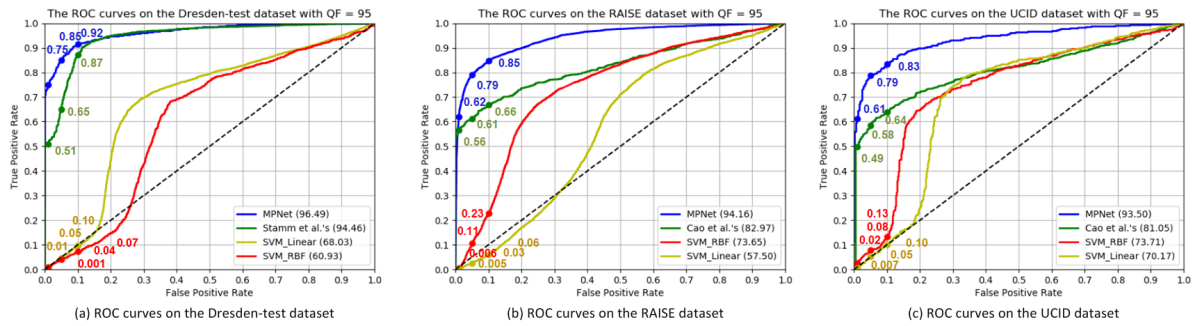


Fig. 4. ROC curves of compared methods on Dresden-test, RAISE and UCID datasets with $QF = 95$. We present the AUC score in the legend.

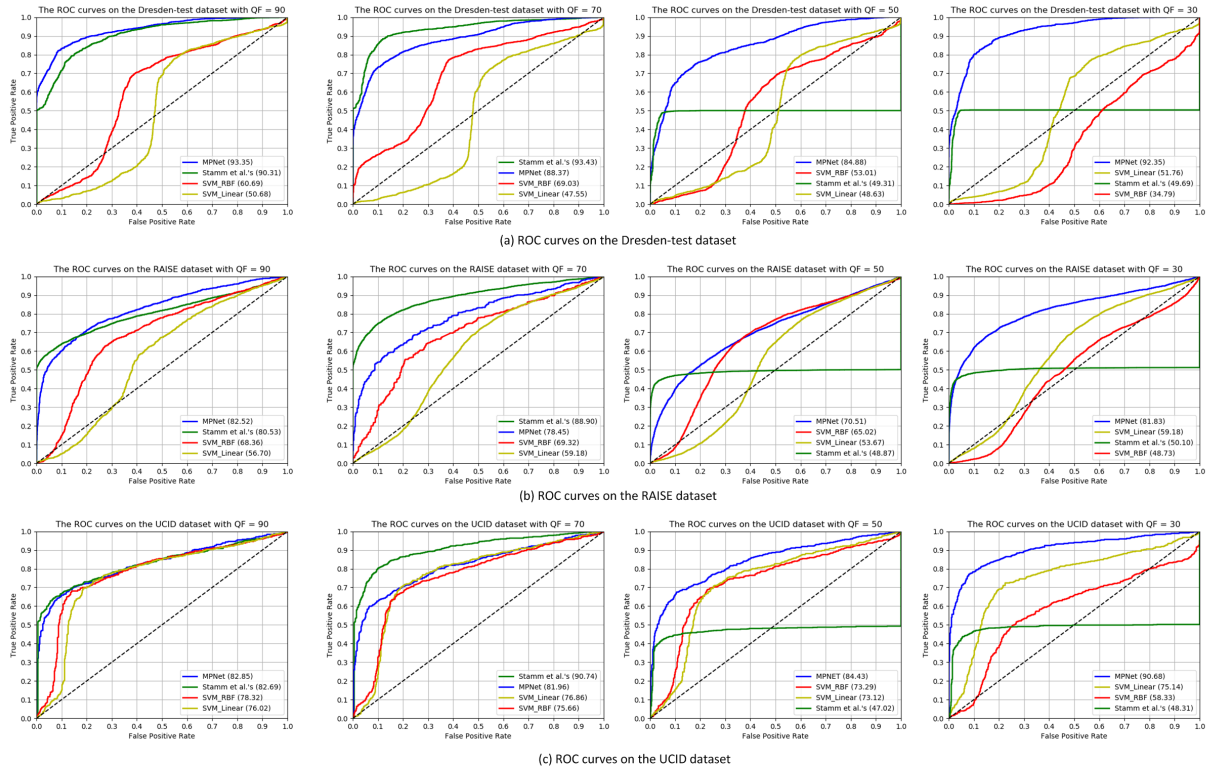


Fig. 5. ROC curves of compared methods on the (a) Dresden-test, (b) RAISE and (c) UCID datasets with different quality factors ($QF = 90, 70, 50, 30$). We present the AUC score in the legend.

the fully-connected layer and output the label based on the mean softmax score among all the paths, denoted as MPNet-mean. From the results in Table II, the aggregation layers can learn the discriminative feature integrated from multiple paths to detect contrast enhancement effectively.

IV. CONCLUSION

In this paper, we propose a new method to detect global contrast enhancement based on a deep multi-path network. The network can exploit more discriminative features in consecutive bins to represent peaks and gaps, even when the original peak and gap features have been diminished by JPEG compression. Moreover, the multi-path module can further improve the accuracy of forensics detection in different com-

pression quality. Experimental results on the Dresden, RAISE and UCID datasets show the proposed deep model works well for contrast enhancement detection, especially when the image undergoes JPEG compression with middle/low quality. In the future work, there are several directions for research. First, we plan to explore the performance of our method against anti-forensic techniques [13], [14]. Second, we would expand our method to detect local contrast enhancement.

ACKNOWLEDGMENT

This work was supported in part by the National Natural Science Foundation of China under Grant 61472388 and Grant 61771341, in part by the US Defense Advanced Research Projects Agency under Grant FA8750-16-C-0166.

TABLE II

THE PERFORMANCE COMPARISON ON THE DRESDEN-TEST, RAISE AND UCID DATASETS FOR DIFFERENT QUALITY FACTORS (*i.e.*, $QF = 90, 70, 50, 30$). THE BEST PERFORMER IS HIGHLIGHTED IN BOLD FONT.

JPEG QF = 90							JPEG QF = 70						
Dresden-test	MPNet	MPNet-mean	VGG-2	VGG-3	VGG-2D	VGG-3D	Dresden-test	MPNet	MPNet-mean	VGG-2	VGG-3	VGG-2D	VGG-3D
AUC score	93.35	88.03	93.83	94.24	92.54	93.75	AUC score	88.37	80.96	85.53	85.25	84.74	83.02
$P_d(P_{fa} = 0.01)$	0.6283	0.2141	0.5815	0.7176	0.5436	0.6797	$P_d(P_{fa} = 0.01)$	0.4358	0.1192	0.2919	0.4777	0.2901	0.3588
$P_d(P_{fa} = 0.05)$	0.7210	0.6138	0.7388	0.7712	0.7338	0.7662	$P_d(P_{fa} = 0.05)$	0.5977	0.4314	0.5073	0.5547	0.5122	0.4911
$P_d(P_{fa} = 0.1)$	0.8326	0.7227	0.8281	0.8326	0.8002	0.8186	$P_d(P_{fa} = 0.1)$	0.7333	0.5910	0.6350	0.6317	0.6066	0.5943
RAISE	MPNet	MPNet-mean	VGG-2	VGG-3	VGG-2D	VGG-3D	RAISE	MPNet	MPNet-mean	VGG-2	VGG-3	VGG-2D	VGG-3D
AUC score	81.67	76.08	81.17	79.57	79.49	80.12	AUC score	78.13	73.34	75.56	73.43	75.25	74.85
$P_d(P_{fa} = 0.01)$	0.4146	0.3328	0.2924	0.0101	0.2178	0.3225	$P_d(P_{fa} = 0.01)$	0.2285	0.2812	0.1289	0.0098	0.1104	0.1055
$P_d(P_{fa} = 0.05)$	0.5145	0.4615	0.4849	0.1268	0.4682	0.4833	$P_d(P_{fa} = 0.05)$	0.4023	0.4062	0.3555	0.0605	0.3398	0.3301
$P_d(P_{fa} = 0.1)$	0.5848	0.5179	0.5709	0.3268	0.5625	0.5631	$P_d(P_{fa} = 0.1)$	0.5176	0.4395	0.4805	0.1816	0.4707	0.4648
UCID	MPNet	MPNet-mean	VGG-2	VGG-3	VGG-2D	VGG-3D	UCID	MPNet	MPNet-mean	VGG-2	VGG-3	VGG-2D	VGG-3D
AUC score	83.63	81.17	81.55	81.27	80.30	81.23	AUC score	84.13	81.63	80.95	77.55	79.92	80.53
$P_d(P_{fa} = 0.01)$	0.5417	0.5221	0.4896	0.0238	0.4596	0.4831	$P_d(P_{fa} = 0.01)$	0.5039	0.4974	0.4323	0.0077	0.3646	0.3737
$P_d(P_{fa} = 0.05)$	0.6003	0.5807	0.5859	0.1190	0.5859	0.5820	$P_d(P_{fa} = 0.05)$	0.5885	0.5898	0.5573	0.0913	0.5885	0.5352
$P_d(P_{fa} = 0.1)$	0.6419	0.6341	0.6289	0.2751	0.6549	0.6367	$P_d(P_{fa} = 0.1)$	0.6536	0.6445	0.6042	0.1992	0.6328	0.5977
JPEG QF = 50							JPEG QF = 30						
Dresden-test	MPNet	MPNet-mean	VGG-2	VGG-3	VGG-2D	VGG-3D	Dresden-test	MPNet	MPNet-mean	VGG-2	VGG-3	VGG-2D	VGG-3D
AUC score	84.88	77.33	76.17	75.23	75.24	74.63	AUC score	92.35	85.16	88.48	89.03	88.59	87.64
$P_d(P_{fa} = 0.01)$	0.2444	0.0444	0.1083	0.3477	0.1074	0.2556	$P_d(P_{fa} = 0.01)$	0.4135	0.0856	0.3114	0.4860	0.3357	0.4007
$P_d(P_{fa} = 0.05)$	0.4453	0.2684	0.2796	0.4118	0.3052	0.3956	$P_d(P_{fa} = 0.05)$	0.6200	0.3689	0.5273	0.5820	0.5709	0.5530
$P_d(P_{fa} = 0.1)$	0.6490	0.4939	0.4364	0.4916	0.4375	0.4749	$P_d(P_{fa} = 0.1)$	0.8013	0.6283	0.6819	0.7026	0.7081	0.6685
RAISE	MPNet	MPNet-mean	VGG-2	VGG-3	VGG-2D	VGG-3D	RAISE	MPNet	MPNet-mean	VGG-2	VGG-3	VGG-2D	VGG-3D
AUC score	73.68	69.28	66.07	64.13	65.94	63.89	AUC score	82.31	74.95	78.23	74.96	78.93	74.96
$P_d(P_{fa} = 0.01)$	0.1602	0.1248	0.0858	0.0037	0.0822	0.0948	$P_d(P_{fa} = 0.01)$	0.3145	0.2128	0.2843	0.0066	0.2572	0.0066
$P_d(P_{fa} = 0.05)$	0.3789	0.2977	0.2441	0.0229	0.2420	0.2824	$P_d(P_{fa} = 0.05)$	0.4963	0.3825	0.4531	0.0387	0.4480	0.0387
$P_d(P_{fa} = 0.1)$	0.4635	0.3956	0.3412	0.0798	0.3599	0.3645	$P_d(P_{fa} = 0.1)$	0.5895	0.4701	0.5372	0.1424	0.5712	0.1424
UCID	MPNet	MPNet-mean	VGG-2	VGG-3	VGG-2D	VGG-3D	UCID	MPNet	MPNet-mean	VGG-2	VGG-3	VGG-2D	VGG-3D
AUC score	85.80	83.49	82.36	80.38	80.67	81.78	AUC score	92.07	85.17	89.46	86.88	90.02	88.67
$P_d(P_{fa} = 0.01)$	0.5273	0.5365	0.3711	0.0078	0.3841	0.3060	$P_d(P_{fa} = 0.01)$	0.5977	0.5547	0.5013	0.0119	0.5391	0.4297
$P_d(P_{fa} = 0.05)$	0.6302	0.6328	0.5781	0.0727	0.5638	0.5273	$P_d(P_{fa} = 0.05)$	0.7396	0.6862	0.6706	0.1675	0.7057	0.6484
$P_d(P_{fa} = 0.1)$	0.6849	0.6927	0.6484	0.2701	0.6302	0.6237	$P_d(P_{fa} = 0.1)$	0.8060	0.7578	0.7487	0.5198	0.7956	0.7370

REFERENCES

- [1] G. Cao, Y. Zhao, and R. Ni, "Forensic estimation of gamma correction in digital images," in *ICIP*, 2010, pp. 2097–2100.
- [2] M. C. Stamm and K. J. R. Liu, "Forensic detection of image manipulation using statistical intrinsic fingerprints," *TIFS*, vol. 5, no. 3, pp. 492–506, 2010.
- [3] G. Cao, Y. Zhao, R. Ni, and X. Li, "Contrast enhancement-based forensics in digital images," *TIFS*, vol. 9, no. 3, pp. 515–525, 2014.
- [4] L. Wen, H. Qi, and S. Lyu, "Contrast enhancement estimation for digital image forensics," *TOMM*, 2018.
- [5] T. Pevný and J. J. Fridrich, "Detection of double-compression in JPEG images for applications in steganography," *TIFS*, vol. 3, no. 2, pp. 247–258, 2008.
- [6] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, 2014. [Online]. Available: <http://arxiv.org/abs/1409.1556>
- [7] S. Shalev-Shwartz, Y. Singer, N. Srebro, and A. Cotter, "Pegasos: primal estimated sub-gradient solver for SVM," *Math. Program.*, vol. 127, no. 1, pp. 3–30, 2011.
- [8] T. Gloe and R. Böhme, "The 'dresden image database' for benchmarking digital image forensics," in *ACM SAC*, 2010, pp. 1584–1590.
- [9] D. Dang-Nguyen, C. Pasquini, V. Conotter, and G. Boato, "RAISE: a raw images dataset for digital image forensics," in *ACM MMSys*, 2015, pp. 219–224.
- [10] G. Schaefer and M. Stich, "UCID: an uncompressed color image database," in *Storage and Retrieval Methods and Applications for Multimedia*, 2004, pp. 472–480.
- [11] M. D. Zeiler, "ADADELTA: an adaptive learning rate method," *CoRR*, vol. abs/1212.5701, 2012. [Online]. Available: <http://arxiv.org/abs/1212.5701>
- [12] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *AISTATS*, 2010, pp. 249–256.
- [13] G. Cao, Y. Zhao, R. Ni, and H. Tian, "Anti-forensics of contrast enhancement in digital images," in *MM&Sec*, 2010, pp. 25–34.
- [14] C. W. Kwok, O. C. Au, and S. H. Chui, "Alternative anti-forensics method for contrast enhancement," in *IWDW*, 2011, pp. 398–410.