

# Recolored Image Detection via a Deep Discriminative Model

Yanyang Yan<sup>ID</sup>, Student Member, IEEE, Wenqi Ren<sup>ID</sup>, Member, IEEE,  
and Xiaochun Cao<sup>ID</sup>, Senior Member, IEEE

**Abstract**—Image recoloring is a technique that can transfer image color or theme and result in an imperceptible change in human eyes. Although image recoloring is one of the most important image manipulation techniques, there is no special method designed for detecting this kind of forgery. In this paper, we propose a trainable end-to-end system for distinguishing recolored images from natural images. The proposed network takes the original image and two derived inputs based on illumination consistency and inter-channel correlation of the original input into consideration and outputs the probability that it is recolored. Our algorithm adopts a convolutional neural network (CNN)-based deep architecture, which consists of three feature extraction blocks and a feature fusion module. To train the deep neural network, we synthesize a data set comprised of recolored images and corresponding ground truth using different recoloring methods. Extensive experimental results on the recolored images generated by various methods show that our proposed network is well generalized and very robust.

**Index Terms**—Recoloring detection, convolutional neural network.

## I. INTRODUCTION

NOWADAYS, millions of photographs are produced by various devices and distributed by newspapers, televisions, and websites every day. Many legal, governmental and scientific organizations use digital images as evidence of specific events to make critical decisions. Unfortunately, with the development of low-cost and high-resolution digital cameras and sophisticated photo editing softwares, it is simple to perform image manipulations and the detection of forged images is much difficult through human vision. This challenges the reliability of digital images/photographs as real-world events. Accordingly, image forensic techniques for forged images detection are necessary.

Manuscript received July 29, 2017; revised January 9, 2018 and April 2, 2018; accepted April 15, 2018. Date of publication May 7, 2018; date of current version July 12, 2018. This work was supported in part by the National Key Research and Development Plan under Grant 2016YFB0800603, in part by the National Natural Science Foundation of China under Grant 61332012, Grant U1636214, and Grant U1605252, and in part by the Beijing Natural Science Foundation under Grant 4172068. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Stefano Tubaro. (*Corresponding author: Xiaochun Cao.*)

The authors are with the State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, China (e-mail: yanyanyang@iie.ac.cn; rwq.renwenqi@gmail.com; caoxiaochun@iie.ac.cn).

Color versions of one or more of the figures in this paper are available online at <http://ieeexplore.ieee.org>.

Digital Object Identifier 10.1109/TIFS.2018.2834155



Fig. 1. Can you identify which one is recolored? (a) presents an authentic image while (b) is a recolored image generated by [4]. Three different regions in (a) are recolored: the sky region, the sea area and the bridge. Note it is hardly to tell which one is recolored though human vision system.

Image recoloring, *i.e.*, color transferring, is one of the most common image operations in photo editing [1]. Usually, satisfying color transfer algorithms [1]–[3] apply the color characteristic of a target image to a source image and generate a recolored result that human cannot distinguish. One such example is shown in Figure 1. Figure 1(a) shows an authentic image and Figure 1(b) is a recolored image generated by the recoloring method [4]. The recolored image in Figure 1(b) has three different regions with (a): the sky region, the sea area, and the bridge. However, both the light blue sky in Figure 1(a) and the deep blue sky in (b) are equally authentic in human vision system. Although decent recolored images may leave no visual clues as shown in Figure 1(b), they may alter the underlying image consistencies. Although numerous methods have been proposed for image forensics, such as splicing [5], copy-move [6] and enhancement [7]. To the best of our knowledge, there are no forensics methods specially designed for color transferring even if altering the color of an image is one of the most common tasks in image processing [1]. Therefore, it is necessary to design approaches for recoloring detection. In this work, we take advantages of two consistencies as well as the original input image to distinguish whether an image is recolored.

Previous forged image detection approaches [8]–[11] focus on statistical relationships of hand-crafted appearance features between the original and tampered images. For example, Stamm and Liu [10] show that pixel value mappings leaves behind artifacts and detect enhancement by observing the intrinsic fingerprints in the pixel value histogram. However, these state-of-the-art methods are limited by the hand-designed priors or heuristic cues which may be less effective for some images. For instance, the method proposed in [10] is not likely

to detect tampered images if the pixel value histogram after tampering keeps smooth.

In this paper, we propose an end-to-end deep discriminative neural network to distinguish natural images from recolored images, which captures more comprehensive features. Our network employs inter-channel images and illumination map [12] as well as the input image as the inputs for our proposed network. We select these derived inter-channel images and illumination map as inputs since they have potential effectiveness for forgeries detection [11], [13]. Therefore, these derived inputs can provide additional information in addition to the original input. For training our proposed network, we use three color transfer methods [1]–[3] to automatically generate our training dataset. In addition, to evaluate our proposed model, we also generate a dataset in which the recolored images are generated by a variety of color transfer methods [1]–[4], [14]–[16] and establish a manual recolored image dataset. We will release this dataset publicly for future recoloring detection research.

The main contributions are summarized as follows.

- We are the first attempt to distinguish recolored images from natural images using a deep learning framework.
- We analyze the inter-channel correlation and illumination consistency for natural images which may not hold after the color transfer operation. Based on these two properties, we propose a deep discriminative model for recoloring detection.
- We generate a large-scale and high-quality training dataset for training the proposed network and create a benchmark dataset consisting of 100 skillfully recolored images and the corresponding 100 original photographs for testing.

The rest of the paper is organized as follows. In Section II, we briefly review the related work in several relevant respects. An overview of our deep discriminative model is given in Section III-C, including our analysis of the employed inter-channel correlation and illumination consistency. Extensive experimental results are presented in Section IV and Section VI. Section VI draws conclusions.

## II. RELATED WORK

Our purpose is to train a deep discriminative network for color transfer detection. Accordingly, we discuss the most relevant algorithms including forgery detection methods, color transfer approaches in this section.

### A. Forgery Detection Methods

Forgery detection methods intend to verify the authenticity of images and can be broadly classified into two classes: active authentication [17]–[22] and passive authentication [10], [11], [23].

In active authentication techniques, data hiding techniques are employed where some codes are embedded into the images during generation. These codes are used for further verifying to authenticate the originality of

image. Active authentication methods can be further classified into two types: digital signatures and digital watermarking. Watermarking embeds watermarks into images at the time of image acquisition while digital signatures embed some secondary information extracted from images at the acquisition end into the images. Lots of work has been proposed in both digital watermarking [17]–[19] and digital signatures [20]–[22]. For example, two image authentication algorithms are proposed in [19] to embed an image digest based on error diffusion halftoning technique, into the image in the Integer Wavelet Transform domain and the Discrete Cosine Transform domain, respectively. Lu and Liao [20] construct a structural digital signature using image content information in the wavelet transform domain for image authentication. The main drawback of these approaches remains that they must be inserted at the time of recording, which limits these approaches to specially equipped digital cameras. In addition, the prior information is necessary for an authentication process.

Passive authentication, also called image forensics, has no requirement for prior information. Digital image forensics are based on the assumption that tampering is likely to alter the underlying statistics and distinguish authenticity of an image by detecting these inconsistencies. Most algorithms first divide the input image into various overlapping blocks of different shape and then the feature extraction from each block takes place. Then, the sorting is done based on the features. Lastly, some morphological operations are applied to detect the forged region. Various techniques have been used to detect forgery, such as DWT [24], DCT [25], SVD [26], SIFT [27], LLE [28] and HGOM [29]. Passive techniques can be further classified as forgery dependent methods [5], [6], [30] and forgery independent methods [31]. Forgery independent methods detect forgeries independent of forgery type or can deal with various kinds of forgeries. For instance, a unified framework for determining image integrity is presented by Chen *et al.* [31] using a stochastic fingerprint of imaging sensors named photo-response nonuniformity noise. In contrast, forgery dependent methods are designed to detect an only certain type of forgeries such as splicing and copy-move. Rao *et al.* [5] detect the presence of splicing based on the inconsistencies in motion blur. Since forgery dependent methods focus on exploiting the unique characteristic for a specific task, these methods usually have better performance on a specific forgery detection task. In this work, we propose a forgery dependent method that is designed for recoloring detection.

### B. Color Transfer Approaches

Recent advances in digital image processing and enhancement techniques have made new and useful applications possible. One involves color manipulation, which challenges the reliability of digital images by generating high-quality composite recolored images.

One commonly used kind of methods for transferring the color is example-based recoloring based on the statistics of the color distribution in images. Reinhard *et al.* [1] propose a color transfer method by globally transferring colors. They apply

a simple statistical analysis to imposing one image's color characteristics on another in the *Lab* color space. The color transferring can effectively and efficiently generate a convincing output. A refined probabilistic model is used in [14] to further improve this technique. To better perform nonlinear color adjustments, Pitie *et al.* [3] utilize an N-dimensional probability density function and employ a post-processing algorithm to maintain the gradient field of the original image. Beigpour and van de Weijer [2] present a physical model of the image formation and apply to color transferring, making the results more realistic. All the above methods require an example image as input and we call this type of methods example-based recoloring.

Another kind of recoloring methods is based on edit propagation, which means drawing scribbles on different regions and propagating these edits to pixels automatically. This technique for propagating user edits is introduced in [32] firstly. An and Pellacini [16] extend this work by properly approximating the affinities between all pixels. Chen *et al.* [33] propose a sparsity-based edit propagation by using sparse dictionary learning for accelerating and saving memory. Palette-based recoloring methods have been proposed recently. A probabilistic factor graph model is developed by Lin *et al.* [34] to learn the properties of example patterns for coloring 2D patterns. Recently, Chang *et al.* [4] extract a color palette of an image by clustering and create a useful tool for recoloring by editing a color palette.

Although these recoloring algorithms may leave no visual clues, these methods may alter the underlying image consistencies. In this work, we take advantages of two consistencies to distinguish whether an image is recolored.

### III. OUR METHOD

In this section, we first present an overview including the reason we utilize the neural network and the derived inputs (the Difference Images (DIs) and the Illuminant Map (IM)), followed by the introduction and analysis of two related properties in Section III-B. Then the network architecture and details of our proposed deep discriminative model are given in Section III-C.

#### A. Overview

Existing forgery detection methods adopt some description techniques to combine the information attained by evidence estimators. However, every description technique has its own limitations and drawbacks. Recently, CNNs have shown an explosive popularity in image classification [35], [36] and other computer vision tasks [37], [38]. Traditional neural networks employ the original image in RGB channels as the input since it contains information about the picture such as color and structural features.

In this paper, we use three feature extractors and a feature fusion module to learn forgery-relevant features. The flowchart of our proposed approach is shown in Figure 2. We adopt the original image as one of the input branches like traditional neural networks. Additionally, we derive DIs and IM as two pieces of evidence of image recolored detection

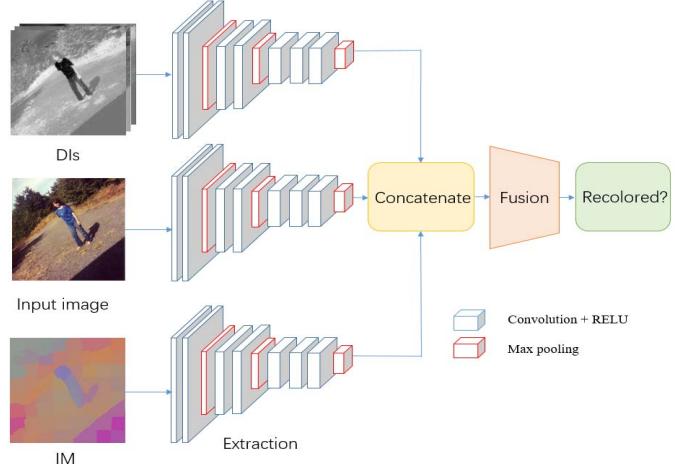
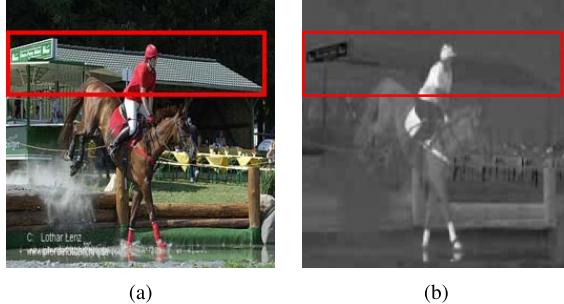


Fig. 2. Overview of our proposed approach. Given an image to be judged, the difference images (DIs) and the illuminant map (IM) are calculated firstly. Then the DIs and IM together with the input image in RGB channels are served as the inputs of our deep neural network. The network backbone is based on the VGG network and outputs a two-dimensional vector for distinguishing the input is recolored or not. Detailed configurations are outlined in Table I.

based on the observations that images may not maintain the inter-channel correlation or illuminant consistency after the recoloring process. These two pieces of evidence are employed as two additional input branches together with the original image. The network architecture can be found in Section III-C. Since the learned features are based on a data-driven approach, they are able to describe the intrinsic properties of forgery formation and help distinguishing the authenticity of an image. After extracting forgery-relevant features, we use a feature fusion network to refine these features and output the probability of authenticity. Based on this premise, we evaluate the proposed algorithm on forged images generated by various color transfer methods and the images collected through the Internet.

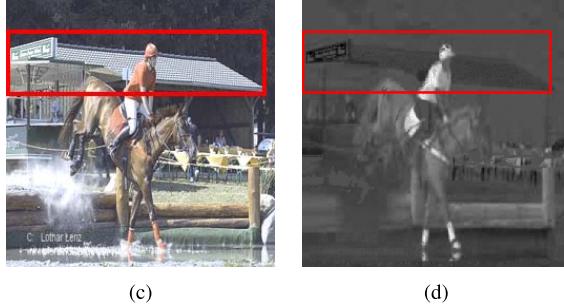
#### B. Derived Evidences

1) *Inter-Channel Correlation*: Most commercial digital cameras are equipped with an image sensor, charge-coupled device (CCD) or complementary metal-oxide-semiconductor (CMOS) and acquire the color information of each pixel using a CFA [39]. For example, the Bayer array [40], the most frequently used CFA, consists of four channels: red, blue, and two green channels. The green pixels are sampled on a quincunx lattice while the red and blue pixels are sampled on rectilinear lattices. As a result, the captured images by such cameras include specific correlations which are likely to be destroyed during manipulation. Instead of analyzing the property of one special CFA pattern, we focus on the common correlations among a range of CFA algorithms. Gunturk *et al.* [41] have shown that high-frequency components across image color channels are strongly correlated and similar. For most images, the correlation coefficients range from 0.98 to 1. In addition, this correlation has been widely used in CFA de-mosaicking [42]–[44]. We exploit this property to distinguish recolored images by DIs. The DIs can be



(a)

(b)



(c)

(d)

Fig. 3. The original and recolored images and their  $R - G$  DIs. (a) Original image. (b)  $R - G$  DI of (a). (c) Recolored image. (d)  $R - G$  DI of (c).

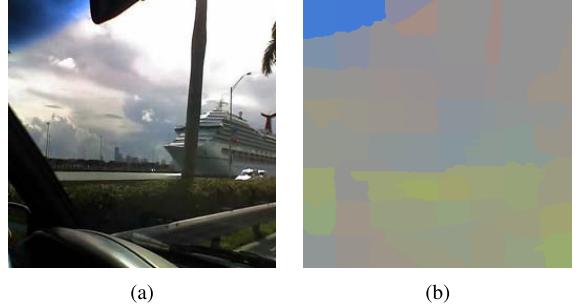
formally described as  $I_{c1} - I_{c2}$ , where  $c1, c2 \in \{R, G, B\}$  and the  $c2$  channel usually employs the green (G) color channel. Take the inter-channel correlation into consideration, the DIs can be given by

$$I_{c1} - I_{c2} = I_{c1}^l + I_{c1}^h - I_{c2}^l - I_{c2}^h \approx I_{c1}^l - I_{c2}^l \approx f_{LPF}(I_{c1} - I_{c2}) \quad (1)$$

where  $I_{c1}^h \approx I_{c2}^h$  due to the similarity of high-frequency components [41],  $h$  and  $l$  denote the high-frequency and low-frequency components of image color channels and  $f_{LPF}$  is a low-pass filter. As we can observe from Eqn. (1), a difference image (DI) from natural images is approximately equivalent to itself after passing through a low-pass filter. Therefore, compared to the original color channels, the DIs are smoother due to the lack of edges or details.

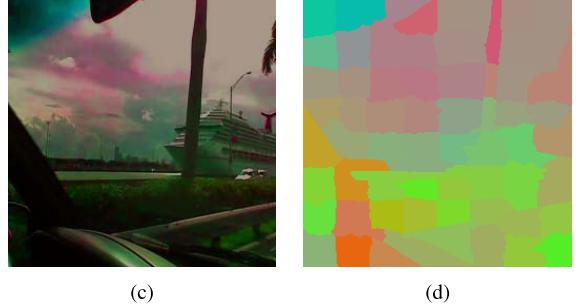
Figure 3(a) and Figure 3(b) show the contrast between natural image and corresponding DI of  $R - G$ . After manipulation operation, the inter-channel correlation may not exist as well as Eqn. (1) because the change of values in different color channels are often different. In this situation, the DIs not only contain low-frequency components but also some high-frequency components. As shown in Figure 3(d), the DI of  $R - G$  of the recolored image contains much more details than the original one in Figure 3(b). For example, the textures of the roof in Figure 3(d) are more clear than the one in Figure 3(b). In addition, some artifacts may arise in the DI after the color transfer operation, especially in edge regions like the back of the rider in Figure 3. Therefore, it will be useful to utilize this inter-channel correlation for our discriminative task. In this work, for dimensional uniformity, we calculate the DIs of  $R - G$ ,  $B - G$ , and  $R - B$  for an image and stack them in the channel dimension as one input branch of our network.

2) *Illuminant Consistency*: Illumination consistency has been widely used in forgery detection over a decade,



(a)

(b)



(c)

(d)

Fig. 4. Comparison of the illuminant maps of natural and recolored images. (a) Natural image. (b) IM of (a). (c) Recolored image. (d) IM of (c).

especially for splicing. Illumination-based methods are mainly grouped into two approaches: geometry-based and color-based approaches. Geometry-based methods look for inconsistencies in light source positions [45]–[47] and color-based methods focus on inconsistencies in the estimated light color [48]–[50]. Recently, a CNN-based algorithm is developed in [51] for the estimation of the color of the illuminant in RAW images.

Due to the fact that the changes of pixels in one image are not exactly identical during the color transfer process, it may be hard to maintain the illuminant consistency. Therefore, we utilize the illuminant consistency as another property in our discriminative model. Riess and Angelopoulou [49] propose an estimate strategy, which first segments the image into super-pixels with similar color and then utilizes an illuminant color estimator for local estimation at each super-pixel. In this work, we employ the same strategy and a new image called IM is derived.

IM represents the illuminant color of the input image and has the same dimension to the original RGB image. The values at each pixel denote the corresponding estimated illuminant color at this position. In general, the illuminant colors in a neighborhood should be close due to the illuminant consistency. However, image recoloring cannot maintain the illuminant consistency since the changes of pixels are not identical. As is shown in Figure 4(d), the IM has obvious blocking artifacts after the color transfer process and the illuminant consistency is destroyed.

In this paper, we use the Generalized Grayworld Estimates (GGE) proposed by van de Weijer *et al.* [12] as the illuminant color estimator. The method is based on the Grey-Edge hypothesis that assumes the average edge difference is achromatic in a scene. Let  $\mathbf{f}(x) = (R(x), G(x), B(x))^T$  denote the RGB value of a pixel at location  $x$ . For a Lambertian surface, the image values are dependent on the

light source. Then,  $\mathbf{f}(x)$  can be formed by

$$\mathbf{f}(x) = \int_{\omega} e(\lambda, x)s(\lambda, x)c(\lambda)d\lambda \quad (2)$$

where  $\omega$  is the visible spectrum of the light,  $\lambda$  is the wavelength of the light,  $e(\lambda, x)$  denotes the spectrum of the illuminant,  $s(\lambda, x)$  denotes the object surface reflectance and  $c(\lambda) = (R(\lambda), G(\lambda), B(\lambda))$  is the camera sensitivities functions. Van de Weijer *et al.* proposed to estimate the illuminant color as

$$k\mathbf{e}^{n,p,\sigma} = \left( \int \left| \frac{\partial^n \mathbf{f}^\sigma(x)}{\partial x^n} \right|^p dx \right)^{\frac{1}{p}} \quad (3)$$

where  $k$  is a scale factor,  $|\cdot|$  the absolute value,  $\partial$  the differential operator,  $\mathbf{f}^\sigma(x)$  denotes the observed intensity at coordinate  $x$ , which is smoothed. Eqn. (3) describes a framework for low-level based illuminant estimation through the incorporation of three parameters:

- Derivative order  $n$ : to determine if the estimation algorithm is a GreyWorld or a Grey-Edge method. The Grey-World methods are corresponding to the *RGB* values while the Grey-Edge methods are corresponding to the spatial derivatives of order  $n$ . All the methods are based on the assumption that the average value of the illuminants can be extended to the absolute value of the derivative sum of the image.
- Minkowski norm  $p$ : to assign the corresponding weights of the multiple measurements (intensities or derivatives) for the final illuminant color estimation instead of simply adding.
- Gaussian smoothing  $\sigma$ : to smooth the image prior by processing with a Gaussian kernel of standard deviation  $\sigma$ .

Note that Eqn. (3) computes the  $\mathbf{e}$  separately for each color channel. It is more robust than the original gray world algorithm in [52]. In addition, the Minkowski norm emphasizes larger measurements over smaller measurements, resulting in better exploited specular edges. Instead of using special image description techniques like Carvalho *et al.* [11], such as SASI [53], LAS [54], ACC [55] and so on, to extract illuminant features, in this work, we employ a deep CNNs-based to exploit illuminant mismatches among the objects in the image.

### C. Algorithmic Details

1) *Network Architecture*: As mentioned in Section III-B, the recolored images have different representation in DIs and IM. These two properties can be used for distinguishing whether a photo is recolored. As is shown in Figure 2, given an image to be judged, we first calculate the DIs and the IM based on [12] for the input. Then, we use the original image in RGB channels, the DIs and IM as the inputs in our network.

The backbone is based on the recent VGGnet [56], which is a 16-layer model. The convolutional layers mostly have very small  $3 \times 3$  filters, which outperforms larger filters [56]. Our network contains three phases: feature extraction, fusion and the final classification step, which are labeled in Figure 2.

TABLE I

SPECIFICATIONS OF THE PROPOSED NEURAL NETWORK. EACH CONVOLUTIONAL LAYER IS FOLLOWED BY A ReLU LAYER. THE CONVOLUTIONAL LAYER PARAMETERS ARE DENOTED AS “CONV-<FILTER SIZE>-<THE NUMBER OF CHANNELS>”

ConvNet Configuration		
DIs (224*224)	Original image (224*224 RGB)	IM (224*224)
conv3-64	conv3-64	conv3-64
conv3-64	conv3-64	conv3-64
maxpool	maxpool	maxpool
conv3-128	conv3-128	conv3-128
conv3-128	conv3-128	conv3-128
maxpool	maxpool	maxpool
conv3-256	conv3-256	conv3-256
conv3-256	conv3-256	conv3-256
conv3-256	conv3-256	conv3-256
maxpool	maxpool	maxpool
concat		
conv3-512		
conv3-512		
conv3-512		
maxpool		
conv3-512		
conv3-512		
conv3-512		
FC-4096		
FC-4096		
FC-2		
Soft-max		

In the feature extraction phase, we extract the features of each input using the first three convolutional stages of the VGGnet. This phase is equal to description techniques in traditional methods. The parameters for different inputs are not shared. In the fusion phase, we first connect the features extracted in the front phase by a concatenate layer. Then the remained two stages of the VGGnet are applied to the connected features, followed by two 4096-dimension fully connected layers. Compared to traditional methods, this phase is used to replace the feature selection or integration part. Finally, a fully connected layer whose output is a two-dimension vector and a soft-max layer make up the classification phase. The efficiency of our proposed algorithm will be discussed detailedly in Section IV. In addition, detailed configurations of the proposed network are outlined in Table I. The convolutional layer parameters are denoted as “conv-<filter size>-<the number of channels>” in Table I. For example, the kernel size and output channels of *conv3-64* are  $3 \times 3$  and 64, respectively.

2) *Implementation*: During training, we use a batch size of 10, and patch size of  $224 \times 224$ . We use the Stochastic Gradient Descent (SGD) [35] for optimizing. The learning rate starts from 0.0001 and is divided by 10 when the error plateaus. We employ a weight decay of 0.0005 and a momentum of 0.9. For all the results reported in the paper, we train the network for 40 epoch, which takes about 42 hours on an NVidia K40 GPU, and report all the experimental results at 40 epochs. The convergence plot of the proposed network is given in Figure 5. As can be seen, loss value decreases obviously in the first 20 epochs and reaches the stability

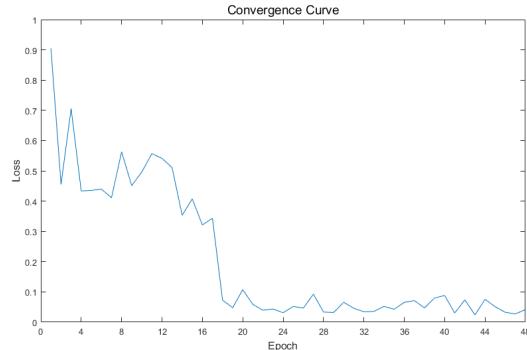


Fig. 5. Convergence curve of the proposed discriminative model on the validation dataset.

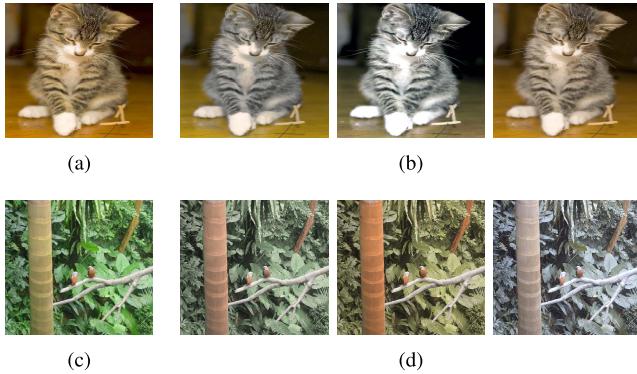


Fig. 6. Examples selected from the training set. The first column shows a natural indoor and an outdoor image, which followed by three recolored images generated by methods [1]–[3], respectively. It can be hard to tell which image is the original one if there exist no annotations. (a) Indoor image. (b) Recolored images of (a). (c) Outdoor image. (d) Recolored images of (c).

around 40<sup>th</sup> epoch on the validation set. Since our approach judges an image in a single forward pass, it is computationally very efficient. Using an NVidia K40 GPU, we can process a 512 × 512 image within 0.1s, which indicates that our model has the ability to deal with large data.

#### IV. EXPERIMENTAL RESULTS

**Training Data.** The training set is an essential component of the network. We use the VOC PASCAL 2012 dataset which contains 17125 images including both indoor and outdoor photographs. Since edit propagation and palette based recoloring methods require artificial manipulation and are inappropriate for generating a large number of training data, we only use the example-based recoloring methods to generate training data. Given a color transferring algorithm, the target image and the source image are randomly selected from the VOC PASCAL 2012 dataset. In this paper, three different color transferring methods [1]–[3] are employed to generate training data. Figure 6 shows some examples of recolored images generated by these methods. As our recoloring detection is a binary classification task, we need a balance between the positive and negative examples in training data. In this work, given an original photograph  $I$ , we randomly select one recoloring method to generate the recolored image. Therefore, the ratio between the positive and negative examples is 1,

which is the most appropriate for binary classification using the neural network.

Not that although our method is trained using 224 × 224 patches, the proposed algorithm could be used for a full-frame image in the testing stage. Given a high-resolution input, we first crop the image to several 224 × 224 parts with overlap, from which we can recover the entire image. Then, our network will judge these cropped 224 × 224 parts independently. Finally, the image is considered to be recolored if one of the cropped parts is judged to be recolored by our network.

For convenience, we refer the proposed Recolored Detection Network as RecDeNet in the following.

#### A. Quantitative Evaluation on Benchmark Datasets

After training our discriminative model, we evaluate the RecDeNet on a testing dataset. The testing images are also generated by applying three methods [1]–[3] to images from the VOC PASCAL 2012 dataset. In addition, The testing data have no overlap with the training data in terms of the target images. Apart from the accuracy, the area under ROC curve (AUC) is also reported to evaluate the classification performance. In this dataset, the accuracy reaches 86.89% and the AUC achieves 94.29%, suggesting that our RecDeNet performs well.

Figure 7 shows the visual feature maps about original and recolored images on the first *conv3-64* layer of different input branches. Note that the learned feature maps contain various features, which reflects in having different degrees of response in the region. In addition, although the original and recolored images look familiar, their feature maps are quite different. Therefore, our RecDeNet is capable of replacing image description techniques. Examples of feature maps of DIs are shown in the last column of Figure 7, which are generated by the same kernel. Notice that the responses in edge regions of recolored maps are higher than those of original photos, revealing that some artifacts are produced especially in edge regions after recoloring and this convolution kernel probably has some connection with edges.

Furthermore, a k-fold cross-validation is performed based on the semantic classes where  $k = 3$ . The PASCAL dataset contains the information about 20 semantic classes. We split these classes into three major categories, in which two major categories make up the training and validation set while the left one makes up the testing set. The training and testing processes are repeated three times by selecting different two major categories and we focus on the average performance. Our RecDeNet still performs well with the average accuracy 84.98% and the average AUC 93.01%, which reveals the robustness of our methods.

*1) New Synthetic Dataset:* To further demonstrate the practicability of our framework, we synthesize a new dataset including 100 authentic photographs and corresponding synthetic recolored images. We first crawl the original photographs from the website, then employ a variety of color transferring methods [1]–[4], [14]–[16] to generate the recolored images. Note that edit propagation and palette based

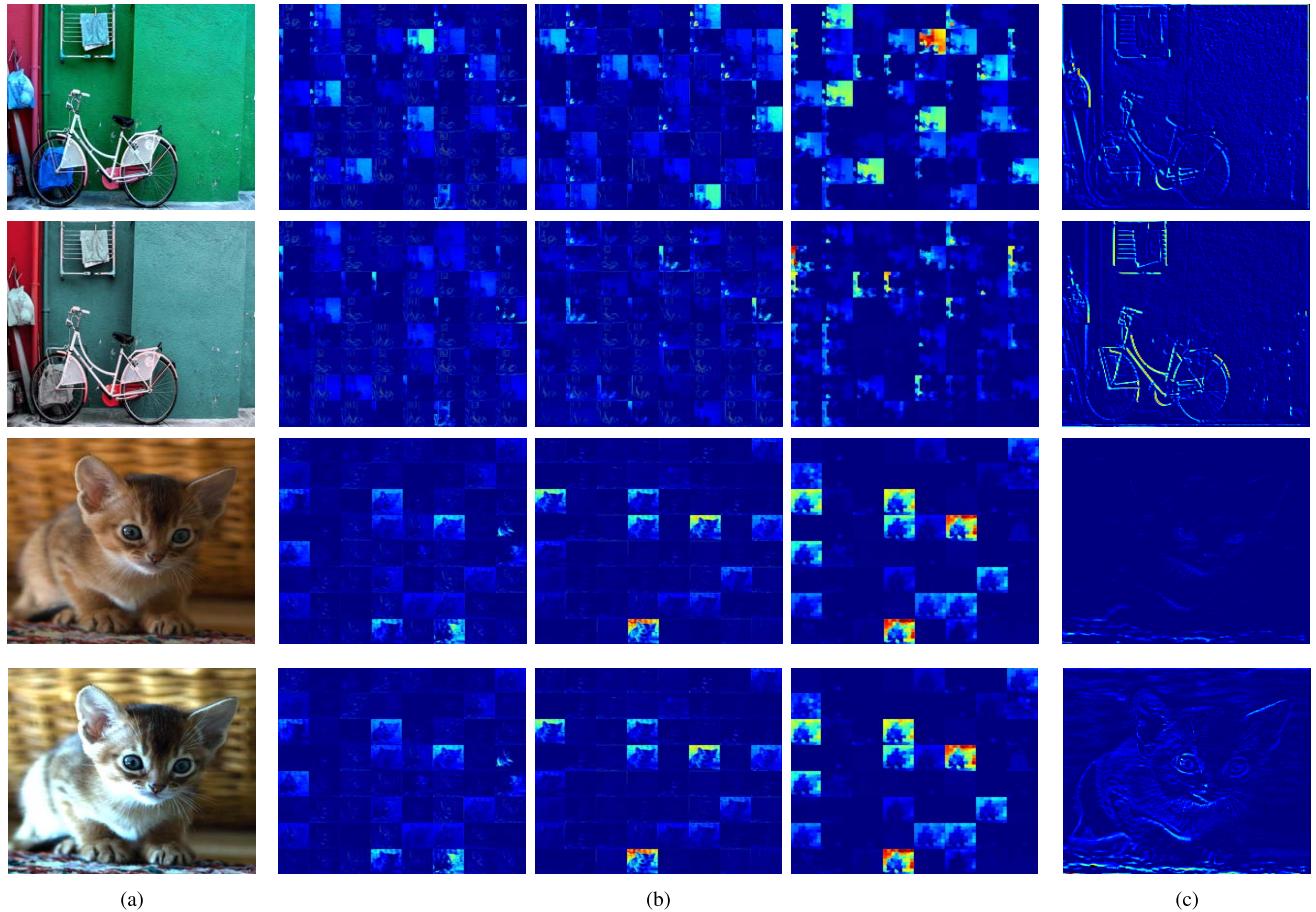


Fig. 7. Feature maps on the first *conv3-64* layer of different input branches. The first images in the second and fourth rows are recolored. The neural network learns various useful features for distinguishing recolored images. With these diverse features automatically learned from the proposed algorithm, our RecDeNet could effectively distinguish authentic and recolored images. The last column shows the examples of feature maps of DIs, which are generated by the same convolution kernel. (a) Image pairs. (b) Feature maps on the first *conv3-64* layer. Columns from left to right are corresponding to the RGB channels, DIs and IM of (a), respectively. (c) Selected examples of feature maps of DIs using the same convolutional kernel.

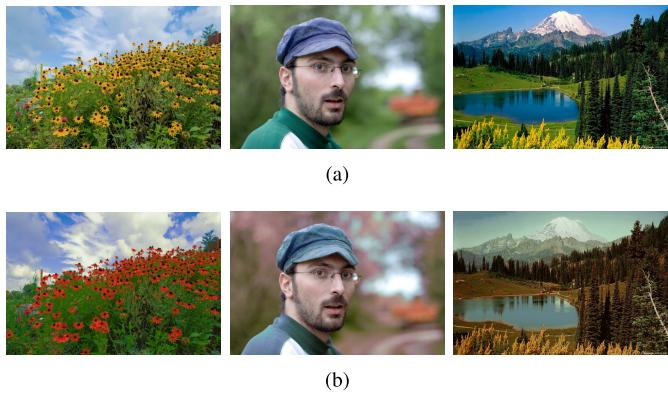


Fig. 8. Example image pairs randomly selected from our synthetic dataset.  
 (a) Original photos. (b) Recolored images.

recoloring methods are also used to generate recolored images in this section for testing the generalization ability of the RecDeNet. Figure 8 shows some examples of authentic and the corresponding recolored image pairs in our dataset. Most of the recolored images in the synthetic dataset appear quite realistic, it is hard to distinguish them from authentic photographs by human vision. We utilize our trained RecDeNet to

judge the images in the synthetic dataset. The accuracy reaches 83.50% and the AUC achieves 90.75%, which reveals that our RecDeNet is effective for various types of color transferring methods.

**2) Human Perceptual Performance:** Because of the effectiveness of color transferring techniques in recent years, distinguishing an image as recolored or not can be very difficult for humans. In this section, 30 volunteers are asked to evaluate each image pair in our dataset for human performance. Participants in the experiment were shown a series of pairs of images. Each pair consists of an authentic photo next to a recolored version, produced by one of the color transferring algorithms. In addition, the positions L/R of recolored images are randomly switched. Participants were asked to click on the photo which in their opinion contains fake colors generated by a recoloring algorithm. If it is difficult to judge which is the recolored one, participants can choose the “other” button to denote “difficult to judge”. Before starting the actual experiment, participants completed two practice trials on which they were given feedback. In the testing stage, participants were given unlimited time to respond. No feedback was given during the actual experiment. When counting the quantization results, we set the probability of being recolored according

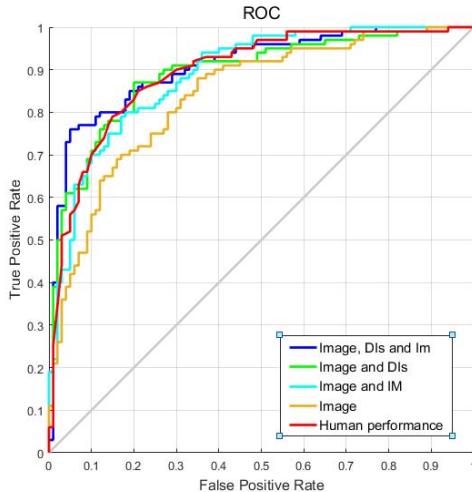


Fig. 9. ROC curves provided by the experiments and human performance on our synthetic dataset. Our RecDeNet achieves better performance than humans. The comparisons between results of different inputs reveal that the DIs and the IM are helpful.

to human choices. For one image pair, the photo which is selected will be given the probability 1.00 while the probability of left one is 0.00. However, both two images will be given probability 0.50 if participants choose the “other”. We use MOS to aggregate scores for each image pairs. The link for human perceptual performance is here.<sup>1</sup>

We compare the results to the ground truth labels, the accuracy achieves 81.50%, which is 2.00% lower than the performance of our proposed RecDeNet. The disparity is consistent with the AUC performance, participants achieve a score of 89.53%, 1.22% lower than ours, suggesting our RecDeNet performs better than the human judgment on this dataset. The ROC curves of human results and RecDeNet are shown in Figure 9 and the red line denotes the human performance.

Figure 9 also contains the results generated by neural networks with different inputs. The baseline only employs the original image as the input and other two networks contain two input branches: 1) the original image and corresponding DIs; and 2) the original image and the IM. The comparisons in Figure 9 demonstrates the reliability of our proposed RecDeNet as well as the efficiency of the derived inputs of inter-channel correlation and illuminant consistency.

In Figure 10, we show a visual representation of the rankings including the performance of the RecDeNet and volunteers on our synthetic dataset. From top to bottom in Figure 10, randomly selected images are shown according to the prediction scores of the proposed RecDeNet. Each row corresponds to a score interval. For instance, the recolored probabilities of images, obtained by RecDeNet, are between 0~25% in the first row. For comparison, human judgment scores are indicated by color borders. We use four kinds of color borders (red, yellow, green and blue) to denote four

<sup>1</sup>[https://docs.google.com/forms/d/e/1FAIpQLSd\\_J1clvUuqLtz3x9RZyGQSTPUJQwsyqwwb5G1ffEGO5zlow/viewform?usp=sf\\_linkLink](https://docs.google.com/forms/d/e/1FAIpQLSd_J1clvUuqLtz3x9RZyGQSTPUJQwsyqwwb5G1ffEGO5zlow/viewform?usp=sf_linkLink)

human judgment rankings (recolored probabilities between 0~25%, 25~50%, 50~75% and 75~100%), respectively. In addition, the ground truth label of an image is attached to the top right corner. Choosing the probability 0.50 as the threshold, both RecDeNet and humans distinguish accurately for a majority of images compared to the ground truth labels. However, misjudgments exist for both RecDeNet and human performance. Although our proposed RecDeNet and participants perform closely in terms of evaluation criterion, each has its own merits and drawbacks. On the one hand, humans have mastery of general knowledge about object colors and the neural network is limited by the training data. On the other hand, the general knowledge can sometimes mislead the judgment, like the first and second images in the fourth row in Figure 10. In contrast, the neural network can learn the most manipulation-relevant features and perform favorably against the human visual system as shown in Figure 9.

### B. Newly Collected Data

Since color transferring methods have been widely used in human society, we further collect a new dataset that contains 80 recolored photos which are performed manually. Some of the photographs are produced by mobile APPs while others are downloaded from the websites such as Photoshop tutorial websites. All the downloaded images are mentioned that they are recolored.

Some examples randomly selected from our dataset are shown in Figure 11. Note it is hard to tell whether these images are recolored just by human vision. The detection accuracy of human vision is 55.50%. In contrast, our proposed RecDeNet achieves 68.75%, which suggests that our discriminative model has good capability to detect recolored images by various recoloring tools automatically.

We note that the accuracy of the newly collected dataset is a bit lower than the results in Section IV-A. This is mainly due to that the coverage of our training data is not sufficient for these images. However, the detection accuracy is still higher than human vision system by up to 13.25% since some recoloring-relevant features have been learned by the proposed RecDeNet automatically.

## V. ANALYSIS AND DISCUSSIONS

We analyze the proposed RecDeNet about generalization, architecture and failure case in this section.

### A. Applicability and Generalization

The experimental results on the testing dataset in Section IV-A are based on the condition that image pairs in the training data and testing set are generated by the same methods. In this section, we generate training data and testing data by different recoloring methods to validate the applicability and generalization of our RecDeNet. We use two of three color transferring approaches [1]–[3] to synthesize training data and use the same algorithms as well as the left method to generate two different testing sets as shown in Table II. We do not consider employing only one color

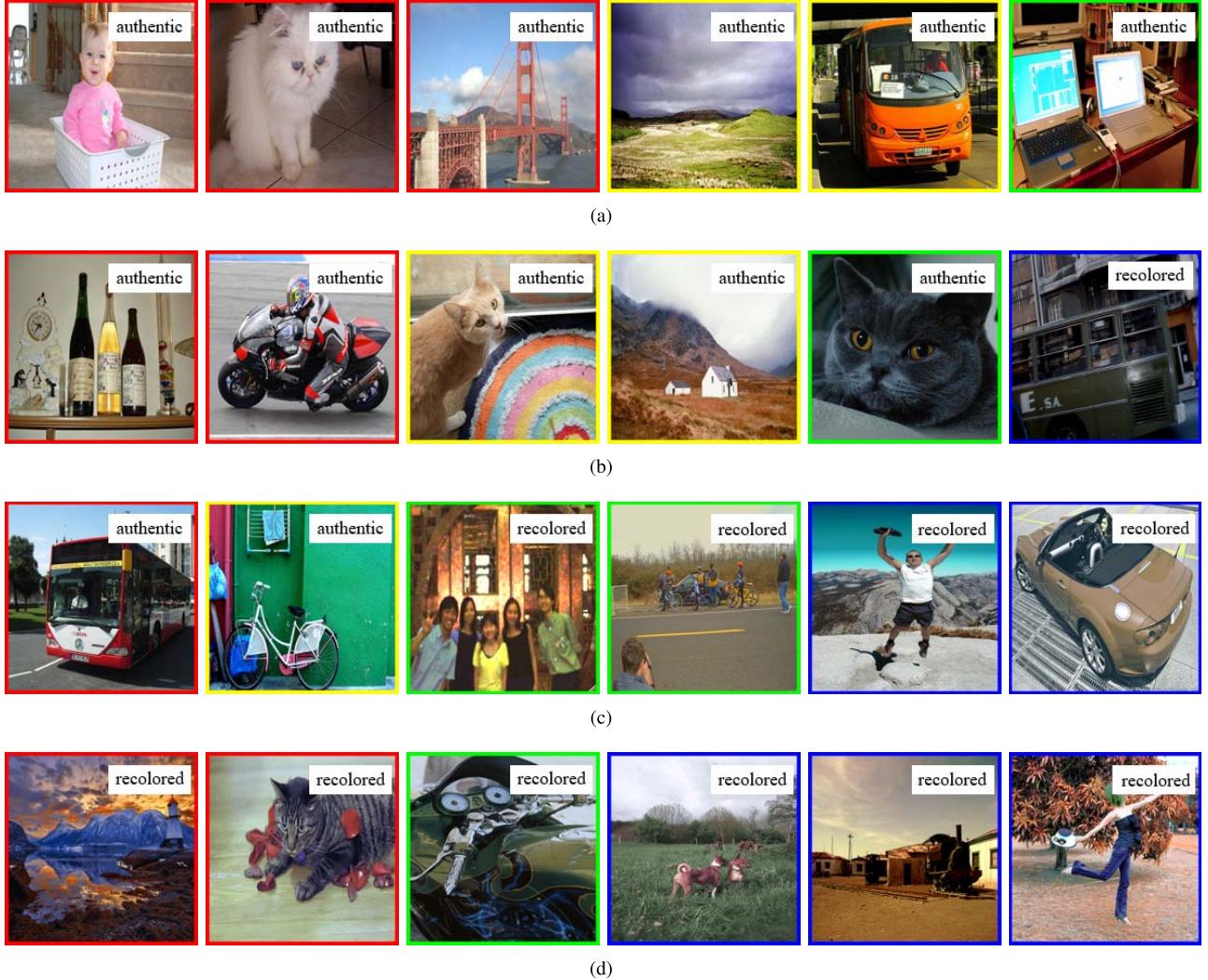


Fig. 10. Images ranked according to our ReDeNet’s recolored prediction scores. Each row corresponds to a score interval. Row 1: 0~25% interval (most natural), 2: 25~50%, 3: 50~75%, 4: 75~100% (most likely to be recolored). In each interval, the images are randomly selected from our dataset. The border color illustrates the average human score of recolored probability for each image (red: 0~25%; yellow: 25~50%; green: 50~75%; blue: 75~100%). In addition, the ground truth of an image is labeled in the top right corner. (a) The recolored probabilities are between 0~25% by RecDeNet. (b) The recolored probabilities are between 25~50% by RecDeNet. (c) The recolored probabilities are between 50~75% by RecDeNet. (d) The recolored probabilities are between 75~100% by RecDeNet.



Fig. 11. Example images randomly selected from our collected recolored dataset.

transfer method to produce image pairs for the training data. The reason behind is the deep discriminative model may learn some special features about the color transfer pattern to judge images as recolored or not, which results in unsatisfying applicability.

Table II shows the detection accuracy of the proposed RecDeNet with various datasets. Note that when the methods used for training and testing are different, both the accuracy and AUC are a little lower than those in the other conditions, suggesting that the deep network still learns some

manipulation-relevant features about the color transferring pattern even we only use two recoloring methods for training. However, even in this premise, our RecDeNet still performs well with the accuracy 75.56% and the AUC 85.87% at least, which reveals the robustness of the deep discriminative model for recolored image detection.

### B. Architecture

We use the front three stages of the VGGnet to extract the features of each input and concatenate these features, followed

TABLE II

RESULTS OF RecDeNets WITH DIFFERENT TRAINING AND TESTING DATASETS. THE NUMBERS IN THE DATASET CELLS REPRESENT THE CITE NUMBER OF USED RECOLORING METHODS

Training set	Testing set	Accuracy(%)	AUC(%)
[1], [2]	[1], [2]	87.40	93.81
	[3]	77.68	87.56
[2], [3]	[2], [3]	88.82	95.52
	[1]	75.56	85.87
[1], [3]	[1], [3]	83.34	90.46
	[2]	78.63	86.72

TABLE III

RESULTS OF RecDeNets WITH DIFFERENT INPUTS, SUGGESTING THE PRACTICABILITY OF THE INTER-CHANNEL CORRELATION AND THE ILLUMINATION CONSISTENCY

Inputs	Accuracy(%)	AUC(%)
image	77.48	85.30
image, DIs	84.22	92.43
image, IM	83.46	91.46
image, DIs, IM	86.89	94.29

by two convolution stages. In this section, we explore different architectural designs of the network and study the relations between recoloring performance and factors like different input branches concatenate layers, networks and illuminant estimation algorithms.

1) *DIs and IM*: As is shown in Figure 2, the inputs of our RecDeNet are the DIs and the IM together with the original image. Here, we set the baseline as the condition which only employs the original image as the input. Additionally, we also add only one branch of the DIs or IM to the baseline. By comparing the results of different inputs in Table III, it can be driven obviously that our RecDeNet achieves the highest accuracy and AUC when both the DIs and IM are added to the baseline for the same training and testing dataset. We note that only adding the DIs or IM to the baseline improves the discriminative effect more or less compared to the baseline. For example, the AUC value can be increased by up to 6.74% by adding the DIs. These indicate that two added branches are helpful and assistant to each other. Furthermore, these results also drive us for a more comprehensive model by adding other properties as branches. Figure 12 shows the ROC curves with different inputs. Note that both the DIs and IM improve the performance of recoloring detection. Therefore, it is necessary to concatenate these three branches for better performance.

2) *Concatenate Layer*: In this subsection, we examine the network sensitivity to different concatenate layer. In previous experiments, we concatenate the features of three inputs at the third pooling layer. Specifically, we concatenate features at the different pooling layers. We use *RecDeNet-c1*, *RecDeNet-c2*, *RecDeNet-c3*, and *RecDeNet-c4* to denote concatenating features at the first, second, third and fourth pooling layers, respectively. Table IV shows the results by concatenating the features in different stages while having five stages in total for the fair comparison. Comparing the results of *RecDeNet-c1*, *RecDeNet-c2*, and *RecDeNet-c3*, we note that using deeper networks to extract features of each input achieve higher

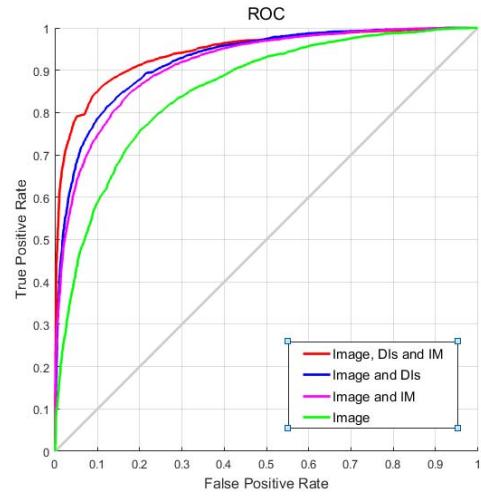


Fig. 12. **ROC Curve**. Proposed methods with different inputs on the testing set. Notice that adding only one branch of the DIs or IM to the baseline will help the recoloring detection. Our RecDeNet achieves the best performance when employing the DIs and the IM together with the original image as inputs.

TABLE IV

RESULTS OF RecDeNets WITH VARIOUS BACKBONES, WHICH CONCATENATE THE FEATURES IN DIFFERENT LAYERS WHILE THE TOTAL STAGES REMAIN THE SAME. BOTH THE FEATURE EXTRACTION AND FUSION STAGES ARE IMPORTANT

	Concatenate layer	Accuracy(%)	AUC(%)
<i>RecDeNet-c1</i>	pooling1	81.57	89.74
<i>RecDeNet-c2</i>	pooling2	84.33	92.23
<i>RecDeNet-c3</i>	pooling3	86.89	94.29
<i>RecDeNet-c4</i>	pooling4	85.25	93.33

accuracy and AUC when the number of total stages is the same. It benefits from more specific parameters to be trained. However, the results of *RecDeNet-c4* are a little lower than of *RecDeNet-c3*, which seems not meet the aforementioned property. It mainly because the convolution layers following the connected features are also essential for compromising the exacted features. In addition, deeper networks for each input correspond to larger models as well as longer training time. Above all, we adopt the backbone *RecDeNet-c3* in Table IV as our final architecture of RecDeNet, with accuracy 86.89% and AUC 94.29%.

3) *VGGnet vs AlexNet*: In this section, we compare our RecDeNet against the traditional AlexNet [35]. The network using VGGNet and AlexNet are referred to as *RecDeNet-VGG* and *RecDeNet-Alex*, respectively. For the fair comparison, the inputs to AlexNet is the same as the inputs to the VGGnet. Furthermore, we use the first three convolution layers in *RecDeNet-Alex* to extract features of inputs and use the left part to act on the connected features, which keeps consistency with the *RecDeNet-VGG*.

The results are shown in Table V, where the *RecDeNet-Alex-pre* and *RecDeNet-VGG-pre* model employ pre-trained models. The accuracy and AUC of *RecDeNet-Alex* are 16.12 and 14.90 points lower than the *RecDeNet-VGG*, respectively, which is similar to the situation when our networks employ

TABLE V

RESULTS OF RecDeNets BASED ON DIFFERENT NETWORKS

	Network	Accuracy(%)	AUC(%)
<i>RecDeNet-Alex</i>	AlexNet	70.77	79.39
<i>RecDeNet-VGG</i>	VGGnet	86.89	94.29
<i>RecDeNet-Alex-pre</i>	AlexNet	73.99	82.72
<i>RecDeNet-VGG-pre</i>	VGGnet	88.07	95.22

TABLE VI

COMPARISONS OF RecDeNets BASED ON DIFFERENT ILLUMINANT ESTIMATION ALGORITHMS

Test data	Illuminant estimation algorithm	Accuracy(%)	AUC(%)
PASCAL 2012 dataset	Riess <i>et al.</i> [52]	86.89	94.29
	Gijsenij <i>et al.</i> [58]	88.98	95.77
New synthetic dataset	Riess <i>et al.</i> [52]	83.50	90.75
	Gijsenij <i>et al.</i> [58]	86.50	90.76

pre-trained models. It is probably due to the smaller filters and the deeper network architecture has stronger ability to learn features. But in addition, although only a few layers are used to extract features of inputs or to compromise the exacted features in *RecDeNet-Alex*, our proposed basic frame is still helpful for recolored images detection, which suggests the ability of deep neural network.

*4) Illuminant Estimation Algorithms:* It is natural to use some more recent illuminant estimation methods as like [51] and [58] to improve the illuminant and thus the final recoloring detection accuracy. Here, we adopt a better illuminant estimation algorithm proposed in [57] in our RecDeNet. The comparisons of the detection accuracy and AUC are shown in Table VI. As shown, the results based on [49] and [57] are very similar (even the recent work [57] can yield a little bit better results than [49]), which demonstrate the robustness of our algorithm.

### C. Comparisons With State-of-the-Art Methods

Most passive authentication algorithms are designed in block-level, which are not suitable for image-level detection. However, we compare our method against a state-of-the-art histogram-based method [10], a CNN-based method [58] and a rich-features-based method [59]. We use the default parameters according to [1] and set the classification number as 2 for the CNN-based method [2]. Figure 13 shows the ROC curves generated by [10] and [58] and our proposed RecDeNets with different inputs. As shown, our RecDeNet containing image, DIs and IM achieves the best performance. The detailed results are listed in Table VII. Since the histogram-based method [10] only considers the information of histogram, other image information such as content and gradients are not utilized. Therefore, [10] does not perform well towards the recolor detection problem. The results of [58] are better than [10] on the PASCAL 2012 dataset since this method employs CNNs to extract a large number of error-relevant features. However, when it comes to the new synthetic dataset, the CNN-based method [2] performs little worse than [1], which is opposite to the results on PASCAL 2012 dataset. The main reason may

TABLE VII

COMPARISONS OF RecDeNet AGAINST HISTOGRAM-BASED METHOD [10] AND CNN-BASED METHOD [58]

Test data	Methods	Accuracy(%)	AUC(%)
PASCAL 2012 dataset	[10]	-	62.89
	[58]	62.01	70.99
	RecDeNet	86.89	94.29
New synthetic dataset	[10]	-	79.90
	[58]	63.50	73.69
	RecDeNet	83.50	90.75

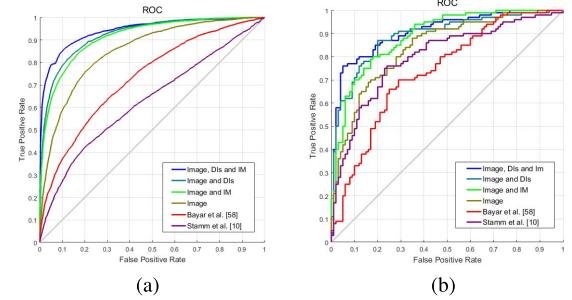


Fig. 13. ROC Curves provided by histogram-based method [10], CNN-based method [58] and our proposed methods with different inputs on the PASCAL testing set and our new synthetic dataset. (a) ROC Curves on the PASCAL testing set. (b) ROC Curves on our new synthetic dataset.

TABLE VIII

COMPARISONS OF RecDeNets AND RICH-FEATURES-BASED ALGORITHM

Test data	Methods	Accuracy(%)
PASCAL 2012 dataset	RecDeNet	86.89
	Rich-features-based	66.99
New synthetic dataset	RecDeNet	83.50
	Rich-features-based	69.00
Newly collected data	RecDeNet	65.00
	Rich-features-based	37.50

be that the new synthetic dataset contains only 200 images, which is in the range of the decision rule of [10]. Once the image number is large, like the PASCAL 2012 dataset, it is hard to take a suitable decision threshold. Furthermore, our RecDeNet is superior to [58] due to better information carriers and more effective network architecture. For the rich-features-based method, we extract the SCRMQ1 features by the method in [59] and train an SVM for classification. Table VIII shows the comparison results. Note that our method performs much better than the rich-features-based method both on the PASCAL 2012 dataset and our new synthetic dataset, which demonstrates the efficiency of our RecDeNet. Above all, these comparisons also validate the difficulty-level of the used dataset.

### D. Failure Cases

In this work, although our proposed RecDeNet achieves good performance, our method is less effective for some cases. In this paper, the recolored images in the training set are generated by [1]–[3]. These adopted methods all belong to the example-based recoloring algorithms. However, edit propagation [16], [32], [33] and palette [4], [34] based recoloring



Fig. 14. Examples of failure cases. (a) False negative examples. (b) False positive examples.

methods are not used since they are difficult to be used to generate a large number of recolored images. Therefore, our trained RecDeNet may be less effective for detecting recolored images produced by these kinds of methods, which is consistent with the results in Section IV-A. The accuracy of the testing data generated by example-based methods [1]–[3] has a 3.39% higher than the testing set generated by various recoloring methods [1]–[4], [14]–[16] based on different strategies. In addition, some failure examples are shown in Figure 14. The false negative examples denote original photos which are classified to recolored images by RecDeNet, while the false positive examples are recolored images which are not detected. In future work, we will address this problem by exploring more manipulation-relevant features.

## VI. CONCLUSION

In this work, we present a novel deep learning approach for recolored image detection. Both the inter-channel correlation and the illumination consistency are employed to help the feature extraction. We elaborate the design principle of our RecDeNet and systematically validate the rationality by running a number of experiments. Furthermore, two recolored datasets with different sources are created and the high performance of our RecDeNet demonstrates the effectiveness of the model. We hope our simple yet effective RecDeNet will serve as a solid baseline and help future research in recolored images detection. Our future work will focus on designing a more effective network architecture and searching for some high-level cues for better distinguishing.

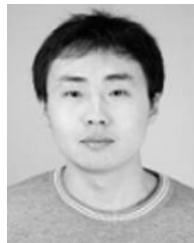
## REFERENCES

- [1] E. Reinhard, M. Adhikmin, B. Gooch, and P. Shirley, “Color transfer between images,” *IEEE Comput. Graph. Appl.*, vol. 21, no. 5, pp. 34–41, Sep./Oct. 2001.
- [2] S. Beigpour and J. van de Weijer, “Object recoloring based on intrinsic image estimation,” in *Proc. ICCV*, Nov. 2010, pp. 327–334.
- [3] F. Pitié, A. C. Kokaram, and R. Dahyot, “Automated colour grading using colour distribution transfer,” *Comput. Vis. Image Understand.*, vol. 107, nos. 1–2, pp. 123–137, 2007.
- [4] H. Chang, O. Fried, Y. Liu, S. DiVerdi, and A. Finkelstein, “Palette-based photo recoloring,” *ACM Trans. Graph.*, vol. 34, no. 4, 2015, Art. no. 139.
- [5] M. P. Rao, A. N. Rajagopalan, and G. Seetharaman, “Harnessing motion blur to unveil splicing,” *IEEE Trans. Inf. Forensics Security*, vol. 9, no. 4, pp. 583–595, Apr. 2014.
- [6] G. Muhammada, M. Hussain, and G. Bebis, “Passive copy move image forgery detection using undecimated dyadic wavelet transform,” *Digit. Invest.*, vol. 9, no. 1, pp. 49–57, 2012.
- [7] G. Cao, Y. Zhao, R. Ni, and X. Li, “Contrast enhancement-based forensics in digital images,” *IEEE Trans. Inf. Forensics Security*, vol. 9, no. 3, pp. 515–525, Mar. 2014.
- [8] X. Pan and S. Lyu, “Region duplication detection using image feature matching,” *IEEE Trans. Inf. Forensics Security*, vol. 5, no. 4, pp. 857–867, Dec. 2010.
- [9] X. Zhao, J. Li, S. Li, and S. Wang, “Detecting digital image splicing in chroma spaces,” in *Proc. Int. Workshop Digit. Watermarking*, 2010, pp. 12–22.
- [10] M. C. Stamm and K. J. R. Liu, “Forensic detection of image manipulation using statistical intrinsic fingerprints,” *IEEE Trans. Inf. Forensics Security*, vol. 5, no. 3, pp. 492–506, Sep. 2010.
- [11] T. Carvalho, F. A. Faria, H. Pedrini, R. da S. Torres, and A. Rocha, “Illuminant-based transformed spaces for image forensics,” *IEEE Trans. Inf. Forensics Security*, vol. 11, no. 4, pp. 720–733, Apr. 2016.
- [12] J. van de Weijer, T. Gevers, and A. Gijssenij, “Edge-based color constancy,” *IEEE Trans. Image Process.*, vol. 16, no. 9, pp. 2207–2214, Sep. 2007.
- [13] J. S. Ho, O. C. Au, J. Zhou, and Y. Guo, “Inter-channel demosaicking traces for digital image forensics,” in *Proc. IEEE Int. Conf. Multimedia Expo*, Jul. 2010, pp. 1475–1480.
- [14] F. Pitie and A. C. Kokaram, “The linear monge-kantorovitch linear colour mapping for example-based colour transfer,” in *Proc. IET CVMP*, 2007, pp. 1–9.
- [15] M. Grogan, M. Prasad, and R. Dahyot, “L2 registration for colour transfer,” in *Proc. Eur. Signal Process. Conf.*, Aug./Sep. 2015, pp. 1–5.
- [16] X. An and F. Pellacini, “AppProp: All-pairs appearance-space edit propagation,” *ACM Trans. Graph.*, vol. 27, no. 3, 2008, Art. no. 40.
- [17] R. Chamlawi, A. Khan, and I. Usman, “Authentication and recovery of images using multiple watermarks,” *Comput. Elect. Eng.*, vol. 36, no. 3, pp. 578–584, 2010.
- [18] G. S. Spagnolo and M. De Santis, “Holographic watermarking for authentication of cut images,” *Opt. Lasers Eng.*, vol. 49, no. 12, pp. 1447–1455, 2011.
- [19] L. Rosales-Roldan, M. Cedillo-Hernandez, M. Nakano-Miyatake, H. Perez-Meana, and B. Kurkoski, “Watermarking-based image authentication with recovery capability using halftoning technique,” *Signal Process. Image Commun.*, vol. 28, no. 1, pp. 69–83, 2013.
- [20] C.-S. Lu and H.-Y. M. Liao, “Structural digital signature for image authentication: An incidental distortion resistant scheme,” *IEEE Trans. Multimedia*, vol. 5, no. 2, pp. 161–173, Jun. 2003.
- [21] X. Wang, J. Xue, Z. Zheng, Z. Liu, and N. Li, “Image forensic signature for content authenticity analysis,” *J. Vis. Commun. Image Represent.*, vol. 23, no. 5, pp. 782–797, 2012.
- [22] M. Sengupta and J. Mandal, “Authentication through hough transformation generated signature on G-let D3 domain,” *Procedia Technol.*, vol. 10, pp. 121–130, Dec. 2013.
- [23] Z.-P. Zhou and X.-X. Zhang, “Image splicing detection based on image quality and analysis of variance,” in *Proc. Int. Conf. Edu. Technol. Comput.*, Jun. 2010, pp. 242–246.
- [24] C.-M. Pun, X.-C. Yuan, and X.-L. Bi, “Image forgery detection using adaptive oversegmentation and feature point matching,” *IEEE Trans. Inf. Forensics Security*, vol. 10, no. 8, pp. 1705–1716, Aug. 2015.
- [25] Y. Cao, T. Gao, L. Fan, and Q. Yang, “A robust detection algorithm for copy-move forgery in digital images,” *Forensic Sci. Int.*, vol. 214, nos. 1–3, pp. 33–43, 2012.
- [26] J. Zhao and J. Guo, “Passive forensics for copy-move image forgery using a method based on DCT and SVD,” *Forensic Sci. Int.*, vol. 233, nos. 1–3, pp. 158–166, 2013.
- [27] L. Jing and C. Shao, “Image copy-move forgery detecting based on local invariant feature,” *J. Multimedia*, vol. 7, no. 1, pp. 90–97, 2012.
- [28] Z. Junhong, “Detection of copy-move forgery based on one improved lle method,” in *Proc. 2nd Int. Conf. Adv. Comput. Control*, Mar. 2010, pp. 547–550.

- [29] C.-M. Hsu, J.-C. Lee, and W.-K. Chen, "An efficient detection algorithm for copy-move forgery," in *Proc. 10th Asia Joint Conf. Inf. Secur.*, May 2015, pp. 33–36.
- [30] Y. Guo, X. Cao, W. Zhang, and R. Wang, "Fake colorized image detection," *IEEE Trans. Inf. Forensics Security*, vol. 13, no. 8, pp. 1932–1944, Aug. 2018.
- [31] M. Chen, J. Fridrich, M. Goljan, and J. Lukás, "Determining image origin and integrity using sensor noise," *IEEE Trans. Inf. Forensics Security*, vol. 3, no. 1, pp. 74–90, Mar. 2008.
- [32] A. Levin, D. Lischinski, and Y. Weiss, "Colorization using optimization," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 689–694, 2004.
- [33] X. Chen, D. Zou, J. Li, X. Cao, Q. Zhao, and H. Zhang, "Sparse dictionary learning for edit propagation of high-resolution images," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2014, pp. 2854–2861.
- [34] S. Lin, D. Ritchie, M. Fisher, and P. Hanrahan, "Probabilistic color-by-numbers: Suggesting pattern colorizations using factor graphs," *ACM Trans. Graph.*, vol. 32, no. 4, 2013, Art. no. 37.
- [35] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2012, pp. 1097–1105.
- [36] C. Szegedy et al., "Going deeper with convolutions," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2015, pp. 1–9.
- [37] W. Ren, S. Liu, H. Zhang, J. Pan, X. Cao, and M.-H. Yang, "Single image dehazing via multi-scale convolutional neural networks," in *Proc. Eur. Conf. Comput. Vis.*, 2016, pp. 154–169.
- [38] W. Ren et al., "Gated fusion network for single image dehazing," *CoRR*, vol. abs/1804.00213, Apr. 2018.
- [39] G. K. Birajdar and V. H. Mankar, "Digital image forgery detection using passive techniques: A survey," *Digit. Invest.*, vol. 10, no. 3, pp. 226–245, 2013.
- [40] B. E. Bayer, "Color imaging array," U.S. Patent 3 971 065, Jul. 20, 1976.
- [41] B. K. Gunturk, Y. Altunbasak, and R. M. Mersereau, "Color plane interpolation using alternating projections," *IEEE Trans. Image Process.*, vol. 11, no. 9, pp. 997–1013, Sep. 2002.
- [42] J. E. Adams, Jr., and J. F. Hamilton, Jr., "Adaptive color plan interpolation in single sensor color electronic camera," U.S. Patent 5 629 734, May 13, 1997.
- [43] X. Wu and N. Zhang, "Primary-consistent soft-decision color demosaicing for digital cameras," *IEEE Trans. Image Process.*, vol. 13, no. 9, pp. 1263–1274, Sep. 2004.
- [44] D. D. Muresan and T. W. Parks, "Demosaicing using optimal recovery," *IEEE Trans. Image Process.*, vol. 14, no. 2, pp. 267–278, Feb. 2005.
- [45] M. K. Johnson and H. Farid, "Exposing digital forgeries by detecting inconsistencies in lighting," in *Proc. Workshop Multimedia Secur.*, New York, NY, USA, 2005, pp. 1–10.
- [46] M. K. Johnson and H. Farid, "Exposing digital forgeries through specular highlights on the eye," in *Proc. Int. Conf. Inf. Hiding*, 2007, pp. 311–325.
- [47] W. Fan, K. Wang, F. Cayre, and Z. Xiong, "3D lighting-based image forgery detection using shape-from-shading," in *Proc. Signal Process. Conf.*, 2012, pp. 1777–1781.
- [48] S. Gholap and P. K. Bora, "Illuminant colour based image forensics," in *Proc. IEEE Region 10 Conf. TENCON*, Nov. 2008, pp. 1–5.
- [49] C. Riess and E. Angelopoulou, "Scene illumination as an indicator of image manipulation," in *Proc. Int. Conf. Inf. Hiding*, 2010, pp. 66–80.
- [50] X. Wu and Z. Fang, "Image splicing detection using illuminant color inconsistency," in *Proc. 3rd Int. Conf. Multimedia Inf. Netw. Secur.*, Nov. 2011, pp. 600–603.
- [51] S. Bianco, C. Cusano, and R. Schettini, "Single and multiple illuminant estimation using convolutional neural networks," *IEEE Trans. Image Process.*, vol. 26, no. 9, pp. 4347–4362, Sep. 2017.
- [52] G. Buchsbaum, "A spatial processor model for object colour perception," *J. Franklin Inst.*, vol. 310, no. 1, pp. 1–26, Jul. 1980.
- [53] A. Çarkacıoğlu and F. Yarman-Vural, "SASI: A generic texture descriptor for image retrieval," *Pattern Recognit.*, vol. 36, no. 11, pp. 2615–2633, 2003.
- [54] B. Tao and B. W. Dickinson, "Texture recognition and image retrieval using gradient indexing," *J. Vis. Commun. Image Represent.*, vol. 11, no. 3, pp. 327–342, 2000.
- [55] J. Huang, S. R. Kumar, M. Mitra, W.-J. Zhu, and R. Zabih, "Image indexing using color correlograms," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 1997, pp. 762–768.
- [56] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *CoRR*, vol. abs/1409.1556, Sep. 2014.
- [57] A. Gijsenij, T. Gevers, and J. van de Weijer, "Improving color constancy by photometric edge weighting," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 5, pp. 918–929, May 2012.
- [58] B. Bayar and M. C. Stamm, "A generic approach towards image manipulation parameter estimation using convolutional neural networks," in *Proc. 5th ACM Workshop Inf. Hiding Multimedia Secur.*, 2017, pp. 147–157.
- [59] M. Goljan, J. Fridrich, and R. Cogranne, "Rich model for steganalysis of color images," in *Proc. IEEE Int. Workshop Inf. Forensics Secur. (WIFS)*, Dec. 2014, pp. 185–190.



**Yanyang Yan** received the B.E. degree from the University of Science and Technology of China, Hefei, China. He is currently pursuing the Ph.D. degree with the Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China. His research interests include image/video deblurring, enhancement, and related vision problems.



**Wenqi Ren** received the Ph.D. degree from Tianjin University, Tianjin, China, in 2017. From 2015 to 2016, he was a joint-training Ph.D. Student with the Electrical Engineering and Computer Science Department, University of California at Merced. He is currently an Assistant Professor with the Institute of Information Engineering, Chinese Academy of Sciences, China. His research interests include image/video analysis and enhancement. He received the Tencent Rhino Bird Elite Graduate Program Scholarship and CVPR Doctoral Consortium in 2017.



**Xiaochun Cao** received the B.E. and M.E. degrees in computer science from Beihang University, Beijing, China, and the Ph.D. degree in computer science from the University of Central Florida, Orlando, FL, USA. He has been a Professor with the Institute of Information Engineering, Chinese Academy of Sciences, Beijing, since 2012. After graduation, he spent about three years at ObjectVideo Inc., as a Research Scientist. From 2008 to 2012, he was a Professor with Tianjin University, Tianjin, China. He has authored and co-authored over 120 journal and conference papers. He is a fellow of the IET. His dissertation was nominated for the University of Central Florida's University-Level Outstanding Dissertation Award. In 2004 and 2010, he was a recipient of the Piero Zamperoni Best Student Paper Award at the International Conference on Pattern Recognition. He is on the Editorial Boards of the *IEEE TRANSACTIONS ON IMAGE PROCESSING*, the *IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY*, and the *IEEE TRANSACTIONS ON MULTIMEDIA*.