IDENTIFYING PHOTOREALISTIC COMPUTER GRAPHICS USING CONVOLUTIONAL NEURAL NETWORKS

*In-Jae Yu, Do-Guk Kim, Jin-Seok Park, Jong-Uk Hou, Sunghee Choi, and Heung-Kyu Lee**

Korea Advanced Institute of Science and Technology
School of Computing
(ijyu[1], dgkim[2], jspark[3], juheo[4])@mmc.kaist.ac.kr, (sunghee,heunglee)@kaist.ac.kr†

ABSTRACT

As computer graphics technology advances, it is becoming increasingly difficult to determine whether a given picture was taken by camera or via computer graphics. In this work, we propose a method to using simple CNN structures to identify photorealistic computer graphics (PRCG) using convolutional neural networks (CNN). This network trained to identify the source of image patches. We showed the network without pooling layer showed 98.2% accuracy, which is 2.1% higher than the result of using conventional object-recognition network. Testing random patches from image, the accuracy of identifying image reached 98.5%. Furthermore, it is possible to detect the photograph-PRCG synthesized regions from the image.

Index Terms— Digital Forensics, Image Source Identification, Convolutional Neural Networks, Photo-Realistic Computer Graphics

1. INTRODUCTION

Camera and computer graphics technology has been developing simultaneously. The development of digital camera technology has dramatically improved the performance of the built-in camera in the mobile phone, and now everyone with a cell phone can shoot high-resolution or full HD images. At the same time, computer graphics technology keep developing. With the development of rendering software, the 3D modeling of the scene has become more sophisticated and the results has became more realistic after rendering. In addition, with the development of GPU technology, the level of real-time graphics such as computer game has also improved greatly. Therefore, it is very easy to get a graphic image that looks similar to a real photograph if the user requires it, and the quality is also very high. We call these images as photorealistic computer graphics (**PRCG**). As shown in **Fig. 1**, the current graphics technology has developed to describe a direction of the light and texture of the objects.

*Corresponding Author

(a) PRCG (b) Photograph

Fig. 1: Examples of PRCG and photograph images

So far, studies have been conducted to identify PRCG and photographs. They classified these images using machine learning techniques such as Support Vector Machine (**SVM**) after calculating statistical properties in images. Ng et al. [1] used feature vector consisting of 33 dimensional power spectrum, 24-dimensional local patch features, and 72-dimensional high order wavelet coefficients. Lyu et al. [2] proposed a method of using wavelet models of natural images. A 216-dimensional feature vector generated by constructing QMF pyramid in each color channel of the image was learned through SVM. Wang et al. [3] proposed 70-dimensional contourlet transformation and homomorphic filtering features. Peng et al. [4] proposed 31-dimensional texture and statistic features and achieved over 97% accuracy for both PRCG and photograph using LIBSVM [5].

Although existing PRCG identification methods use different statistical features, but there is no clear relationship between them. Features such as third or fourth order wavelet coefficients may be appropriate for PRCG detection after feature calculation, but there is insufficient explanation as to why such differences exist. On the other hand, when extracting features, existing techniques work on the entire image, or split the image, and extract feature from each image block to merge into a single feature. Depending on the size of image, it is difficult to determine whether the technology works well. Also, when photograph and PRCG are synthesized into one images, these methods are difficult to make accurate judgments.

In this work, we propose a method using convolutional neural networks (**CNN**) to identify PRCG. This network was trained to classify input image patches to PRCG or photo-

graph. Unlike the existing studies, the features required for the training are learned by the network. In addition, since the trained network identifies only for a small image patch, not an entire image, it can perform two functions depending on its use. First, based on the high accuracy of network on the image patch, it is possible to identify whether the image is PRCG with small number of patches from the image. Second, it is possible to detect the area where PRCG is synthesized, by testing a very large number of patches in a constant grid.

The rest of this paper is organized as follows: The proposed method is described in Section 2. The experiment and discussion is presented in Section 3. The conclusion and future works is contained in Section 4.

2. THE PROPOSED METHOD

2.1. Network training process

Fig. 2 describes the training process. In **preprocess phase**, image patches are randomly extracted from the photograph and PRCG dataset. Size of each patch is 32 $\times$32 $\times$ 3. The photograph database consisted of images taken from different digital cameras and mobile phones. The PRCG dataset consisted of images collected from the web such as PRCG competition and dataset from the previous works. The number of patches selected from the photograph and PRCG dataset were same.

In **network traning phase**, a CNN is trained to classify the input patch into photograph and PRCG. The network was trained using the prepared patch data.

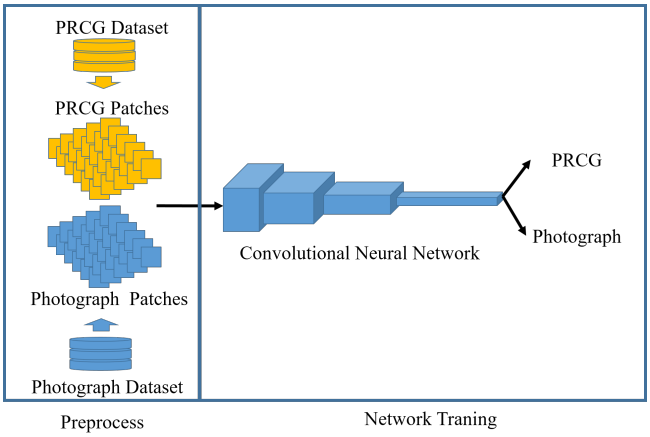


Fig. 2: Training process

2.2. PRCG identification

The previous methods used an entire image or a very large part of the image in training and test phase. Therfore, previoius methods directly made decision on image whether it is PRCG or not.

In our method, the trained network judges only for tiny patch of image. To test the image, we pick image patches randomly from the images. The image is determined to be PRCG if the ratio of the patches classifed to PRCG is over 50%. Overall process is described in **Fig. 3**.

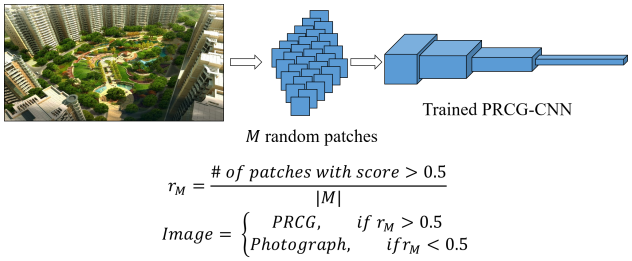


Fig. 3: PRCG identification

2.3. PRCG-photograph synthesized area detection

It is possible to detect area where PRCG and photograph are synthesized from the image using PRCG-CNN. Rather than randomly extracting patches from the image, the patch is extracted at regular intervals and tested with the learned network. Each grid was colored **white** when it was classified as PRCG, and otherwise it was colored **black**.

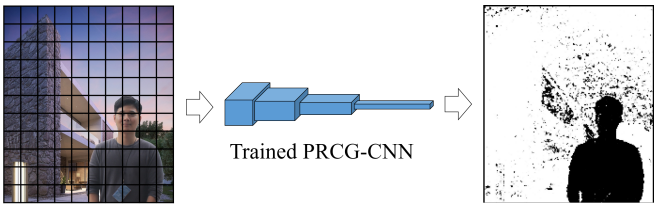


Fig. 4: Synthesized area detection

2.4. Network models used in training

The basic structure of the networks is VGG-net [6]. We replaced dropout [7] with batch normalization [8]. Because the size of image patches is much smaller than that of the ImageNet dataset [9], we used a structure that reduced the depth and filter size of the existing VGG-net.

We used one more network model, which consist of just convolutional layers without pooling layer. The reason of using such network is that max-pooling layer is not suitable for training low-level difference compare to ordinary classification task that use high level features. Thus, we removed pooling layer.

The two networks are referred to as **Type 1** and **Type 2** , respectively. **Fig. 5** represents the detailed structure of each network. The difference between the two networks is that Type 2 has no pooling layer and padding is removed from

each convolutional layer to reduce the number of training parameters.

Type 1	Type 2
Conv 3 × 3 ×32	Conv 3 × 3 ×32
Conv 3 × 3 ×32	Conv 3 × 3 ×32
Max-pooling 2 ×2	Conv 3 × 3 ×64
Conv 3 × 3 ×64	Conv 3 × 3 × 64
Conv 3 × 3 × 64	Conv 3 × 3 × 128
Max-pooling 2 ×2	Conv 3 × 3 × 128
Conv 3 × 3 × 128	FC, output: 1024
Conv 3 × 3 × 128	FC, output: 1024
Max-pooling 2 ×2	FC, output: 2
FC, output: 1024	Softmax
FC, output: 1024	
FC, output: 2	
Softmax	

Fig. 5: Architecture of two networks

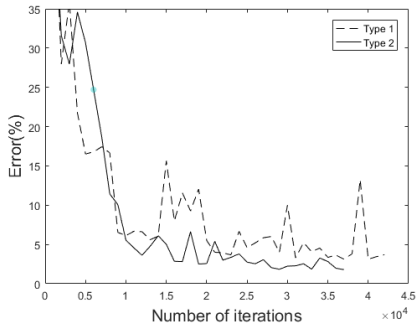
3. EXPERIMENTS AND DISCUSSIONS

Photograph dataset consisted of 1000 JPEG images taken from several digital cameras and cell phones. PRCG dataset consisted of jpeg 1000 images from Columbia Image Dataset [10], and images from the photorealism competition conducted monthly by the 3d rendering softwares (e.g. 3D Studio Max, Maya, Blender). Recently rendered images have a resolution from HD to FullHD, while the images in the Columbia dataset has very low resolution. For each photograph and PRCG dataset, the number of images used for training and test were 750 and 250, respectively. Image patches were extracted from the datasets to use as input to the network. The number of patches used to training and test were 200,000 and 80,000, respectively. The computer used in the experiment consisted of i7-6770K 4GHz CPU and NVIDIA Geforce GTX 1070 GPU with 8GB memory.

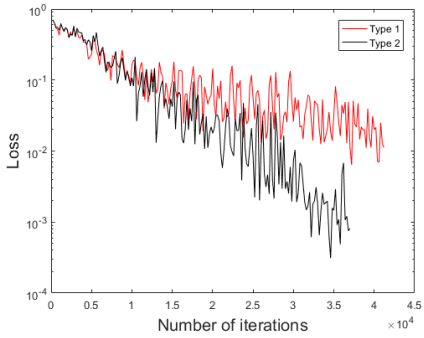
3.1. Training result of the network

We used Caffe [11] for experiment. All the learning parameters (e.g. learning rate, batch size) applied to the two networks were same. **Fig 6**(a) represents the performance of two networks. The test error of each network is reduced to 3.81%, 1.78%, respectively. In the case of testing 1000 patches randomly selected from each image (Section 2.2), it was possible to make accurate decision on 98%. The rate of mis-classifying the patch extracted from the PRCG was much lower than that of the photograph.

Fig 7(a) is result of paste photograph-oriented object into PRCG. It shows that the network correctly detects the synthesized region from the image.



(a) Test error of each networks



(b) Training loss of each networks

Fig. 6: Training/Test result of each network: there is no significant difference in the error reduction tendency, but Type 2 showed a stable error reduction. Training loss of Type 2 decreased much more rapidly.

3.2. The effect of eliminating pooling

Fig 6(b) shows that the training loss between two networks is 100 times different. By applying patches that are extracted in constant grid from the image to the trained network showed that the Type 1 is less robust to edge area (**Fig. 8**). This is because passing through the pooling layer loses association between adjacent pixels. Although pooling enhance the performance of object recognition by finding hierarchically high-level features and speed up the training, but it is not suited to digital forensics.

3.3. Robustness on image processing

Fig 9 shows the result of robustness test to image processing on photographs. In the case of PRCG, almost all of the patches were classifed to PRCG before image processing, which did not change after processing. On the other hand,

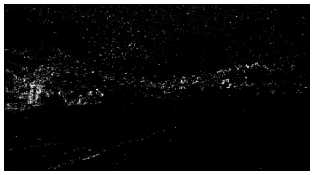
(a) Synthesized image

(b) Detection result

Fig. 7: Trained network succesfully detects synthesized region from the image

(a) Photograph

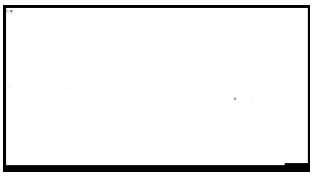
(d) PRCG



(b) Result of Type1

(e) Result of Type1

(c) Result of Type2



(f) Result of Type2

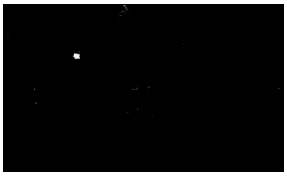
Fig. 8: Result of test on patches with constant grid from photograph and PRCG

in the case of photographs, image processing significantly loses the characteristics of photography, which significantly increases the rate of inaccurate network decision.

If the the patches classified to PRCG is uniformly distributed throughout the image, it is able to use the network to detect the synthesized region, but the distribution is skewed according to the image content. Therefore, the current network is not suitable to detect synthesized region if the image has been processed.

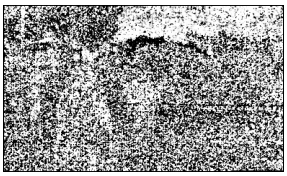
4. CONCLUSION

In this paper, we proposed a method for identifying PRCG using convolutional neural networks. We trained network to classify the tiny image patch into photograph and PRCG. Both object-recognition network (Type 1) and network with-

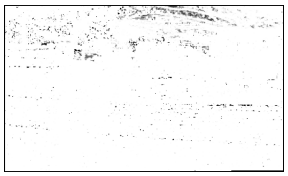


(a) Photograph

(b) Resized to half



(c) JPEG 90



(d) JPEG 70

Fig. 9: Test on photograph for robustness to image processing

out pooling (Type 2) showed excellent results, while Type 2 showed less sensitive performance to edge and plain regions. Furthermore, with the high accuracy of network, it was possible to detect synthesized regions from the image. However, the proposed network was not robust to image processing such as resizing and JPEG compression. For future works, we plan to design a network having high accuracy, and robust to various image processing. We also plan to extend this work to apply for various forensic tasks.

Acknowledgement

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MSIP) (No. 2016R1A2B2009595)

5. REFERENCES

[1] Tian-Tsong Ng and Shih-Fu Chang, "Classifying photographic and photorealistic computer graphic images using natural image statistics," 2004.

[2] Siwei Lyu and Hany Farid, "How realistic is photorealistic?," *IEEE Transactions on Signal Processing*, vol. 53, no. 2, pp. 845–850, 2005.

[3] Xiaofeng Wang, Yong Liu, Bingchao Xu, Lu Li, and Jianru Xue, "A statistical feature based approach to distinguish prcg from photographs," *Computer Vision and Image Understanding*, vol. 128, pp. 84–93, 2014.

[4] Fei Peng, Jiao-ting Li, and Min Long, "Identification of natural images and computer-generated graphics based on statistical and textural features," *Journal of forensic sciences*, vol. 60, no. 2, pp. 435–443, 2015.

[5] Chih-Chung Chang and Chih-Jen Lin, "Libsvm: a library for support vector machines," *ACM Transactions*

on Intelligent Systems and Technology (TIST), vol. 2, no. 3, pp. 27, 2011.

[6] Ken Chatfield, Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman, "Return of the devil in the details: Delving deep into convolutional nets," *arXiv preprint arXiv:1405.3531*, 2014.

[7] Nitish Srivastava, Geoffrey E Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting.," *Journal of Machine Learning Research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[8] Sergey Ioffe and Christian Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167*, 2015.

[9] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei, "ImageNet Large Scale Visual Recognition Challenge," *International Journal of Computer Vision (IJCV)*, vol. 115, no. 3, pp. 211–252, 2015.

[10] J. Hsu T.-T Ng, S.-F. Chang and M. Pepeljugoski, "Columbia photographic images and photorealistic computer graphics dataset," Tech. Rep. 205-2004-5, ADVENT, Columbia University, 2004.

[11] Yangqing Jia, Evan Shelhamer, Jeff Donahue, Sergey Karayev, Jonathan Long, Ross Girshick, Sergio Guadarrama, and Trevor Darrell, "Caffe: Convolutional architecture for fast feature embedding," *arXiv preprint arXiv:1408.5093*, 2014.