Splicing Detection And Localization In Satellite Imagery Using Conditional GANs

Emily R. Bartusiak[1], Sri Kalyan Yarlagadda[1], David Güera[1], Fengqing M Zhu[1], Paolo Bestagini[2], Stefano Tubaro[2], and Edward J. Delp[1]

[1]Department of Electrical and Computer Engineering, Purdue University, West Lafayette
[2]Department of Electronics, Information and Bioengineering, Politecnico di Milano, Italy

Abstract

The widespread availability of image editing tools and improvements in image processing techniques allow image manipulation to be very easy. Oftentimes, easy-to-use yet sophisticated image manipulation tools yields distortions/changes imperceptible to the human observer. Distribution of forged images can have drastic ramifications, especially when coupled with the speed and vastness of the Internet. Therefore, verifying image integrity poses an immense and important challenge to the digital forensic community. Satellite images specifically can be modified in a number of ways, including the insertion of objects to hide existing scenes and structures. In this paper, we describe the use of a Conditional Generative Adversarial Network (cGAN) to identify the presence of such spliced forgeries within satellite images. Additionally, we identify their locations and shapes. Trained on pristine and falsified images, our method achieves high success on these detection and localization objectives.

1 Introduction

Proper communication at both the public and personal level is key to the healthy development of human civilization. Over the years the means of communication has evolved, and in the present day the Internet is the most popular and important platform for communication. Many social media systems have developed using the Internet and they provide a very cheap and effective way to express and shares one's ideas with the rest of the world. While an effective communication system for sharing information could help us become more informed and connected as a society, it could also be used to spread misinformation to achieve a nefarious objective. Hence, it is of paramount importance that we verify and authenticate the shared data on these systems.

While there are many ways of communicating ideas, such as speech, symbols, and written text, images are today one of the most popular means. Unfortunately, manipulating images has become very easy. Tools such as GIMP and Photoshop can be used to manipulate images in a wide variety of ways and they are easily accessible to the general public. To address this problem, the forensic community has developed a wide variety of tools to detect various kinds of image forgeries [17, 16, 19]. While most of the images shared on the internet come from consumer cameras and smart-phones, other types of imagery such as satellite images are also very important in business and government applications and thus posing new problems for the forensic community [7, 5].

With the increase in the number of satellites equipped with imaging sensors and the technological advancements made in satellite imaging technology, high resolution images of the ground are becoming popular. It is now possible to not only access these overhead images from public websites [1] but also to buy custom satellite imagery of specific locations. Just like any other image, satellite images can also be doctored. While the forensic community has been developing tools to address forgeries of all types, they have been biased towards imagery captured from consumer cameras and smartphones [9, 15, 11, 10]. The nature of acquisition of satellite imagery is quite different from that of images from consumer cameras hence it's important that forensic tools be developed that specifically target satellite imagery.

In the recent years, some methods [13, 20, 8] for satellite image forgeries have been developed. In [13] the authors have proposed an active forensic method based on watermarks to verify the authenticity of a satellite image. While watermarks are an effective way of ascertaining whether an image is forged or not, their absence renders such methods ineffective. In [8] the authors have proposed a passive method based on machine learning to detect inpainting in satellite images. The authors in [20] have proposed a method based on deep learning to detect splicing in satellite images. They employ Generative Adversarial Networks (GANs) [12] to learn a compact representation of pristine satellite images and use it to detect splicing of various sizes.

In this paper, we discuss the detection and localization of splicing in satellite images. Splicing refers to replacement of pixels of a region of the image to add or remove an object. We employ a Conditional Generative Adversarial Network (cGAN) to learn a mapping from a satellite image to its splicing mask. The trained cGAN operates on a satellite image of interest and outputs a mask of the same resolution that is indicative of the likelihood of a pixel belonging to a spliced region. Our cGAN's architecture is an extension of the popular pix2pix [14]. Our approach differs from [20] because we are trying to learn a direct mapping from an image to its forgery mask and to achieve this we provide both pristine and spliced images to train our model, while the authors in [20] only use pristine images for training as they are trying to learn a compact representation of the pristine data and use it to identify forgeries.

We use the dataset proposed in [20] to validate our method. We report both the localization and detection performance.

2 Problem Formulation

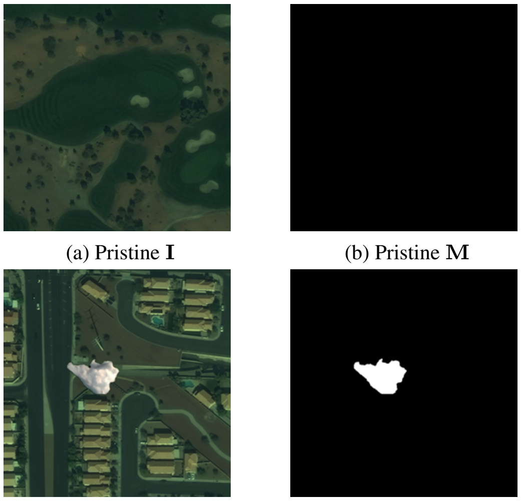
We investigate the following two specific objectives in this paper, forgery detection and localization. *Detection* refers to the goal of determining if an RGB satellite image $\mathbf{I}$ has been modified via splicing. It is a binary classification problem where images can be considered *forged*, if they have been modified, or *pristine*, if not. *Localization* refers to the image segmentation goal of identifying each pixel in a forged image that belongs to the spliced entity, otherwise known as the *forgery*. These goals are defined in a similar manner to those outlined in [20].

Forgery masks $\mathbf{M}$ are used to help us visualize and determine the outcomes for these objectives. For an image $\mathbf{I}$, a forgery mask $\mathbf{M}$ of the same dimensions shows the forgery in $\mathbf{I}$ (if it exists). In other words, for a satellite image $\mathbf{I}(x, y)$ where (x, y) specifies the coordinate location of a pixel in $\mathbf{I}$, the corresponding forgery mask $\mathbf{M}$ is comprised of values defined as

$$\mathbf{M}(x, y) = \begin{cases} 255 & \text{if } \mathbf{I}(x, y) \text{ is forged,} \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

Therefore, the shape, size, and location of a forgery in an image $\mathbf{I}$ can be ascertained from the mask $\mathbf{M}$ if it contains white pixel values (i.e., 255). At an extreme, an entirely white mask ($\mathbf{M} \neq 0$) indicates that every pixel in $\mathbf{I}$ has been manipulated, whereas an entirely black mask ($\mathbf{M} = 0$) represents a pristine image.

Our approach is to train a cGAN to create $\hat{\mathbf{M}}$, an estimate of the forgery mask $\mathbf{M}$. $\mathbf{I}$ is considered doctored if $\hat{\mathbf{M}} \not\approx 0$, meaning that a forgery is detected in it and is comprised of the pixels located at $\{(x, y) : \hat{\mathbf{M}}(x, y) \neq 0\}$. On the other hand, the image $\mathbf{I}$ is considered pristine if no forgery is detected, indicated by $\hat{\mathbf{M}} \approx \mathbf{0}$. Examples of satellite images $\mathbf{I}$ and their corresponding ground truth forgery masks $\mathbf{M}$ can be seen in Figure 1.



(a) Pristine $\mathbf{I}$

(b) Pristine $\mathbf{M}$

(c) Forged $\mathbf{I}$

(d) Forged $\mathbf{M}$

Figure 1: Image - mask $\{\mathbf{I}, \mathbf{M}\}$ pairs. (a) and (c) portray two example satellite images under analysis. (b) and (d) illustrate their corresponding ground truth masks. A pristine image's mask is entirely black, like (b). (d) displays a mask that contains a forgery. The cGAN will try to create mask estimates that resemble masks (b) and (d).

3 Method

In this section we describe our technique for splicing detection and localization. Additional details about the general cGAN concepts reported in this section can be found in [14]. We train our cGAN on both pristine and forged images to learn a mapping from an input image $\mathbf{I}$ to a forgery mask

$\mathbf{M}$. It consists of two parts: a generator G and a discriminator D. Figure 2 shows the overall cGAN architecture.

The generator G has a 16-layer U-net architecture (8 encoder layers, 8 decoder layers) with skip connections [18]. When G is presented with an image $\mathbf{I}$, it computes an estimated forgery mask $\hat{\mathbf{M}}$, defined as $\hat{\mathbf{M}} = G(\mathbf{I})$. The generator's objective is to create $\hat{\mathbf{M}}$ that is close to the true $\mathbf{M}$. Meanwhile, the discriminator D is trained to differentiate between the true input-mask pairs $\{\mathbf{I}, \mathbf{M}\}$ and synthesized input-mask pairs $\{\mathbf{I}, \hat{\mathbf{M}}\}$ coming from the generator. In a cGAN, the generator and the discriminator are coupled via a loss function as defined in equation 4. During the course of training the discriminator forces the generator to produce masks that are not only close to the ground truth but also good enough that the discriminator cannot distinguish them from the ground truth thus making the generator do a better job.

The discriminator, D has an architecture of a 5-layer CNN that does binary classification on masks. Sometimes, a true image-mask pair $\{\mathbf{I}, \mathbf{M}\}$ is presented to D. Other times, an image-mask estimate pair $\{\mathbf{I}, \hat{\mathbf{M}}\}$ is presented. In both cases, the image under analysis $\mathbf{I}$ is presented to the discriminator D along with either a true forgery mask $\mathbf{M}$ or a synthesized forgery mask $\hat{\mathbf{M}}$. D divides the input into patches of size 70x70 pixels. It then classifies each patch as forged or pristine, assigning labels 0 and 1 respectively. The values for all of the patches are averaged to determine the classification for the entire input. The following equations describe the two cases outlined in this paragraph:

$$D(\mathbf{I}, \hat{\mathbf{M}}) = D(\mathbf{I}, G(\mathbf{I})) = 0, \tag{2}$$

$$D(\mathbf{I}, \mathbf{M}) = 1. \tag{3}$$

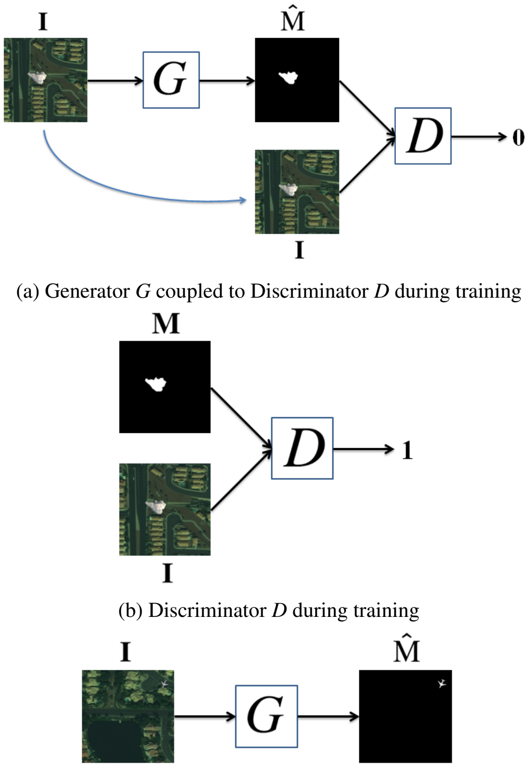
The generator G and the discriminator D compete in a min-max game, training and improving each other over time. The coupled loss function of the network is described in the following equations:

$$\begin{aligned} \mathcal{L}_{\text{cGAN}}(G, D) = &\mathbb{E}_{\mathbf{I},\mathbf{M}}[\log(D(\mathbf{I}, \mathbf{M}))] + \\ &\mathbb{E}_{\mathbf{I},\mathbf{z}}[\log(1 - D(\mathbf{I}, G(\mathbf{I}))]. \end{aligned} \tag{4}$$

So far, we have described a network in which the generator G learns to create masks $\hat{\mathbf{M}}$ that could be mistaken for real forgery masks by D. However, this does not ensure that the synthesized masks will correctly show forgeries in images. For example, $\hat{\mathbf{M}}$ may "fool" D and be classified as an authentic mask for $\mathbf{I}$ without resembling its ground truth mask. In such a case, $\hat{\mathbf{M}} \not\approx \mathbf{M}$. Therefore, we impose an additional constraint on the generator so that it learns to reconstruct the ground truth masks of training images, i.e., $\hat{\mathbf{M}} \approx \mathbf{M}$. This can be achieved by training G to minimize reconstruction loss $\mathcal{L}_{\text{R}}$ between $\hat{\mathbf{M}}$ and $\mathbf{M}$. Since our task is to primarily classify every individual pixel into two classes i.e. forged or pristine we chose $\mathcal{L}_{\text{R}}$ to be the binary cross-entropy (BCE) loss. This is different with respect to the classic pix2pix which uses L_1 as their $\mathcal{L}_{\text{R}}$. We later on verify in our experiments that BCE is indeed a better choice over L_1. The total loss function of the cGAN is denoted as:

$$\mathcal{L} = \mathcal{L}_{\text{cGAN}} + \lambda \mathcal{L}_{\text{R}}. \tag{5}$$

Once training is complete, the generator G is capable of producing masks $\hat{\mathbf{M}}$ that are realistic and close to $\mathbf{M}$. To test new images under analysis, the discriminator is not considered, and the generator is used to produce mask estimates.



(a) Generator G coupled to Discriminator D during training

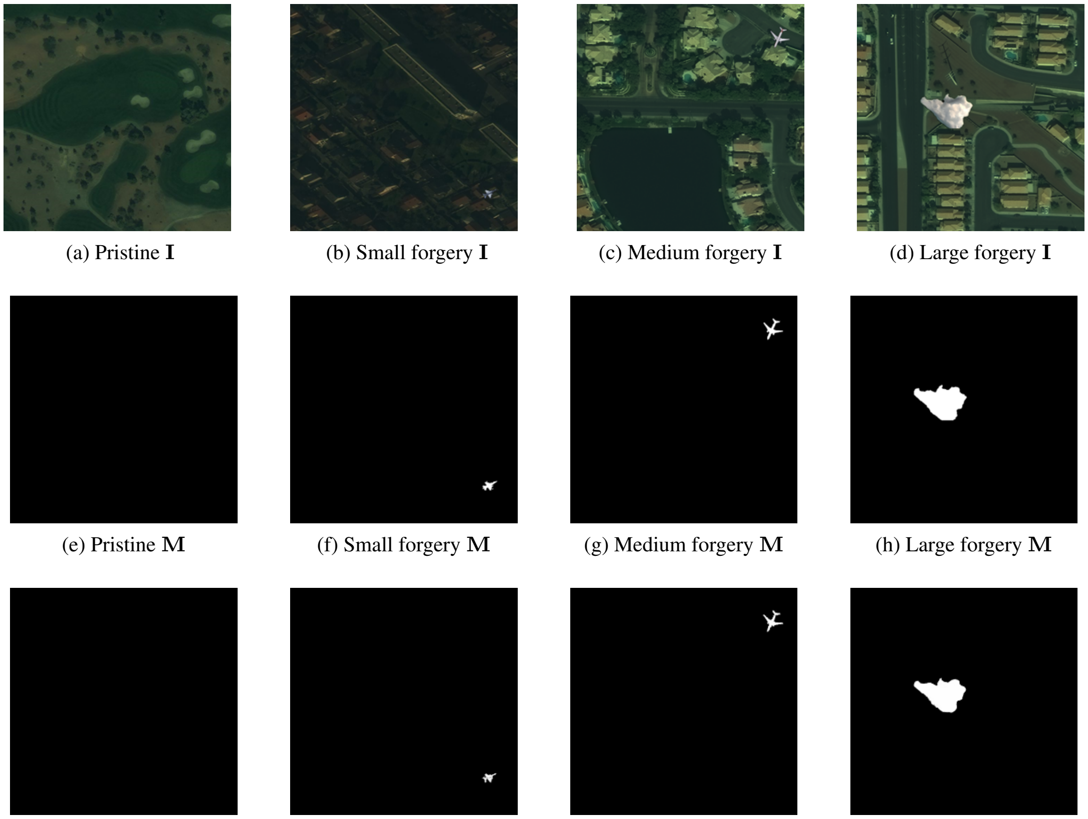
(b) Discriminator D during training

(c) Generator G after training

Figure 2: cGAN architecture. (a) shows the *G-D* training configuration, where G produces mask estimate $\hat{\mathbf{M}}$ and presents it to D for evaluation. D attempts to classify non-authentic $\{\mathbf{I}, \hat{\mathbf{M}}\}$ pairs as 0. (b) depicts D during training when presented with a true mask $\mathbf{M}$. It attempts to classify true $\{\mathbf{I}, \mathbf{M}\}$ pairs as 1. The model resembles (c) after training is complete.

4 Experimental Validation

In this section, we report the details of our experiments. First, we describe the image dataset. Next, training strategies are discussed. Finally, we present experimental results and analysis.



(a) Pristine $\mathbf{I}$ (b) Small forgery $\mathbf{I}$ (c) Medium forgery $\mathbf{I}$ (d) Large forgery $\mathbf{I}$

(e) Pristine $\mathbf{M}$ (f) Small forgery $\mathbf{M}$ (g) Medium forgery $\mathbf{M}$ (h) Large forgery $\mathbf{M}$

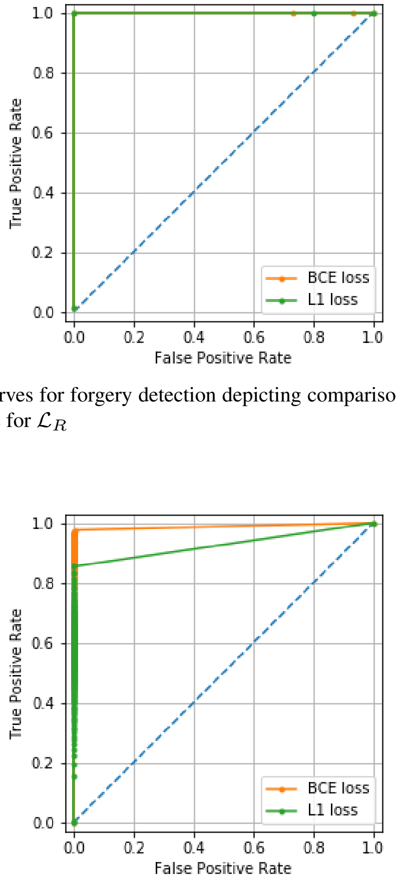
(i) Pristine $\hat{\mathbf{M}}$ (j) Small forgery $\hat{\mathbf{M}}$ (k) Medium forgery $\hat{\mathbf{M}}$ (l) Large forgery $\hat{\mathbf{M}}$

Figure 3: Input images, ground truth masks, and generated mask estimates. Each column presents a set of $\{\mathbf{I}, \mathbf{M}, \hat{\mathbf{M}}\}$ examples for different types of images.

We utilized the dataset presented in [20] for our experiments. It contains color images of overhead scenes from a satellite and their corresponding ground truth forgery masks. Each image-mask pair is defined as $\{\mathbf{I},\ \mathbf{M}\}$ and has resolution 650×650 pixels. The images were adapted from ones originally provided by the Landsat Science program [2, 3] run jointly by NASA [4] and US Geological Survey (USGS) [6]. To create forged images, objects such as airplanes and clouds were spliced into some of the images at random locations. These doctored images fall into one of three size categories (small, medium, or large) based on the approximate dimensions of the forgery they contain relative to the patch dimensions (64x64 pixels) used by the discriminator *D* to analyze a mask. Small forgeries are approximately 32x32 pixels; medium forgeries are approximately 64x64 pixels, and large forgeries are approximately 128x128 pixels. The remaining satellite images were left as pristine. For our purposes, pristine and small-forgery samples underwent data augmentation to increase the size of the training dataset. Augmentation methods included rotating pristine and small-forgery $\mathbf{I}$, $\mathbf{M}$ pairs by multiples of 90°and flipping them about the vertical and horizontal center axes. This produced our dataset $\mathcal{D}$, which contains 344 total $\{\mathbf{I},\ \mathbf{M}\}$ pairs. Also, 158 pairs contain small forgeries, 32 pairs contain medium forgeries, 31 pairs contain large forgeries, and 123 are pristine. These subsets of $\mathcal{D}$ are denoted as $\mathcal{D}_S$, $\mathcal{D}_M$, $\mathcal{D}_L$, and $\mathcal{D}_P$, respectively. Examples are shown in Figure 3.

The dataset $\mathcal{D}$ was split into three sets for training, validation, and testing. The training dataset $\mathcal{D}_{train}$ contains 128 $\mathcal{D}_S$ pairs and 90 $\mathcal{D}_P$ pairs. The validation set $\mathcal{D}_{validation}$ has 32 $\mathcal{D}_S$ pairs and 18 $\mathcal{D}_P$ pairs. The final dataset, $\mathcal{D}_{test}$, consists of 32 $\mathcal{D}_M$, 31 $\mathcal{D}_L$, and 15 $\mathcal{D}_P$ pairs. By creating disjoint training/validation and evaluation datasets, we observe how well a trained model extends to new forgery sizes. It was hypothesized that small forgeries might pose the biggest challenge to the network, so they compose the training and validation sets. The cGAN was trained for 200 epochs using the Adam optimizer with an initial learning rate of 0.0002. The reconstruction loss $mathcalL_R$ coefficient λ was set to 100. After training, the model that performed the best on $\mathcal{D}_{validation}$ was selected to use for testing.

We did both visual and numerical analysis of the results to determine the effectiveness of our proposed method. Figure 3 contains examples of mask estimates $\hat{\mathbf{M}}$ produced by G and their corresponding ground truth masks $\mathbf{M}$. It shows that the model produces mask estimates of both pristine and forged images that very closely resemble the ground truth masks, i.e., $\hat{\mathbf{M}} \approx \mathbf{M}$. Thus, we can clearly see if a forgery is present in an image $\mathbf{I}$ and, if so, its various properties. A numerical analysis of the results further verifies this.



(a) ROC curves for forgery detection depicting comparison between BCE and L_1 loss for $\mathcal{L}_R$

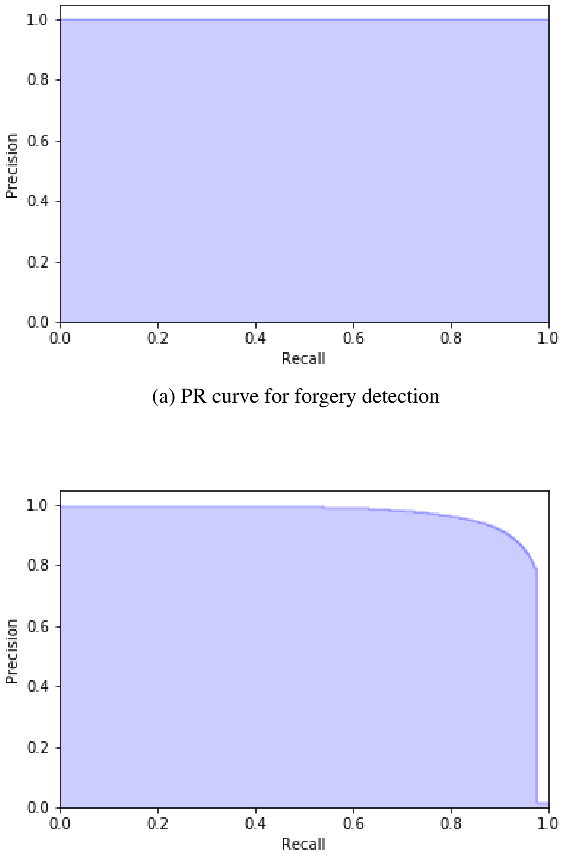
(b) ROC curves for forgery localization depicting comparison between BCE and L_1 loss for $\mathcal{L}_R$

Figure 4: ROC curves for detection and localization of spliced forgeries

To evaluate forgery detection, the average pixel value of a mask estimate is defined as

$$\hat{\mathbf{M}}_{avg} = \frac{1}{X \cdot Y} \sum_{x=1}^{X} \sum_{y=1}^{Y} \hat{\mathbf{M}}(x, y), \qquad (6)$$

where $X \times Y$ is the image resolution. Then, binary thresholding with threshold T is used to determine whether the image under analysis $\mathbf{I}$ is pristine or forged. As described above, an image is considered pristine when $\hat{\mathbf{M}} \approx 0$. From a thresholding standpoint, this is achieved when $\hat{\mathbf{M}}_{avg} < T$. Otherwise, $\mathbf{I}$ is labeled as forged. Figure 4 shows the receiver operating characteristic (ROC) curves that reveal the performance of different thresholds T. It also illustrates model performances achieved when using BCE loss and L_1 loss for reconstruction. The area under the curve (AUC) for both BCE and L_1 loss are 1.000, indicating that it is possible to achieve perfect detection accuracy with thresholding. These results are further verified by the precision-recall (PR) plot in Figure 5 for a model using BCE loss. It too indicates that perfect detection is possible with our 2-class model, as its average precision score is also 1.000.



(a) PR curve for forgery detection

(b) PR curve for forgery localization

Figure 5: PR curves for detection and localization of spliced forgeries

To assess forgery localization, a similar evaluation process occurs; however, only for images in which forgeries are detected. Their mask estimates $\hat{\mathbf{M}}$ are thresholded and then undergo a pixel-wise comparison to to their corresponding ground truth masks $\mathbf{M}$. Figure 4 also shows ROC curves for localization for different thresholds. In this case, a performance difference in BCE $\mathcal{L}_R$ versus L_1 $\mathcal{L}_R$ is observed.

BCE yields a higher AUC value of 0.988 in comparison to L_1, which achieves an AUC of 0.927. The PR curve (again using BCE loss) with an average precision score of 0.953 confirms that localization results are very good.

5 Conclusions

In this paper, we propose a forensic image analysis method based on a cGAN for splicing detection and localization in satellite images. The proposed technique exploits a data driven approach, thus learns how to distinguish forged regions from pristine ones directly from the available training data.

Results show that the developed methodology accomplishes both tampering detection and localization with incredibly high accuracy on the used dataset. Moreover, it is interesting to notice how the proposed solution is able to generalize to forgeries of different size than those seen during training.

While the results of this experiment are very good, it would be interesting to see how the technique performs on different types of forgeries, as well as on datasets containing images coming from different satellites, to further test the method generalization capability.

References

[1] 15 free satellite imagery data sources. GIS Geography *http://gisgeography.com/free-satellite-imagery-data-list*, (Accessed on 12/01/2018).

[2] Landsat on AWS. Amazon Web Services Inc. *https://aws.amazon.com/public-datasets/landsat/*, (Accessed on 12/01/2018).

[3] Landsat science. National Aeronautics and Space Administration *https://landsat.gsfc.nasa.gov/*, (Accessed on 12/01/2018).

[4] NASA. National Aeronautics and Space Administration *https://www.nasa.gov/*, (Accessed on 12/01/2018).

[5] Satellite images show clearly that russia faked its MH17 report. Mashable *http://mashable.com/2015/05/31/russia-fake-mh17-report*, (Accessed on 12/01/2018).

[6] Usgs.gov — science for a changing world. U.S. Geological Survey *https://www.usgs.gov/*, (Accessed on 12/01/2018).

[7] Conspiracy files: Who shot down MH17? BBC News *http://www.bbc.com/news/magazine-35706048*, April 2016 (accessed January 1, 2018).

[8] L. Ali, T. Kasetkasem, F. G. Khan, T. Chanwimaluang, and H. Nakahara. Identification of inpainted satellite images using evalutionary artificial neural network (EANN) and k-nearest neighbor (KNN) algorithm. *Proceedings of the IEEE International Conference of Information and Communication Technology for Embedded Systems*, May 2017. Chonburi, Thailand.

[9] M. Barni, A. Costanzo, and L. Sabatini. Identification of cut&paste tampering by means of double-JPEG detection and image segmentation. *Proceedings of the IEEE International Symposium on Circuits and Systems*, May 2010. Paris, France.

[10] L. Bondi, S. Lameri, D. Güera, P. Bestagini, E. J. Delp, and S. Tubaro. Tampering detection and localization through clustering of camera-based CNN features. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 1855–1864, July 2017. Honolulu, HI.

[11] D. Cozzolino, G. Poggi, and L. Verdoliva. Splicebuster: A new blind image splicing detector. *Proceedings of the IEEE International Workshop on Information Forensics and Security*, Nov. 2015. Rome, Italy.

[12] I. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, Cambridge, MA, 2016.

[13] A. T. S. Ho, X. Zhu, and W. M. Woon. A semi-fragile pinned sine transform watermarking system for content authentication of satellite images. *Proceedings of the IEEE International Geoscience and Remote Sensing Symposium*, January 2005. Seoul, Korea.

[14] P. Isola, J. Y. Zhu, T. Zhou, and A. A. Efros. Image-to-image translation with conditional adversarial networks. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5967–5976, July 2017. Honolulu, HI.

[15] M. Kirchner and T. Gloe. Forensic Camera Model Identification. In *Handbook of Digital Forensics of Multimedia Data and Devices*. John Wiley & Sons, Ltd, 2015.

[16] A. Piva. An overview on image forensics. *ISRN Signal Processing*, 2013:22, November 2013.

[17] A. Rocha, W. Scheirer, T. Boult, and S. Goldenstein. Vision of the unseen: Current trends and challenges in digital image and video forensics. *ACM Computing Surveys*, 43:1–42, October 2011.

[18] O. Ronneberger, P. Fischer, and T. Brox. U-Net: Convolutional networks for biomedical image segmentation. *Proceedings of the International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 234–241, October 2015. Munich, Germany.

[19] M. C. Stamm, Min Wu, and K. J. R. Liu. Information forensics: An overview of the first decade. *IEEE Access*, 1:167–200, May 2013.

[20] S. K. Yarlagadda, D. Güera, P. Bestagini, F. M. Zhu, S. Tubaro, and E. J. Delp. Satellite image forgery detection and localization using GAN and one-class classifier. *CoRR*, abs/1802.04881, 2018.