Hiding Private Information in Images From AI

Hanyu Xue*, Bo Liu*, Ming Ding‡, Li Song§, Tianqing Zhu*

*School of Computer Science, University of Technology Sydney, Australia.
Email: hanyu.xue@student.uts.edu.au, {bo.liu, tianqing.zhu}@uts.edu.au
‡Data61, CSIRO, Australia. Email: ming.ding@data61.csiro.au.,
§Department of Electronic Engineering, Shanghai Jiao Tong University, Shanghai, China. Email: song_li@sjtu.edu.cn.

***Abstract*—Privacy protection attracts increasing concerns these days. People tend to believe that large social platforms will comply with the agreement to protect their privacy. However, photos uploaded by people are usually not treated to achieve privacy protection. For example, Facebook, the world's largest social platform, was found leaking photos of millions of users to commercial organizations for big data analytics. A common analytical tool used by these commercial organizations is the Deep Neural Network (DNN). Today's DNN can accurately identify people's appearance, body shape, hobbies and even more sensitive personal information, such as addresses, phone numbers, emails, bank cards and so on. To enable people to enjoy sharing photos without worrying about their privacy, we propose an algorithm that allows users to selectively protect their privacy while preserving the contextual information contained in images. The results show that the proposed algorithm can select and perturb private objects to be protected among multiple optional objects so that the DNN can only identify non-private objects in images.**

***Index Terms*—privacy, object detection, deep learning**

I. Introduction

In such an era of data sharing, many people would like to share their life photos and videos on social software with friends or strangers. For example, Every 60 seconds on Facebook 136,000 photos are uploaded [1]. However, people may not have noticed that these images and videos contain a large amount of private information [2], [3], [4] such as the faces, vehicle license plates, locations, email-addresses, etc. If such information is used by adversaries, it may have a detrimental effect on the users [3]. Meanwhile, the newly emerging deep learning techniques further increases the privacy risks for online photo sharing. Artificial intelligence (AI) aided by deep learning methods can automatically collect and detect private and sensitive information from social networks. For example, DNNs can automatically search meaningful information in images and exploit an outcome to perform targeted advertisements [5]. DNNs can even extract user's private information, such as fingerprints [6], addresses, family members [7], [8], etc. This brings more risks to personal privacy, while the traditional privacy-preserving method seems powerless when facing the large-scale deep learning tools. Therefore, the development of image privacy protection methods is in urgent need, especially when considering AI as the adversary.

Privacy protection for unstructured data such as image is much more complicated compared with that for structured data. Traditional image privacy protection research assumes humans as an adversary. "Blurring", "pixelation" and "mosaic" are the most commonly used methods. For example, Viola et al. [9] used a sliding window detector to identify and blur the license plates in Google Street View images. Researchers start to consider the case where AI acts as an adversary very recently. The fundamental idea is to generate a small but intentional worst-case disturbance to an original image, which misleads deep neural networks (DNNs) without causing a significant difference perceptible to human eyes. The perturbed image is called an "adversarial example" [10] and the specially generated noise is named adversarial perturbations (AP). A few papers have discussed the potential of AP in privacy protection in different applications including image classification [11] and face recognition [12]. In [13], the authors proposed a novel stealth algorithm, which makes all the objects invisible to DNNs in an image. These works cannot solve our problem thoroughly due to several reasons: First, there is generally multiple private objects in the images, especially for social network images. Second, the revision of the images should be as small as possible and limited to the private information, so as to preserve the utility of the images.

To overcome the above-mentioned problems, we proposed a framework for image private information protection in this paper. It consists of three major steps: i) defining the privacy information in images, ii) identifying the private objects and their positions in image, and iii) image privacy protection using adversarial noise. Specifically, for image privacy protection, we propose to add adversarial perturbations to the sensitive parts of the images so that the private information can be hidden while the rest parts of the images are still visible to AI.

In summary, the contributions of this paper are as follows:

- Developing an image privacy protection framework to hide private information from AI, while the applied privacy protection is imperceptible to human eyes.
- Proposing an adversarial perturbation-based image privacy protection scheme, such that it can hide multiple private objects in the image while having a minor impact on the non-private objects.

The rest of the paper is organized as follows. Section II discusses the system model and formulates the research problem. In Section III, an AP-based image privacy protection scheme is proposed. Section IV shows our experimental results. Finally,

the results are concluded in Section V.

II. System Model and Problem Formulation

In this section, we present the system model used in our paper and formulate the research problem.

A. System Model

As shown in Fig. 1, our proposed image privacy protection framework consists of three major parts: object detection, image privacy definition, image privacy protection.

1) Object Detection: the input image $\mathbf{X}$ will first pass the object detection module.

There are many existing frameworks for object detection, among which Faster R-CNN [14] is a widely used framework that has been cited frequently in this research area. Therefore, we adopt Faster R-CNN as our object detection module. As shown in Fig. 2, the Faster R-CNN detects the region containing objects by three submodules.

- Feature Extractor: a traditional convolutional neural network to perform the feature extraction.
- Region Proposal Network (RPN): RPN finds the object regions by scans the image using different size anchors (The area RPN scans) in a slide window fashion. The outputs of RPN include a series of anchors A_a, as well as pre-classifier result P_a, i.e.:

$$A_{rpn} = (A_a|P_a) = \left(\begin{array}{cccc|c} x_1 & y_1 & w_1 & h_1 & p_1 \\ x_2 & y_2 & w_2 & h_2 & p_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_\alpha & y_\alpha & w_\alpha & h_\alpha & p_\alpha, \end{array}\right) \quad (1)$$

  where x_i, y_i, w_i, h_i represent the up left corner x-coordinate, y-coordinate and width, height of anchors, respectively. i is the index of the anchor ($i = 1, 2, \ldots, \alpha$). $P_a = (p_1, p_2, \ldots, p_\alpha)^T$ denotes the probabilities of anchors being positive.
- Regions of Interest (ROI) Classifier: ROI classifier output contains the location and size of each proposed region, and the probability of anchors being a class (e.g. cat, dog, face):

$$A_{roi} = \left(\begin{array}{cccc|ccc} x_{a1} & y_{a1} & w_{a1} & h_{a1} & p_{11} & \ldots & p_{1m} \\ x_{a2} & y_{a2} & w_{a2} & h_{a2} & p_{21} & \ldots & p_{2m} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \ddots & \vdots \\ x_{an} & y_{an} & w_{an} & h_{an} & p_{n1} & \ldots & p_{nm} \end{array}\right), \quad (2)$$

  where n is the number of anchors that ROI proposed ($n \leq \alpha$). $x_{aj}, y_{aj}, w_{aj}, h_{aj}$ are the coordinate and size information of ROI proposed anchors. $p_{11}, \ldots, p_{nm}$ (noted as P_{ROI}) are the probability of n anchors belonging to m class respectively.

Finally, the output of the object detection module is represented as:

$$C(\mathbf{X}) = \left(\begin{array}{cccc|c} x_{a1} & y_{a1} & w_{a1} & h_{a1} & c_1 \\ x_{a2} & y_{a2} & w_{a2} & h_{a2} & c_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{an} & y_{an} & w_{an} & h_{an} & c_n \end{array}\right), \quad (3)$$

where

$$\forall j \in (1, n) : c_j = \begin{cases} \arg\max_i p_{ni}, & 1 \leq i \leq m \\ c_{bg}, & \forall p_{ni} \leq threshold \end{cases}$$

It worth noting that Faster-RCNN treats background as a class, i.e., c_{bg}. $threshold$ is used to deal with the unrecognizable area that may appear. If the probability of all classes is less than $threshold$, it is recognized as the background.

2) Image Privacy Definition: In this module, we first define what object in the image contains individuals' private information. According to the General Data Protection Regulation (GDPR) [15], anything that can be used as a personal identifier should be treated as private information. Therefore, we propose that private objects in images should include:

- Personal identity - license plate, phone number, address, etc.
- Biometrics - face, calendar data, fingerprints, retinal scans, photos, etc.
- Electronic records - cookies, IP locations, mobile device IDs, social network activity records

According to this definition, all classes in the object detection output are divided into two subsets: $\mathbf{C}_{private}$ is the set of private classes, and $\mathbf{C}_{non-private}$ includes non-private classes.

3) Image Privacy Protection: a small adversarial perturbation $\delta\mathbf{X}$ targeting on private objects is applied to generate the privacy-free image $\mathbf{X}^{pr}$, so that only non-private information can be detected when passing $\mathbf{X}^{pr}$ through an object detector, i.e.,

$$C(\mathbf{X}^{pr}) = \left(\begin{array}{cccc|c} x_{a1} & y_{a1} & w_{a1} & h_{a1} & c_1^{pr} \\ x_{a2} & y_{a2} & w_{a2} & h_{a2} & c_2^{pr} \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ x_{an} & y_{an} & w_{an} & h_{an} & c_n^{pr} \end{array}\right), \quad (4)$$

where $\forall c_j \in \mathbf{C}_{private} : c_j^{pr} = c_{bg}$.

B. Problem Formulation

Based on the above-described framework, our target is to fool the network by changing the class of the private objects to bg, while the non-private objects are recognized as their original classes. Meanwhile, the added noise δX should be small so that it is imperceptible for humans. Hence, the problem can be formulated as follows:

$$\arg\min_{\delta\mathbf{X}} \|\delta\mathbf{X}\|_2 \quad (5)$$

$$\text{s.t.: } \forall c_j \in \mathbf{C}_{private} : c_j^{pr} = c_{bg} \quad (6)$$

$$\forall c_j \in \mathbf{C}_{non-private} : c_j^{pr} = c_j \quad (7)$$

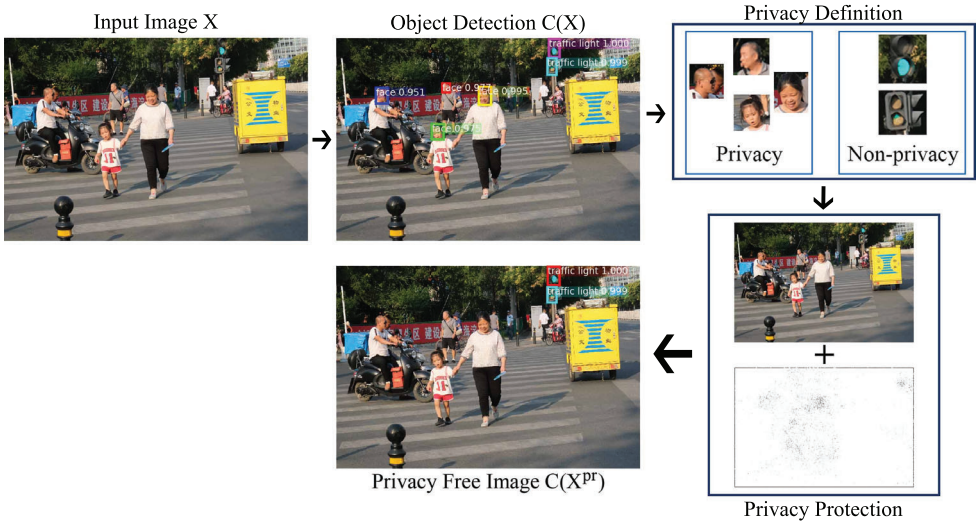

Fig. 1. Image privacy protection framework.

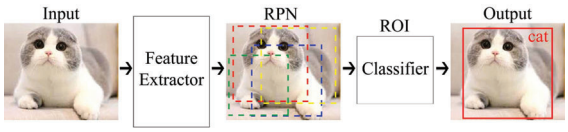

Fig. 2. The Faster R-CNN framework.

III. Adversarial Perturbation based Image Privacy Protection Algorithm

In order to solve the image privacy protection problem, we proposed an AP-based image privacy protection algorithm in this section, along with the metrics that can be used to evaluate the performance of the algorithm.

A. AP-based Image Privacy Protection

Fig. 3 gives the flow chart of our algorithm. The input image is sent to the object detector along with the generated noise and then the output objects are divided into three categories (Background, Non-privacy objects, Privacy objects). Initially, the added noise is 0 and the object detector will find all objects in the image. Next, we replace the label of the private objects with the background and then put it into the loss function to calculate the gradient. Then the noise is updated according to the gradient. Finally, a perturbed image is generated, in which all privacy objects are treated as background by the object detector.

The key part of the algorithm is to trick the classification loss ($\mathcal{L}_{cls}$) so as to mislead the object detector recognizing the privacy objects to background. We define our new loss function as shown in Eq. (8) to mislead the classifier so that it will reckon all private objects as background:

$$\mathcal{L}_{cls} = \frac{1}{n_a}\sum_i En(p_i, p_i^*) + \lambda \left\|\mathbf{X} - \mathbf{X}^{pr}\right\|_2, \tag{8}$$

where $p_i = [p_{i1}, \ldots, p_{im}]$ is the probability of the content of an anchor being recognized as each class. p_i^* is one-hot encoded ($p_i^* = [0, 0, \ldots, 1, \ldots, 0, 0]$), in which 1 appears in the position where we set the class as the correct class. p_i^* will be generated according to ground truth label if the object is non-private, while it will be changed to the background if the object is private. n_a is the number of anchors in the image so that the entropy will be averaged over all anchors. Next, we can use $\mathcal{L}_{cls}$ to generate the perturbation, using the fast gradient sign method (FGSM) [10].

Using the targeted FGSM, the perturbation can be calculated in the direction of the gradient:

$$\delta\mathbf{X} = -\epsilon \text{sign}(\nabla_{\mathbf{X}}\mathcal{L}_{cls}) = -\epsilon \text{sign}(\frac{\mathcal{L}_{cls}}{\partial \mathbf{X}}), \tag{9}$$

where ϵ is the step parameter that scales the noise. Therefore, the generated image will be:

$$\mathbf{X}^{pr} = \mathbf{X} + \delta\mathbf{X} = \mathbf{X} - \epsilon \text{sign}(\frac{\mathcal{L}_{cls}}{\partial \mathbf{X}}) \tag{10}$$

In practice, one step FGSM is usually not enough, so we can use an iterative version as shown in Alg. 1.

B. Evaluation Metrics

In order to measure the performance of our proposed methods, we introduce the following metrics from three different aspects:

1) Distortion metrics: Two distortion metrics are used to measure the amount of noises added to the original image.

- L_2 computes the Euclidean distance between original and perturbed examples, i.e., $L_2 = \left\|\mathbf{X}^{pr} - \mathbf{X}\right\|_2 = \left\|\delta\mathbf{X}\right\|_2$
- Average L_p Distortion ALD_p [16]: $ALD_p = \frac{\left\|\mathbf{X}^{pr} - \mathbf{X}\right\|_p}{\left\|\mathbf{X}\right\|_p}$. We use ALD_∞ to measure the maximum change in all dimensions of adversarial perturbations in the simulation.

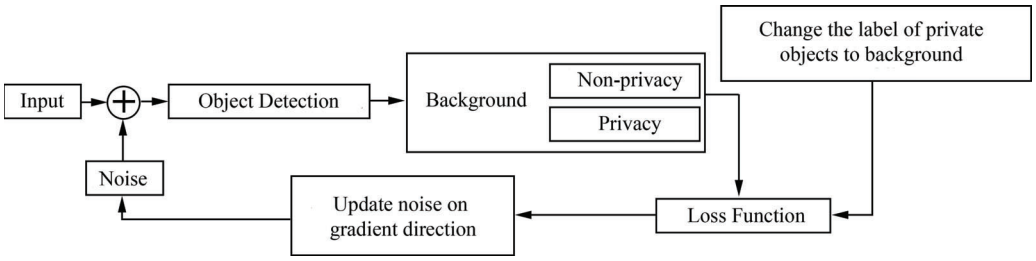

Fig. 3. Diagram of AP-based image privacy protection algorithm.

Algorithm 1: AP-based image privacy protection algorithm.

Parameters: Noise scalar ϵ.
Iteration number N.
Input: The original image $\mathbf{X}$.
Output: The released privacy-preserving image $\mathbf{X}^{pr}$.
Initialization: Overall noise $\delta\mathbf{X} = 0$, $\mathbf{X}_0^{pr} = \mathbf{X}$.
for $1 \leq n \leq N$ **do**
  $\delta\mathbf{X}_{n-1} = -\epsilon \text{sign}(\nabla_{\mathbf{X}} \mathcal{L}_{cls})$;
  $\delta\mathbf{X} = \delta\mathbf{X}_{n-1} + \delta\mathbf{X}$;
  $\mathbf{X}_n^{pr} = \delta\mathbf{X}_{n-1} + \mathbf{X}_{n-1}^{pr}$;
end
$\mathbf{X}^{pr} = \mathbf{X}_n^{pr}$.

2) Structural Similarity (SSIM): SSIM is a method used to measure the similarity between two digital images. Compared with the traditional image quality measurement methods, such as peak signal-to-noise ratio (PSNR) and mean squared error (MSE), SSIM can better match the human judgment of image quality [17][18]. It can be used to quantify the extent that the perturbation is invisible to human eyes.

3) Private Information Hiding Ratio: $R = \bar{r}_p + \lambda \bar{r}_a$, where $\bar{r}_p$ and $\bar{r}_a$ are the average hiding ratio or keeping ratio of privacy objects and non-privacy objects respectively, λ is set to 0.5 for better illustration. It is used to measure whether our method can hide the proper objects in images.

IV. Experiment and Discussions

In this section, we show our experimental results.

A. Experiment Settings

In our experiment, we use the images from the data set provided by Tribhuvanesh et al. [3]. The data set is originated from the VISPR data set. The authors selected images containing private information and pixel-annotated using 24 privacy attributes. In our experiment, we choose faces and license plates as privacy items. Hence, we filtered the images with these two annotations from the data set. And filter some non-private data from the original data set. Then, we added more street view images containing faces and license plates into the training data set for better performance.

Here we use Faster R-CNN as the object detector, Faster R-CNN requires that the input image shape is square ($1024 *$ 1024 is the suggested size). The original data set contains a large number of large size images (e.g. 7000x6000), so we make standardization on our training data set before training for better training performance. The model was trained on one GPU card GeForce GTX 2080Ti.

B. The Experiment Results

Fig. 4 shows an example of our proposed algorithm. The left column shows the detection result of the original image, the next two columns are the detection result after the proposed privacy treatment and the added perturbation, respectively. As can be seen in the figure, without privacy protection, all objects in the images can be detected by a standard Faster R-CNN. After adding the adversarial noise, the detector cannot detect the privacy objects, including faces and car plates while the non-sensitive features (e.g. traffic lights) can still be detected. The adversarial perturbation in the images are generated in range R_p ($R_p \in (-2, 2)$). In order to display the noise in image, we normalize the R_p to R_P ($R_P \in (0, 255)$). Observing the relative location of objects in the original image, the noise dots concentrate on the regions contain objects. By adding adversarial perturbation in such a small range (e.g.R_p), human eyes can hardly recognize the difference. But the object detector has been successfully fooled.

Now we compare the performance of our proposed algorithm with Blur and Mosaic. As shown in Fig. 5, the Blur and Mosaic's "thickness" has been carefully adjusted to just hide the sensitive information. But human eyes can easily notice those changes. Our method, while deceiving the detector from the private objects, greatly preserves the non-private information in the original image, so that naked eyes can hardly see the difference.

Next, we measure the effectiveness of our approach using the metrics mentioned in Section III.B. Table I shows the performance comparison measured by distortion metrics ($L2$ and ALD_p). The adversarial noise (AD Noise) are generated in range R_p ($R_p \in (2, 2)$). Blur and Mosaic noise thickness has been modified to barely hide the sensitive information from Object Detector. Compare with Blur, our method is 73.5% and 79.2% lower in the $L2$ and ALD_p average scores, respectively. Also, our method is superior to Mosaic in both $L2$ and ALD_p, i.e., our algorithm is 81.4% lower in $L2$ and 85.2% lower in ALD_p.

Table II presents the results measured by the SSIM metric.

Fig. 4. Illustration of AP-based image privacy protect algorithm.

Fig. 5. The detection results after privacy protection: (a) Image without Protection; (b) Blur; (c) Mosaic; (d) AP-based.

TABLE I
$L2$ AND ALD_p SCORE COMPARED WITH CLASSICAL METHODS

		Original	Blur	Mosaic	AD Noise
$L2$	1	0	4111	4153	765
	2	0	2008	3730	776
	3	0	3426	6168	778
	4	0	1980	2386	731
	Average	0	2881.2	4109.2	762.5
ALD_p (10^{-2})	1	0	5.55	6.27	0.97
	2	0	2.29	4.82	0.68
	3	0	4.59	6.67	0.69
	4	0	2.19	2.81	0.70
	Average	0	3.655	5.143	0.76

A higher score means a smaller distortion of the image. The performance of our method has increased by 192.5% compared to blur, which is an increase of 474.8% compared to Mosaic.

TABLE II
THE SSIM SCORE COMPARED WITH CLASSICAL METHODS

	Original	Blur	Mosaic	AD Noise
1	1	0.548	0.286	0.998
2	1	0.536	0.243	0.995
3	1	0.442	0.121	0.997
4	1	0.472	0.191	0.998
5	1	0.634	0.182	0.998
6	1	0.478	0.235	0.996
Average	1	0.518	0.210	0.997

From the above results, we can see that our method gains

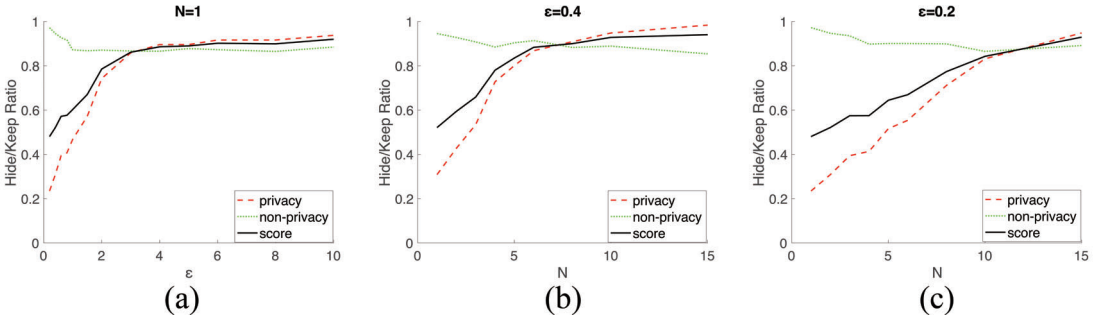

Fig. 6. The hide/keep ratio; (a) Fixed iteration number $N = 1$; (b) Fixed $\epsilon = 0.4$; (c) Fixed $\epsilon = 0.2$

performance improvement in balancing privacy protection and information preservation, compared with classical methods. Finally, we measure the privacy protection efficiency by running the test data set with different iteration numbers (N) and noise scalar (ϵ). The adversarial noise thickness is related to the iteration numbers (N) and noise scalar (ϵ). The noise range R_p is related to $N \times \epsilon$. So, we use noise range as an index to measure our adversarial noise thickness. As can be seen in Fig. 6, the private information hiding rate is proportional to the noise thickness, while the non-sensitive objects keeping ratio slightly decreases with the increase of thickness. Fig. 6.(a) gives the change in hiding ratio with an increase of ϵ. It shows that a very small amount of noise thickness: $R_p \in (-3, 3)$ (out of 255) is enough to achieve an over 90% high hiding ratio. Fig. 6.(b) and Fig. 6.(c) show that under the same thickness (R_p), a smaller ϵ achieves a relatively higher hiding ratio, but it needs more iterations.

V. Conclusion and Discussion

The recent advancement of artificial intelligence exacerbates the privacy concern, especially for images that contain a variety of personal information. In order to solve this problem, we proposed an image privacy protection framework against AI, using an AP-based privacy protection algorithm. Our results show that private objects in images can be well protected while non-private information is preserved by adding a small amount of noise. Therefore it can protect image privacy while preserving the image utility. Moreover, the noise added can hardly be detected by naked eyes, which lens more practical value of the proposed algorithm in real-life employment.

VI. Acknowledgments

This work is supported by the Australian Government through the Australian Research Council's Linkage Projects funding scheme (LP180101150).

References

[1] Zephoria, "Top 20 Facebook Statistics - Updated June 2019." [Online]. Available: https://zephoria.com/top-15-valuable-facebook-statistics/

[2] T. Orekondy, B. Schiele, and M. Fritz, "Towards a Visual Privacy Advisor: Understanding and Predicting Privacy Risks in Images," in *IEEE International Conference on Computer Vision (ICCV)*, 2017.

[3] T. Orekondy, M. Fritz, and B. Schiele, "Connecting Pixels to Privacy and Utility: Automatic Redaction of Private Information in Images," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

[4] S. J. Oh, M. Fritz, and B. Schiele, "Adversarial Image Perturbation for Privacy Protection – A Game Theory Perspective," in *International Conference on Computer Vision (ICCV)*, 2017.

[5] W. Meng, X. Xing, A. Sheth, U. Weinsberg, and W. Lee, "Your online interests: Pwned! a pollution attack against targeted advertising," in *Proceedings of the 2014 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2014, pp. 129–140.

[6] Phys.org, "Japan researchers warn of fingerprint theft from 'peace' sign," jan 2017. [Online]. Available: https://phys.org/news/2017-01-japan-fingerprint-theft-peace.html

[7] E. Tseng, "Computer-vision-assisted location accuracy augmentation," Apr. 5 2016, uS Patent 9,305,024.

[8] W.-H. Li, F.-T. Hong, and W.-S. Zheng, "Learning to learn relation for important people detection in still images," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[9] P. Viola and M. J. Jones, "Robust real-time face detection," *International journal of computer vision*, vol. 57, no. 2, pp. 137–154, 2004.

[10] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," *arXiv preprint arXiv:1412.6572*, 2014.

[11] B. Liu, M. Ding, T. Zhu, Y. Xiang, and W. Zhou, "Adversaries or allies? privacy and deep learning in big data era," *Concurrency and Computation: Practice and Experience*, p. e5102.

[12] B. Liu, J. Xiong, Y. Wu, M. Ding, and C. M. Wu, "Protecting multimedia privacy from both humansand ai," in *in Proc. IEEE International Symposium on Broadband Multimedia Systems and Broadcasting*, 2019.

[13] Y. Liu, W. Zhang, and N. Yu, "Protecting privacy in shared photos via adversarial examples based stealth," *Security and Communication Networks*, vol. 2017, 2017.

[14] S. Ren, K. He, R. B. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," *CoRR*, vol. abs/1506.01497, 2015. [Online]. Available: http://arxiv.org/abs/1506.01497

[15] *Presidency of the Council: "Compromise text. Several partial general approaches have been instrumental in converging views in Council on the proposal for a General Data Protection Regulation in its entirety. The text on the Regulation which the Presi.* [Online]. Available: http://data.consilium.europa.eu/doc/document/ST-9565-2015-INIT/en/pdf

[16] X. Ling, S. Ji, J. Zou, J. Wang, C. Wu, B. Li, and T. Wang, "Deepsec: A uniform platform for security analysis of deep learning model."

[17] H. R. Sheikh, M. F. Sabir, and A. C. Bovik, "A statistical evaluation of recent full reference image quality assessment algorithms," *IEEE Transactions on image processing*, vol. 15, no. 11, pp. 3440–3451, 2006.

[18] Z. Wang and A. C. Bovik, "Mean squared error: Love it or leave it? a new look at signal fidelity measures," *IEEE signal processing magazine*, vol. 26, no. 1, pp. 98–117, 2009.