EFFICIENT PRIVACY PRESERVING VIOLA-JONES TYPE OBJECT DETECTION VIA RANDOM BASE IMAGE REPRESENTATION

Xin Jin[1], *Peng Yuan*[1,2], *Xiaodong Li*[1], *Chenggen Song*[1], *Shiming Ge*[3,*], *Geng Zhao*[1], *Yingya Chen*[1]

[1]Beijing Electronic Science and Technology Institute, Beijing 100070, China
[2]Xidian University, Xi'an 710071, China
[3]Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100095, China

ABSTRACT

A cloud server spent a lot of time, energy and money to train a Viola-Jones type object detector [1] with high accuracy. Clients can upload their photos to the cloud server to find objects. However, the client does not want the leakage of the content of his/her photos. In the meanwhile, the cloud server is also reluctant to leak any parameters of the trained object detectors. 10 years ago, Avidan & Butman introduced *Blind Vision*, which is a method for securely evaluating a Viola-Jones type object detector. Blind Vision uses standard cryptographic tools and is painfully slow to compute, taking a couple of hours to scan a single image. The purpose of this work is to explore an efficient method that can speed up the process. We propose the *Random Base Image (RBI) Representation*. The original image is divided into random base images. Only the base images are submitted randomly to the cloud server. Thus, the content of the image can not be leaked. In the meanwhile, a random vector and the secure Millionaire protocol are leveraged to protect the parameters of the trained object detector. The RBI makes the integral-image enable again for the great acceleration. The experimental results reveal that our method can retain the detection accuracy of that of the plain vision algorithm and is significantly faster than the traditional blind vision, with only a very low probability of the information leakage theoretically.

***Index Terms*—** Blind Vision, Random Base Image, Privacy Preserving, Object Detection

1. INTRODUCTION

Recently, widespread smart phones with cameras enable people to shot images and videos nearly anytime and anywhere. Millions of surveillance cameras including the driving recorders captures images and videos every second. All these conveniences devices are producing the large-scale visual media data, which is considered as the *biggest big data*.

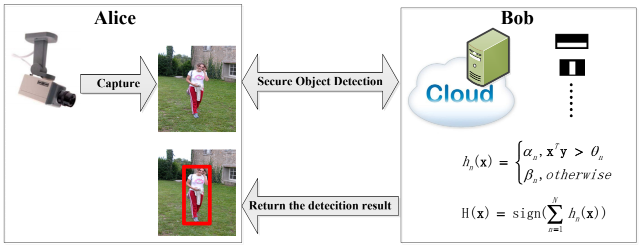
Alice
Capture
Secure Object Detection
Bob
Cloud
$h_n(\mathbf{x}) = \begin{cases} \alpha_n, \mathbf{x}^T\mathbf{y} > \theta_n \\ \beta_n, otherwise \end{cases}$
Return the detecition result
$\mathrm{H}(\mathbf{x}) = \mathrm{sign}(\sum_{n=1}^{N} h_n(\mathbf{x}))$

Fig. 1. Alice would like to detect objects in a collection of sensitive surveillance images she own. Bob has an object detection algorithm that he is willing to let Alice use, for a fee, as long as she learns nothing about his detector. Alice is willing to use Bob's detector provided that he will learn nothing about her images, not even the result of the object detection operation [2].

Due to the limited storage space of these terminal devices, large-scale visual media data is being uploaded and stored in the cloud servers. Not only the storage, but also the processing of large-scale visual media data are being outsourced to the cloud servers.

The cloud servers have some strong algorithms such as face/object detection, face/object recognition, intelligent video surveillance. Nowadays, people can easily find all the faces in their photos stored in the cloud servers using the powerful face detection algorithms maintained by the cloud servers. However, the cloud servers are always third party entities. Thus the privacy of the users' visual media data may be leaked to the public or unauthorized parties.

In the meanwhile, the powerful cloud services for visual media analysis and processing need a lot of money, data and time from the cloud server producers. The cloud servers are also reluctant to leak any parameters of the trained models or some protected details of their algorithms with copyrights.

Thus, the privacy of both the content of the visual media from the clients and the parameters of the vision algorithms from the cloud servers should be protected. 10 years ago, Avi-

This work is partially supported by the National Natural Science Foundation of China (Grant NO.61402021, 61402023, 61640216), the Science and Technology Project of the State Archives Administrator (Grant NO. 2015-B-10), the open funding project of State Key Laboratory of Virtual Reality Technology and Systems, Beihang University (Grant NO. BUAA-VR-16KF-09), and the Fundamental Research Funds for the Central Universities (NO. 2016LG03, 2016LG04). *Corresponding author: geshiming@iie.ac.cn

978-1-5090-6067-2/17/$31.00 ©2017 IEEE

dan & Butman introduced *Blind Vision* [2], which is a method for securely evaluating a Viola-Jones type face detector. Blind Vision uses standard cryptographic tools and is painfully slow to compute, taking a couple of hours to scan a single image.

After that, rich literatures have been proposed in this field. The cryptographic tools such as secret sharing (SS) [3], security multi-party computation (SMC) [4], homomorphic encryption (HE) [5], garbed circuit (GC) [6], Chaotic System (CS) [7] are heavily used. Plenty of computer vision applications have been modified to the privacy preserving or secure versions such as private face detection [8], face recognition [4], content based image retrieval [9], visual media search on public datasets [10], intelligent video surveillance [3, 5, 6, 7].

However, most of these work rely heavily on cryptographic tools, which are painfully slow to compute or need bit by bit interaction between the clients and the cloud servers. In this paper, we revisit the *Blind Vision* [2] and attempt to make the blind vision towards *cryptographic-free*, without losing the security properties. We use randomness and only a little cryptographic operations to protect the visual media data of the clients and the parameters of the trained models in the cloud servers.

A novel image representation called *Random Base Image(RBI)* representation is proposed. In this work, we also investigate the object detection in the cloud. We apply our RBI to the famous Viola and Jones object detection method and propose a novel blind object detection method. We separate an image into random base images. The weight of each base image is only known by the client. The base images are sent randomly to the cloud server. The cloud server cannot recover anything from the random base images. A random vector and the secure Millionaire protocol [2] are leveraged to protect the parameters of the trained object detector. The RBI makes the integral-image enable again for the great acceleration. The experimental results reveal that our method is significantly faster than the traditional blind vision, with only a very low probability of the information leakage theoretically.

2. SECURE OBJECT DETECTION

In this section we develop a secure object detector with the random base image representation.

2.1. Notations

Our scenario and the notations are the same as that of traditional Blind Vision [2], as show in Figure 1. Denote some L dimensions finite field F that is large enough to represent all the intermediate results. Denote by X the image that Alice owns. A particular detection window within the image X will be denoted by $x \in F^L$ and x will be treated in vector form. Bob owns a strong classifier of the form

$$H(\mathbf{x}) = \mathrm{sign}(\sum_{n=1}^{N} h_n(\mathbf{x})), \tag{1}$$

where $h_n(\mathbf{x})$ is a threshold function of the form

$$h_n(\mathbf{x}) = \begin{cases} \alpha_n & \mathbf{x}^T\mathbf{y_n} > \theta_n \\ \beta_n & \text{otherwise,} \end{cases} \tag{2}$$

and $y_n \in F^L$ is the hyperplane of the threshold function $h_n(\mathbf{x})$. The parameters $\alpha_n \in F, \beta_n \in F$ and $\theta_n \in F$ of $h_n(\mathbf{x})$ are determined during training; N is the number of weak classifiers used.

2.2. The Random Base Image Representation

The core idea of our RBI is to separate the original image into some random base images with fixed weights. The original image can be recovered by all the base images. The sparse representation can be considered as the one has such ability. However, they need another image dataset for learning the base images. Further more, there could be reconstruction error. Thus, we fix the weights and randomize the base images themselves.

The detection window $\mathbf{x}$ can be represented as:

$$\mathbf{x} = \sum_{i=0}^{M-1} w_i \mathbf{B}_i, \tag{3}$$

where $\mathbf{B}_i$ is the base image with weight w_i. As is shown in Figure 2, each base image has a fixed weight. The base image itself is randomly determined. The number of the base image is set to $M = 256$. Thus, each base image can be a binary image, which is easy for network transfer and fast to compute. In addition, there are $256!$ permutation of the base image which is not easy to guess. The process of the RBI generation is described in Algorithm 1.

2.3. Secure Object Detection with RBI

2.3.1. Secure Object Classifier Protocol

The core of our method is the secure object classifier protocol as is described in Algorithm 2 and Figure 3. For secure object detection, Alice first divides the test image $\mathbf{X}$ into Q detection windows $\{\mathbf{x}_1, \mathbf{x}_2, ...\mathbf{x}_Q\}$. Then the detection windows are randomly sent to Bob as the inputs of the secure face classifier protocol one by one. Using the Algorithm 2, Alice and Bob know which detection windows are the target objects. Because the detection windows are randomly sent to Bob, only Alice learns the location of all the detected faces

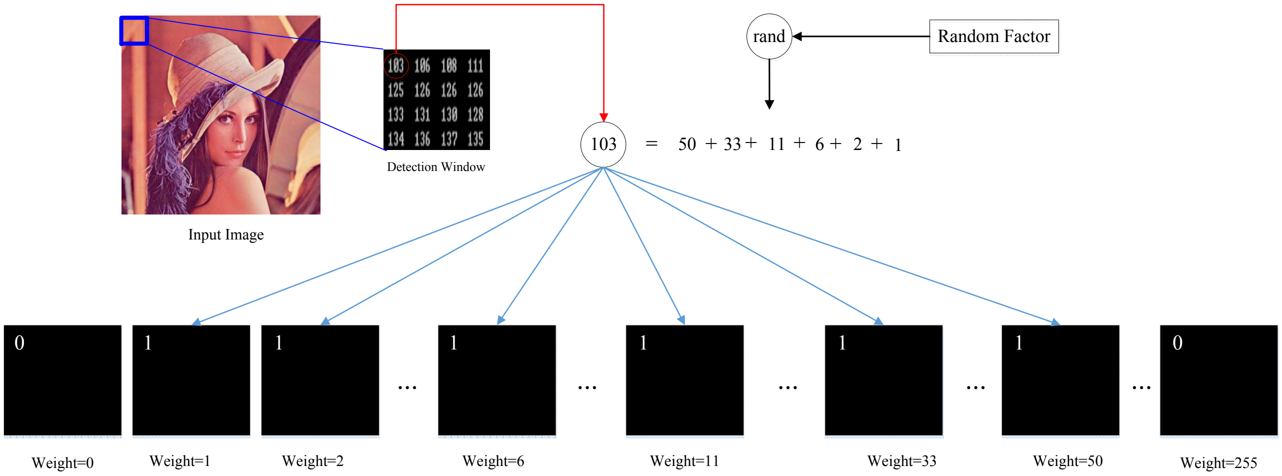

Fig. 2. The proposed Random Base Image Representation. For each single pixel p_x in the detection window x ($p_x \in [0, 255]$), we randomly select S numbers $p_x^j, j = 1, 2, ..., S, p_x^j \in [0, p_x]$ to represent $p_x = \sum_{j=1}^{S} p_x^j$. We fix the number of the base image to $M = 256$ and set each weight $w_i = i$. The initial value of each pixel in each base image is set to 0. Then we set the corresponding pixel in each of the ${p_x^j}^{th}$ base image B_i to 1: $p_{B_i} = 1, i = p_x^j$.

```
Algorithm 1 Random Base Image Factorization
Input:
    The detection window $\mathbf{x}$ from the client image $\mathbf{X}$
Output:
    $M$ random binary base images
    $\mathbf{B} = \{\mathbf{B}_0, \mathbf{B}_1, ..., \mathbf{B}_{M-1}\}$
Alice creates $M$ binary images with all the pixels initialized to $0$. Each binary image has the same size as that of the detection window $\mathbf{x}$.
The weight of each binary image is set to $w_i = i, i = 0, 1, ..., M-1$.
Alice sets a index $j = 1$. For each pixel $p_x$ in $x$, Alice repeat the following 3 steps until $p_x = 0$.
    (1) Alice generates a random number $p_x^j \in [0, p_x]$.
    (2) Set the corresponding pixel in the ${p_x^j}^{th}$ base image $B_i$ to 1: $p_{B_i} = 1, i = p_x^j$.
    (3) $p_x = p_x - p_x^j, j = j + 1$.
return $\mathbf{B} = \{\mathbf{B}_0, \mathbf{B}_1, ..., \mathbf{B}_{M-1}\}$.
```

```
Algorithm 2 Secure Object Classifier Algorithm with RBI
Input:
    (1) Alice has input detection window $\mathbf{x} \in F^L$
    (2) Bob has a strong classifier of the form $H(\mathbf{x}) = \text{sign}(\sum_{n=0}^{N-1} h_n(\mathbf{x}))$
Output:
    (1) Alice has the result $H(\mathbf{x})$ and nothing else
    (2) Bob learns nothing about the detection window $\mathbf{x}$
```

in the original image. Bob does not learn the contents including where the faces are in the image of Alice. Alice learns nothing about the parameters of the face detector of Bob.

The body of Algorithm 2 is described as follows:

- (1): Alice factorizes the detection window x into M random base images $\mathbf{B} = \{\mathbf{B}_0, \mathbf{B}_1, ..., \mathbf{B}_{M-1}\}$ with weight $w = \{w_0, w_1, ..., w_{M-1}\} = \{0, 1, ..., M-1\}$ through Algorithm 1.
- (2): Alice randomly shuffles the weight w to w'. The random base images $\mathbf{B}$ are permuted with the same order of that of $w' = \{w'_0, w'_1, ..., w'_{M-1}\}$ to $\mathbf{B}' = \{\mathbf{B}'_0, \mathbf{B}'_1, ..., \mathbf{B}'_{M-1}\}$, which is sent to Bob.
- (3): In one cascade, Bob has N weak classifiers with parameter vectors $\mathbf{y} = \{\mathbf{y}_0, \mathbf{y}_1, ...\mathbf{y}_{N-1}\}$. Bob randomly add K fake weak classifiers and set their parameters α and β to zero. Bob randomly shuffles the $N + K$ true and fake weak classifiers to form $\mathbf{y}' = \{\mathbf{y}'_0, \mathbf{y}'_1, ...\mathbf{y}'_{N+K-1}\}$. Then, Bob generates $N + K$ random positive numbers $s = \{s_0, s_2, ..., s_{N+K-1}\}$. For each parameter vector $\mathbf{y}'_n \in \mathbf{y}'$. Bob and Alice repeat the following 3 steps.
  - (3.1): Bob computes the feature responses for all the base image $\mathbf{B}'_m$ in $\mathbf{B}'$ by $F_m(n) = \mathbf{B}'^T_m \mathbf{y}'_n, m = 0, 1, ..., M-1$. All the M responses of base images $\mathbf{B}'$ on each parameter vector $\mathbf{y}'_n$ are sent back to Alice.
  - (3.2): Alice computes the feature responses of the detection window $\mathbf{x}$ by $F(n) = \sum_{m=0}^{M-1} F_m(n) w'_m$.

  - (3.3): Alice and Bob use the secure Millionaire protocol [2] to determine which number is larger: $F(n)$ or θ_n. Bob send $\alpha_n + s_n$ or $\beta_n + s_n$ to Alice. Alice store it as c_n.
- (4): Alice and Bob use the secure Millionaire protocol [2] to determine which number is larger: $\sum_{n=1}^{N+K} c_n$ or $\sum_{n=1}^{N+K} s_n$. If Alice has a larger number then x is positively classified, otherwise x is negatively classified.

2.3.2. Security

The protocol protects the security of both parties. The protocol protects the contents of the image from Alice and the parameters of the face detector from Bob. We analyse the security of Algorithm 2 in the following paragraph.

- From Alice to Bob
  - In step 2, Alice send randomly shuffled base images to Bob. Bob only knows the randomly generated base images and do not know the weight of each base image. The probability of guessing out the right permutation is $1/M!$. Even Bob guesses out the right permutation, he does not know the weight of each base image. Thus, it is almost impossible for Bob to recover the detection window of Alice.
  - In the 3th sub-step of step 3 and the step 4. Alice and Bob engage in secure Millionaire protocol [2]. so Bob can learn nothing about Alices data.
- From Bob to Alice
  - In the 1st sub-step of step 3, Alice can not learn the number of the weak classifiers N or the true filters from the received feature responses. The true filters are obfuscated by the fake filters.
  - In the 3rd sub-step of step 3, Alice and Bob engage in a secure Millionaire protocol so Alice only learns if $F(n) > \theta_n$. She can not learn anything about the parameter θ_n. Moreover, at the end of the Millionaire protocol Alice learns either $\alpha_n + s_n$ or $\beta_n + s_n$. In both cases, the real parameter (α_n or β_n) is obfuscated by the random number s_n.
  - In step 4, Alice and Bob use the secure Millionaire protocol to determine which number is larger: $\sum_{n=1}^{N} c_n$ or $\sum_{n=1}^{N} s_n$. If Alice has a larger number then x is positively classified, otherwise x is negatively classified.
- Multiple Cloud Servers
  - The M random base images can be also sent to multiple cloud server with the same object detector to increase security.

2.3.3. Complexity and Efficiency

The complexity of the protocol is $O(M(N+K)L)$, where M is the number of the base images. N and K are the numbers of the true and fake weak classifiers, respectively. L is the dimensionality of the detection window x.

Unlike the traditional Blind Vision [2], in which the OT operation is used extensively, the proposed method only use OT operation to compare 2 numbers. In the secure dot-product protocol, each pixel of each detection window uses a OT_1^{256} operation, which needs 1 RSA encryption and 256 RSA decryption with 128-bit long encryption keys. We leverage our M random images, whose computation is much faster than the RSA encryption and decryption operations.

In addition, in the traditional Blind Vision [2], they convert the integral-image representation to regular dot-product operation, a step that clearly slows down their implementation as they no longer take advantage of the integral-image representation. In our RBI based protocol, the integral-image representation is enabled again, which accelerates the computation obviously.

3. EXPERIMENTS

We convert the Viola-Jones type object detector [1, 11] to our secure object detector. We implement our RBI based object detector using Microsoft Visual Studio 2012 and OpenCV 2.4.3/10. [1] package for computer vision in a 64 bits Windows 7 operating system. The hardware configuration is 3.5GHz AMD A10 Pro-7800 R7 CPU with 12 compute Cores and 8GB Memory.

The face detector is from the OpenCV 2.4.3 package and consists of a cascade of 22 rejectors, where each rejector is of the form presented in Eq. 1. The first rejector consists of 3 weak classifiers. The most complicated rejector consists of 213 weak classifiers. There is a total of 2135 weak classifiers. We also test the nose detector, the eye detector and the full body detector from OpenCV 2.4.10.

3.1. The Detection Accuracy

We test our secure face detector in 3 face detection datasets: The Face Detection Dataset (FDDB) [12], The Face96 Dataset [13], and The FEI Face Database [14].

We randomly select 100 face images from each of the 3 datasets. The detection accuracy (88.46%) of our secure face detector is the same as that of the OpenCV 2.4.3 face detector (88.46%).

The nose and the eye detectors are tested on the FDDB dataset [12]. The full body detector is tested on the INRIA Person dataset [15]. We randomly select 100 images from each of the 2 datasets. The detection accuracy of our secure

[1] http://opencv.org/

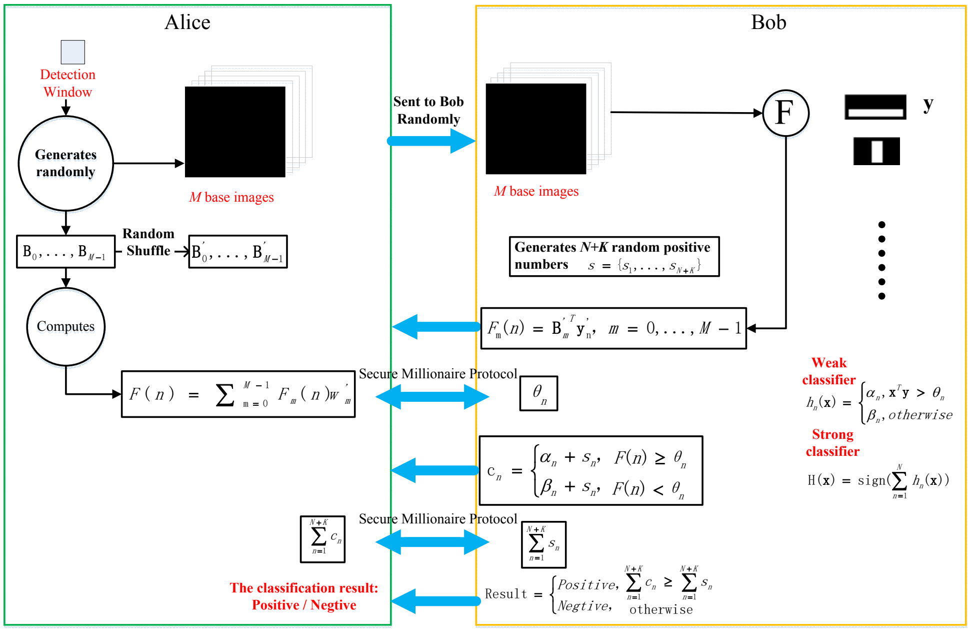
Alice
Detection Window
Generates randomly
M base images
$B_0, \ldots, B_{M-1}$
Random Shuffle
$B'_0, \ldots, B'_{M-1}$
Computes
$F(n) = \sum_{m=0}^{M-1} F_m(n) w'_m$
$\sum_{n=1}^{N+K} c_n$
The classification result: Positive / Negtive
Sent to Bob Randomly
Secure Millionaire Protocol
Secure Millionaire Protocol
Bob
M base images
F
y
Generates N+K random positive numbers $s = \{s_1, \ldots, s_{N+K}\}$
$F_m(n) = B'^{T}_m y'_n, \; m = 0, \ldots, M-1$
θ_n
$c_n = \begin{cases} \alpha_n + s_n, & F(n) \geq \theta_n \\ \beta_n + s_n, & F(n) < \theta_n \end{cases}$
$\sum_{n=1}^{N+K} s_n$
$\text{Result} = \begin{cases} Positive, & \sum_{n=1}^{N+K} c_n \geq \sum_{n=1}^{N+K} s_n \\ Negtive, & \text{otherwise} \end{cases}$
Weak classifier
$h_n(\mathbf{x}) = \begin{cases} \alpha_n, & \mathbf{x}^T \mathbf{y} > \theta_n \\ \beta_n, & otherwise \end{cases}$
Strong classifier
$H(\mathbf{x}) = \text{sign}(\sum_{n=1}^{N} h_n(\mathbf{x}))$

Fig. 3. The proposed secure object classifier.

object detectors is the same as that of the OpenCV 2.4.10 nose, eye and full body detectors.

3.2. Comparison with Other Methods

We compare our method with the Viola-Jones method implemented by the OpenCV package and the method of the traditional Blind Vision [2]. 50 test images with size of 100 × 100 are randomly selected from each of the 3 datasets. The average running time is shown in Table 1. All the methods are running in client and server mode. For the Viola-Jones, Alice send the original image to Bob. Then, Bob runs the Viola-Jones method and return the detected windows to Alice. Our method is slower than the Viola-Jones method, which is running on plain images without protecting any privacy. According to the traditional Blind Vision method [2], the time-consuming OT operation is heavily used and the integral-image representation is disabled. Thus, they have to take a couple of hours to scan a single image, which is painfully slow. Although in our method, the only information that Bob learns is that how many faces are in the image of Alice, our cryptographic-free method is significantly faster than the previous work towards practical usage of blind vision applications.

In addition, we compare our method with the Viola-Jones method implemented by the OpenCV package and the method of the traditional Blind Vision [2]. 50 test images with size of 100 × 100 are randomly selected from each of the FDDB and the INRIA Person datasets. The average running times are shown in the last 3 rows of Table 1. All the methods are running in client and server mode.

4. CONCLUSIONS AND DISCUSSIONS

We propose a novel random base image representation (RBI) for efficient object detection applications. The traditional blind vision method applies secure multi-party techniques to vision algorithm. Their method reveals no information to either party at the expanse of heavy computation load. Our method is an attempt towards cryptographic-free. Alice learns nothing about the parameters of the face detector of Bob. Bob does not know the contents of the image of Alice. The only information may be leaked is that Bob have a probability $1/M!$ to guess out the right permutation of the base images. This is just a theoretical event. Even Bob guesses out the right permutation, he does not know the weight of each base image. Thus it is almost impossible for Bob to learn the information of the detection window of Alice. Because the heaviest cost of OT operation in the secure dot-production of [2] is avoided by our RBI based dot-production, the Millionaire version protocol of ours need much less time than the traditional blind vision protocol does.

There are several extensions to this work. First is the need to accelerate the secure blind vision to practical use, i.e. to reduce the time cost to near that of the vision algorithm without security consideration. Second is to make both the training and the test blind. This will make the client users to upload more visual data to the cloud without worrying about the privacy leakage.

Dataset	**Our**	**Our + Comm. Delays**	**VJ** [11]	**VJ** [11] **+ Comm. Delays**	**Blind Vision** [2]
FDDB-face	143.852s	380.992s	0.380s	0.843s	A couple of hours [8]
Face96	173.471s	477.635s	0.358s	0.809s	
FEI	152.701s	414.522s	0.363s	0.827s	
FDDB-nose	113.398s	294.845s	0.372s	0.836s	A couple of hours [8]
FDDB-eye	85.754s	240.111s	0.496s	0.912s	
INRIA Person	80.204s	224.571s	0.333s	0.771s	

Table 1. Average running time comparison with the Viola-Jones method [11] and the Blind Vision method [2] on the FDDB [12], the Face96 [13] and the FEI [14] datasets. In the second and fourth columns, we simulate Alice and Bob on one PC without communication delays. The third and fifth columns report the time costs in a private cloud environment.

5. REFERENCES

[1] Paul A. Viola and Michael J. Jones, "Robust real-time face detection," in *IEEE 8th International Conference On Computer Vision ICCV 2011, Vancouver, British Columbia, Canada, July 7-14, 2001*, 2001, p. 747.

[2] Shai Avidan and Moshe Butman, "Blind vision," in *Computer Vision - ECCV 2006, 9th European Conference on Computer Vision, Graz, Austria, May 7-13, 2006, Proceedings, Part III*, 2006, pp. 1–13.

[3] Maneesh Upmanyu, Anoop M. Namboodiri, Kannan Srinathan, and C. V. Jawahar, "Efficient privacy preserving video surveillance," in *IEEE 12th International Conference on Computer Vision, ICCV, Kyoto, Japan, September 27 - October 4*, 2009, pp. 1639–1646.

[4] Margarita Osadchy, Benny Pinkas, Ayman Jarrous, and Boaz Moskovich, "Scifi - A system for secure face identification," in *31st IEEE Symposium on Security and Privacy, S&P 2010, 16-19 May 2010, Berleley/Oakland, California, USA*, 2010, pp. 239–254.

[5] Hosik Sohn, Konstantinos N. Plataniotis, and Yong Man Ro, "Privacy-preserving watch list screening in video surveillance system," in *Advances in Multimedia Information Processing - PCM 2010 - 11th Pacific Rim Conference on Multimedia, Shanghai, China, September 21-24, 2010, Proceedings, Part I*, 2010, pp. 622–632.

[6] Chun-Te Chu, Jaeyeon Jung, Zhicheng Liu, and Ratul Mahajan, "strack: Secure tracking in community surveillance," in *Proceedings of the ACM International Conference on Multimedia, MM'14, Orlando, FL, USA, November 03 - 07, 2014*, 2014, pp. 837–840.

[7] Xin Jin, Kui Guo, Chenggen Song, Xiaodong Li, and et al., "Private video foreground extraction through chaotic mapping based encryption in the cloud," in *MultiMedia Modeling - 22nd International Conference, MMM 2016, Miami, FL, USA, January 4-6, 2016, Proceedings, Part I*, 2016, pp. 562–573.

[8] Shai Avidan and Moshe Butman, "Efficient methods for privacy preserving face detection," in *Advances in Neural Information Processing Systems 19, Proceedings of the Twentieth Annual Conference on Neural Information Processing Systems, Vancouver, British Columbia, Canada, December 4-7, 2006*, 2006, pp. 57–64.

[9] Jagarlamudi Shashank, Palivela Kowshik, Kannan Srinathan, and C. V. Jawahar, "Private content based image retrieval," in *2008 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2008), 24-26 June 2008, Anchorage, Alaska, USA*, 2008.

[10] Giulia C. Fanti, Matthieu Finiasz, and Kannan Ramchandran, "One-way private media search on public databases: The role of signal processing," *IEEE Signal Process. Mag.*, vol. 30, no. 2, pp. 53–61, 2013.

[11] Paul A. Viola and Michael J. Jones, "Robust real-time face detection," *International Journal of Computer Vision*, vol. 57, no. 2, pp. 137–154, 2004.

[12] Vidit Jain and Erik Learned-Miller, "Fddb: A benchmark for face detection in unconstrained settings," Tech. Rep. UM-CS-2010-009, University of Massachusetts, Amherst, 2010.

[13] Libor Spacek, "Collection of facial images: Faces96," `http://cswww.essex.ac.uk/mv/allfaces/faces96.html`.

[14] Carlos E. Thomaz and Gilson Antonio Giraldi, "A new ranking method for principal components analysis and its application to face image analysis," *Image Vision Comput.*, vol. 28, no. 6, pp. 902–913, 2010.

[15] Navneet Dalal and Bill Triggs, "Histograms of oriented gradients for human detection," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2005), 20-26 June 2005, San Diego, CA, USA*, 2005, pp. 886–893.