IEEE Open Journal of the Computer Society

Received 5 October 2022; accepted 24 October 2022. Date of publication 1 November 2022;
date of current version 30 November 2022. The review of this article was arranged by Associate Editor Celimuge Wu.

Digital Object Identifier 10.1109/OJCS.2022.3218559

Efficient Video Privacy Protection Against Malicious Face Recognition Models

ENTING GUO [1], PENG LI [1] (Senior Member, IEEE), SHUI YU [2] (Senior Member, IEEE), AND HAO WANG [3] (Senior Member, IEEE)

[1]School of Computer Science and Engineering, University of Aizu, Aizuwakamatsu 965-8580, Japan
[2]School of Computer Science, University of Technology Sydney, Ultimo, NSW 2007, Australia
[3]Department of Computer Science, Norwegian University of Science and Technology, 7034 Trondheim, Norway

CORRESPONDING AUTHOR: PENG LI (e-mail: pengli@u-aizu.ac.jp).

This work was supported in part by Chinese Scholarship Council (CSC).

ABSTRACT The proliferation of powerful facial recognition systems poses a serious threat to user privacy. Attackers could train highly accurate facial recognition models using public data on social platforms. Therefore, recent works have proposed image pre-processing techniques to protect user privacy. Without affecting people's normal viewing, these techniques add special noises into images, so that it would be difficult for attackers to train models with high accuracy. However, existing protection techniques are mainly designed for image data protection, and they cannot be directly applied for video data because of high computational overhead. In this paper, we propose an efficient protection method for video privacy that exploits unique features of video protection to eliminate computation redundancy for computational acceleration. The evaluation results under various benchmarks demonstrate that our method significantly outperforms the traditional methods by reducing computation overhead by 35.5%.

INDEX TERMS Computation reuse, deep learning, video privacy.

I. INTRODUCTION

Deep Learning (DL) shows great promises for facial recognition [1], [2], which enables various intelligent services (e.g., medical records, smart communication) while posing a threat to personal privacy. With the popularity of social networks, people share their images or videos on Twitter or Facebook [3], [4], [5]. By using these data available on social platforms, attackers with modest resources can build highly accurate facial recognition models without people's awareness [5], [6], [7]. Therefore, it is important to protect user privacy hidden in public images and videos from unauthorized facial recognition trackers.

Some methods have been proposed to avoid facial recognition by deforming images, but they degrades user experiences [5], [8], [9]. Others require users to wear specific clothing with patterns that interfere with the recognition model [10], [11]. Some protection methods [12], [13] need the information of attack models to generate protection data. As one of the state-of-the-art privacy protection techniques, Fawkes [14], [15] considers both visual effects and privacy for images. Specifically, the faces in images are masked by a special kind of matrices, called cloaks. These cloaks are designed in a sophisticated way, so that people cannot distinguish masked images and original ones with human vision capability. Meanwhile, even though these masked images are fed to malicious models, the model training cannot converge to high accuracy. Despite the promises of Fawkes for images, it does not work well for video protection because of its high computational overhead. Masking a single face takes about 400 seconds on a mid-range GPU, and a one-minute video with 60 frames per second needs 10 hours.

In this paper, we find that there exists a large amount of computation redundancy in the video masking process. Motivated by this finding, we propose to accelerate face masking for videos by reusing some cloaks. Specifically, we first locate faces in videos through the key positions of the eyes and mouths using the MTCNN method [15]. Then, we propose a matrix affine technique to transform cloaks based on the relationship of key positions, avoiding re-generating cloaks

for every video frames. The main contributions of this paper are listed as follows:

- We find that existing privacy protection methods against malicious face recognition have high computational overhead for videos. The main overhead comes from the process of generating cloaks for faces in video frames.
- We propose a novel method to reuse existing cloaks, instead of generating new ones to reduce computational overhead, so that video protection can be accelerated.
- We implement our method on TensorFlow and use well-known video data sets for performance evaluation. The results of experiments show that 35.5 % computation can be saved.

The rest of our paper is organized as follows. We first introduce the background and discuss the motivation in Section II. Section III presents the algorithm design. Then, we show performance evaluation in Section IV. Finally, we draw the conclusion in Section V.

II. BACKGROUND AND RELATED WORK

The current protection techniques are outlined in this section. Additionally, we preview three aspects, including metrics, sorts of attackers, and protection methods. The current study extends numerous types of applications and focuses on the human vision and machine recognition aspects. Attacker types are separated into authorized model and unauthorized model based on the relationship between attackers and protectors. Additionally, we list the protections against the unauthorized model.

A. METRICS OF HUMAN AND MACHINE

An increasing number of apps now use this recognition method as DL develops. Modifying images has resulted in several applications, whether they be from a human or machine perspective. Applications in the arts and entertainment change how people perceive themselves. Moreover, applications that avoid face or sign recognition can result from machine perspective analysis of photos. Applications that scan for malware tampering or circumvention also consider the two aforementioned factors.

Celebrity faces are replaced in videos with Deepfakes using DL, and the edited content is then made available online [16], [17]. Applications for this technology are innovative and useful. Examples include the accurate video dubbing of foreign films, historical figure recreations for educational purposes, and virtual dressing rooms when shopping. Although the resultant artifacts are imperceptible to humans, DL analysis makes them easy to spot. Some research look for certain artifacts to identify deepfakes.

Additionally, certain information is not readily detectable by humans but is recognized by machines. In addition, the machine view can be utilized to conceal image or video details without impacting the structure data. Using the real-time video streaming analytics (VSA) front-end, the privacy enhancement system (PECAM) is a flexible privacy-enhancing system [18]. To carry out the privacy enhancement transformation, the authors offer a unique security enhancement cycle-consistent generative adversarial network (GAN) [17], [19]. In PECAM, for instance, each vehicle's license plate number is regarded as private information when used for video surveillance. The monitoring video's finer details are masked by GAN before being uploaded to the server. The structural information is also kept, and it has been empirically demonstrated to be secure and reversible. Consequently, it is usually possible to determine the application of road condition and accident analysis [20], [21]. Moreover, PECAM can retrieve the data for in-depth analysis. For privacy protection, this method works effectively. To get the same visual impression in the context of this paper however, is challenging.

Some programs alter images and videos so they look good from both a human and a machine perspective. The backdoor attack is a typical example, in which the attacker controls the training of the DL model by supplying specific data and labels. This method is also analyzed following as protecting privacy via poisoning attacks. After the DL model is impacted by backdoor data, the data containing the relevant inputs are classified into predetermined categories. Since the change of the images or videos is the user's personal conduct in this article's context, other people's data cannot be impacted. Furthermore, personal information is shared on social media networks where tags cannot be changed. Therefore, backdoor's approach is not completely applicable to our scenario. In summary, we compare two parallel metrics in existing applications and list some applications. Moreover, the shortcomings of the above methods for the scenario in our article are compared.

B. ATTACKER TYPES

Recent studies have shown that DL models can memorize the information of training data [22], [23]. Due to the possibility that these attacks could reveal sensitive information about training dataset participants, training models with differentially private (DP) is becoming more popular [24], [25]. Note that these methods presume a stable model trainer and are not appropriate for unauthorized model trainers. This DP scheme is shown as:

$$Pr[M(x) \in S] \leq e^{\mathcal{E}} Pr\left[M(x') \in S\right] + \delta \tag{1}$$

where δ is a positive real number for relaxation and $\mathcal{E}$ is the scale coefficient of the protection method. The underlying premise of DP is that, if the impact of randomly altering database entries is sufficiently limited, the statistical characteristics that result cannot be utilized to deduce the content of a single record [26], [27]. In general, an algorithm is a DP algorithm if it is impossible to determine whether the result of the algorithm uses data about a specific person.

DP is usually used to query data [28], [29]. The cutting-edge approach uses the concept of inquiry to video in order to safeguard privacy. Privid first divides the long videos into short videos of equal length and counts the appearance time of the object [30]. After that, the object occurrence time is sorted into the same database records. The authors use statistics to measure the privacy of each object and the information

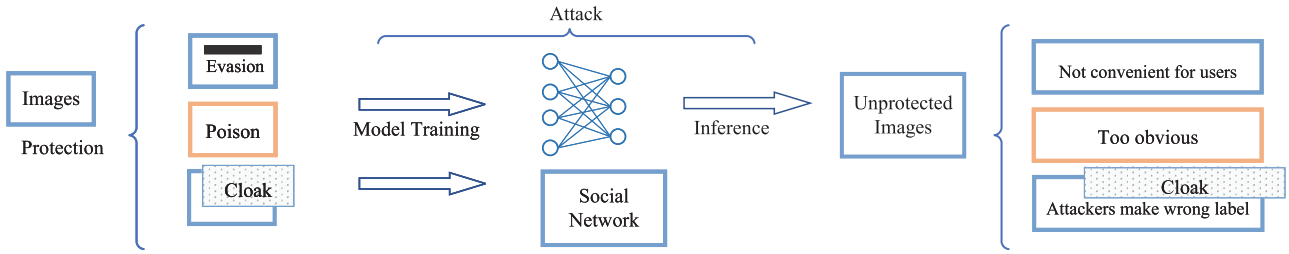


FIGURE 1. There are three ways to protect privacy on images. 1) Evasion protection methods cover a portion of the face with the suitable decoration. 2) Poison protection methods generate the images to interfere with the attackers. 3) Cloaking protection methods go a step further and modify the image at the pixel level.

available in a single query. Each access can only return results that have been processed, such as the quantity and duration of objects. Using this method, Privid divides continuous video into discrete query pieces and then increases the DP of the query results [26], [30]. This method mostly applies to tracking statistics, not to private social platform videos. We have no control over how the video is edited or how the inquiry process works. Methods based on DP are therefore unrelated to the subject matter of this paper. Not the amount of data a query can access at once needs to be regulated, but rather the effect that an unauthorized attacker could have after training the data.

C. PROTECTING PRIVACY METHODS

We provide a preview of cloaks which protect privacy through the pixel-level changes. As shown in Fig. 1, there are three ways to protect privacy on images. Protecting privacy via evasion attacks uses the appropriate decoration to cover part of the face. Poison in protection methods generate the images to interfere the attackers. Cloaking protection methods modifies the image at the pixel-level. In order to protect user privacy from unauthorized attackers, many techniques use attacks against DL models. This scenario can be seen as a reversal of the traditional roles of attacker and protector, where the user are the attackers and a third party tracker with unauthorized tracing is the protector.

1) PROTECTING PRIVACY VIA EVASION ATTACKS

This type of technology requires the user to wear the appropriate decoration, which is not suitable for normal use [22], [31]. To evade tracking, this kind of methods need adequate white box access to the attacker model. The recognition effect of the tracker is calculated as the optimization objective. Therefore, the scope of application is small and easy to be defended [32], [33]. The other kind of evasion method changes the original image obviously, which will affect the normal use.

2) PROTECTING PRIVACY VIA POISONING ATTACKS

Another way to avoid DL model attack is to interfere with their training [31], [33]. A typical one is the backdoor attack, in which the attacker guides the training process of DL model by generating specific data and labels. Model corruption attacks actively attack the tracker, which will easily lead the tracker to use more advanced attack methods. In fact, although it is difficult to eliminate the influence of backdoor data in the model, it is not difficult to detect attacks [34]. On the other hand, the label is the owner of image under face recognition task. Since our images or videos are published on social platforms, they can only be protected by clean label methods. Clean label attacks does not change the label of the data, but by modifying the original data to protect the effect.

3) PROTECTING PRIVACY VIA CLOAK

Protecting privacy via cloak is more suitable for the scenario of individual privacy [10], [14], [32]. Firstly, it can be modified for a user's personal data, Secondly, it has good generality and can deal with a wide range of models. Third, the concealment of abnormal detection is better. Different from backdoor attacks, this approach does not trigger the wrong classification by having the model record a particular input-output pair. This particular input-output pair is independent of the original recognition task. Protection method via cloak is equivalent to modifying the input-output pair in the recognition task.

III. ALGORITHM DESIGN

This section describes threat model and assumptions as shown in Fig. 2. The method for locating key positions of the face is then demonstrated. By combining the consideration of visual effects and privacy protections, we formulate the loss function. We then go over how to re-generate the cloaks using adjacent frames.

A. THREAT MODEL AND ASSUMPTIONS

In this section, we present the threat model and assumptions for both users and trackers as shown in Fig. 1. Then we analyse the intermediate results of face recognition models, which are called intermediate features. We follow existing work's assumptions about the computing power, where users can obtain intermediate features [13], [14].

Users: Users' goal is to share their images or videos on social platforms while preventing facial recognition by unauthorized facial recognition trackers. In addition, changes of visual effects should be constrained, to maintain regular usage [14], [34], [35]. Therefore, users should apply data pre-processing for protection before uploading. The pre-processed data can lead trackers to train faulty models that fails to recognize user faces. Unfortunately, such pre-processing is computationally

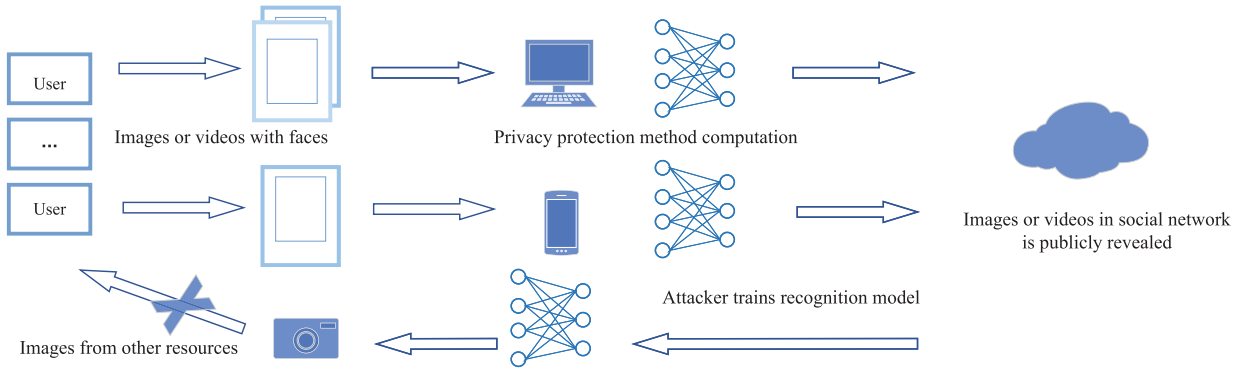
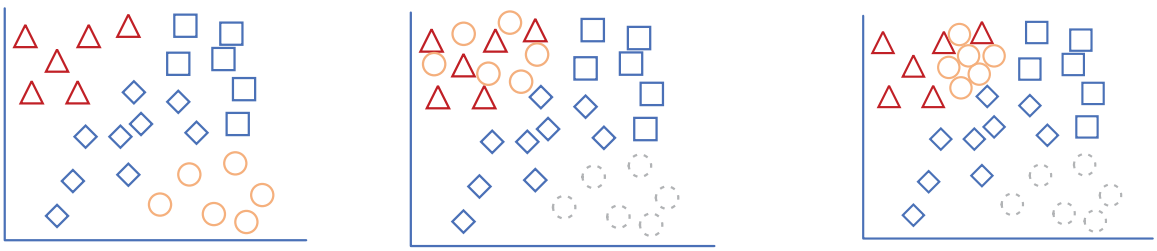


FIGURE 2. Users would like to publish their videos or images on social media platforms but they don't want unauthorized face recognition trackers to be able to identify who they are. Trackers have sufficient processing power. They can train the recognition model using a vast data set available on social media.



(a) Original feature space (b) Feature space of pre-processed individual images (c) Feature space of pre-processed frames in a video

FIGURE 3. The operation principle of the protection method in feature space is visually demonstrated. In the figure, we show an example of a dataset with four people. Classes are distinguished by different shapes, where the circle is the protected class. (a) The data is spread over locations. (b) The protection method changes data features. (c) The feature distribution of video is more concentrated than individual images.

expensive, especially for videos, which brings high computation overhead for users. Therefore, we have the following design goals for the protection method.

- The protection method should not affect the visual effects of images or videos.
- The malicious models trained on the pre-processed data cannot recognize user's faces.
- The computational overhead of the protection method should be acceptable for users.

Tracker: We assume that trackers have sufficient computing power. They can access large data sets or use pre-trained feature extractors through transfer learning. The intermediate feature of faces in the data set are located in the same high-dimensional space, which is called feature space. As a result of the large amount of data in the feature space, we have the chance to confuse our own data with other data. The attacker that only identifies a single user is out of this paper's scope. At the same time, we mainly consider the case, where social platform is the primary source of personal data. Although, in reality, user's data might also be leaked from other sources. If the user provide enough pre-processed data, the effect of real data provided by other sources is negligible.

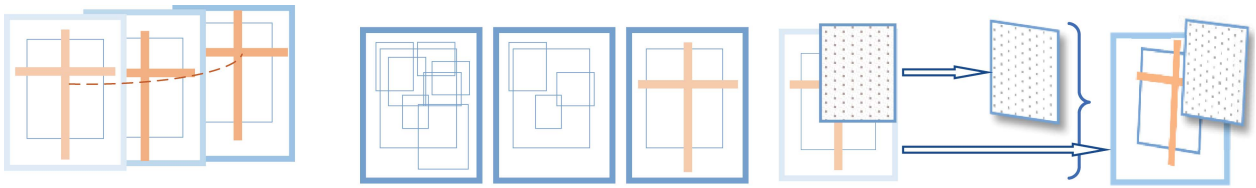
Feature Space: We explain the intuition of the protection method using an example with four classes in the feature space as shown in Fig. 3. Four users' data are mapped to different locations in the feature space. Note that feature space has multiple dimensions; for visual purposes, we present a two-dimensional one here. The same user's data is distributed across nearby regions as a result of their similar features. The orange circle user pre-processes his data using the protection method. To obfuscate the malicious model, the protection method moves the user's data to the locations of another user, called target class. In our example, the user data is moved closer to the red triangle class. Due to the large variation of features, even if the tracker selects a different model, it won't typically assign the orange circle data to its real class.

Moreover, the relationship between the data belonging to the same user must be considered. In fact, the identical person's data are not entirely mapped in the same location in feature space. Weather, angle, and camera equipment can also have an impact on the intermediate features of the same person. In a given video, these outside variables tend not to drastically alter. As a result, compared to individual images, the intermediate elements in a video frame are more closely distributed as shown in Fig. 3(c).

B. PROTECTION PROCESS

1) LOCATING FACE POSITIONS

When pre-processing video data, the video is first divided into frames. Inspired by MTCNN [15], we use a positioning model



(a) Relations of frames in a video (b) Locating face Positions (c) Efficient cloak generation method for continuous frames

FIGURE 4. Video image cloaks are related to one another. The key positions are tracked by the model for position detection. Based on key positions, we apply an affine transformation method for cloaks.

cascaded of three deep convolutional neural networks (CNN) to locate faces and their landmarks as shown in Fig. 4(b). First, using shallow CNN, the positioning model quickly generates candidate windows, each of which has a chance of locating the face. Second, a more sophisticated CNN refines the candidate windows and filters out many of them. Finally, CNN outputs the face location and refines the results. Note that the positioning model specifically outputs the location of the face together with the of the eyes and mouth, called key positions. Later in the cloak generation section, we will detail how to identify the relationship between adjacent cloaks through key positions.

2) JOINT OPTIMIZATION OF VISUAL EFFECTS AND PRIVACY PROTECTION

The basic aspect of the our protection method is to superimpose the faces by a pixel matrix, called cloak. The cloak influences the frame in both visual effect and the position of intermediate features. We measure the visual effect changes using Structure Similarity Metric (SSIM) [14], [31], [36]. SSIM assesses how similar two frames are, utilizing brightness, contrast, and structure as three different dimensions as follows:

$$\begin{aligned} l(x,y) &= \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1}, \\ c(x,y) &= \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2}, \\ s(x,y) &= \left(\frac{1}{\sqrt{N-1}}\frac{x-\mu_x}{\sigma_x}\right).\left(\frac{1}{\sqrt{N-1}}\frac{y-\mu_y}{\sigma_y}\right). \end{aligned} \tag{2}$$

where μ_x and μ_y represents the mean of x and y, respectively. C_1 and C_2 are the divided zero protections, which are constant time the value range of the images. σ_x and σ_y stand for variance of x and y, respectively.

Then the SSIM can be obtained by multiplying the above brightness, contrast, and structure similarity as follows:

$$\begin{aligned} SSIM(x,y) &= l(x,y)\cdot c(x,y)\cdot s(x,y), \\ &= \frac{\left(2\mu_x\mu_y + C_1\right)\left(2\sigma_{xy} + C_2\right)}{\left(\mu_x^2 + \mu_y^2 + C_1\right)\left(\sigma_x^2 + \sigma_y^2 + C_2\right)}. \end{aligned} \tag{3}$$

Note that the SSIM [14] value is proportional to the similarity between two frames.

While using SSIM to measure visual effects, we use the changes in intermediate feature to reflect protection effects. We train a CNN facial recognition model to extract the intermediate features of the frame. The frame's dimensionality can be swiftly decreased using the convolutional layer. We basically need to intercept the convolutional layer's output as an intermediary feature in our proposed protection method. It is straightforward for us to acquire the Minkowski Distance [37] from the intermediate features in the lower dimensions, called feature distance. The output of the convolutional layer [38] on the position (k, x, y) is computed with the region of inputs according to the following equation:

$$\begin{aligned} F(x) &= \sum_{c}^{C}\sum_{m}^{F_w}\sum_{n}^{F_h} W_{(c,m,n,k)} * In_{(c,x+m,y+n)} + b_j, \\ &0 \le k \le K, 0 \le x \le I_w - F_w, 0 \le y \le I_h - F_h \\ Dt &= \left(\sum_{c}^{C}\sum_{m}^{F_w}\sum_{n}^{F_h} |F\left(x_a\right)_{(m,n,k)} - F\left(x_b\right)_{(m,n,k)}|^p\right)^{\frac{1}{p}}. \end{aligned} \tag{4}$$

where the convolution layer is defined by the weight $W_{(c,m,n,k)}$ of height F_h, width F_w and channel C in network. The convolution layer scans the space of the inputs with height I_h and width I_w. p is the order of Minkowski Distance. The convolutional layer's parameters are frozen and no longer changed again after the model is converged. We can map the frames to their positions in the feature space using the frozen recognition model. In addition, simply using a portion of the network for computation leads to lower costs, which may be managed within the user's tolerance range [11], [13], [14].

To sum up, the optimization strategy of the protection method should combine the visual effects and feature distance. The optimization objective is:

$$\begin{aligned} L_f &= -Dt\left(F(x), F\left(x_m\right)\right) + \lambda_1 Dt\left(F\left(x_m\right), F\left(x_p\right)\right), \\ L_s &= \lambda \max\left(|Ds\left(x, x_m\right)| - \rho, 0\right) \\ &\quad - \lambda_2 max\left(|Ds\left(x_m, x_p\right)| - \rho, 0\right), \\ &\quad \min L_f - \lambda_3 L_s. \end{aligned} \tag{5}$$

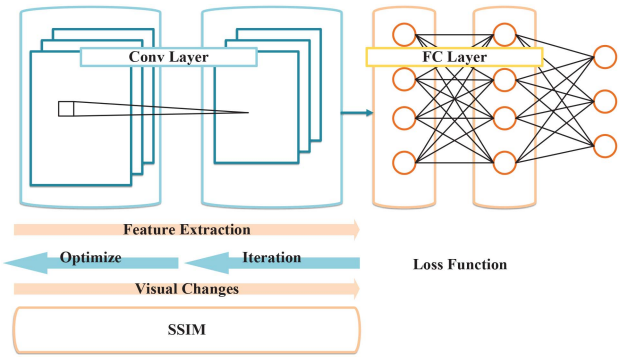


FIGURE 5. The facial recognition model is frozen during optimization.The loss function is directly differentiated against the parameters in the cloak.The iterative gradient descent procedure gradually updates the cloaks to the ideal value.

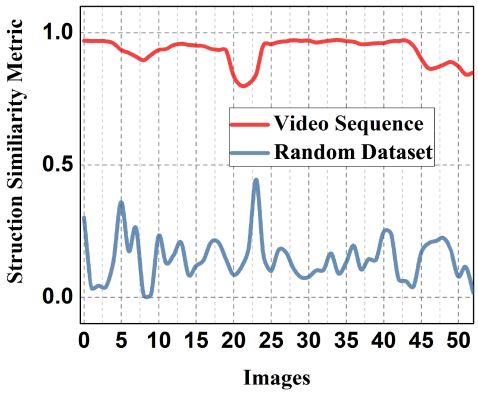


FIGURE 6. We choose a number of continuous frames from a video, then we use SSIM to assess how similar they are. We choose a few other people's individual photos at random for comparison. Continuous videos typically have SSIM values that are higher than single images.

where F is the feature extractor to translate the frames to the feature space. $\lambda \max(|Ds(x, x_m)| - \rho, 0)$ calculates SSIM, which is the measure of the view effects. In order to improve the consistency of the video, we consider the feature distance and SSIM between adjacent frame x_p and recent frame x_m. The image is mistakenly assigned to different categories by shifting its location within the feature space. Therefore, using these pre-processed data as guidelines, malicious face recognition likewise produces inaccurate results. The loss function is directly differentiated against the parameters in the cloak in Fig. 5. A random pixel matrix, which is the same size as the face, serves as the initiation of a cloak. By using an iterative gradient descent procedure, the cloaks are gradually updated to the optimal value.

3) EFFICIENT CLOAK GENERATION

There are many similarities between continuous frames in a video, including similar positions in feature space and visual effects. As shown in Fig. 6, we select continuous frames from a video and measure their similarity using SSIM. For comparison, we randomly select images of several people from the well-known VGG2 data [14], [39]. The SSIM value is often greater than 0.7 in continuous videos. Even identical images from the same person have a low similarity rate of typically less than 0.5. The average similarity between the random images is 0.25.

Fig. 7 presents the optimizations process of individual images and frames in a video. The process of random initialization and optimization of individual images is shown in Fig. 7(a). The central part is the final optimization point. The protection method uses gradient descent to update the parameters of the cloak during training. The iterative processes are depicted in the figure by the arrows. For a computation to be efficient, initialization is a crucial step in the optimization process. The number of iterations is increased by random or improper initialization. However, the traditional optimization process fails to take into account how the video's frames relate to one another, which results in numerous unnecessary iterations as shown in Fig. 7(b).

To re-generate the cloak for the recent frame, we use the one from the previous frame. After the iteration process of the previous frame, the cloak is reused for the subsequent frame as the initialization in Fig. 7(c). Ideally, the optimization result of the previous frame is close to the next frame. In the actual process, if the actions of the two frames is unconspicuous, this intuitive approach works well. However, the face in the video have some obvious rotation, which degrades the performance of the above method. Therefore, we propose a matrix affine transformation method to track the face rotation, which will be detailed explain in the following subsection.

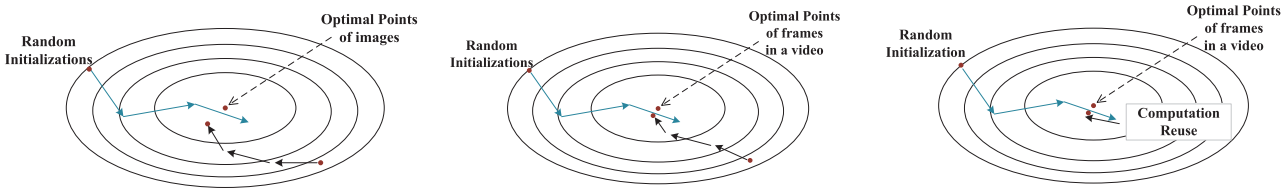
4) RELATIONSHIP BETWEEN CONTINUOUS FRAMES

We re-generate the cloak based on the prior one using an affine transformation method [38]. The key positions, such as the corners of the mouth and the eyes, are first located from the continuous frames. Additional computational expenses can be avoided by incorporating the task of finding key positions within the positioning model. As a result, we create the affine transformation matrix as follows:

$$\overrightarrow{f(P)f(Q)} = \varphi\left(\overrightarrow{PQ}\right),$$
$$\begin{bmatrix}\vec{y}\\1\end{bmatrix} = \begin{bmatrix} A & | \vec{b} \\ 0 \ \ldots \ 0 & 1 \end{bmatrix}\begin{bmatrix}\vec{x}\\1\end{bmatrix} \tag{6}$$

where P, Q are the points before affine in $\overrightarrow{f(P)f(Q)}$. y is selected from key positions. A and b are affine matrices that determine the changes of each point.

The corners of the mouth and the centers of the eyes are the key positions that the affine transformation method uses to identify the plane of the face in a recent frame. Once affine matrices have been established, we can utilize the affine transformation method as given in. The pixels of the frame are affined in this manner [38]. The areas where the pixels in the new frame do not match those in the old frame are filled using area interpolation. The cloaks are created to match the updated frames following the affine transformation method. Therefore, the optimization process is accelerated.



(a) Optimization process of individual images (b) Traditional optimization process of frames in a video (c) optimization process using computation reuse method of frames in a video

FIGURE 7. The optimization process of the image and video data is visually demonstrated. The optimization target of each image is represented as the central points, respectively. (a) Two images are protected independently. (b) The objects of consecutive frames in a video has high similarity. (c) The clocks in the video frames is reused.

Algorithm 1: Privacy Protection.

Input Video
for images $r = 1, \ldots, m$ **do**
  Using positioning model to locate the face in the image
  Store key positions generated by positioning model
  if the Cloak of $r - 1$ exist **then**
    `Affine` the Cloak of by the key positions
  else
    Randomly initializes Cloak
  end if
  while loss $\geq$ threshold **do**
    Iteratively optimize Cloak
  end while
  `Output` protected images
end for
Restore the image to video
`Output` protected video

5) VIDEO PROTECTION PROCEDURE

Frames are removed from the input video at the beginning of protection method in Algorithm 1. The pixel matrix of the face is then cuts out once the positioning model has first identified the key positions on each frame as shown in Fig. 8 . During this procedure, key positions for each image are noted. The protection mechanism randomly initializes the cloak's parameters while dealing with the first frame. Otherwise, the affine transformation matrix is generated by extracting the key positions from the preceding and most recent frames. The loss function in this protection method is made up of visual effects and feature distance. The cloak's parameters are iteratively updated by the optimization function until the loss function is below the threshold, as shown in Algorithm 1. Each frame is protected and ultimately converted to a video in the manner described above.

IV. EVALUATION

We conduct extensive experiments to evaluate the performance of the proposed method. This section evaluates the overhead of the protection method for videos. Moreover, some variants of data are chosen for comparison. Finally, we analyze three loss function indexes to reflect the visual effect and protection effect.

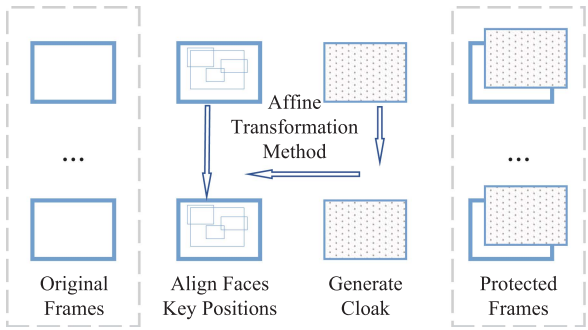
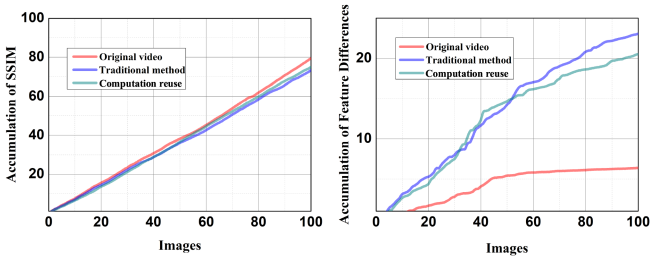


FIGURE 8. The affine approach uses three key positions to establish the face's plane in the most recent frame. We employ the affine transformation method.The cloak's pixels are affined in this manner. When the pixels in the new frame do not match those in the old frame, area interpolation is employed to fill those gaps. The cloaks are created to match the updated frames following the affine procedure.

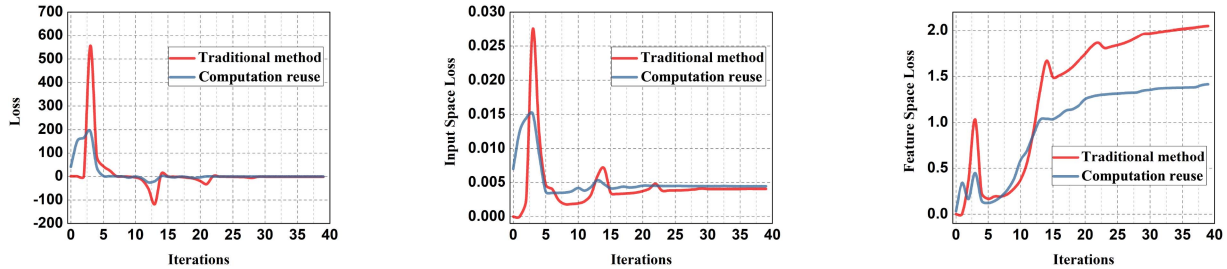


(a) SSIM Relations in a video (b) Feature distance in a video

FIGURE 9. Relation metrics of frames in videos.

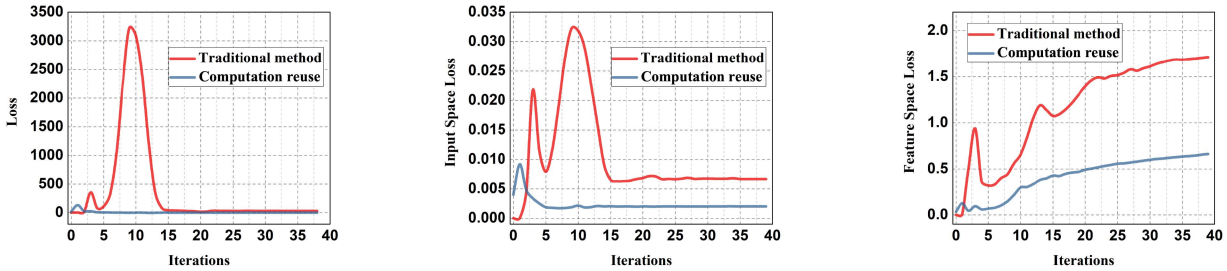
A. ENVIRONMENT

We deploy our experiment on TensorFlow [39], which runs with 4 $\times$ 2.8 GHz Intel Core i7 CPU, 8 GB memory and Nvidia GeForce RTX 3060 GPU. After careful consideration of the experimental scale, we choose the well-known British original drama data for protection [14]. We deploy the



(a) Optimization Loss (b) Input Space Loss (c) Feature Space Loss

FIGURE 10. Evaluation on short videos.



(a) Optimization Loss (b) Input Space Loss (c) Feature Space Loss

FIGURE 11. Evaluation on long videos.

TABLE 1. Protection Success Rate in Face Recognition Platforms

Face Recognition API	Fawkes	Our Method
Amazon Rekognition Face Verification	100%	100%
Face++ Face Search API	100%	100%

well-performance protection system [14] and list their respective calculations. Each configuration is arranged according to previous work [12], [36].

Metrics: Fawkes simultaneously optimized two different modifications. Users' visual effects are unaffected. The attacker's training model may be misled by the pre-processed videos, misclassifying the real user data. As a result, there are two separate indicators to measure these objectives. The visual differences between videos taken before and after protection are represented by SSIM. A lower value means that the protection method has less of an influence on the image or video's routine use. The protective effect measurement is the other component. We continue to use the feature extractor in previous work [14]. Through the pre-trained model, the feature extractor compares the feature distance between two frames. The method's improved protective impact is generally shown by the noticeable feature change. The total loss function, which is inversely proportional to space loss and square of input loss, is summarized at the end. In addition, our method can achieve the same image protection efficiency as Fawkes in Table 1.

Hyper-parameters: Our approach includes several hyper-parameters. We undertake a thorough analysis to determine the ideal hyper-parameter settings in order to improve the performance of the computation reuse approach. The selection of hyper-parameters mainly refers to previous experimental results [14], [39] and the results obtained in actual use. The first of the hyper-parameters is the number of iterations. Each cloak is initialized, either randomly in the conventional method or through the computation reuse process by mapping the cloak from the previous frame. The cloaks are iteratively optimized after initialization, with a maximum value of 40 iterations set for each image. Typically, the iteration results can be finished inside this upper limit. On the other hand, we established a threshold and used it to gauge calculation speed throughout the comparison process. When the rate of optimization within the threshold accelerates, the aforementioned number of iterations can be reduced to boost the optimization rate.

B. EVALUATION

We demonstrate the effects of the traditional method and the computation reuse method on the data's visual effects in Fig. 9(a), Compared to existing methods, our method gets the similarity between frames closer than traditional method to the original video. As shown in Fig. 9(a), both methods have an impact on the feature continuity of the original video. Our method is considerably more similar to the original video.

The values of the loss metrics are shown in Fig. 10. We contrasted the two examples' video processing outcomes based on the duration of the videos. The first one is for seconds-long videos, while the second one is for minutes-long ones. Firstly, the general situation of picture protection is observed. In the initial iteration of the process, the loss function is negligible. Since the cloak's value is so negligible, it has no discernible visual effect on the frames. The other side doesn't really affect the characteristic space much. Adadelta is the optimization function we select [37]. So that the loss function can increase quickly at roughly 2–5 iterations, this optimization technique accumulates and updates in the direction of the first iterations. Adadelta can prevent the initial optimization process from staying in the local best quality. Firstly, observe the three indicators in the short video shown in Fig. 10(a). For input space loss shown in Fig. 10(b), due to random initialization, the traditional method suffers a minor loss at the beginning. It is quickly optimized, nevertheless, to bring about several alterations. After optimization, we simultaneously stay online and vary till around 30 iterations have passed. Feature space Loss presents the same trend in Fig. 10(c), but in the end, loss has undergone more notable alterations than computational reuse. Unpredictability before and after the frames, of course, could influence visual effects. Based on the optimization loss results above, the computation reuse can be utilized below the threshold in fewer iterations, saving 33.4 % computational costs.

The outcomes are comparable to the short video, as seen from the perspective of the extended video, as shown in Fig. 11. Finding optimal values is made easier by using computation reuse. Moreover, the benefits were more obvious than with short videos. Similar frames are more common during the calculating process. The previous calculation results are comparatively stable and can be utilized immediately for calculating the next frame. The final experimental result, however, continued to use the maximum value of iteration as the standard rather than the average result since we did not consciously enhance the judgment of the similarity of the number of related frames in order to lessen the computing burden. It is important to note that the iterative option uses the iterative setting rather than the loss function's threshold value. Because the loss function can occasionally oscillate above and below the threshold value, altering the outcome of the protection.

V. CONCLUSION

Pre-processing of images containing personal data is crucial for maintaining DL privacy. We present a video study and do an empirical analysis of the affine relations for cloaks. We also consider the limitations of earlier protection methods, which are limited to individual image protection. As a result, the protection procedure is improved by the affine technique. We adapt the cloak to the ongoing frames using key positioning model. Our evaluation shows that our method works much better than the conventional ways; for instance, the 35.5% computation consumption is saved.

REFERENCES

[1] M. Abadi et al., "Deep learning with differential privacy," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, 2016, pp. 308–318.

[2] N. Carlini and D. Wagner, "Adversarial examples are not easily detected: Bypassing ten detection methods," in *Proc. 10th ACM Workshop Artif. Intell. Secur.*, 2017, pp. 3–14.

[3] D. Ambra et al., "Why do adversarial attacks transfer? explaining transferability of evasion and poisoning attacks," in *Proc. 28th USENIX Conf. Secur. Symp.*, 2019, pp. 321–338.

[4] N. Aaron et al., "Level playing field for million scale face recognition," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2017, pp. 7044–7053.

[5] T. Li and L. Lin, "AnonymousNet: Natural face de-identification with measurable privacy," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops*, 2019, pp. 56–65.

[6] M. Fredrikson, S. Jha, and T. Ristenpart, "Model inversion attacks that exploit confidence information and basic countermeasures," in *Proc. 22nd ACM SIGSAC Conf. Comput. Commun. Secur.*, 2015, pp. 1322–1333.

[7] S. T. Jan, J. Messou, Y.-C. Lin, J.-B. Huang, and G. Wang, "Connecting the digital and physical world: Improving the robustness of adversarial attacks," in *Proc. 33rd AAAI Conf. Artif. Intell. 31st Innov. Appl. Artif. Intell. Conf. 9th AAAI Symp. Educ. Adv. Artif. Intell.*, 2019, Art. no. 119.

[8] Q. Sun, A. Tewari, W. Xu, M. Fritz, C. Theobalt, and B. Schiele, "A hybrid model for identity obfuscation by face replacement," in *Proc. Eur. Conf. Comput. Vis.*, 2018, pp. 553–569.

[9] Y. Wu, F. Yang, Y. Xu, and H. Ling, "Privacy-protective-GAN for privacy preserving face de-identification," *J. Comput. Sci. Technol.*, vol. 34, no. 1, pp. 47–60, 2019.

[10] S. Thys, W. V. Ranst, and T. Goedemé, "Fooling automated surveillance cameras: Adversarial patches to attack person detection," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 49–55.

[11] Z. Wu, S.-N. Lim, L. S. Davis, and T. Goldstein, "Making an invisibility cloak: Real world adversarial attacks on object detectors," in *Proc. Eur. Conf. Comput. Vis.*, 2020, pp. 1–17.

[12] A. Shafahi et al., "Poison frogs! Targeted clean-label poisoning attacks on neural networks," in *Proc. 32nd Int. Conf. Neural Inf. Process. Syst.*, 2018, pp. 6103–6113.

[13] C. Zhu, W. R. Huang, H. Li, G. Taylor, C. Studer, and T. Goldstein, "Transferable clean-label poisoning attacks on deep neural nets," in *Proc. 36th Int. Conf. Mach. Learn.*, 2019, pp. 7614–7623.

[14] S. Shan, E. Wenger, J. Zhang, H. Li, H. Zheng, and B. Y. Zhao, "Fawkes: Protecting privacy against unauthorized deep learning models," in *Proc. 29th USENIX Conf. Secur. Symp.*, 2020, Art. no. 90.

[15] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint face detection and alignment using multitask cascaded convolutional networks," *IEEE Signal Process. Lett.*, vol. 23, no. 10, pp. 1499–1503, Oct. 2016.

[16] Y. Mirsky and W. Lee, "The creation and detection of deepfakes: A survey," *ACM Comput. Surv.*, vol. 54, no. 1, pp. 1–41, 2021.

[17] L. A. Gatys, A. S. Ecker, and M. Bethge, "Image style transfer using convolutional neural networks," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2016, pp. 2414–2423.

[18] H. Wu et al., "PECAM: Privacy-enhanced video streaming and analytics via securely-reversible transformation," in *Proc. 27th Annu. Int. Conf. Mobile Comput. Netw.*, 2021, pp. 229–241.

[19] O. Gafni, L. Wolf, and Y. Taigman, "Live face de-identification in video," in *Proc. IEEE/CVF Int. Conf. Comput. Vis.*, 2019, pp. 9377–9386.

[20] F. Mo et al., "DarkneTZ: Towards model privacy at the edge using trusted execution environments," in *Proc. 18th Int. Conf. Mobile Syst. Appl. Serv.*, 2020, pp. 161–174.

[21] M. Xu, X. Zhang, Y. Liu, G. Huang, X. Liu, and F. X. Lin, "Approximate query service on autonomous IoT cameras," in *Proc. 18th Int. Conf. Mobile Syst. Appl. Serv.*, 2020, pp. 191–205.

[22] C. Song, T. Ristenpart, and V. Shmatikov, "Machine learning models that remember too much," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, 2017, pp. 587–601.

[23] N. Carlini et al., "Towards evaluating the robustness of neural networks," in *Proc. IEEE Symp. Secur. Privacy*, 2017, pp. 39–47.

[24] C. Dwork, "Differential privacy: A survey of results," in *Proc. Int. Conf. Theory Appl. Models Comput.*, 2008, pp. 1–19.

[25] Z. Yang et al., "Neural network inversion in adversarial setting via background knowledge alignment," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, 2019, pp. 225–240.
[26] N. Johnson, J. P. Near, and D. Song, "Towards practical differential privacy for SQL queries," *Proc. VLDB Endowment*, vol. 11, no. 5, pp. 526–539, 2018.
[27] F. D. McSherry, "Privacy integrated queries: An extensible platform for privacy-preserving data analysis," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, 2009, pp. 19–30.
[28] F. Bastani et al., "MIRIS: Fast object track queries in video," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, 2020, pp. 1907–1921.
[29] Z. Cai, M. Saberian, and N. Vasconcelos, "Learning complexity-aware cascades for deep pedestrian detection," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2015, pp. 3361–3369.
[30] F. Cangialosi, N. Agarwal, V. Arun, S. Narayana, A. Sarwate, and R. Netravali, "Privid: Practical, privacy-preserving video analytics queries," in *Proc. 19th USENIX Symp. Netw. Syst. Des. Implementation*, 2022, pp. 209–228.
[31] J. Steinhardt, P. W. Koh, and P. Liang, "Certified defenses for data poisoning attacks," in *Proc. 31st Int. Conf. Neural Inf. Process. Syst.*, 2017, pp. 3520–3532.
[32] O. Suciu, R. Marginean, Y. Kaya, H. Daume III, and T. Dumitras, "When does machine learning generalized transferability for evasion and poisoning attacks," in *Proc. USENIX Conf. Secur. Symp.*, 2018, pp. 1299–1316.
[33] Y. Wu et al., "DeltaGrad: Rapid retraining of machine learning models," in *Proc. 37th Int. Conf. Mach. Learn.*, 2020, pp. 10355–10366.
[34] B. Wang et al., "Neural cleanse: Identifying and mitigating backdoor attacks in neural networks," in *Proc. IEEE Symp. Secur. Privacy*, 2019, pp. 707–723.
[35] B. Wang, Y. Yao, B. Viswanath, H. Zheng, and B. Y. Zhao, "With great training comes great vulnerability: Practical attacks against transfer learning," in *Proc. 27th USENIX Conf. Secur. Symp.*, 2018, pp. 1281–1297.
[36] M. Sharif, S. Bhagavatula, L. Bauer, and M. K. Reiter, "Accessorize to a crime: Real and stealthy attacks on state-of-the-art face recognition," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, 2016, pp. 1528–1540.
[37] O. Suciu et al., "When does machine learning fall? generalized transferability for evasion and poisoning attacks," in *Proc. 27th USENIX Conf. Secur. Symp.*, 2018, pp. 1299–1316.
[38] G. Singh, R. Ganvir, M. Püschel, and M. Vechev, "Beyond the single neuron convex barrier for neural network certification," in *Proc. 33rd Int. Conf. Neural Inf. Process. Syst.*, 2019, Art. no. 1352.
[39] M. Abadi et al., "TensorFlow: A system for large-scale machine learning," in *Proc. 12th USENIX Conf. Operating Syst. Des. Implementation*, 2016, pp. 265–283.

ENTING GUO received the master's degree with the Nanjing University of Posts and Telecommunications, Nanjing, China, in 2020. He is currently working toward the Ph.D. degree from the School of the Division of Computer Science, University of Aizu, Aizuwakamatsu, Japan. His research interests include AI systems, and AI security and privacy.

PENG LI (Senior Member, IEEE) received the B.S. degree from the Huazhong University of Science and Technology, Wuhan, China, in 2007, and the M.S. and Ph.D. degrees from the University of Aizu, Aizuwakamatsu, Japan, in 2009 and 2012, respectively. He is currently a Senior Associate Professor with the University of Aizu. He has authored or coauthored more than 100 papers in major conferences and journals. His research interests mainly include cloud/edge computing, Internet-of-Things, distributed AI systems, and AI security and privacy. He was the recipient of the Young Author Award of IEEE Computer Society Japan Chapter in 2014, Best Paper Award of IEEE TrustCom 2016, and Best Paper Award of IEEE Communication Society Big Data Technical Committee in 2019. He supervised students to win the First Prize of IEEE ComSoc Student Competition in 2016. Dr. Li was also the recipient of the 2020 Best Paper Award of IEEE Transactions on Computers. Dr. Li is the Editor of IEEE OPEN JOURNAL OF THE COMPUTER SOCIETY, and *IEICE Transactions on Communications*.

SHUI YU (Senior Member, IEEE) received the Ph.D. degree from Deakin University, Burwood, VIC, Australia, in 2004. He currently is a Professor with the School of Computer Science, University of Technology Sydney, Ultimo, NSW, Australia. He initiated the research field of networking for big data in 2013, and his research outputs have been widely adopted by industrial systems, such as Amazon cloud security. He has authored or coauthored four monographs and edited two books, more than 500 technical papers, including top journals and top conferences, such as IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, IEEE TRANSACTIONS ON COMPUTERS, IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY, IEEE TRANSACTIONS ON MOBILE COMPUTING, IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, IEEE TRANSACTIONS ON EMERGING TOPICS IN COMPUTING, IEEE/ACM TRANSACTIONS ON NETWORKING, and INFOCOM. His research interests include Big Data, security and privacy, networking, and mathematical modeling. His h-index is 66. He is currently serving a number of prestigious editorial boards, including IEEE COMMUNICATIONS SURVEYS AND TUTORIALS as an Area Editor, *IEEE Communications Magazine*, IEEE INTERNET OF THINGS JOURNAL, and so on. He was a Distinguished Lecturer of IEEE Communications Society (2018–2021). He is a Distinguished Visitor of IEEE Computer Society, a Voting Member of IEEE ComSoc Educational Services Board, and an Elected Member of Board of Governor of IEEE Vehicular Technology Society.

HAO WANG (Senior Member, IEEE) is currently an Associate Professor and the Head of the Big Data Laboratory, Department of ICT and Natural Sciences, Norwegian University of Science and Technology, Trondheim, Norway. He was a Researcher with IBM Canada, McMaster, and St. Francis Xavier University, Antigonish, NS, Canada, before he moved to Norway. His research interests include Big Data analytics and industrial Internet of Things, high-performance computing, safety-critical systems, and communication security. He has authored more than 60 papers in the IEEE TVT, GlobalCom 2016, Sensors, the IEEE Design & Test, and Computer Communications. He is a Member of the IEEE IES Technical Committee on Industrial Informatics. He was a TPC Co-Chair of the IEEE DataCom 2015, IEEE CIT 2017, and ES 2017, and a Reviewer of journals, such as the IEEE TKDE, TBD, TETC, T-IFS, and ACM TOMM.