Efficient Privacy Protection in Video Surveillance by StegoScrambling

Natacha Ruchaud
Multimedia department
Eurecom
Biot Sophia antipolis, France
Email: ruchaud@eurecom.fr

Jean Luc Dugelay
Multimedia department
Eurecom
Biot Sophia antipolis, France
Email: dugelay@eurecom.fr

***Abstract*—This paper presents a near lossless reversible system which allows a user to protect privacy in video surveillance (or still images) by replacing sensitive RofIs (Regions of interest) by its edges using steganography while keeping visual quality needed for security almost in real time. Our proposed system outperforms the state-of-the-art methods by fulfilling four criteria which are near lossless Reversible, Fast Computation (integrated in real time), Usability (preserves shape and motion of people to still recognize events) and Privacy protection (no more possibility to recognize people). The effectiveness of the proposed filter has been demonstrated using some face recognition algorithms on different images.**

I. Introduction

The growth in the adoption of video surveillance systems emphasizes the need of privacy protection techniques. Significant recent improvements have been made in the field of pedestrian detection [1], face detection, human recognition and image enhancement. Also the improvement in image sensors helps those identification techniques (e.g. a person can be recognized even far away from a camera). This is why, questions about respect of privacy become more and more important.

Some applications to protect privacy have been created. As examples, privacy can be protected by pixelization, blurring or black masking with FacePixelizer [1] on Google plus, ObscuraCam [2] on Android, and also by scrambling in the JPEG compression domain with Scrambling JPEG tool [3]. Other works have been performed to hide identity like morphing [2], warping [2], scrambling [3] but they are rarely used in practice because of their complexity or their applications which degrade the visual quality needed for security.

To respect privacy in video surveillance, a filter should satisfy four criteria:

-Protection of privacy,

-Usability which means that the visual quality of the protected video enables for further computer vision tasks as recognize events in the scene (i.e. people walking, running, fighting, stealing...),

-Reversibility which means the possibility to reverse the process by authorized people owing a key,

-Real-time integration.

None of existing privacy filters satisfies all criteria for video surveillance as shown in Table I. Indeed, the more identity is hidden with masking, warping and morphing filters the lower is the quality of the recovering (using the inverse process) contrary to scrambling which is always reversible. Nevertheless, scrambling is not able to recognize events consequently this filter cannot fulfil privacy protection in video surveillance.

TABLE I. Summary of Existing Filters

Filter	Pricacy	Usability	Reversibility	Time
`Pixelization/Blur`	Yes	Yes	No	Fast
`Masking/Warping`	Yes	No	No	Fast
`Morphing`	Yes	No	No	Low
`Scrambling`	Yes	No	Yes	Low
`Scrambling JPEG`	Yes	No	Yes	Fast
`StegoScrambling`	Yes	Yes	Yes	Fast

Our proposed system allows privacy protection in real time while keeping the shapes which lets recognition of events in the scene. Moreover, our privacy filter is near lossless reversible, and self reversible (Automatic localisation of filter) contrary to other reversible filters which need to store RofIs localisation.

A preliminary version of our approach was, first, designed within the context of the Mini-drone Video Privacy Task at MediaEval Benchmark 2015 [4] but it did not run in real time and RofIs (Regions of Interest) have been annotated manually for the challenge. A video, where this preliminary approach [5] is applied, can be downloaded at http://www.eurecom.fr/ruchaud/StegoScrambled.avi.

Our current method runs in real time on videos from smartphone (Android) using the IP address of the camera or on drone (AR.Drone 2.0. Parrot) by integrating our algorithm in the APIs of CV Drone (OpenCV + AR.Drone) [4].

II. Implementation system

A. Regions of Interest (RofIs)

RofIs (e.g. people bounding boxes) are automatically detected by Dalal & Triggs method [1]. Moreover, since the detection rate is low, we added some conditions using the temporal information. First, the search area is shrunk to the area close to the previous RofI (if there is no people detection

[1] http://www.facepixelizer.com/
[2] https://guardianproject.info/apps/obscuracam/
[3] http://ltslinux18.epfl.ch/scramble/
[4] https://github.com/puku0x/cvdrone/wiki/API-reference

in the previous frame, the search area is the entire image). Second, if nobody is detected in the current frame, whereas there was a detection in the previous frame, we assume that the person does not move far away from the previous RofI therefore the current RofI is set to the previous. Nevertheless, after more than twenty successive frames, where nobody is detected, we assume that the person is no more present.

B. Generate a seed

User has to provide a string, denoted *passphrase*. Each letter of this *passphrase* is converted into a number according the ASCII code. And the Least Significant Bit (LSB) of the first thirty two numbers are concatenated. Consequently, a number between 1 and 2^{32}, named the seed, is obtained.

C. StegoScrambling

With a pseudo-random number generator (PRNG), controlled by a seed generated in 2.2, numbers are generated randomly. An XOR is computed between the 6 Most Significant Bit (MSB) of the RofI and the random numbers in order to hide information as in equation 1.

$$XORImg(i) = RofI(i) \oplus RandNums(i), \forall i \qquad (1)$$

with *i* the bit position and each bit $\in \{0, 1\}$.

In parallel, an edge detector is applied on the RofI which returns a bi-level image. Finally, the 2 MSBs of the edge image (pixels intensity is either 192 or 0) are inserted in the 2 MSBs of the resulting image and the 6-bit of the XOR image (pixels intensity between 0 and 63) are integrated in the LSB of the resulting image as in the equation 2. Therefore, only edge images are visible by viewers which enable to recognize events.

$$PrivacyImg = \sum_{i=0}^{5} XORImg(i) * 2^i + \sum_{i=6}^{7} EdgeImg(i) * 2^i, \qquad (2)$$

Figure 1 illustrates the workflow of the proposed method and shows an example of an entire privacy image.

D. Inverse StegoScrambling

To recover the original RofI, the authorized people have to know the *passphrase* which was used to generate the seed in the same way than 2.2.

As previously, with a PRNG, controlled by the seed, numbers are generated randomly. An XOR is computed between the 6 LSBs of the privacy image and the random numbers as in equation 3. The result of the XOR is the recovered image.

$$RecoveredImg = \sum_{i=2}^{7} (PrivacyImg(i-2) \oplus RandNums(i)) * 2^i \qquad (3)$$

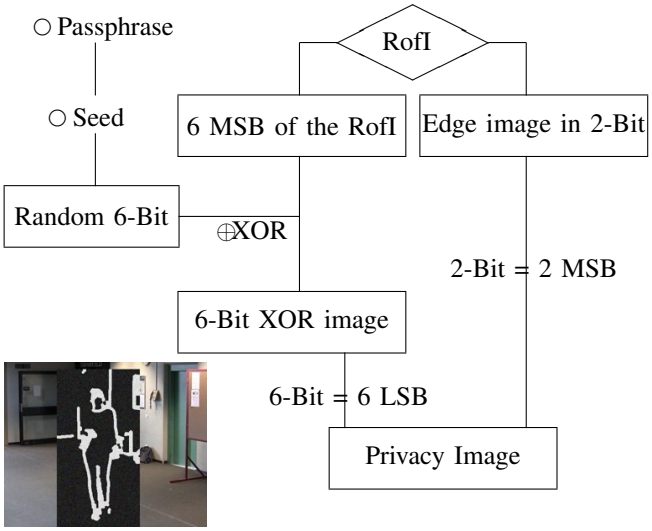

Fig. 1. Workflow of the proposed process

III. Evaluation

Two LSBs of each pixel intensity are removed from original RofI. Thus, at most each pixel intensity of the recovered RofI decreases of three compared to those of the original RofI which has no effect for human vision and a minimal impact for machine.

Our method has been evaluated in terms of reversibility and privacy with some face recognition algorithms. Results clearly show that rate of face recognition is almost null when our filter is applied. Moreover, face recognition for recovered and original images perform almost equally.

On average, an image is shown in 0.547 seconds by our system with an image resolution of 640 * 480. More specifically, people detector takes 0.53 seconds per image and StegoScrambling 0.017 seconds per image. The amount of time consuming is mainly due to people detector and can be explained by the fact that we use only CPU. Nevertheless, we are going to integrate our process using GPU thus, the system should be at least five times faster.

IV. Conclusions

We have presented a new method for protecting privacy on videos from smart-phone, camera or sensors of drone by combining scrambling, steganography and cryptography.

Our reverse process is not enough robust against compression. We currently work on the integration of this process in JPEG and MPEG compression domain.

References

[1] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, vol. 1. IEEE, 2005, pp. 886–893.

[2] P. Korshunov and T. Ebrahimi, "Towards optimal distortion-based visual privacy filters," in *IEEE International Conference on Image Processing*, no. EPFL-CONF-197087, 2014.

[3] A. Melle and J.-L. Dugelay, "Scrambling faces for privacy protection using background self-similarities," in *Image Processing (ICIP), 2014 IEEE International Conference on*. IEEE, 2014, pp. 6046–6050.

[4] A.Badii, T.Ebrahimi, P.Koshunov, JL.Dugelay, C.Fedorczak, T.Piatrik, V.Eiselein, A.Al-Obaidi, and N.Ruchaud, "The 2015 droneprotect task: Mini-drone video privacy task," 2015. [Online]. Available: http://www.multimediaeval.org/mediaeval2015/droneprotect2015/

[5] N. Ruchaud and J.-L. Dugelay, "Privacy protection filter using stego-scrambling in video surveillance." In MediaEval 2015 Workshop, Wurzen, Germany, Sept, 2015.