ParisTech
INSTITUT DES SCIENCES ET TECHNOLOGIES
PARIS INSTITUTE OF TECHNOLOGY

TELECOM ParisTech

2018 ENST 0009

EDITE - ED 130

Doctorat ParisTech

T H È S E

pour obtenir le grade de docteur délivré par

TELECOM ParisTech
Spécialité « Signal et Images »

présentée et soutenue publiquement par

Natacha RUCHAUD

16 Février 2018

Privacy Protection, Preserving the Utility of Visual Surveillance

Directeur de thèse: **Jean-Luc DUGELAY**

Jury

M. Marc ANTONINI, Directeur de Recherche CNRS, Laboratoire I3S, France	Rapporteur
Mme. Azza OULED-ZAID, Professeur, Université de Tunis El Manar, Tunisie	Rapporteur
M. Frédéric DUFAUX, Directeur de Recherche, TELECOM ParisTech, France	Président de Jury
M. Laurent GIULIERI, Directeur de Recherche, Digital Barriers, France	Examinateur
M. Jean-Luc DUGELAY, Professeur, Eurecom, France	Directeur de Thèse

THÈSE

TELECOM ParisTech
école de l'Institut Télécom - membre de ParisTech

Abstract

Privacy Protection, Preserving the Utility of Visual Surveillance

by Natacha Ruchaud

Due to some tragic events such as crime, bank robberies and terrorist attacks, an unparalleled surge in video surveillance cameras has occurred in recent years. In consequence, our daily life is overseen everywhere (e.g. on the street, in stations, in shops and in the workplace). For example, on average, people living in London can be caught on cameras more than 300 times a day. At the same time, automatic processing technology and quality of sensors have advanced significantly, which has even enabled automatic detection, tracking and identification of individuals. With the proliferation of video surveillance systems and the progress in automatic recognition, privacy protection is now becoming a significant concern.

Video surveillance is intrusive because it allows the observation of certain information that is considered as private (i.e., identity or some characteristics such as age, race, gender). Nowadays, some processing technologies are able to limit the intrusiveness of surveillance by automatically hiding private information. Indeed, a number of studies have recently begun to focus on the protection of personal privacy, especially the identity, but they are seldom taken care of crucial criteria required for the surveillance (e.g. reversibility, the utility of the surveillance, compliant with the compression standards).

For a long time, privacy and safety of people were viewed as mutually exclusive factors. One could not preserve the utility of visual surveillance and protect the privacy at the same time, both are very significant, though. While privacy is essential to freedom, surveillance is a major actor for our safety. The purpose of this thesis is to find technological solutions to the issue of privacy protection of individuals while preserving the utility of the surveillance (i.e. leaving the understanding of the scene and enabling the re-identification of a person in case of an incident). Indeed, an ideal surveillance system should protect the personal privacy of individuals while still providing a high level of the utility of visual surveillance. Existing methods have issues to manage this trade off, usually, the increase in utility of surveillance brings about a significant decrease in personal privacy.

To address privacy concerns regarding digital image or video surveillance cameras, we propose one main concept: using the most important information to preserve the ability to recognize actions while protecting individual identities by encrypting and hiding their original information in the least important information of the data. This fulfils a better trade off between privacy and safety of people compared to the existing methods in that domain. We integrate this idea, first, in the spatial/pixel domain, and, then, in the transform/frequency domain to be compliant with the classical compression standards such as JPEG and H.264/AVC. Indeed, nowadays almost all images and videos are compressed. Moreover, the data that we encrypt is reversible with a secret key being available for authorized people only.

Résumé

En raison d'événements tragiques tels que la criminalité, les vols et les attentats terroristes qui persistent dans le monde, nous avons pu observer, en particulier ces dernières années, un déploiement important des systèmes de surveillance (notamment via l'installation de caméras dans la rue, dans les gares, dans les magasins et même sur les lieux de travail). Par exemple, un Londonien peut passer devant des caméras, en moyenne, plus de 300 fois par jour. En parallèle, les techniques de traitement automatique ainsi que la diversité et la qualité des capteurs ont considérablement progressé, ce qui a permis la détection automatique, le suivi et l'identification des individus. Avec la prolifération des systèmes de surveillance et les progrès de la reconnaissance automatique, la protection de la vie privée devient, par conséquent, une préoccupation importante.

La vidéo surveillance est intrusive car elle permet d'observer certaines informations considérées comme privées (par exemple, l'identité ou des caractéristiques telles que l'âge, l'ethnicité, le genre). De nos jours, il existe des techniques dont le but est de protéger la vie privée en masquant automatiquement les informations sensibles. En effet, depuis plusieurs années de nombreuses études ont été effectué dans le domaine de la protection de la vie privée, mais elles ne prennent pas en compte d'important critères liés à la surveillance (réversibilité, préservation de l'utilité de la surveillance, respect des normes de compression).

La vie privée et la sécurité des personnes ont longtemps été considérées comme des facteurs mutuellement exclusifs (contradictoires). On ne pouvait pas préserver l'utilité de la surveillance et protéger la vie privée en même temps pourtant les deux sont très importants. La vie privée est essentielle à notre liberté tandis que la surveillance est un acteur majeur de notre sécurité. Le but de cette thèse est de trouver des solutions technologiques qui répondent à cette problématique : respecter la vie privée des individus tout en préservant l'utilité de la surveillance (c-à-d, être capable de reconnaître les actions de la scène et pouvoir ré identifier une personne en cas d'incident). En effet, un système de surveillance idéal devrait protéger la vie privée des individus tout en offrant un haut niveau d'utilité de la surveillance. Les méthodes existantes ont du mal à gérer ce compromis, généralement, l'augmentation de l'exploitabilité de la surveillance entraîne une diminution significative du respect la vie privée.

Pour répondre aux préoccupations de confidentialité concernant les caméras de vidéo surveillance ou des images numériques, nous proposons de préserver la capacité à observer des actions grâce à la manipulation des informations les plus importantes tout en protégeant l'identité, en cryptant et en cachant leurs informations originales dans les informations les moins importantes. Ainsi, nous obtenons un meilleur compromis, par rapport aux méthodes existantes du domaine, entre la vie privée et la sécurité des personnes (c-à-d, l'utilité de la surveillance). Nous intégrons ce concept d'abord dans le domaine spatial / des pixels, puis dans le domaine des fréquences pour être conforme aux standards de compression classiques tels que JPEG et H.264/AVC. En effet, de nos jours, presque toutes les vidéos sont compressées. De plus, les données que nous cryptons sont réversibles grâce à une clé secrète connue uniquement par les personnes autorisées.

Acknowledgements

Working as a PhD student in EURECOM was a great experience that would have not be achieved without the help, guidance and support of many people, who I would like to acknowledge here.

First and foremost, I would like to acknowledge my supervisor Prof. Jean-luc Dugelay for giving me the opportunity to join his team as a PhD. at EURECOM / Telecom ParisTech. Throughout my Ph.D. he provided ingenious ideas and encouraging support. He created a vastly positive and enthusiastic working atmosphere that fuelled self-motivation and ambition.

I would like to thank my committee members, the reviewers Prof. Mark Antonini and Prof. Azza Ouled Zaid, and furthermore the examiner Dr. Frédéric Dufaux for their precious time in reviewing this manuscript, and in sharing positive insight and guidance.

I owe my deepest gratitude to my parents, Véronique Ruchaud and Eric Ruchaud, for their unwavering encouragement, devotion and unconditional love in this long but fascinating way. I also thank, Ponpon, my cat who through his eyes gave me courage. I do not forget my spouse, Nicolas Roux, who supported me through the hardships.

My warmest thanks to my colleagues who supported me during my Ph.D. Precisely, I would like to thank Chiara, Julien, Katy, Stéphanie, Khawla, Ihsen, Grigory, Massimiliano, Pasquale, Valeria, Pramod, Pepe, Hector and many others for their vital support and for sharing their brilliance and creativity with me. Also, I thank all those working at EURECOM, they made my stay there very pleasant. Lastly, special thanks to my friends for their unwavering friendship, moral and infinite support.

The research presented in this thesis was supported by the European project VIDEOSENSE. That is why, I would like to express my deepest appreciation for members of this Group led by Prof. Atta Badi.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Context and Motivation

Most of the people post and share photos and videos online of private or public events (e.g., the birthday of your child, concerts, etc.). This new trend, filming or taking photographs everywhere, emphasizes the need of privacy protection techniques.

On the other hand, due to a serie of terrorist attacks at the beginning of the 21st century and increasing criminal activities in recent years, video surveillance system (i.e., CCTV cameras) is becoming part of daily life and is a major component of many security systems. Many CCTV cameras are needed to record critical areas continuously and, then, video can convey an enormous amount of information that can be considered as sensitive. According to a Canadian study[1], there is an impressive growth in the adoption of video surveillance systems in public transport, parking facilities, shopping places, airports, banks, schools, workplaces, hospitals, city centers/streets. The Table 1.1 illustrates the percentage of CCTV depending on the place. According to this study, public/open street surveillance (42%) represents the most common place for CCTV surveillance. The workplace and public transit (e.g., trains, buses) also represent the most likely location to find a video surveillance network. Surprisingly, the presence of surveillance cameras in parking garages, banks, retail stores, and schools, is not as high as expected.

TABLE 1.1: Locations of CCTV surveillance, % according to the total number of camera

Public/Open Street	Airport	School	Hospital	Bank	Transit	Border	Business/Workplace	Parking Garage	Other
42 %	0.5 %	5.6 %	0.6 %	3%	6%	0.4%	22.5%	2%	16.8%

The increased surveillance of citizens in public space affects individual privacy rights (in terms of the likelihood of a CCTV image rendering accurate identification). Privacy is the major problem of visual surveillance. About 90% of all adults in the United States are concerned about their privacy with 26% feeling that they have lost their privacy already [12].

[1] http://www.cjc-online.ca/index.php/journal/article/view/2200/3033

Mobile devices as well as CCTV include more and more high-resolution cameras. In particular, the deployment of the highly mobile and versatile drones or wearable camera for surveillance, gives rise to new challenges for civil liberties, privacy and safety. While we increasingly use cameras, the resolution of visual sensors (e.g., 4k, HD) and the performance of video processing algorithms (e.g., identity recognition) are continually upgrading. We are seeing spectacular progress in the field of automatic recognition in images and video processing (e.g., face and body detection, and identity recognition). For example, in [13], authors demonstrate that accuracy of face recognition algorithms has improved up to 30 percent within a period of three years. These efficient automatic image analyses (e.g. recognition of people, vehicles, animals or bags) are integrated into CCTV (Closed-Circuit TeleVision) systems (e.g., Clear View[2]).

The new generation of visual sensors (e.g., 4k, HD) contributes to increase performance attached to those detections/identification techniques (e.g., a person can be recognized even far away from a camera). The powerful video analytic tools combined with pervasive networks of dense cameras highlight failures in the privacy policy. Indeed, if everything we do can be analyzed and be collected, the privacy of people under surveillance is being threatened [14].

On the one hand, video surveillance may help to increase our safety (e.g. deterrent for criminals and deliver evidence to investigate and solve crimes). On the other hand, the privacy of citizens is constantly invaded, and there is a fear that the gathered information can be misused.

Solutions to alleviate privacy concerns from the use of CCTVs, already exist. For example, using a black mask to block out a PIN number entry for ATM security cameras, or to protect private property for outdoor security cameras. However, protecting the privacy of people is more complex given that the monitoring of their actions should not be hampered. The current privacy protection methods that most people probably already saw on TV, are the blurring or the pixelization (e.g., Google Street View, FacePixelizer [3], ObscuraCam [4] on Android) but they are not reversible (i.e., users cannot recover the original video). Technological solutions for this issue have been proposed, but none of them provides adequate privacy protection that fulfills all the criteria needed to be compliant with the surveillance. Specifically, recent works mostly bypass the tasks of the compression and the reversibility or have difficulties to manage the trade-off between the privacy protection and the utility preservation of visual surveillance.

The purpose of this thesis is to provide a contribution to solve the main research question, which was formulated as follows:

How can we protect the privacy of individuals while maintaining the utility of the surveillance?

During this PhD thesis work, our study focus on protecting privacy with the aim of, making compliant the monitoring in video surveillance that are two conflicting objectives. In particular, we intend:

-To address the problem of privacy facing the expansion of cameras.

[2] http://www.clearview-communications.com/cctv/facial-recognition-video-analytics
[3] http://www.facepixelizer.com/
[4] https://guardianproject.info/apps/obscuracam/

-To investigate today's state-of-the-art on the existing privacy protection methods, and their criteria that define the level of privacy protection as well as the one of the utility of visual surveillance.

-To prove the vulnerability of existing privacy protection filters.

-To propose a reversible approach that keeps the motion of the body while hiding both, the identity and the gender information.

-To propose a generic privacy filter which fulfils all criteria needed in video surveillance. In particular, a method integrated in the widely adopted standards: JPEG and H.264/AVC.

-To prove that our process performs the best trade-off between the privacy protection and the utility preservation of video surveillance, compared to the other recent methods of the domain.

One challenge raised in this thesis is to preserve the utility of the surveillance while protecting the privacy at any resolution without ignoring robustness against potential attacks (i.e. someone tries to reverse the process without authorization). The next Section 1.2 presents the contributions of this thesis in the field of privacy protection preserving surveillance.

1.2 Achievements of the thesis

In this thesis, we focus on the problems related to privacy protection preserving surveillance.

Nowadays, the surveillance is mostly done by humans, but more and more it exists tools to automatically detect and classify diverse objects/tasks. Thus, we studied the differences between an objective and subjective evaluation when applying privacy protection filters. We have selected the PETA datasets that contain several databases of pedestrians and we perform the evaluation of gender detection on the original images and on the ones that are protected by a privacy filter. To evaluate the gender detection, we apply a Convolutional Neural Network (i.e., objective evaluation) and made a Crowdsourcing (i.e., subjective evaluation, survey done by human vision). Note that we use the same data for the testing set in both evaluations. We could have thought that the human would have better results than the machine, but, surprisingly, we conclude that both operate almost equally. Therefore, in all experiments of this thesis, we use the objective evaluation. This work was published at ADIP (SPIE) 2015.

In another study, we proved that our privacy is not well protected by the current proposed filters such as the pixelization, the Gaussian blur or even the black masking. We propose a whole architecture to de-anonymized faces (i.e., recover the identity of the face on which a privacy filter has been previously applied). It consists of, first, detecting the presence of a privacy filter, then, detecting its type as well as its strength (e.g., pixelization of size 4) and finally performing a de-anonymization well-defined depending on the identified type and the strength of the privacy filter. This architecture was published at Electronic Imaging 2016.

After showing the weaknesses of the current privacy filters, we design a first reversible (only for authorized people) method to protect the privacy while keeping the shape of the body. In the spatial domain,

we encrypt and shift the Most Significant Bits (MSBs) of the original pixels from a RoI (Region of Interest) to the Least Significant Bits (LSBs). Then, we insert the bits from the edge of this RoI (shape of the body) into the MSBs in order to keep the scene understandable. This method was published within the context of the *Mini-drone Video Privacy Task* at MediaEval Benchmark 2015 and an improved version was presented at WIFS (IEEE) 2015. We also design a new version of the shape of the body using body contours reshaping such as the gender information is no more visible as well as the identity. This work was published at ISBA (IEEE) 2017. This privacy filter is reversible but is not robust against some manipulations, in particular lossy compression. De facto, many applications cannot use then that tool. Indeed, nowadays almost all images and videos are compressed, therefore, image processing algorithms must be compliant with at least the most widespread standards such as JPEG and H.264/AVC.

Therefore, we propose to integrate our approach in these standards. The algorithm operates in the Discrete Cosine Transform (DCT) domain to allow compatibility with the compression of the JPEG and H.264/AVC standards. For each sensitive area of the picture (i.e. area where privacy needs to be protected), the proposed algorithm uses the low-frequency coefficients of the DCT to display a privacy preserved version of the region and the high-frequency coefficients to hide the majority of the original information. Moreover, we encrypt this original information. Our approach allows authorized users to nearly recover the original image from the hidden information. Hence, this method ensures privacy at any image size (which is not the case for other existing similar methods) while preserving the minimum of information required by the surveillance (e.g., recognition of sports in the scene). Our proposed system is near lossless reversible for authorized people who own the key, and secure against brute force, parrot and replacement attacks. Moreover, the method is compliant with the JPEG and H.264/AVC standard. The integration on the standard JPEG was published at ICME (IEEE) 2016 and the one on the standard H.264/AVC at EUSIPCO 2017. An extended version of the second one was published at CVPR (IEEE) 2017 and was presented in French as a poster at GRETSI 2017.

1.3 Thesis Outline

In the first phase of this thesis, we mainly investigate the motivation to protect the privacy, the criteria associated with video surveillance and the existing privacy filters. The **Chapter 2** is dedicated to present related works in visual privacy protection. First, we start by explaining the detection of the region of interest (i.e. the area to protect). Then, we present the existing privacy filters, their advantages and their disadvantages. During our studies, we notice a lack of tools to measure the efficiency of a privacy preserving system. Thus, we introduce the evaluations that are used to assess the criteria of a privacy protection method preserving visual surveillance.

In the **Chapter 3**, we study the differences between an objective and a subjective evaluation of the gender recognition task in images when processed by privacy protection filters. The results show that they perform equally.

In the **Chapter 4**, we propose an architecture that de-anonymized faces protected by common privacy methods (i.e., the pixelization, the Gaussian Blur, the black masking) and therefore prove the weakness

of these filters. To construct this architecture, we have combined several popular feature representations (i.e., HoG, LBPH, PCA) with a linear SVM classifier.

The idea developed in the **Chapter 5 and 6** is to keep the main information required by the surveillance in the most important coefficients while we encrypt and hide the original coefficients in the least important coefficients.

- In the **Chapter 5**, we focus on the design of a new privacy protection filter in the spatial domain (i.e. pixel domain) that keeps the shape of the body. The process is reversible only for authorized people. We encrypt the original bits and hide them in the least significant bits. To preserve the motion of the body, we insert the edge/shape information that can be modified or not (depending on whether we want to hide the gender information or not), in the most significant bits.
- Following the same idea, in the **Chapter 6**, we propose to work with the DCT coefficients in order to integrate the process in the standards JPEG and H.264/AVC. We encrypt and hide the original DCT coefficients in the AC coefficients (i.e., color variations). To keep a minimum information required by the surveillance, we use and adjust the most important DCT coefficient, the DC (i.e., the mean color).

In the **Chapter 7**, we conclude about the presented works, highlight its limitations and suggest new research directions.

Chapter 2

Visual privacy protection: Related work and Background

2.1 Introduction

In this Chapter, we first describe how we detect the area where the privacy needs to be protected. Then, we examine the work related to the research topics that the current thesis deals with. In particular, we cite the advantages and disadvantages of the existing methods in privacy protection and, then, we highlight our contributions. We notice that few metrics in this domain where proposed (i.e., measuring the level of the utility of visual surveillance and privacy in a given area). Therefore, we also present a detailed review of the criteria as well as the assessment protocols that we follow in this thesis to evaluate the efficiency of a privacy filter compliant with the surveillance.

2.2 Detection of sensitive areas (RoIs: Regions of Interest)

We generally detect the sensitive area (i.e., the region in the image where the privacy should be protected), and then, apply a privacy filter on it. The efficiency of the privacy protection methods is strongly dependent on the performance of the detection of the region of interest (RoI). Indeed, if the detector misses a sensitive part, the identity is no more protected. On the contrary, if the detector selects an area that does not need to be protected, the application of the privacy protection will damage this part for nothing.

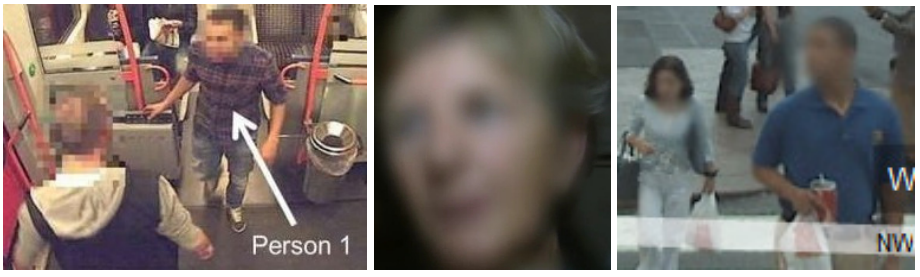
We use either a face or a body detector because we want to protect the identity of people that is usually deduced from the body appearance. Mostly, privacy filters apply their protection to the face only. However, body can offer additional information such as gender, gait or even personal accessories. Based on this information, it has been proven that we can deduce the identity of a person [15].

In this thesis, we use the Viola–Jones [16] face detector (Haar + Integral Images + Adaboost + Cascade) to extract faces and the Dalal and Triggs [17] people detector (Histograms of oriented gradients + SVM) to extract bodies.

We either store the bounding boxes of the RoI in a metadata (e.g., a text file) or inside the image itself or in the header of a compression standard. In the last case, note that the decoder of the header should be aware of this step.

2.3 Current methods in privacy protection

As you probably already have noticed, broadcasting, printed media and some surveillance systems apply obfuscation methods to damage the pixel data and provide anonymity. As we show in the Figure 2.1, they mostly use either pixelization (image subsampling), blurring (smoothing the image with, e.g., Gaussian filter with large variance), or masking by a solid colored rectangle. In the following, we describe with more details these ad-hoc distortions, with x and y pixel coordinates and $I(x, y)$ the original image.



(a) (b) (c)

FIGURE 2.1: Privacy of faces is protected (a) by a pixelazation filter in "20 minutes" a French magazine, by a blurring filter (b) in "crimes" a French program and in (c) Google Street view.

- We apply a **Masking** filter with the following formula:

$$IblackMask(x, y) = I(x, y) \times (1 - \alpha) + color \times \alpha \tag{2.1}$$

  with $color$, the color of the solid rectangle that merges with the original image, and α representing the opacity. The higher is α the stronger is the impact of the filter.

- The convolution of an original image ($I(x, y)$) and a Gaussian function ($G(x, y)$) gives as a result a blurred image, denoted **Gaussian blur** ($IgaussianBlur(x, y)$):

$$G(x, y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}, IgaussianBlur(x, y) = I(x, y) \times G(x, y), \tag{2.2}$$

  where σ is the standard deviation of the Gaussian distribution. This function modifies each pixel of an image using neighbouring pixels. For example, blurring is used in Google Street View [18] to hide human faces and licence plates.

- **Pixelization** [19] can be perceived as a downsampling of the image without modifying the size. Applying a pixelization reduces the number of distinct pixel values in the RoI by replacing a square block of pixel values (of size N) with their averaged value. The pixelization of the image I(x, y) is given by:

$$Ipixelization(x,y) = \frac{1}{N^2}\sum_{i=0}^{N-1}\sum_{j=0}^{N-1} I(\lfloor\frac{x}{N}\rfloor + i, \lfloor\frac{y}{N}\rfloor + j), \tag{2.3}$$

  where N is the block size. Pixelating is commonly used in television to preserve the anonymity of suspects, witnesses or bystanders.

- A **Noising** filter adds a kind of noise (e.g., Gaussian noise, Salt and Pepper noise, Poisson noise) to the original image:

$$IgaussianNoise(x,y) = I(x,y) + n(x,y) \tag{2.4}$$

  with n(i,j) a random variable.

We show, in the Figure 2.2, images on which we applied these previous privacy filters. YouTube [1] creates an application which allows users to protect their own videos with a blurring filter. Authors, in [20], proposed to distinguish intentionally-captured persons (ICPs) from the non-ICPs, and then, they blur or mask with a solid colored rectangle the regions of non-ICPs.

However, these common approaches are far to be ideal/optimal privacy filters and compliant with surveillance due to some unfulfilled crucial criteria:

- In [21], results indicate that participants were still able to recognise some of the pixelated and blurred faces in videos and static images.
- Pixelating and blurring filters are easy to defeat. Indeed, reconstruction attacks can be perpetuated by super resolution techniques [22, 23]. In addition, when training a parrot recogniser using the pixelated and blurred images [24], the recognition becomes efficient. Indeed, high recognition rates are obtained (near 100%) despite looking somewhat/somehow de-identified to humans. Additionally, we prove, in the Chapter 4, that we can restore obscured face images (i.e., faces on which privacy protection is applied) by automatically extracting the type and the strength of the privacy filter and applying the image restoration method associated.
- The original information is irreversibly lost in the process. Thus, either you have to keep a copy of the original data (without the privacy filter) or you could never recognize a person's identity again in case of need.

[1] https://www.youtube.com/

FIGURE 2.2: Privacy filters. From left to right: original body, black masking, Gaussian blur, pixelization, Gaussian Noise, K-means, morphing, warping, encryption.

2.4 Advanced privacy protection methods

In this section, we introduce the work performed during the last years to design an efficient privacy filter and we highlight their advantages as well as their disadvantages according to essential criteria in privacy and utility preservation of the surveillance.

K-means

The K-means clustering filter [25] aims to partition the color value of pixels into N clusters in which each color value belongs to the cluster with the nearest mean value using the Euclidian distance.

The K-means clustering algorithm [26] chooses N colors (r, g, b) randomly as the assumed centroids at the beginning. Then, it minimizes the Euclidean distance between each pixel color and these centroids. Finally, original pixel values are replaced by the corresponding centroid values. The strength of K-means is determined by the number of clusters. This privacy filter has the same drawbacks as the traditional one: not reversible and easy to restore.

Morphing

The idea behind the morphing filter [27] is to find an average face/body image between the source and the target faces/bodies according to a given interpolation level. The source face/body corresponds to the face/body of the individual whose identity must be preserved. The target face/body is any generic human face/body.

A simple morphing approach consists in applying the following formula (similar to the one of the masking filter):

$$IMorphing(x, y) = I(x, y) * (1 - \alpha) + target * \alpha \tag{2.5}$$

α representing the opacity. The higher α is, the stronger is the impact of the filter.

In [27], Korshunov and Ebrahimi introduce a more advanced morphing. The method divides both images into Delaunay triangles [28] and transforms the vertices of the source image to the vertices of the target image. Pixel intensities are also interpolated with respect to a second parameter. Its security can be ensured by encrypting the key points, the interpolation level and the pixel interpolation values for each triangle. However, as the algorithm begins with triangulating face images, it may fail to work in cases where the faces are not captured from 'ideal' angles.

The stronger the morphing is, the higher is the estimation error of the inverse application. We need a stronger morphing to protect the privacy, therefore, in this case, the morphing is seldom reversible.

Photorealistic

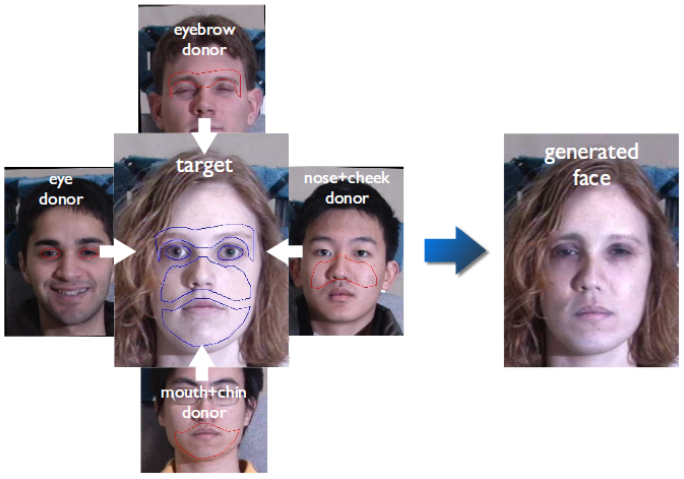


FIGURE 2.3: Generated face from the face components of donors. Reprinted from [1].

In [1], faces are de-identified by substituting their own face components (e.g., eyes, chin, nose, etc.) with the donors' ones, in such a way that the appearance of the generated faces is as close as possible to original faces while the identities are made unrecognizable. This method is highly dependent on the detection of facial components.

Warping

Warping is a geometrical transformation and is performed by applying a mathematical function to a set of coordinates in the image. Warping can preserve (depending on the strength level) general aspects of a person. Nevertheless, the strength level has to be strong because human vision is robust against geometrical transformations. Moreover, the stronger the warping, the higher the estimation error of the inverse application is, as authors demonstrated in [29]. Therefore, warping is partially reversible.

Color transformation

FIGURE 2.4: Applying false colored image. Reprinted from [2].

Authors, in [2], transform the original color palette of an image into a different color palette. The method is reversible only for authorized people because they encrypt the color palette. However, the traits of a person are still visible. Thus, using an edge extraction or an efficient Convolutional Neural Network (CNN) as a feature descriptor, the identities would probably be recognized.

Inpainting / Object removal

Inpainting methods aim to remove sensitive regions of an image by filling the left gap with the corresponding background. In [3] and [30], authors hide privacy information inside image itself and use an inpainting method to refill the privacy-sensitive regions. Despite the strong protection of privacy, the actions of people are invisible that is not compliant at all with surveillance.

Body shape filling

In the work of [31], they provide a robust background subtraction to detect people and their shadow, and then apply an existing privacy protection method on them. The system allows authorized people to recover the original information of selected individuals using robust tracking methods. This approach was designed only for static cameras, and is very dependent on the extraction of the exact body contour that is more challenging to obtain on moving cameras.

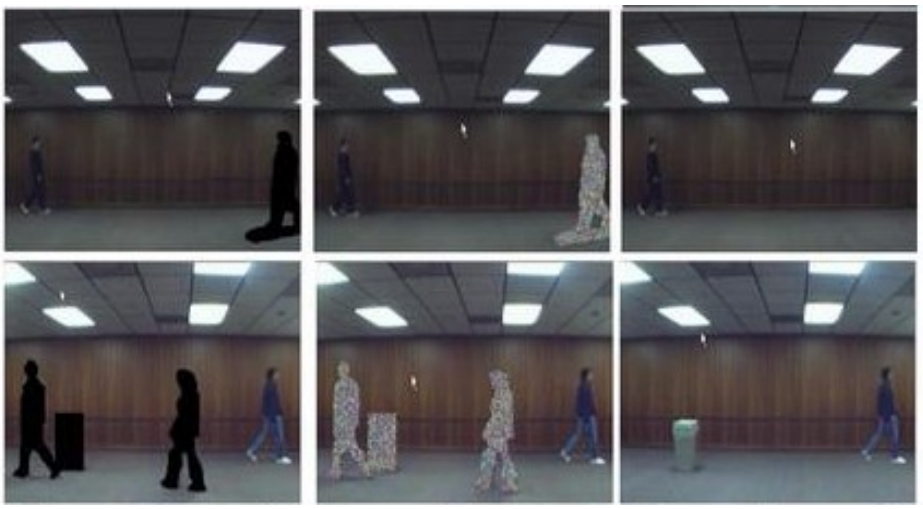


FIGURE 2.5: Applying a black masking, a scrambling and an inpainting methods on the exact shape of the region to hide. Reprinted from [3].

One to several

In [32], authors split each frame into a set of random images that are meaningless about the original frame while collectively, they retain all the information. This method does not preserve the visualization of the events.

K-Same family of algorithms

K-Same family of algorithms [24, 33] implement the k-anonymity protection model [34] for face images. K-Same achieves k-anonymity protection by averaging each k-closest faces in a database based on the Euclidean distance, in image space or Principal Component Analysis coefficient space [24]. Then, they add k-copies of the resulting average into this database, therefore, guarantee that the probability of a de-identified face to be correctly recognised by a face recognition software is not higher than 1/k. To manage additional information such as gender or facial expressions, k-Same-Select was introduced in [33]. In [35], authors propose a new concept, denoted ”Controllable Face Privacy", a flexible method that independently controls the amount of identity alteration while keeping unchanged other facial attributes. They incorporated k-anonymity mechanism into their approach.

In [5], authors created "Garp-face" method that consists of protecting privacy while retaining soft biometric traits such as gender, age and race. They use modern facial analysis technologies to determine the gender, age, and race attributes of facial images and preserve these attributes by seeking and merging corresponding facial images (in terms of attributes) through a gallery dataset respecting the k-anonymity protection.

Authors, in [36], adopt a watermarking method that makes the k-same process reversible. The difference between the original and privacy protected image is compressed, encrypted and embedded within the privacy protected image itself.

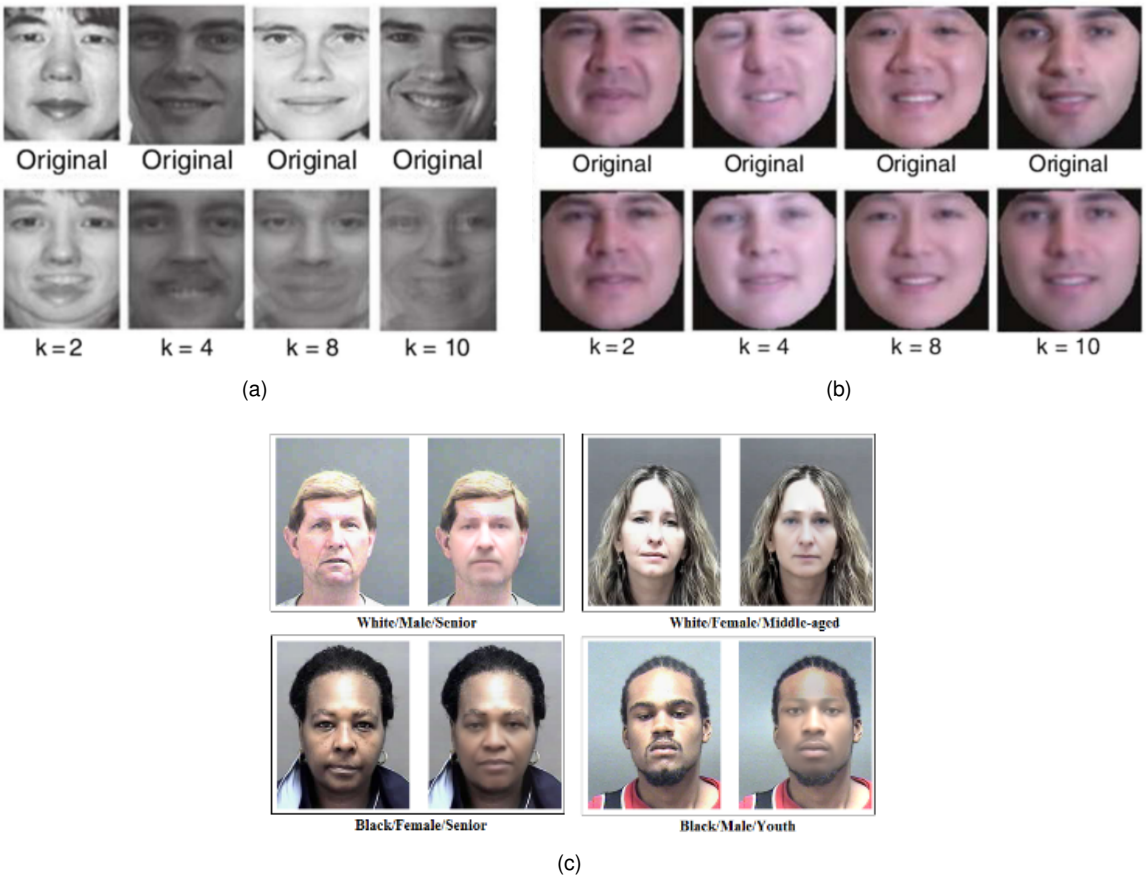


FIGURE 2.6: (a) Protection of faces identity by using the k-Same algorithm where some ghosting artefacts appear due to misalignments in the face set, (b) by using k-Same-M algorithm and (c) by fulfilling the k-anonymity while preserving soft biometric traits. Reprinted from [4, 5].

In [37], authors anonymize faces using the average projection of k face images in every cluster. They select a partition by picking a random face image and selecting the k -1 nearest images (of distinct persons) according to the Euclidean or MahCosine distance metric. Then, for every cluster, they project each of the k images, and they compute the average projection (of the PCA). All images in a cluster are modified to have the same averaged projection such as they visually preserve the recognition. This method allows a human to recognize a person from the image while preventing the face identification by software.

K-Same does not provide k-anonymity privacy protection if a subject is represented more than once in the dataset. In order to address this shortcoming, Gross, Sweeney et al. [4] propose a multi-factor model which unifies linear, bilinear and quadratic models. By using a generative multi-factor model, a face image is factorized into identity and non-identity factors. Afterwards, the de-identification algorithm is applied and the de-identified face is reconstructed using the multi-factor model.

Even if at first sight, k-same algorithm appears as an efficient method in terms of protecting the identity and preserving other traits of the person, they have several drawbacks:

- As face Morphing and Photorealistic, the algorithm requires to use a library containing many face images and it is restricted only for face identity protection (not the whole body). A lot of studies demonstrated that people can be recognized even by their body [38] or their gait [39]. Therefore, protecting only faces, is not enough to prevent the identification.
- They are just appropriate for low-dimensional data due to the curse of dimensionality [40], hence they do not fit most of the multimedia data (e.g., video surveillance data). Moreover, a variety of attacks make many of these methods unreliable [41].

Visual encryption/scrambling to secure data (in Image/Video)

In privacy protection field, encryption methods are often used to allow the reversibility and the security of the process, for authorized people only. By applying encryption on data coefficients (e.g., pixels, DCT), a distorted data is obtained for unauthorised viewers. Only users who have the proper key for the decryption can visualise the original data. Scrambling techniques are a subfield of image encryption, generally based on coefficient permutation methods using a pseudo-random number generator (PRNG). The PRNG generates sequences of random numbers determined by an initial value, called the seed. This algorithm is often used in practice for their speed in number generation and their reproducibility.

Encryption algorithms are commonly used such as Data Encryption Standard (DES), Rivest's Cipher (RC5), Advanced Encryption Standard (AES), Rivest, Shamir and Adleman (RSA). These algorithms guarantee high security level, but unfortunately, they are not suitable for real-time encryption because of their high computation time [42, 43]. Thus, other encryption algorithms [44] have been proposed such as a simple XOR cipher or only encrypting some bits of the data stream, which are more appropriate for real-time.

- **Steganography (Data hiding/embedding)**

  The goal of Steganography is to conceal a secret message within another message called cover message. This message can be a file, an image, a video or a text. Cryptography is first used to encrypt messages and then steganography hides it so that no one suspects it exists.

  The steganography process is the following:

  As inputs, we denote the cover message c, the message m and the key k, and as output, the stego message s: $Embed(c, m, k) = s$. We recover the original message by using the correct key and applying the inverse process: $Recover(s, k) = m$. Authors, in [45], first, encrypt the original data and then use the least significant bits of an image to hide the encrypted data.

- **Block-based pixel cryptography/encryption/scrambling approach**

  The process of PICO (Privacy through Invertible Cryptographic Obscuration) [46, 47] combines cryptographic techniques and image processing approaches provide a practical solution to the critical issue of privacy invasion. Authors, in [48], suggest a subband-adaptive approach for scrambling the regions of privacy-sensitive face in JPEG XR-encoded surveillance video content. In [49], the authors argue that chaos cryptography techniques are more appropriate than traditional ones

such as RSA (Rivest–Shamir–Adleman) when it comes to encrypt large amounts of data. These methods generate a lot of noise which leads to degrade the image.

Even though the proposed methods succeed in protecting privacy and ensuring reversibility, they perform poorly on the intelligibility level (i.e., keep the scene understandable), since the regions of interest are replaced by random noise.

Therefore, Melle and Dugelay [50] introduce a reversible scrambling method which operates in pixel domain using background self-similarities, and preserve the data utility. However, the image can be reconstructed only at the lowest levels of privacy protection, whereas we need a high level to prevent identity recognition, thus, in this case, the original image can never be recovered.

- **Block-based DCT[2] domain cryptography/encryption/scrambling approach**

  Encryption approaches operate either before (e.g., [47]), during (e.g., [51, 52]) or after (e.g., [53]) compression, denoted pre-, in- and post-compression encryption algorithms. Regarding pre-compression encryption, we will not recover the exact original encrypted values due to the lossy compression therefore we cannot fully decrypt them. Post-compression encryption requires an additional step to make sure that the generated bitstream is decodable by a conventional decoder, but it is too complex and has a little added value. Therefore, most of video/image encoding schemes operates during the compression and usually the encryption is executed after the quantization of DCT coefficients and before the entropy encoding, thus, in the DCT domain. For instance, in [52], authors propose a computationally efficient and secure video encryption algorithm by encrypting the DCT coefficient within the MPEG framework.

  - **Within the JPEG standard:**

    Boult, in [47], introduces an encryption algorithm using DES and AES during compression to protect the privacy in JPEG images. The information required for the decrypting process is stored inside the JPEG file header. This information cannot be used without the private key. Chattopadhyay and Boult [46] used this technique for real-time encryption, using uCLinux (i.e., a version of Linux kernel) on the Blackfin DSP architecture (i.e., embedded processors). Finally, in [54], the JPEG image is divided into two parts, the first one is public and the second one is private. The public part of the image is unaltered, whereas in the private one, the most significant DC coefficients are encrypted (similar to the previous encryption) during the encoding process after the quantization step. These previous approaches generate noisy images, therefore they are not suitable for surveillance applications.

    In [6], Yuan and al. create a JPEG scrambling framework with several levels of privacy protection. They propose to flip randomly the signs of the AC[3] coefficients of all YUV components for the weakest level of privacy protection. For medium-level, both DC and AC coefficients of only luminance (Y) component are modified while for the high-level of privacy protection, both DC[4] and AC coefficients of all YUV components are changed. Finally, for ultra-high-level, they apply a bitwise XOR operation between each DC value and a pseudo-random number in addition to the AC coefficients scrambling.

[2] DCT: Discrete Cosine Transform (contains DC and AC coefficients)
[3] The AC coefficients represent color variations across the block.
[4] The DC coefficient represents the average color of the block.

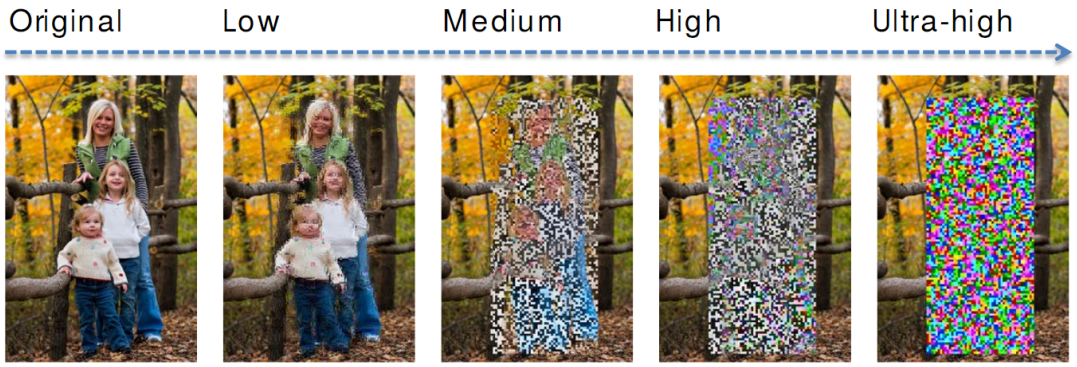


FIGURE 2.7: A JPEG scrambling framework with different levels of privacy protection: Low, Medium, High and Ultra-high. Reprinted from [6].

In [55], authors propose three scrambling-embedding reversible methods for JPEG. One of these methods operates only on the DC coefficients while the other two consider only the AC coefficients.

– **Within the H.264/AVC standard:**

The most currently known standard for video compression is H.264/AVC. We provide more details about this standard in the Appendix. Each block of an image is predicted from previous ones and the difference between the predicted block and the real one, called the residual block, is transformed, quantized and encoded. The following methods, including the one suggested in this thesis, use the non-encrypted blocks to compute these difference (i.e., the motion compensation) before being scrambled.

In H.264/AVC, encrypting blocks directly in the privacy region of a video frame will result in drift error in the non-privacy region due to intra and inter prediction from the privacy region. Therefore, Tong, Dai et al. [51] propose two main methods to prevent such drift error: Mode Restricted Intra Prediction (MRIP) and Search Window Restricted Motion Estimation (SWRME). MRIP is used to prevent the drift error caused by intra prediction when applying the scrambling on the blocks of the RoI. SWRME can remove the drift error due to inter prediction. To prevent the drift error caused by intra prediction when applying the scrambling on the blocks of the RoI, the fundamental idea in the MRIP technique is to restrict the possible intra prediction modes for blocks around the boundary of the privacy region. The excluded intra prediction modes are listed in Tables 2.1 and 2.2. The principle of SWRME consists in forbidding the use of any block in the privacy region of the reference frame to predict a block in the non-privacy region of the current frame. For example, in the Figure 2.8, it is forbidden to use any block in the privacy region of the reference frame, such as B, to predict a block in the non-privacy region of the current frame, such as block A.

Not all existing approaches include a mechanism that prevents this degradation of non-private areas.

In [56], authors design a video privacy protection scheme compliant with the H.264/AVC standard by leveraging the compression sensing (CS) theory.

TABLE 2.1: The excluded intra 4*4 prediction modes of a 4*4 block if its adjacent block is within the privacy region.

Position	The excluded intra prediction modes
T	DC, diagonal down left, diagonal down right, vertical left, vertical right and horizontal down
L	Horizontal, DC, diagonal down right, vertical right, horizontal down and horizontal up
TL	DC, diagonal down right, vertical right and horizontal down
TR	Diagonal down left and vertical right

TABLE 2.2: The excluded intra 16*16 prediction modes of a 16*16 block if its adjacent block is within the privacy region.

Position	The excluded intra 16*16 prediction modes
T	Vertical, DC and plane
L	Horizontal, DC and plane
TL	Plane
TR	None

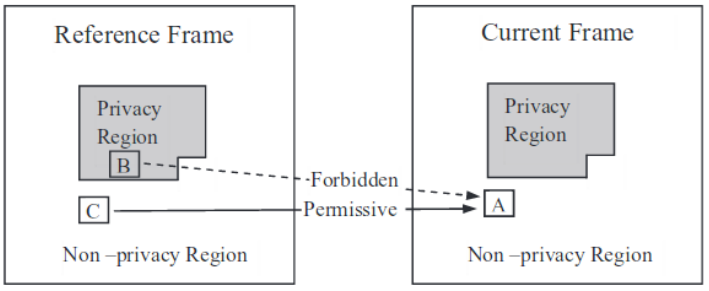
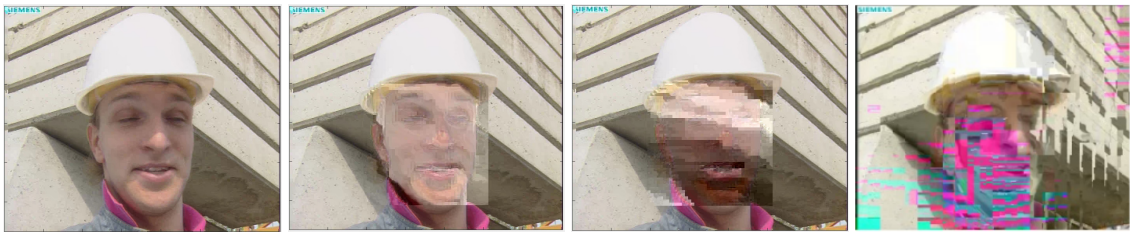


FIGURE 2.8: The inter prediction from the privacy region to the non-privacy region is forbidden.



(a) (b) (c) (d)

FIGURE 2.9: (a) The first frame of the foreman sequence, (b) when applying SNC [7], (c) SNC+IPM [8] and (d) SMV [9]. The last image is reprinted from [9].

Dufaux and Ebrahimi proposed pseudo-random permutation of the non-zero coefficients of each residual block of the private region or pseudo-random inversion of their sign within the MPEG-4 [57] framework and the H.264/AVC [7] standard. In both approaches, [57] and [58], the insertion of the scrambled step is applied on the intra prediction frames of the encoder,

unscrambled data are used in the motion-compensated (MC) prediction loop. Sohn et al. [59] describe the same encryption system for scalable video coding (SVC). In [60], authors prove that in some cases, encrypting AC coefficient signs only is not enough to protect the privacy. Indeed, it produces a relatively weak scrambling effect, especially for high-resolution images.

To enhance the scrambling effect in privacy protection, Wang et al. [8] encrypt the intra prediction modes (IPM) in addition to the signs of the non-zero coefficients (SNC) within the private areas. They also propose a spiral binary mask mechanism to reduce the bit rate overhead incurred by flagging the position of the privacy region. Su et al. [61] directly modify the intra prediction modes (IPM) as well as the motion vector differences (MVD) while embedding their original information in the AC coefficients. Khlif et al. [9] scramble the signs of motion vectors (SMV) using a chaotic cryptography algorithm. In [44], authors prove that for all scrambling schemes that they have tested, even if the motion vector information was not encrypted, the privacy is still protected. However, to have a very high level of privacy protection, they propose to scramble the motion vector. Finally, Peng et al. [62] encrypt the SNC, the MVD and the IPM based on flexible macroblock ordering (FMO) and chaos.

Contrary to the approach suggested in [57] that encrypts only SNC, these previously mentioned methods produce a strong scrambling effect yielding to noisy pictures which hamper the monitoring.

Coefficient scrambling is a common encryption technique widely used within the context of DCT-based compression formats. When we only encrypt the AC coefficients, the privacy is not sufficiently protected, particularly for images of high resolution because the DC coefficients of the residual blocks mainly keep the average color of each block. On the contrary, the images seem to be too noisy when we encrypt both, DC and AC coefficients, which hampers the surveillance. Therefore, encryption and scrambling methods present some limitations in managing the trade-off between the privacy protection and the data utility preservation.

In many works only the face is obscured, but that is not enough to protect visual privacy. We can identify a person with other elements from the image, for instance, using soft attributes like clothes, height, gait [15]. For instance, in a pair-wise constraints identification [63, 64] where faces had been masked, observers were able to identify a person from two different images. In this study, the accuracy of recognition was higher than 80%. Therefore, in this thesis we apply privacy protection methods that protect the identity from the face as well as from the body.

In addition, there is a trade-off between providing privacy and utility of surveillance. For instance, when an image is degraded to protect privacy, information needed for image understanding may also be removed. In other words, privacy protected images have to retain useful information needed for the surveillance.

Most of the works cited previously could not fulfill all the criteria required in privacy protection preserving surveillance. In the next section, we describe these specific criteria and we summarize in the Table 2.3 which of them are fulfilled by the existing privacy filters.

2.5 Criteria and metrics to assess the efficiency of privacy protection preserving surveillance approach

Filter	Privacy protection	Data Utility	Reversibility	Security	Compression	Real Time
Pixelization/Blur	Yes	Yes	No	No	Yes	Yes
Noising	Yes	No	Yes	No	No	Yes
Blacking out/Warping [29]	Yes	No	No	No	Yes	Yes
Morphing [27]	Yes (Only face)	No	No	No	Yes	Yes
GARP-Face [5], Photorealistic [1]	Yes (Only face)	Yes	No	No	Yes	No
Kmeans [25]	Yes	Yes	No	No	Yes	Yes
Color Transformation[3]	NA**	Yes	Yes	No	No	Yes
Inpainting[3]	Yes	No	Yes	Yes	No	Yes
K-Same family [34]	Yes (Only face)	Yes	Yes	No	Yes	No
Body shape filling [31]	NA**	NA**	Yes	Yes	Yes	Yes
One to several [32], PICO[46, 47], Subband scrambling[48]	Yes	No	Yes	Yes	Yes	Yes
Block-based pixel scrambling [50]	Yes	Yes	No	No	Yes	Yes
Block-based DCT domain scrambling, only AC coefficients [6, 57]	No	Yes	Yes	Yes	Yes	Yes
Block-based DCT domain scrambling, DC+AC coefficients [6, 57]	Yes	No	Yes	Yes	Yes	Yes
StegoScrambling * (Chapter 5)	**Yes**	**Yes**	**Yes**	**Yes**	**No**	**Yes**
ASePPI * (Chapter 6)	**Yes**	**Yes**	**Yes**	**Yes**	**Yes**	**Yes**

*Our proposed methods, **NA: not always; very dependent on many other algorithms (e.g., body contour extraction, face recognition)

TABLE 2.3: Summary of privacy protection methods according to the six criteria that are described in [11] and summarized in this Chapter VS the proposed methods of this thesis (in **bold**).

2.5.1 Criteria for an ideal privacy protection filter in surveillance

As, first, listed in [11], an ideal privacy filter, should meet the following six criteria:
i) Privacy protection (i.e., no possibility to recognize people from the face and neither from the body), depending on the cultures or the application, what needs to be protected and what is private are considered differently (some attributes may also be considered as private such as gender, age, clothes),
ii) Data utility preservation, also known as Intelligibility, which means keeping a fair visual quality to be able to recognize events or actions in the scene,
iii) Reversibility, which is the possibility to recover the original images or videos,
iv) Security of the process, the reversibility should be available only for authorized people, and the privacy protection should be robust against attacks,
v) Compression, compliant with the widely used coding standards (e.g., JPEG, Mpeg, H.264/AVC),
vi) Real time processing, required for some applications.

Privacy protection and security of the process (i.e., *i* and *iv*) permit the evaluation of the efficiency of the method in terms of privacy protection, whereas the four other criteria (i.e., *ii, iii* and *v*) determine the efficiency of the method in terms of compatibility with the surveillance.

The mandatory criteria considered for this field are privacy protection on both, face and body, the data utility preservation and also the reversibility (i.e., *i*, *ii* and *iii*). These two last ones are crucial for the safety of people. Indeed, in case of an incident, the identity of a person must be revealed. According to the Table 2.3, no one of the existing methods fulfills these three mandatory criteria.

Existing anonymization methods focus only on few criteria, depending on the exact application as the aforementioned ones introduced in the previous section. Indeed, they fail to satisfy all criteria, as it is shown in Table 2.3.

2.5.2 Objective VS Subjective evaluation

- **Convolutional neural networks (CNN)**

  Basic notions in visual neuroscience [65] lead to the creation of the convolutional and pooling layers in CNN. From 1990s, CNN is used for object detection in natural images (e.g., faces and hands [66]) and for face recognition [67].

  In images, deep neural networks exploit the property of local combinations of edge pattern motifs which can be regrouped into parts. Firstly, local groups of pixels are often highly correlated in images, forming distinctive local patterns that are easily detected. Secondly, the local statistics of images are invariant to location.

  The architecture of a typical CNN is structured as follows: the first few stages are composed of two types of layers: convolutional layers and pooling layers. Units in a convolutional layer are organized in feature maps and each unit is connected to local patches in the feature maps of the previous layer through a set of weights called a filter bank. The result of this filter bank is then passed through a non-linearity such as ReLU. The same filter bank is connected with all units in a feature map. Mathematically, a discrete convolution is the filtering operation performed by a feature map.

  The role of the convolutional layer is to detect local correlations of features from the previous layer, and the role of the pooling layer is to merge semantically similar features into one. The maximum of a local patch of units in one feature map is computed in a typical pooling unit. Usually two or three stages of convolution, non-linearity and pooling are cumulated. Finally, back-propagation gradients allow all the weights in all the filter banks to be trained.

  Nowadays, it exists some CNN architectures such as AlexNet-CNN described in detail in [68]. We generally use AlexNet-CNN as a pre-trained model that we fine-tune it for a specific task.

- **Crowdsourcing**

  Crowdsourcing has shown to be a viable alternative to conventional laboratory-based subjective assessments, especially for cognitive tasks [69]. Crowdsourcing-based evaluation of privacy protection methods for video surveillance has shown good consistency with laboratory-based studies [70]. The crowdsourcing methodology benefits from a large number of participants and at a relatively low cost without requiring a significant commitment from subjects, which are called workers in the crowdsourcing terminology. Workers accept to undertake a task (usually 5-20 minutes per task) and are grouped into larger units, called batches. When the evaluation experiment is over, workers submit their answers. Unlike laboratory-based experiments, crowdsourcing cannot impose specific displays or controlled illumination of surroundings in which assessments take place. However, since standard environment and equipment conditions for surveillance operators have not been established, typical monitors even with different resolutions and color settings are considered as appropriate.

  To display video sequences to different workers and to collect evaluation results, QualityCrowd2 framework [71] and the Microworkers crowdsourcing platform are often selected in order to access online workers from around the world. QualityCrowd2 is an open-source framework designed for quality of experience (QoE) evaluation with crowdsourcing.

In privacy protection domain, we evaluate the criteria either with subjective evaluations that rely on human perception or with objective evaluations based on mathematical operations. We prove, in the Chapter 3, that subjective and objective evaluations obtain similar results, thus, all the experiments of this thesis are done objectively (i.e., using algorithms).

The application of a privacy protection filter preserving surveillance should decrease the accuracy of identity recognition, whereas it should not disrupt the recognition of action and not degrade the quality of the image.

2.5.3 Privacy assessment

Lately the impact of privacy protection tools has been analyzed in surveillance and effective evaluation methodologies [72] have been developed to take into account both the context and the content.

Face identification evaluation

The objective evaluation of several primitive privacy filters has been reported by Newton *et al.*[24], where the authors have shown that these filters cannot adequately prevent efficient face recognition algorithms. The robustness of face recognition and detection algorithms towards primitive distortions is also reported in [73]. Further, Dufaux *et al.* [74] define a framework to evaluate the performance of face recognition algorithms applied to images altered by various privacy protection methods.

In this thesis, we have selected face recognition algorithms in order to objectively evaluate the level of privacy protection. The goal in the privacy protection domain is to decrease the performance of a face recognition tool (i.e., the accuracy should be the closest as possible to zero).

LBPH [75], Eigen [76] and HoG [17, 77] compute the features representation and a SVM classifier trains a model from these features. We describe these methods in the following. Note that for all these face recognition algorithms all images should be of the same size.

- **Histogram of Oriented Gradient**, referred to as **HOG** in this thesis, is widely used in pedestrian recognition [17] as well as for faces [77] and object recognition [78]. First, HOG computes gradients of an image, then computes a histogram of oriented gradients (between *0* and *180°* with *9* bins) on each subpart of an image defined by size of a cell (*[8, 8]* in our experiments) and concatenates all histograms.
- **Local Binary Pattern Histogram** [75], referred to as **LBPH** in this thesis, is often used to recognize texture since LBPH enables to capture very fine grained details in images. A mathematical description of the LBP operator can be given as:

$$LBP(x_c, y_c) = \sum_{p=0}^{P-1} 2^p s(i_p - i_c) \tag{2.6}$$

where (x_c, y_c) is the central pixel with intensity i_c, and i_p being the intensity of the neighbor pixel. s is the sign function defined as:

$$s(x) = \begin{cases} 1 & \text{if } x \geq 0 \\ 0 & \text{else} \end{cases} \tag{2.7}$$

By definition the LBP operator is robust against lighting transformations.

- **Principal Component Analysis (PCA)** [76], referred to as **Eigen** in this thesis, performs face recognition by:

  i) Projecting all training samples into the PCA subspace.
  ii) Projecting the query image into the PCA subspace.
  iii) Finding the nearest neighbor of the projected query image between the projected training images.

- **Support vector machines (SVMs)**

  SVMs analyze data used for classification and regression analysis. Given a set of training examples, each example marked as belonging to one of the two categories, a SVM trains a model, and then assigns new examples to one category or the other. A SVM model is a representation of the examples as points in space, mapped so that the examples of the separate categories are divided by a clear gap that is as wide as possible. New examples are then mapped into that same space and predicted belonging to a category based on which side of the gap they fall.

  In addition to performing linear classification, SVMs can efficiently make a non-linear classification using what is called the kernel trick, implicitly mapping their inputs into high-dimensional feature spaces.

The biometric community has often applied LBPH, Eigen and even HOG with a linear SVM as baseline algorithms to compare the impact of privacy filters on face recognition [77, 79], but also to estimate face recognition before and after a super-resolution method [80]. They train and test the performance of these algorithms using the three following databases: FERET [81], AT&T [82] and SCFace [83] detailed in the following.

- **The FERET Database [81]**

  The FERET database contains many facial images that were gathered independently from the algorithm developers. To maintain a degree of consistency throughout the database, the same physical setup was used in each session, we can only perceive minor variation in images collected on different dates. They collected the FERET database in 15 sessions between August 1993 and July 1996. This leads to 14,126 images that includes 1199 individuals. They also make a second set of images of a person already in the database, usually taken on a different day and for some them, over two years had elapsed between their first and last sittings. This enables researchers to study the changes in a subject's appearance that occur over a year.

- **AT & T - The Database of Faces [82]**

  This database contains a set of face images taken between April 1992 and April 1994 at the AT&T Laboratories, Cambridge. There are 40 distinct subjects, and 10 different images for each subject are available. All the subjects are in an upright and frontal position. The background of all pictures is dark plain. Some images were taken at different times, varying lighting, facial expressions (open/ closed eyes, smiling / not smiling) and facial details (glasses / no glasses).

- **SCFace - Surveillance Cameras Face Database [83]**

  SCFace is a database containing 4160 static images of human faces (in both visible and infrared spectrums) of 130 subjects that were taken in an uncontrolled indoor environment (i.e., in real-world conditions). Five video surveillance cameras of various qualities were used to capture the images.

Recently, with the emergence and the success of Deep Learning algorithms, many face recognition tools have been created using convolutional neural network (CNN) such as OpenFace [84][5].

- **OpenFace** is a face recognition tool in Lua using Torch library. This face recognition tool is trained with a combination of the two largest publicly available face recognition datasets: FaceScrub [85] and CASIA-WebFace [86]. Using this pre-trained neural network model, we test the performance on the LFW Face Database [87], that contains unconstrained face images or on the YouTube Face Database [88] that contains unconstrained face videos. The following describes the steps used for the experiments of this thesis.
  - Detect faces with a pre-trained model from dlib or OpenCV.
  - Align the face. The dlib's real-time pose estimation with OpenCV's affine transformation makes the eyes and the bottom lip appear in the same location on each image.
  - Use FaceNet's architecture [89] based on GoogLeNet style Inception models.
  - Split randomly 10 times the data into a training and testing set. We choose 75% of the database for the training and 25% for the testing.
  - Lastly, we classify with a linear SVM.

- **LFW Face Database [87]**

  Labeled Face in the Wild (LFW) is an unconstrained face database containing 13.233 images from 5.749 individuals. It contains various possible daily conditions of people (i.e., the pose, lighting, race, accessories, occlusions). The background of its images are in natural settings.

- **YouTube Faces Database [88]**

  This database was designed for studying the problem of unconstrained face recognition in videos. It contains 3.425 videos of 1.595 different people. All the videos were downloaded from YouTube. The average number of videos for each subject is 2.15. The shortest clip duration is 48 frames, the longest clip is 6.070 frames, and the average length of a video clip is 181.3 frames. They follow

[5] https://cmusatyalab.github.io/openface/

the example of the LFW Face database to design this collection of videos and the benchmark associated. The specific goal of the YouTube database is to produce a large collection of videos along with labels indicating the identities of a person appearing in each video.

Gender detection evaluation

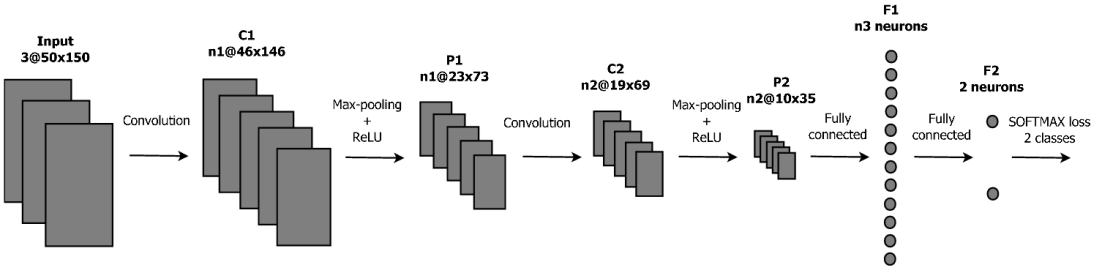


FIGURE 2.10: Architecture of a CNN.

CNNs have recently become the standard of excellence for object detection in natural images, but also for object recognition [68] [90]. Today, CNNs are used for the large variety of computer vision tasks, such as face detection and recognition, gender detection [91].

To evaluate the pedestrian gender recognition, we select the CNN architecture, presented in [92] and in the Figure 2.10. This CNN has 2 convolutional layers (C1 and C2) with 5x5 kernels, each of these layers is followed by max-pooling with a stride of 2 pixels (P1 and P2) and Rectified Linear Unit (ReLU) activations. The model is concluded by 2 fully connected layers (F1 and F2). The F2 layer has 2 neurons (corresponding to the number of classes). The loss is computed by a softmax loss function. As it is depicted in Figure 2.10, this CNN architecture takes three 50x150-dimensional feature maps (red, green and blue channels of an image) as an input. The 1st and 2nd convolutional layers C1 and C2 have n1 and n2 feature maps respectively. They set n1 = n2 = 20 and n3 = 100. In order to augment the training data, they use each training image alongside with its mirrored copy.

After training this CNN, we use the obtained weights to calculate the values of neurons in the F1 layer for all testing images. These neuron values serve as features for the testing images, thus, representing an image by either 100-dimensional vector (using this CNN architecture).

The model is trained on the PETA Database, and a SVM is used for classification. In the experiments of this thesis we have selected two pedestrians databases: PETA [93] and HID [94].

- **Composition of PEdesTrian Attribute (PETA) [93]** contains various people in diverse environments and situations. PETA is a large open-access collection of pedestrian images with several annotations, including age, backpack, hat, jacket, jeans, logo, long hair, gender, muffler, no accessory, plastic bag, sandals, shorts, skirt, sunglasses, trousers, T-shirt, etc. Examples of PETA images are presented in the Figure 2.11.

FIGURE 2.11: PETA dataset.

Originally, the PETA collection consists of 10 datasets of different sizes with a total amount of 19,000 images. Appearances of images significantly vary between different datasets of PETA in terms of image resolutions (from 17x39 to 169x365 pixels), camera angles (pictures are taken either by ground-based cameras, or by surveillance cameras which are set at a certain height) and environments (indoors or outdoors), as shown in the Figure 2.12.

Datasets	#Images	Camera angle	View point	Illumination	Resolution	Scene
3DPeS	1012	high	varying	varying	from 31x100 to 236x178	outdoor
CAVIAR4REID	1220	ground	varying	low	from 17x39 to 72x141	outdoor
CUHK	4563	high	varying	varying	80x160	outdoor
GRID	1275	varying	frontal&back	low	from 29x67 to 169x365	indoor
i-LIDS	477	medium	back	high	from 32x76 to 115x294	indoor
MIT	888	ground	back	high	64x128	outdoor
PRID	1134	high	profile	low	64x128	outdoor
SARC3D	200	medium	varying	varying	from 54x187 to 150x307	outdoor
TownCentre	6967	medium	varying	medium	from 44x109 to 148x332	outdoor
VIPeR	1264	ground	varying	varying	48x128	outdoor
Total = PETA	**19000**	**varying**	**varying**	**varying**	**varying**	**varying**

FIGURE 2.12: Variation in the PETA dataset.

Thanks to the proposed annotations, we can evaluate gender recognition accuracy on the PETA collection of datasets. Unfortunately, PETA contains many similar images of the same person which can considerably bias the resulting prediction rate. For this reason, we remove these recurrence from PETA as well as the images with very low resolutions (when height is less than 120 pixels or width is less than 40 pixels), images with more than one person, and images of babies in strollers. Finally, we are left with 8365 images which is less than half of the initial size of PETA.

- **The Southampton Human ID (HID)** at a distance gait database contains a large ($\sim$ 150 subjects) and small databases ($\sim$ 10 subjects). In this thesis, we only use the small one because we focus on the gender detection. The subjects in the small database walk at different speeds in a laboratory environment wearing a variety of clothes, bags. They are recorded from four different angles.

2.5.4 Evaluation of the Utility preservation of the surveillance

Actions classification

Deepdetect[6] is an efficient tool available online that provides the following image classification models: clothing, bags, footwear, buildings, fabric, gender, age, furnitures, sports and trees. Therefore, we use this tool to evaluate the sports event classification on the UCF Sports dataset [95] presented below. Deepdetect has been trained with 143 sports included nine of the ten sports available on the UCF dataset. Thus, the testing set contains only the nine following categories: acrobatic gym (swing-bench and swing-side), walking, riding a horse, diving, golf, football (kicking), skateboarding, lifting and running.

UCF Sports dataset contains videos of 10 different actions featured in a wide range of scenes and viewpoints, and typically collected from broadcast television channels such as the BBC and ESPN. The dataset includes a total of 150 sequences with the resolution of 720 x 480. The figure 2.13 shows a sample frame of all ten actions, along with their bounding box annotations of the humans shown in yellow.

FIGURE 2.13: UCF Sports dataset.

Image quality measures

The application of privacy protection methods can alter the image quality of the recovered images (when the inverse process of a privacy filter is applied in order to recover the original image). Therefore, we need to assess the quality of recovered images.

[6] http://www.deepdetect.com/

Two metrics have been selected to measure the visual quality of images, the **peak signal-to-noise ratio**, (PSNR) and the **structural similarity** (SSIM). The performance of both, SSIM and PSNR is correlated with human judgments. They are commonly used to evaluate the quality of image reconstruction.

The community of that domain (privacy protection preserving surveillance) uses additional metrics, the **luminance similarity score** (LSS) and the **edge similarity score** (ESS) [96]. We can also use all these metrics to assess the amount of degradation that a privacy protection method induces.

- **The Peak Signal-to-Noise Ratio (PSNR)** measures how much the signal has been corrupted (i.e., the level of degradation of a signal) and is commonly used to evaluate the quality of reconstruction. The higher is the PSNR the better is the quality of the reconstruction.

$$PSNR(I', I) = 10log_{10}(\frac{255^2}{\frac{1}{m*n}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}(I'(i,j) - (I(i,j))^2)}, \tag{2.8}$$

  where I and I' are, respectively, the reference and the alternated images, with a size of $m * n$.

- **The Structural Similarity (SSIM)** measures the similarity between two images. The closer the SSIM is to the value one, the greater the similarity iss. Such objective metric has been designed to be more consistent with the characteristics of the human vision system, with respect to traditional metrics such as PSNR [97].

$$SSIM(Iw', Iw) = \frac{(2\mu_I\mu_{I'} + C_1)(2\sigma_{II'} + C_2)}{(\mu_I^2 + \mu_{I'}^2 + C_1)(\sigma_I^2 + \sigma_{I'}^2 + C_2)}, \tag{2.9}$$

  where Iw and Iw' are two windows, respectively, in the reference and the alternated images, μ_I and $\mu_{I'}$ are the average of Iw and Iw', σ_I^2 and $\sigma_{I'}^2$ are the variance of Iw and Iw' and C_1 and C_2 are two constants to avoid instability.

- **The Edge Similarity Score (ESS)** measures the degree of resemblance of the edge and contour information between two images.

- **The Luminance Similarity Score (LSS)** measures the dissimilarity in luminance between two images.

2.5.5 Robustness against attacks

In this section, we assume that attackers know the regions of interest and the steps of the target privacy protection method.

Brute Force Attack (BFA)

A BFA consists of an attacker trying many combinations with the hope of possibly guessing correctly. In this thesis, instead of evaluating the security of the key (assuming that it is secure enough), we apply the brute force attack to the coefficients because they are encrypted using permutations. The attacker

systematically checks all possible coefficient combinations knowing the algorithm until the correct one is found.

Replacement Attack (RA)

RA [53] implies to set all encrypted values to zero while keeping the unencrypted values. For image and video compression, it consists in extrapolating the encrypted data by error compensation from the previous block using the prediction modes or the motion vectors that are, both, available to the attacker. Thus, this attack confirms that we cannot recover all encrypted blocks by predicting from previous ones.

Parrot Attack (PA)

PA consists to train a model (e.g., an identity recognizer) and test it on images on which the same privacy protection has been applied. Indeed, with the success of deep learning algorithms, it has been proven that if we train an identity recognizer with the degradation applied on images (i.e., the privacy protection method), the identity may be recognized [98].

Chapter 3

Objective VS Subjective evaluation of gender recognition with privacy protection filters

3.1 Introduction

Deep learning-based algorithms have become increasingly efficient in recognition and detection tasks, especially when they are trained on large-scale datasets. Such recent success has led to a speculation that deep learning methods are comparable to or even outperform human visual system in its ability to detect and recognize objects and their features. In this Chapter, we focus on the specific task of gender recognition in images when they have been processed by privacy protection filters (e.g., blurring, masking and pixelization) applied at different strengths. Assuming a privacy protection scenario, we compare the performance of a deep learning algorithm with a subjective evaluation obtained via crowdsourcing to understand how privacy protection filters affect both machine and human vision.

3.2 Objective VS Subjective gender evaluation

This section provides a detailed description of a subjective and objective evaluation of the gender recognition task when applying privacy protection filters. We have selected a Convolutional Neural Networks (CNN) for the objective evaluation, and we did a Crowdsourcing for the subjective one. We use the PETA collection of databases (a complete description of this database is done in the section 2.5.3).

3.2.1 CNN-based gender recognition

Following the work [92], we employ a convolutional neural network for the objective evaluation. In particular, we adopt an architecture proposed by Krizhevsky et al. [68], often denoted as AlexNet. This architecture is presented in Figure 3.1. It consists of five convolutional layers and three fully connected layers. We only fine-tune the model to recognize genders of pedestrians. Therefore, we train practically the same architecture on PETA dataset. The only difference is that in the last fully connected layer we use 2 neurons instead of 1000, since we only have two target classes (men and women).

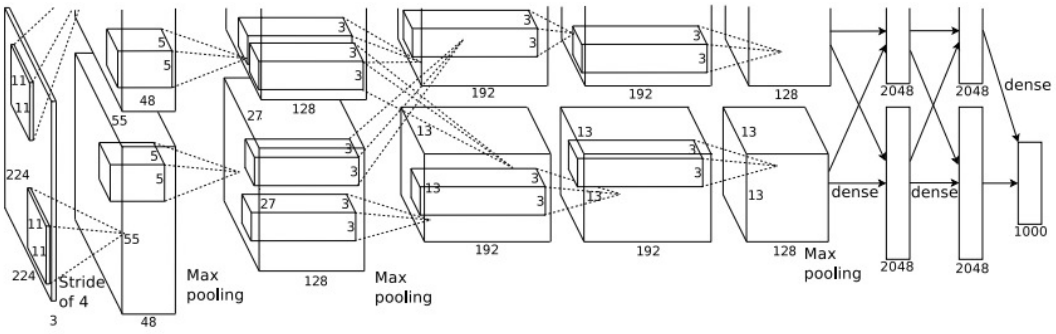


FIGURE 3.1: CNN model.

In order to generalize our CNN on heterogeneous and even completely unseen datasets, we firstly train CNN on the dataset which is composed of training parts (90 %) of CUHK, PRID, GRID, MIT and VIPeR taken together (4488 Male and 2744 Female pedestrian images). Thus, our model is tested on a single big dataset composed of images from the 10 datasets which have not been used in the training set (160 Male and 161 Female pedestrian images). The Table 3.1 summarizes the exact number of training and testing for each dataset.

Dataset	Train size (male + female)	Test size (male + female)
CUHK	$3432 = (2420 + 1012)$	$73 = (34 + 39)$
PRID	$942 = (449 + 493)$	$36 = (16 + 20)$
GRID	$928 = (531 + 397)$	$23 = (10 + 13)$
MIT	$792 = (532 + 260)$	$69 = (41 + 28)$
VIPeR	$1138 = (556 + 582)$	$48 = (24 + 24)$
3DPeS	0	$15 = (10 + 5)$
CAVIAR	0	$9 = (3 + 6)$
i-LIDS	0	$6 = (3 + 3)$
SARC3D	0	$5 = (3 + 2)$
TownCentre	0	$37 = (16 + 21)$
Total	$7232 = (4488 + 2744)$	$321 = (160 + 161)$

TABLE 3.1: Split between training and testing parts per dataset.

We apply five different privacy filters at three different strength levels for each, enunciated and illustrated in the Table 3.2 and in the Figure 3.2. We deeper explain each of this privacy methods, in the sections 2.3 and 2.4.

Filter	Parameter	Strength
Black Masking	opacity	0.5, 0.7, 0.9
Morphing	opacity	0.4, 0.7, 0.9
Pixelization	size of squares	3, 5, 7
Gaussian Blur	standard deviation	2, 4, 6
Kmeans	number of clusters	6, 4, 2

TABLE 3.2: Privacy filter with the strength used in our experiments.

FIGURE 3.2: Privacy filters. From left to right: original body image, images where we apply black masking of opacity 0.5, 0.7 and 0.9, morphing of opacity 0.4, 0.7 and 0.9, pixelization of squares size 3, 5 and 7, Gaussian blur of standard deviation 2, 4 and 6 and Kmeans with number of clusters of 6, 4 and 2.

3.2.2 Crowdsourcing Evaluation

The crowdsourcing assessment (a more precise description is given in the section 2.5.2) aims to check whether a person in a given image can be correctly identified as female or male by an individual, even after the application of a privacy protection filter. For this purpose, each crowdsourcing worker was asked to look at the image of a person and answer the question, “What is the gender of the person?”. The framework QualityCrowd2 performs subjective quality assessment with crowdsourcing. We have selected this tool because it is easily modified to fit our gender recognition task using the provided simple scripting language for batch creation (i.e., quiz) and training sessions.

In total, 300 random images from the PETA dataset were used in the crowdsourcing experiment. They were protected by five different privacy filters at three different strength levels, resulting in $300 \times 5 \times 3 = 4500$ images evaluated in this crowdsourcing study.

To ensure a statistically significant number of evaluations for each image, also taking into account the presence of unreliable subjects (about 50% in a typical crowdsourcing evaluation), 40 subjects were assigned to each image, with a total of 2652 subjects participating in the evaluations.

All versions of the images were randomly distributed among the batches; special care was devoted to guarantee that a particular content was used only once in each batch (i.e., each subject assessed only one version of a given content). Each batch starts with a guideline about what is required from the subject, followed by a training session describing the evaluation procedure. A display brightness test is performed using a method similar to the one described in [99] and permits to estimate the subjects' display settings. Subjects are not allowed to skip any sequence or to avoid answering any question.

Unlike lab-based subjective experiment where all subjects can be observed by experiment operators and its test environment can also be controlled, the major shortcoming of the crowdsourcing-based subjective evaluation is the inability to supervise participants behavior and to restrict their test conditions. When using crowdsourcing for evaluation, there is a risk of including unreliable data into analysis due to the wrong test conditions or unreliable behavior of some workers who try to submit low quality work in order to reduce their effort while maximizing their received payment [99]. For this reason, unreliable workers detection is an inevitable process in crowdsourcing-based subjective evaluation. To identify a worker as 'trustworthy', the following four factors were used in our experiment: *i)* Task completion time; *ii)* Mean observation time per question; *iii)* Observation duration deviation; *iv)* Number of minority answers.

The objective of the first three factors is to filter out the workers who have strange behaviors during their task, because they are either not serious or have poor concentration. The observation time per question is measured as the time from when the question is displayed until the time the answer is given by the worker. The task completion time, mean observation time and observation duration deviation can be calculated using this data. If the task completion time or mean observation time per question is too long compared to their averages of all workers, it can be deduced that they did not take the test seriously or were distracted during their tasks.

To reinforce our decision, unreliable workers were also identified using an approach similar to a typical outlier detection method, commonly used in most subjective quality evaluations. However, typical subjective tests use scoring methods like five-grade evaluation (assess the image quality), and outlier detection is performed on the mean opinion score [100]. Our experiments do not have opinion scores because of the specific privacy-oriented questions we used. Therefore the number of minority answers has been used for the outlier detection instead. The assumption is that a participant who has a lot of different answers compared to the majority of workers is unreliable.

The above unreliable worker detection methods filtered out 198 workers out of total 2652, resulting in 2454 scores used in the analysis.

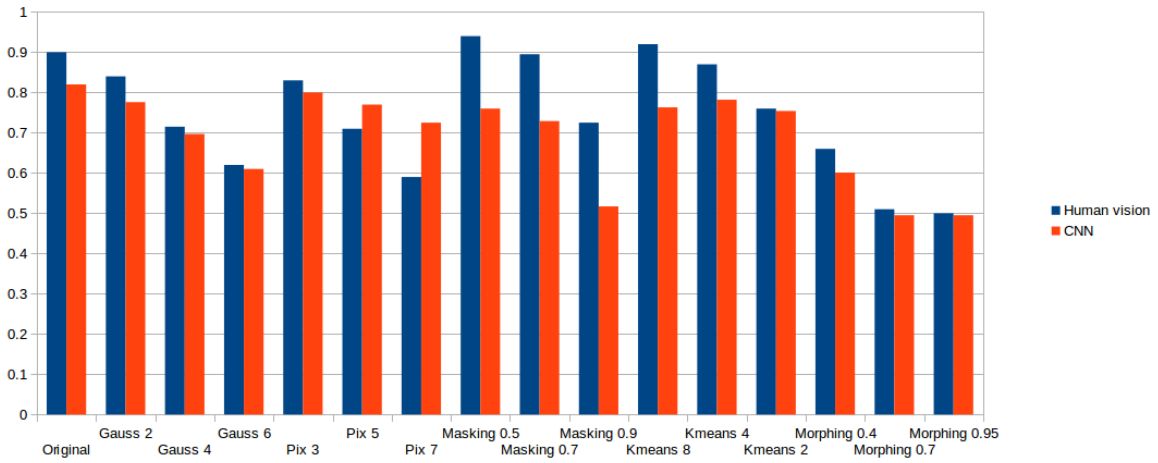


FIGURE 3.3: Accuracy results of human vision and CNN.

3.3 Results and Conclusion

In order to evaluate the performance of the CNN model, we use the mean average precision (MAP). An output of the CNN is binary (“male” or “female”) while the output of the crowdsourcing is ternary (“male”, “female” and “I don’t know”). Therefore, in order to fairly compare the results of the gender recognition based on CNN and crowdsourcing, we assume that 50% of answers “I don’t know” registered during the crowdsourcing had been correct if the response would have been selected randomly.

The Figure 3.3 illustrates the results of the CNN-based gender recognition as well as the ones of the crowdsourcing for original (i.e., no filter) and altered images (application of the privacy filters with different parameter values). CNN shows results close to those by crowdsourcing for Gaussian blurring, K-means and morphing (less than $\sim$ 10% of differences except for Kmeans 8). CNN is more robust to pixelization ($\sim$ 10% better than human vision). In the case of masking, the human vision is $\sim$ 15% better than CNN. Protection of the masking filter really depends on the brightness of the display and its environment. Indeed, some crowdsourcing participants might have increased the luminosity of their computer screens making the test images more visible, whereas the CNN works with values of pixels regardless of the display method. In addition, the human vision adapts to the change of brightness.

In this study, we demonstrated the impact of privacy filters on gender recognition by machine vision algorithms (using a CNN) and by human vision (using a crowdsourcing approach). One might expect the human vision to be more robust to privacy filters than computer vision. Nevertheless, our results show that humans and automatic gender recognition systems perform almost equally.

Chapter 4

Common face anonymization in visual data: are they really protecting our privacy?

4.1 Introduction

Broadcasting, printed media and some surveillance systems apply standard obfuscation methods to corrupt the pixel data and achieve anonymity. These methods are either a masking by a solid colored rectangle, a blurring [1], a pixelization [2], or a noise addition. A detailed description of these popular approaches is given in the section 2.3. In this Chapter, face images where we apply privacy filters, are referred as obscured face images.

The level of privacy protection is controlled by varying parameters. The experiments in [79] demonstrate that, in general, an increase in strength of privacy filters leads to an increase in privacy (i.e., reduction in terms of recognition rate). We illustrate, in the Figure 4.1, the methods that we selected for this study, and we sum up the name of their associated tuning parameters in the Table 4.1.

Even if the application of a privacy filter appears efficient at first sight, we demonstrate in this Chapter that as soon as the category and the strength of this filter are identified, there exist in the literature, current powerful approaches (e.g., by super-resolution techniques [22, 23], by parrot attack [24]), enable to partially cancel the impact of such filters with regards to automatic face recognition. Hence, evaluation is expressed in terms of face recognition rate associated with original, obscured and de-obscured face images.

The domain of image restoration allows to partially recover original face images, referred as de-obscured face images later in this Chapter. Indeed, image restoration improves quality of images or reconstructs

[1] smoothing the image with, e.g., a Gaussian filter with large variance
[2] image subsampling

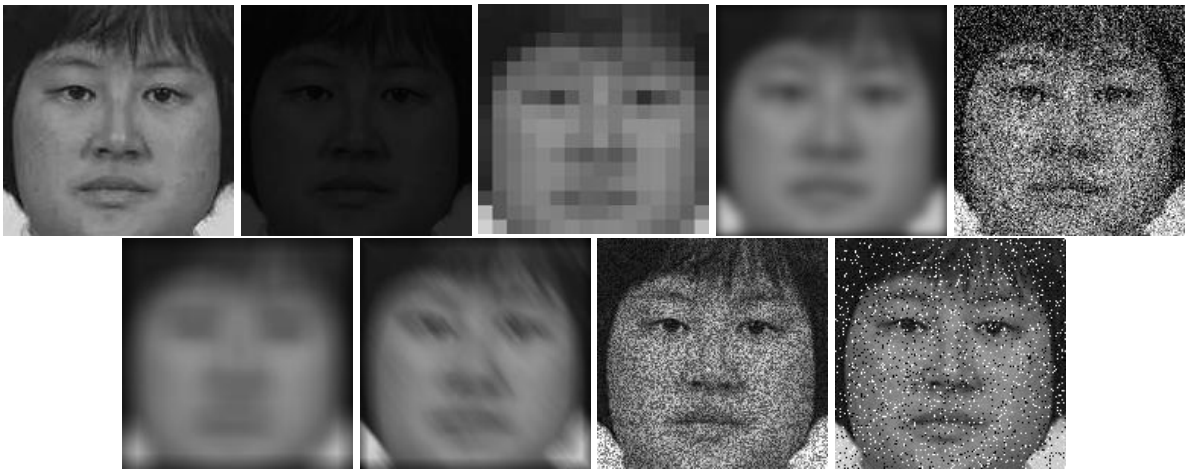


FIGURE 4.1: From left to right, on the top: original faces, black masking, pixelization, Gaussian blur, Gaussian noise; on the bottom: average blur, motion blur, Speckle noise, Salt and Pepper noise.

Filter	Parameter	Strength
Black masking	opacity	0.1, 0.2, 0.3
Pixelization	size of squares	3-10
Gaussian Blur	standard deviation	2-5, 8
Gaussian Noise	standard deviation	0.001, 0.005, 0.01-0.1, 0.3
A circular averaging neighbouring	square matrix of size 2*radius+1	11, 21, 31
Motion Blur	length and angle of the motion	[15, 45], [20, 130]
Speckle noise [101]	standard deviation	0.1, 0.3
Salt and Pepper Noise	noise density	0.1, 0.3, 0.5

TABLE 4.1: Privacy filters with the strength used and the name of their associated parameter. The four last ones are dedicated only for the testing.

corrupted images. Several methods like de-blurring [102], de-noising [103–105], super-resolution [106, 107] are potentially efficient, but require an a priori knowledge about the exact corruption.

To the best of our knowledge, there is no method that detects the category of the filter from obscure face images. Therefore, the key step of this Chapter consists of automatically identifying the category and the associated strength of the filter that have been used to obscure faces. Concerning this preliminary, but mandatory step, we propose the following approach. First, we select a method to classify obscured faces from not obscured faces. Then, a second approach classifies the type of filter. As soon as the filter is identified, a last step would consist of defining the strength. Supposing an a priori knowledge of the category and the strength of the filter, found in the previous steps, we finally demonstrate, in this Chapter, that a de-obscured face operation (i.e., image restoration methods) can be efficiently performed and therefore privacy filters become much less effective.

The rest of the Chapter is organized as follows: in the next section, we explain the proposed obscured face detection against the original face following by the categorization of the filter and the estimation of the filter strength. Then, we describe image restoration methods used to de-obscured face images. In the last section, we evaluate our proposed filter classification, and we demonstrate that the knowledge

of the category in addition to the one of the strength of the privacy filter improves the image restoration. Therefore, the performance of identity recognition on/of recovered faces increases significantly.

4.2 System overview

The proposed method consists foremost of differentiating obscured face images from not obscured faces (i.e., original faces), and then classifying the type of the filter used to obscure faces. We have selected the following methods for the classification: the Histogram of Oriented Gradient (HOG) features, the Principal Component Analysis (PCA or Eigen) or the Local Binary Pattern Histogram (LBPH) features with a linear SVM classifier. We explain these methods in the section 2.5.3. After this classification, we estimate the filter strength. Finally, knowing the type of the filter and his strength, we apply the associated de-obscured face method.

4.2.1 Detection of obscured face images

Original face images and obscured images generate several differences for the oriented gradients. Indeed, pixelization creates more block effects among direction to *0* or *90°*. For noise, all directions got almost the same frequency. Blurring and black masking create dominant orientations.

Therefore, we compute the HOG features with a linear SVM classifier to train a model which differentiates original faces from obscured faces of size *112*92* pixels (i.e., height*width).

4.2.2 Categorization of the filter

This step consists in detecting the type of the filter used to obscure face images. Eigen is more sensitive to light, scale and translation variations that leads to have less robustness against a black filter. Indeed, this filter modifies the darkness of images compared to other filters. LBPH, is employed to classify texture that is appropriate in the present case because privacy filters create some specific texture patterns except the black masking filter that removes textures by masking them.

So instead of making a classification between the black masking, the pixelization, the blurring and the noise filters, we first compute a classification with the Eigen features and a linear SVM between the black masking against all other filters, and then, a classification with LBPH features and a linear SVM to distinguish the three reminder filters.

4.2.3 Estimation of the filter strength

Restoration methods are efficient when not only the exact filter is a priori known, but also the strength that has been used. Therefore, we propose to automatically sort the strength of a filter by the ensuing listed approaches.

To obtain all the following formula (4.2, 4.3, 4.4), we first extract the features associated (i.e., $change\_size$, $edges$, $noise$, defined later) of 300 images taken randomly from Feret, ScFace and AT&T databases, and we find the functions that fit as much as possible the curves generated by the original strength (i.e., $squares\_size$, $standard\_deviation$, $standard\_deviation$) depending on the values of the associated features.

We determine/estimate the strength of the following privacy filters by finding the curve that fits the better as possible the values of original strengths.

On 600 images of Feret, ScFace and AT &T databases (different from the ones used to create the models), we compute the image's darkness (mean of image pixels), the $change\_size$, the $edges$ and the $noise$ and find the formula that better estimates the strength of the black masking (i.e., $opacity$), the pixelization (i.e., $squares\_size$), the blurring (i.e., $standard\_deviation$) and the noising (i.e., $standard\_deviation$). The four graphics in the Figure 4.2, compute the average and the standard deviation of the strengths estimation and compare the results with the original/real ones (the ground truth).

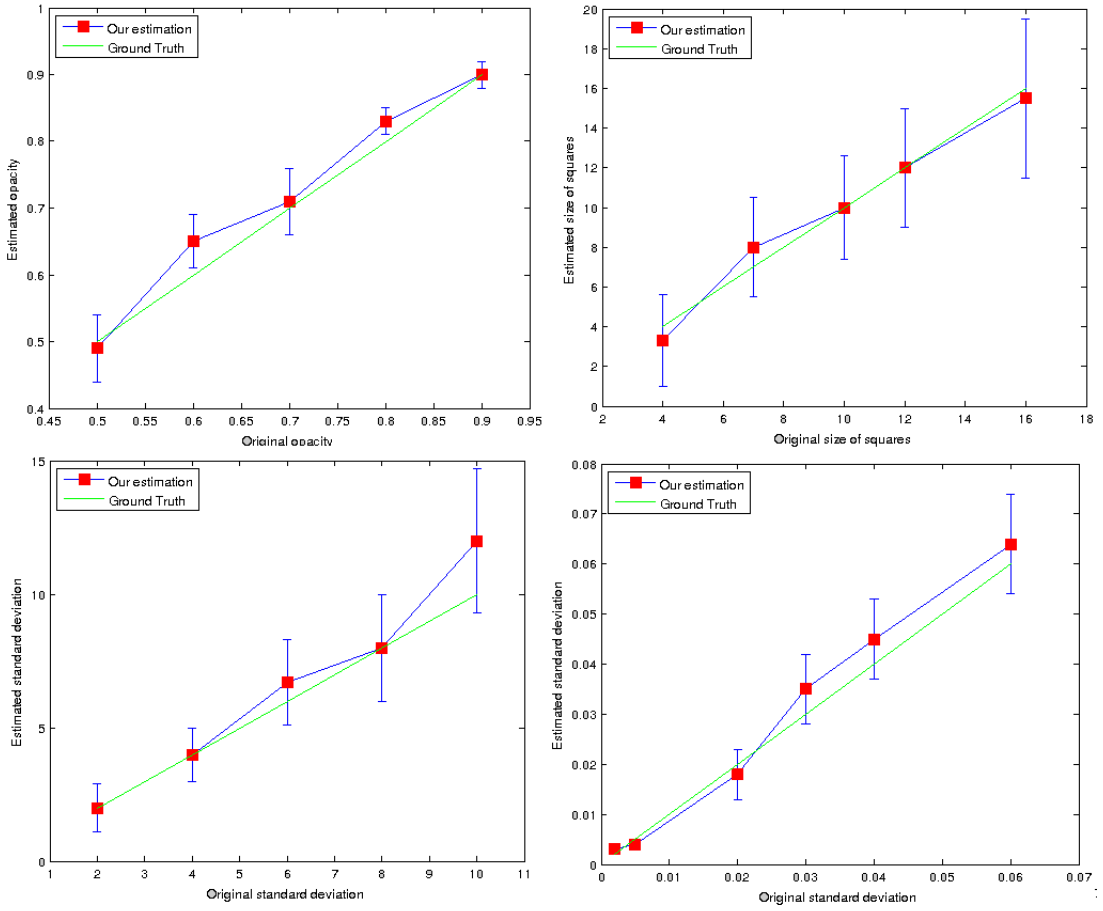


FIGURE 4.2: Results of the strength estimation for, respectively, the black masking, the pixelization, the blurring and the noising filters.

We notice that the predicted strengths do not always correspond exactly to the real ones. However, they provide a good estimation/approximation of the filter strength.

According to these experimentations, we estimate the strength of:

- The **Black masking** filter according to the image's darkness (mean of image pixels). We rewrite the formula of the black masking to find the opacity:

$$\begin{aligned} IblackMask(x,y) &= I(x,y) * (1-\alpha) + color * \alpha \\ \text{if } color \text{ is equal to zero, } &\Rightarrow IblackMask(x,y) = I(x,y) * (1-\alpha) \\ &\Leftrightarrow \alpha = 1 - \frac{IblackMask(x,y)}{I(x,y)} \\ \alpha &= 1 - \frac{mean(IblackMask(x,y))}{mean(I(x,y))} \\ \alpha &= 1 - \frac{mean(IblackMask(x,y))}{127} \end{aligned} \tag{4.1}$$

  where $color$ is the color of the solid rectangle that merges with the original image, and α representing the opacity. The function mean is the average of the intensity of the image. We assume that the mean of the original image should be around 127 because, in our case, the values of an image is between 0 and 255.

- The **Pixelization** filter by counting the number of pixels between each change of color horizontally and vertically, denoted $change\_size$. We average all the $change\_size$ found (it should be close to the size of the pixelization). Then, we find the following relation 4.2 between the size of the squares and the average $change\_size$.

$$squares\_size = -6.642 + 1.43 * mean(change\_size) \tag{4.2}$$

- The **Blurring** filter according to the percentage of edges. Point spread function (PSF) estimation for Image Deblurring [108] is mainly used, but unfortunately this method does not work for all types of blur, in particular motion blur. This is why, we have designed another approach for our work. We apply Canny edge detection and we compute the percentage of the edge in the image, denoted $edges$. Then, we find the following relation between the percentage of the edge and the strength of the blurring (i.e., $standard\_deviation$):

$$standard\_deviation = 16.17 * \exp(-37.8 * edges) \tag{4.3}$$

- The **Noising** filter according to the amount of noise in the detail coefficients of the Discrete wavelet transform (DWT) [103, 105]. First, we compute the wavelet decomposition of the image, two high-pass filters are applied to obtain the diagonal decomposition, named DWT_D, a first one on the rows and the second on the columns. We, then, threshold this diagonal decomposition (negative or positive values) to obtain either 0 or 1 (1 should represent the noise). Therefore, we count the percentage of 1 presented in DWT_D, denoted $noise$. According to this percentage, we estimate the strength of the noise (i.e., $standard\_deviation$) with the following formula:

$$standard\_deviation = 0.003 + \exp(94.35 * noise - 8.75) \tag{4.4}$$

To prove that our proposed filter categorization and strength estimation improve image restoration methods, we test several face recognition algorithms after basic image restoration methods presented in the next section. We apply each of them according to the filter category, with and without the knowledge of the strength. Results in the experimental section 4.3.2, clearly show that strength estimation increases the efficiency of image restoration methods in terms of face recognition performance.

4.2.4 Image restoration

In this section, we describe image restoration methods (listed below) that we have selected for our experiments and their associated strength estimation. When we do not provide the strength estimation, we use a default value. Note that they exist more advance and efficient image restoration approaches, but there are complex. Thus, we select less efficient and less complex methods, but we prove that with them we can recognize the identity.

- **De-black masking**: Remember that the first formula of 4.1 allows to obscure faces with the black masking filter. In our case, $color$ is equal to zero because a black color represents zero as the intensity of a pixel. Thus, the reverse equation is the following:

$$I'(x,y) \sim \begin{array}{r} color = 0 \\ \dfrac{mean(IblackMask(x,y))}{(1-\alpha)} \end{array} \tag{4.5}$$

  We compute the opacity, α, depending on the estimation of the filter strength (as in the section 4.2.3) if known, otherwise we set α to 0.2 that is the average of the parameter value (between 0 and 0.3 to protect privacy).

- **De-pixelization/Super-resolution method 1**: Bicubic interpolation [106] is the simplest and most popular method in super-resolution domain. First, we down sample pixelated face images in a smaller size depending on the estimation of the filter strength (i.e., $squares\_size$) with the algorithm 1, in order to be re expressed as a super-resolution problem. Then, we resize the down sampled face images using the bicubic interpolation to its original size. If the filter strength is unknown, we down sample the image by two to reduce the noise (block effect) whereas preserving image quality and it is the average of the parameter value to protect privacy.

Algorithm 1: Size of the down sample

```
if (squares_size ≤ 4) then
    new_size = original_size * 0.7;
else if (squares_size ≤ 6) then
    new_size = original_size * 0.5;
else if (squares_size ≤ 9) then
    new_size = original_size * 0.3;
else
    new_size = original_size * 0.1;
```

- **De-pixelization/Super-resolution method 2**: We apply a super-resolution by adaptive sparse domain selection and an adaptive regularization [109] on pixelated face images. Depending on the estimation of the filter strength ($squares\_size$), we compute, using algorithm 2, the size of the Gaussian filter (i.e., $size\_gauss$) and the $standard\_deviation$ that are needed to apply the super-resolution method. If the filter strength is unknown, we set $size\_gauss$ to 9 and $standard\_deviation$ to 3 that is the average of the parameter values to protect privacy.

Algorithm 2: Size and standard deviation of the estimated Gaussian filter

```
if (squares_size ≤ 4) then
    size_gauss = 7;
    standard_deviation = 2;
else if (squares_size ≤ 6) then
    size_gauss = 9;
    standard_deviation = 3;
else if (squares_size ≤ 9) then
    size_gauss = 11;
    standard_deviation = 4;
else
    size_gauss = 13;
    standard_deviation = 5;
```

- **De-blurring method 1**: The principle of unsharp method is to compute an edge image *g(x, y)* from an input image *f(x, y)* and *fsmooth(x, y)* a smoothed version of *f(x, y)* (e.g., Gaussian blur):

$$g(x,y) = f(x,y) - fsmooth(x,y) \tag{4.6}$$

And the sharpen image is calculated as follows:

$$fsharp(x,y) = f(x,y) + k * g(x,y) \tag{4.7}$$

where *k* is a scaling constant. We fix k depending on the estimation of the filter strength (i.e, $standard\_deviation$) using algorithm 3. The higher the strength, the higher the value k is. If the filter strength is unknown, we set *k* to 5 that is the average of the parameter values to protect privacy.

Algorithm 3: Setting the parameter k

```
if (standard_deviation ≤ 2) then
    k = 2;
else if (standard_deviation ≤ 3) then
    k = 5;
else
    k = 8;
```

- **De-blurring method 2**: In order to de-blur, an estimation of the unknown blur is performed using maximum a posteriori estimation [110] on blurred faces. Depending on the estimation of the filter

strength ($standard\_deviation$), we compute, with the algorithm 4, the size of the point spread function, denoted PSF and needed to the de-blurring method. If the filter strength is unknown, we set PSF to $[8, 8]$ that is the average of the parameter values to protect privacy.

Algorithm 4: The estimated PSF

```
if (standard_deviation ≤ 2) then
    PSF = [7, 7];
else if (standard_deviation ≤ 3) then
    PSF = [8, 8];
else
    PSF = [9, 9];
```

- **De-noising method 1**: Formula 4.8 and 4.9 represent the Wiener de-noising algorithm [104] which is reiterated on noised face images depending on the estimation of the filter strength (i.e., $standard\_deviation$). The higher the strength the more we reiterate as it is shown in the algorithm 5. If the filter strength is unknown, we reiterate ten times that is the average of the parameter values to protect privacy.

$$b(x, y) = \mu + \frac{\sigma^2 - v^2}{\sigma^2}(f(x, y) - \mu), \tag{4.8}$$

where v is the noise variance deduced from the estimated $standard\_deviation$, *f* a noised image, *b* the de-noising image, *x* and *y* the pixel coordinates.

$$\mu = \frac{1}{NM} \sum_{x,y \in \eta} f(x, y), \sigma^2 = \frac{1}{NM} \sum_{x,y \in \eta} f^2(x, y) - \mu^2 \tag{4.9}$$

with μ the local mean and σ the variance and where η represents the local neighbourhood of each pixel and *N*, *M* the image size. Finally, we compute a bicubic interpolation to, first, reduce and then, enlarge image in order to delete remaining noise.

Algorithm 5: Number of reiteration

```
if (standard_deviation ≤ 0.01) then
    nbReIterations = 4;
else if (standard_deviation ≤ 0.04) then
    nbReIterations = 10;
else if (standard_deviation ≤ 0.08) then
    nbReIterations = 12;
else
    nbReIterations = 15;
```

- **De-noising method 2**: We select a denoising method based on wavelet decompositions [111]. The number of wavelet decompositions depends on the estimation of the filter strength (i.e., $standard\_deviation$), and is calculated with the algorithm 6. If the filter strength is unknown, we set the number of wavelet decompositions to seven that is the average of the parameter values to protect privacy.

Algorithm 6: Number of wavelet decompositions

if $(standard\_deviation \leq 0.01)$ **then**
  $nbrWaveletDecomposition = 2$;
else if $(standard\_deviation \leq 0.04)$ **then**
  $nbrWaveletDecomposition = 7$;
else if $(standard\_deviation \leq 0.08)$ **then**
  $nbrWaveletDecomposition = 10$;
else
  $nbrWaveletDecomposition = 14$;

The Figure 4.3 shows the workflow of the proposed method.

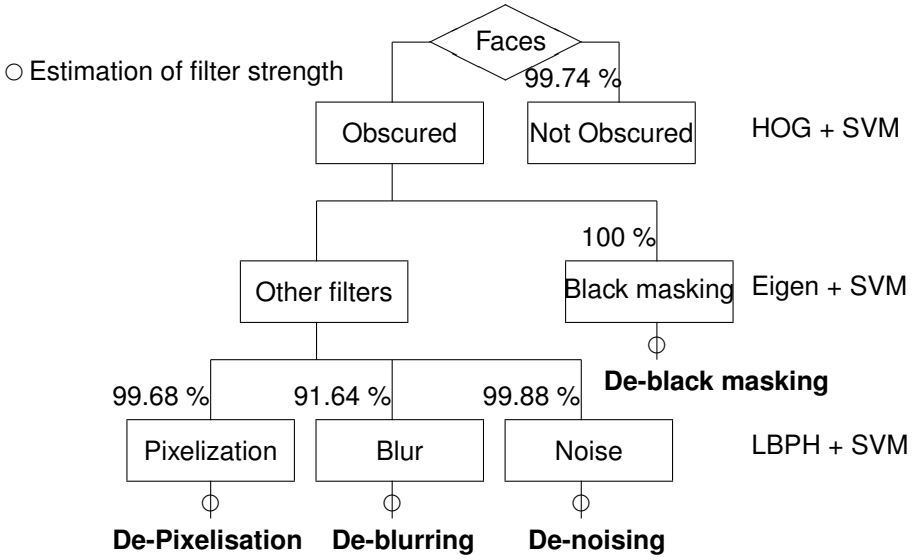


FIGURE 4.3: Workflow of the proposed method

4.3 Experimental results

Our framework has been evaluated in terms of percentage of correct classification (i.e., accuracy). Three popular datasets are selected: Feret [81], ScFace [83] and AT & T [82] (described in the section 2.5.3). Faces have been already cropped using Viola and Jones [16] face detector (described in the section 2.2) and we resize all face images to 127*134 pixels. These images are not compressed. We split the database in two, one for the training and the other for the testing.

4.3.1 Evaluation of filters classification

All steps of filter classification, previously explained, are trained (75 %) using Feret, ScFace and AT & T face datasets whereas only faces from Feret are employed in the testing part (25 %). We apply black masking, pixelization, Gaussian blur and Gaussian noise on original faces with different levels of

strength. The Table 4.2 sums up the number of faces which have been used in the training step for each method.

Detection method	Original	Obscured
HOG + SVM	3 644	11 606

First classification method	Black masking	Other filters
Eigen + Euclidean distance	442	5 967

Second classification	Pix	Blurring	Noising
LBPH + Euclidean distance	1 326	1 768	1 105

TABLE 4.2: Number of faces used in training set.

Our filter classification method has been tested on 13 810 faces which contains original (1149), black masking (2302), pixelized (3138), Gaussian blur (3768) and Gaussian noise (3453) faces with different levels of privacy strength. Note that we did not use all levels of strength in the training set. The confusion matrix in the Table 4.3 represents the percentages of correct and wrong classification. According to these results, the category of some obscured face images is slightly wrongly classified as another one: pixelized faces (0.3%) as blurred faces, blurred faces (8.4%) as original faces and noisy faces (0.1%) as black masking faces. Indeed, pixelization faces, for low strength, look like blurred faces. Blurred faces, for low strength, look like original faces. Noisy faces, for strong strength, are closer to black masking faces because noise affects the luminance of the pictures, thus, perturbs the Eigen algorithm.

In order to test the robustness of our framework, we also apply other types of noise (i.e., the Speckle and Salt and Pepper noise) and blur (i.e., average and motion blur) to original face images. Then, we classify, with our method, the 11 510 other noised and blurred faces. The number of testing faces for other blur and noise are detailed in the Table 4.4 as well as the percentages of correct classification respectively.

4.3.2 Image restoration with and without the estimation of the filter strength

In this part, we only use the Feret dataset. In the training set, we have selected 265 people with 2-8 images per people, 879 images in total, and in the testing set, 112 people with 1-3 images per people, 212 images in total. We apply, on the original face images, black masking (with opacity of 0.1, 0.2, 0.3), pixelization (with averaging size of 3, 4, 5, 6, 7, 8, 9, 10), Gaussian blur (with standard deviation of 2, 3, 4, 5, 8) and Gaussian noise (with variance of 0.001, 0.005, 0.01, 0.02, 0.04, 0.06, 0.08, 0.1, 0.3).

		Prediction				
		Orig	Black	Pix	Blur	Noise
Ground Truth	Orig	99.7%	0.3%			
	Black	0%	100%	0%	0%	0%
	Pix	0%	0%	99.7%	0.3%	0%
	Blur	8.4%	0%	0%	91.6%	0%
	Noise	0%	0.1%	0%	0%	99.9%

TABLE 4.3: Confusion matrix

Types of filters	Average and Motion blur	Speckle and Salt and Pepper noise
Number of face images	5 755	5 755
%	97.85	99.1

TABLE 4.4: Percentages of correct classification for other types of blur and noise.

The robustness of face recognition against privacy filters differs from one algorithm to another one. This is why, we have selected three different face recognition algorithms: LBPH, HOG and Eigen features with a linear SVM, previously explained in the section 2.5.3. They obtain, respectively, 92.45 %, 94.34 % and 93.4 % of accuracy using original face images. Moreover, LBPH and Eigen are often used as baselines by the biometric community to compare the impact of obscuration on face recognition [79], but also to estimate face recognition before and after a Super-resolution method [80].

We illustrate in the Figures **??**, 4.4, 4.6, 4.8, 4.10, the difference between the rate of good recognition for original faces and the rate of good recognition for obscured faces before the image restoration (in blue), after image restoration without strength classification (in yellow and brown) and after the image restoration with strength prediction (in red and green). The lower is the curve the closer performances are to the original face images, and the better is the recognition. According to the Figures 4.4, 4.6, 4.8, 4.10, performance after image restoration with strength classification (in red and green) are the best. For instance, in the Figure 4.4, we remark a strong increase for Eigen face recognition algorithm between before image restoration (in blue) and after applying de-black masking (in red) knowing the strength. In addition, we show in the Figures 4.5, 4.7, 4.9, 4.11, results in images when applying the associated restoration with and without the knowledge of the filter strength.

However, we notice that for de-blurring and de-noising, without strength classification, LBPH algorithm obtains better results. This can be explained because LBPH is not robust against noise, and is sensitive when texture changes. Indeed, the de-noising methods as well as the de-blurring methods when using the strength estimation, add some details which do not exist in the original images therefore the texture changes. If the strength is unknown, the results of the de-blurring and de-noising remain less sharp.

Moreover, as soon as the category of filter is known, we also choose the most appropriate recognition algorithm in terms of robustness against a specific filter (the lower curves in all Figures). In our case, we will select HOG after de-black masking, see Figure 4.4, and Eigen for the other filters after applying their associated image restoration, see Figures 4.6, 4.8, 4.10. By doing this, we obtain similar rates of face recognition between the original face images and the face images after image restoration using this framework (i.e., the curves are close to zero).

Therefore, the proposed filter categorization and the filter strength estimation help to tune image restoration methods. We proved that if we protect the privacy by either black masking, pixelization, blurring or noising filters, the method is no longer effective because our proposed workflow de-anonymized the obscured face images.

4.4 Conclusion and Future work

We have designed a framework which enables, in a first step, to detect the presence of a privacy filter, in the second step, to classify the type of filter (i.e., black masking, a pixelization, blurring and noising) and in the last step, to estimate its strength . Using an appropriate restoration method (i.e., de-black masking, de-pixelization, de-blurring, de-noising), we almost reconstruct the face images and simulations show that the performances of face recognition are closer than the ones obtained for the original faces. Hence, privacy of people can be revealed and is no longer protected.

As future work, we could select better algorithms of image restoration. For instance, the recent deep neural networks successfully perform the tasks of super-resolution [112], de-blurring [113] and de-noising [114]. We could also design an efficient deep neural network to directly classify the category of the filter and their strengths.

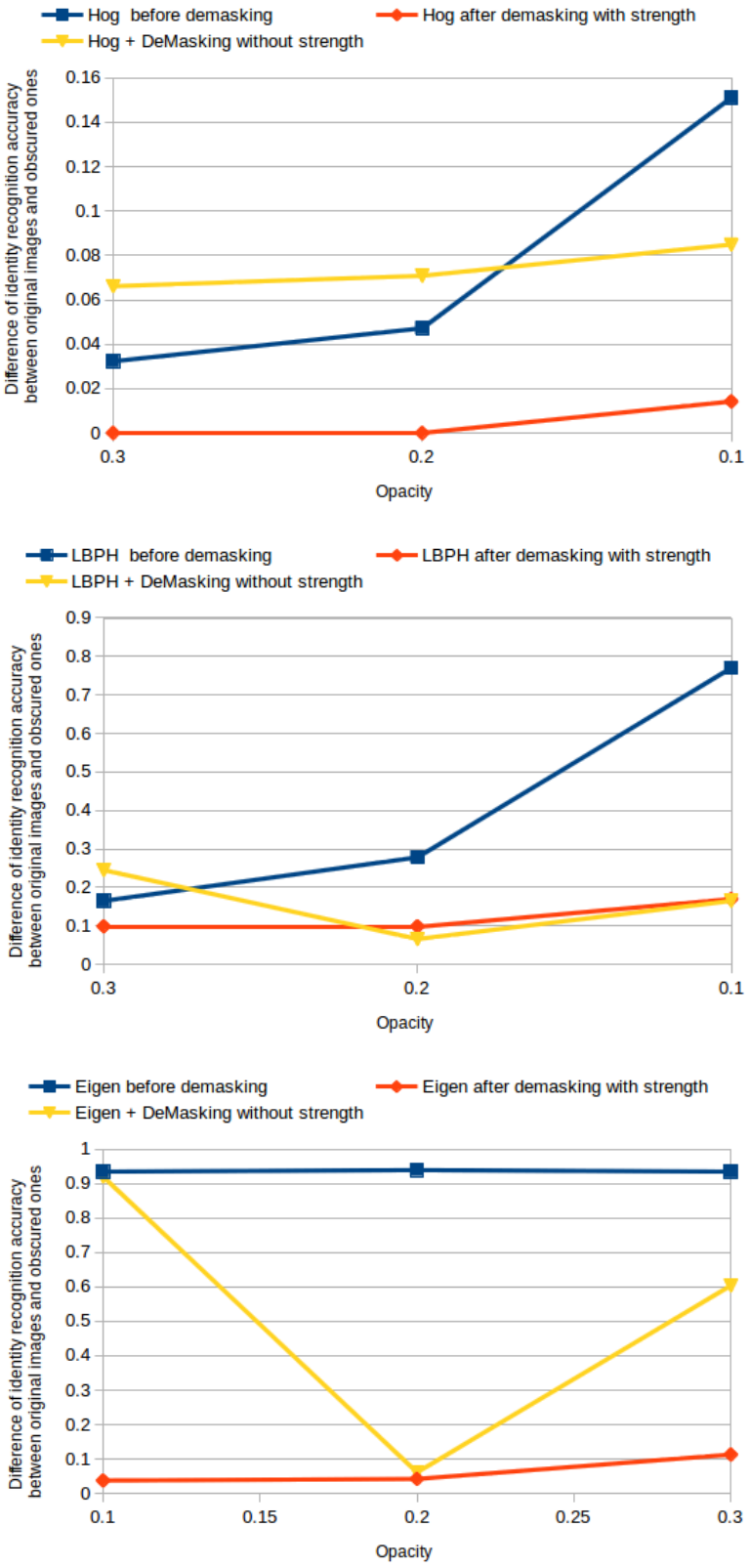


FIGURE 4.4: Impact of the de-black masking method depending on different opacity. The difference between the accuracy (%) of identity recognition on original faces and obscured faces.

(a) (b) (c) (d)

FIGURE 4.5: From left to right, the original face image (a), black masking face image with α = 0.9 (b), de-black masking without (c) and with (d) strength classification.

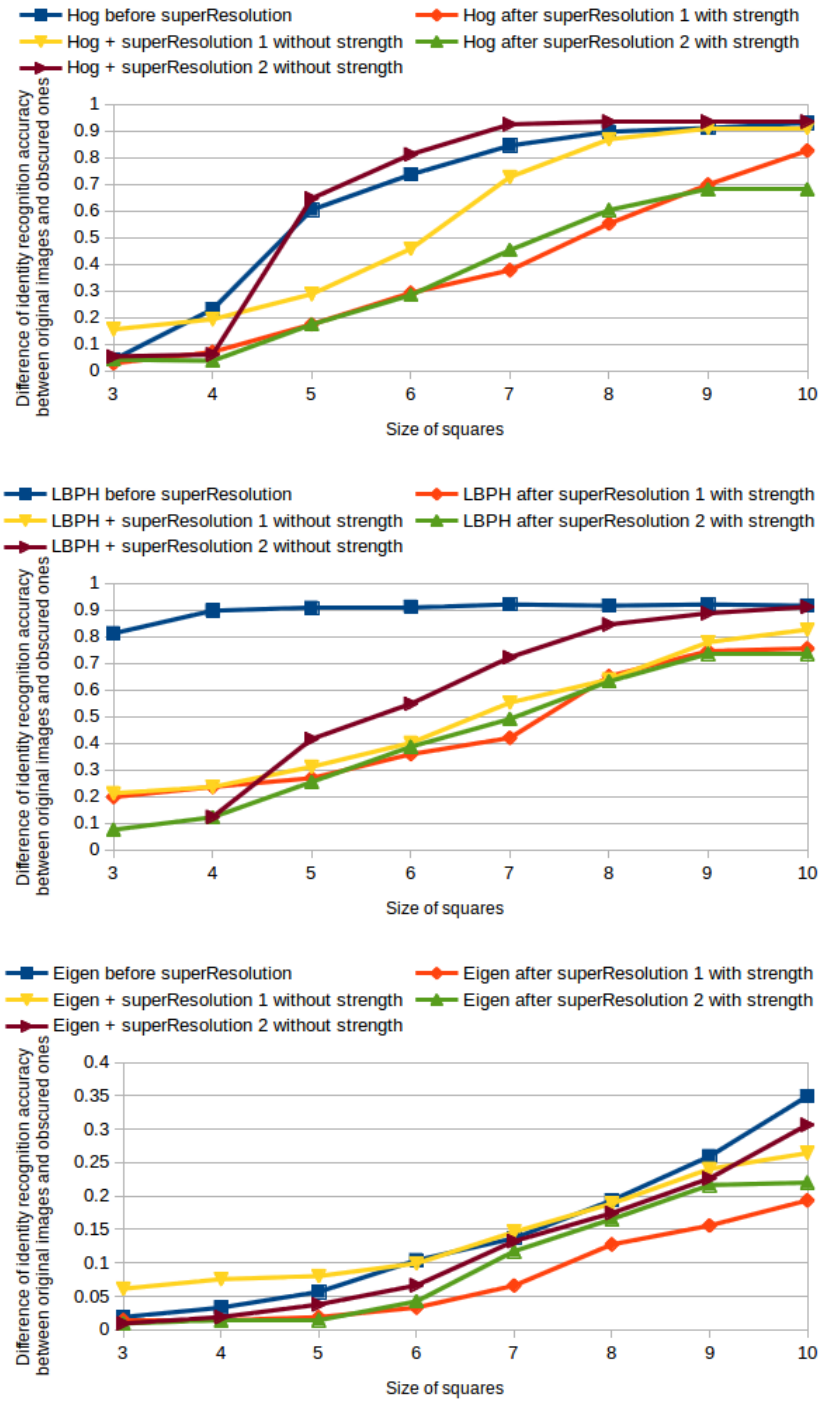
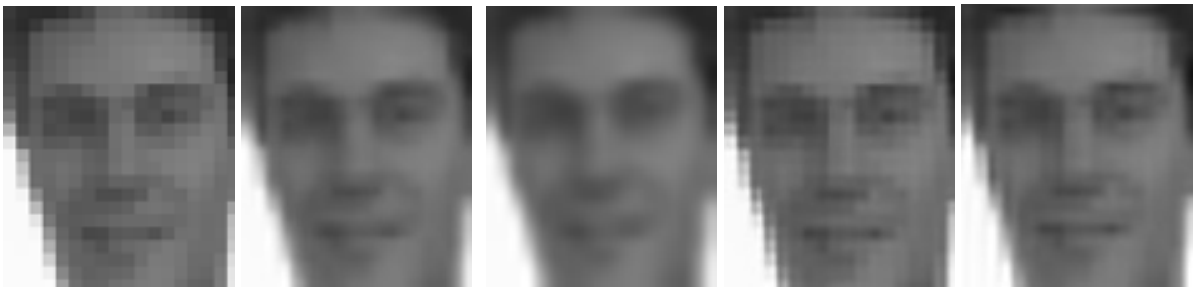


FIGURE 4.6: Impact of the de-Pixelization methods depending on different sizes of squares. The difference between the accuracy (%) of identity recognition on original faces and obscured faces.



(a) (b) (c) (d) (e)

FIGURE 4.7: From left to right, pixelated face image with size of squares = 5 (a), de-Pixelization without (b) and with (c) strength classification for the first method, de-Pixelization without (d) and with (e) strength classification for the second method.

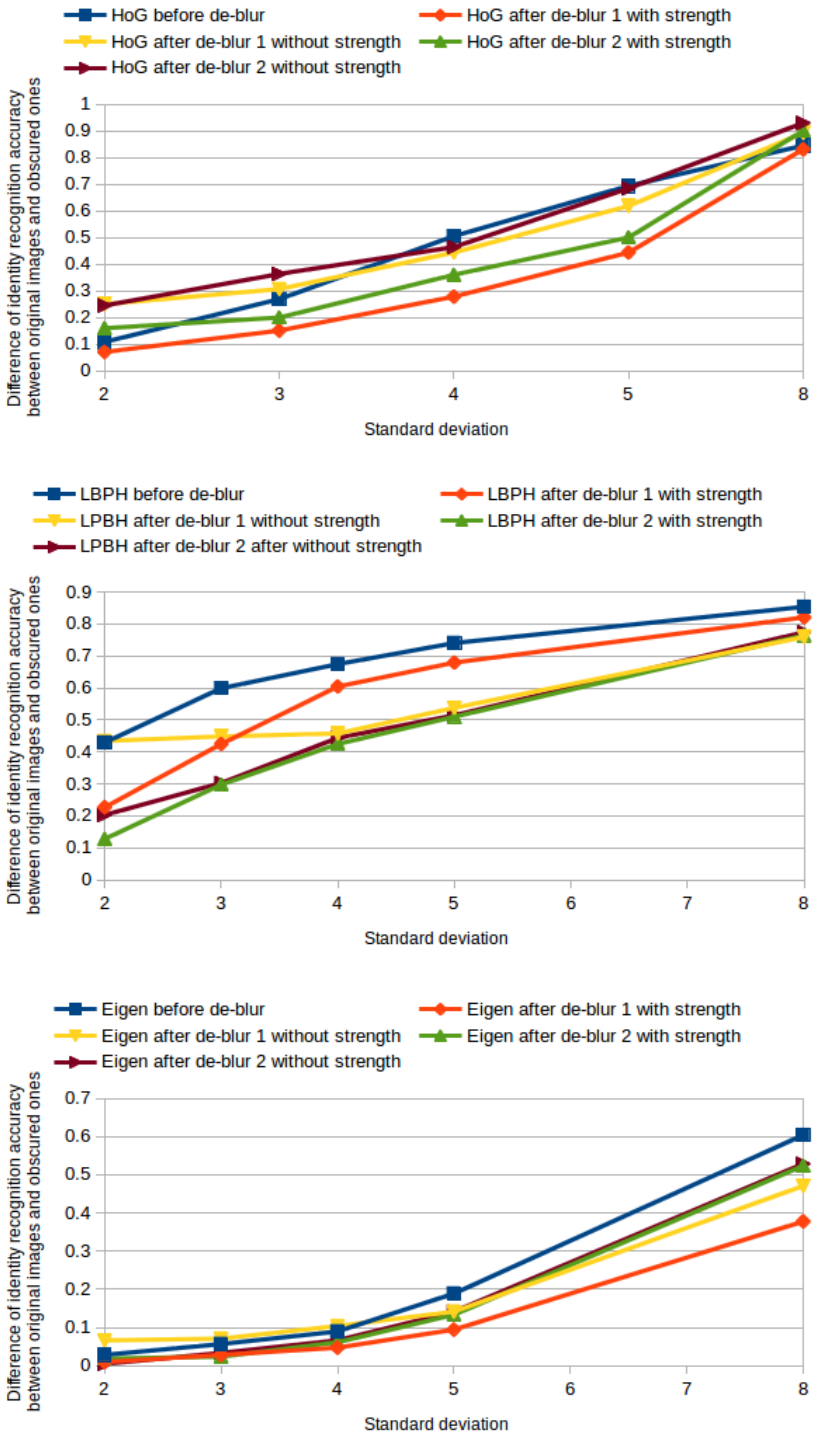


FIGURE 4.8: Impact of the de-blurring methods depending on different standard deviation. The difference between the accuracy (%) of identity recognition on original faces and obscured faces.

(a) (b) (c) (d) (e)

FIGURE 4.9: From left to right, blurred face image with σ = 2 (a), de-blurring without (b) and with (c) strength classification for the first method, de-blurring without (d) and with (e) strength classification for the second method.

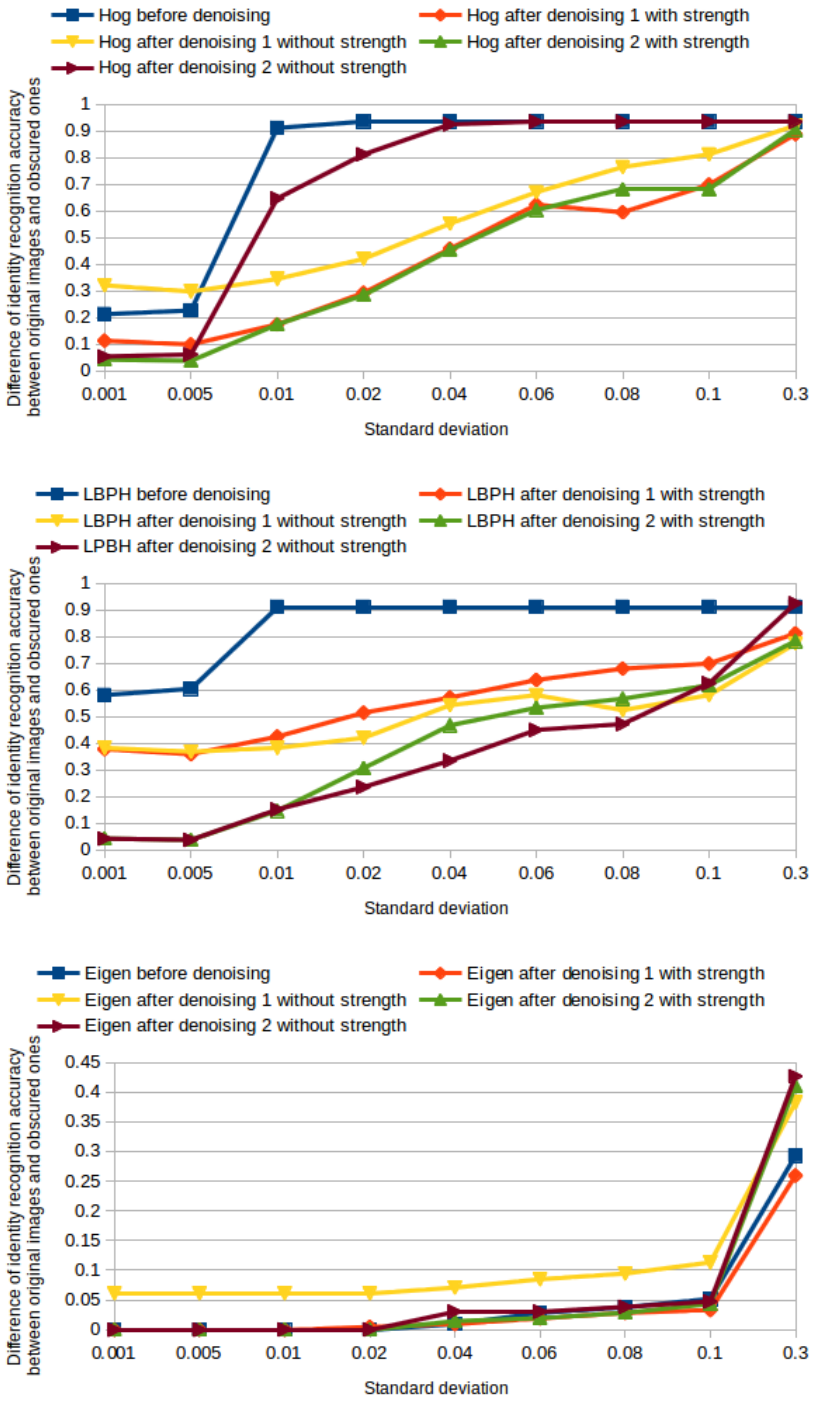


FIGURE 4.10: Impact of the de-noising methods depending on different standard deviation. The difference between the accuracy (%) of identity recognition on original faces and obscured faces.

(a) (b) (c) (d) (e)

FIGURE 4.11: From left to right, noisy face image with σ = 0.01 (a), de-noising without (b) and with (c) strength classification for the first method, de-noising without (d) and with (e) strength classification for the second method.

Chapter 5

Spatial-domain scrambling preserving the utility of visual surveillance

5.1 Introduction

As we introduced in the state-of-the-art Chapter and shown in the Table 2.3, none of the existing privacy filters satisfies all criteria required by the surveillance. In this Chapter, we have designed a new method, named *StegoScrambling* to protect the privacy while preserving the utility of the video surveillance.

Our proposed system allows protecting privacy in video or still images while keeping people motion needed for the surveillance. Therefore, it fulfills the five following criteria: privacy protection (no possibility to identify people), utility of the visual surveillance (keep shape and motion of people to still recognize events), near lossless reversible (possibility to recover the original data), fast computation (run in real time) and security (only authorized people can reverse the process).

In images, we define the number of bits per pixel (bpp) by the number of different colors that usually is 256, thus 8 bpp ($2^0 + 2^1 + 2^2 + 2^3 + 2^4 + 2^5 + 2^6 + 2^7 = 255$). The proposed approach combines a scrambling and a steganography method using the 8 bpp. We apply this combination in a spatial domain by shifting and storing the Most Significant Bits (MSBs) (i.e., the most significant information) of the encrypted pixels from a RoI (Region of Interest) into the Least Significant Bits (LSBs) (i.e., the least significant information). Then, we compute the edge of the RoI (i.e., the shape of the body), and the bits of its pixels replace the MSBs of the resulting image in order to keep the scene understandable.

This method was, first, designed within the context of the Mini-drone Video Privacy Task at MediaEval Benchmark 2015 [115], but this first implementation did not run in real time and RoIs (Regions of Interest) have been annotated manually for the challenge. The last version of this method runs in real time (using GPU) on videos from smart phones (with the IP Webcam application on Android) or on drones (the AR.Drone 2.0. Parrot) by integrating our algorithm in the APIs of CV Drone (OpenCV + AR.Drone) [1].

[1] https://github.com/puku0x/cvdrone/wiki/API-reference

Moreover, RoIs are automatically detected by Dalal and Triggs [17] method (i.e., a people detector). We dedicate the first pixels of the image to store the bounding box of each RoI.

5.2 *StegoScrambling*

5.2.1 Storing the bounding box of each RoI

We first store the number of detected RoI, in the pixel value of the bottom-right corner (for the blue channel) of the original image. The bounding box contains the coordinates of the upper left corner of the RoI denoted as (x, y), and the $width$ and the $height$ of the RoI. Following the algorithm 8 in the Appendix, we insert the four values of each bounding box (i.e., (x, y), $width$ and $height$) in the first pixels of the upper left corner of the original image such as the sum of the three channels (red, green, blue) is equal to the values. Each pixel value of each channel is between 0 and 255. Therefore, we assume that the size of the RoI is not upper than 765.

5.2.2 Generate a pseudo-random numbers (PRNG)

A user must provide a *password* (i.e., a secret key) to allow and secure the reconstruction of the original data (only for users that know the correct secret key). We generate a unique seed from this *password*, and a seed generates a unique sequence of random numbers using a pseudo-random number generator (PRNG). We use the “Mersenne Twister" generator available in Matlab. It is not the goal of this thesis to find the best way to generate a seed so we used a simple method. Each letter of this *password* is converted to a number according to its ASCII code, and the Least Significant Bit (LSB) of each first thirty-two numbers are concatenated (Matlab allows a maximum seed of 2^{32}). Consequently, a number between 1 and 2^{32}, named the seed, is obtained. Multiple seeds can be generated to strengthen the security.

5.2.3 Description of the process

After determining the password and detecting the regions of interest (RoIs), we apply our privacy protection process. We assume that a pixel is coded in 8 bits.

We extract only the six Most Significant Bits (MSBs) of each pixel belonging to the RoI, and we compute an XOR between them and the random numbers (RNs) as in formula 5.1. This protects the original information and allows the reversibility. We keep 75 % of the original bits to have a better trade off between quality of the reconstructed images and quality of the protected ones (for example, if pixels are coded in 16 bits we extract the 12 MSBs).

$$XORImg(x, y, c, i) = RoI(x, y, c, i) \oplus RandNums(x, y, c, i), \forall i \tag{5.1}$$

with *(x, y)* the pixels coordinates, *c* the channel, and *i* the bit position and each bit $\in \{0, 1\}$.

In parallel, we apply an edge detector on the RoI which returns a binary image. Later, in the section 5.3, we explain how this binary image can be designed/computed/conceived differently. We insert the two MSBs of the edge image (i.e. the binary image, pixel intensity is either 192 or 0) into the two MSBs of the privacy image while the six encrypted bits of the XOR image (pixels intensity between 0 and 63) are integrated in the LSBs of the privacy image as in the formula 8.7.

$$PrivacyImg(x, y, c) = \sum_{i=0}^{5} XORImg(x, y, c, i) * 2^i + \sum_{i=6}^{7} EdgeImg(x, y, i) * 2^i, \tag{5.2}$$

The Figure 5.1 illustrates the workflow of the proposed method and the Figure 5.2 at the top right shows an example of an image on which we apply the process.

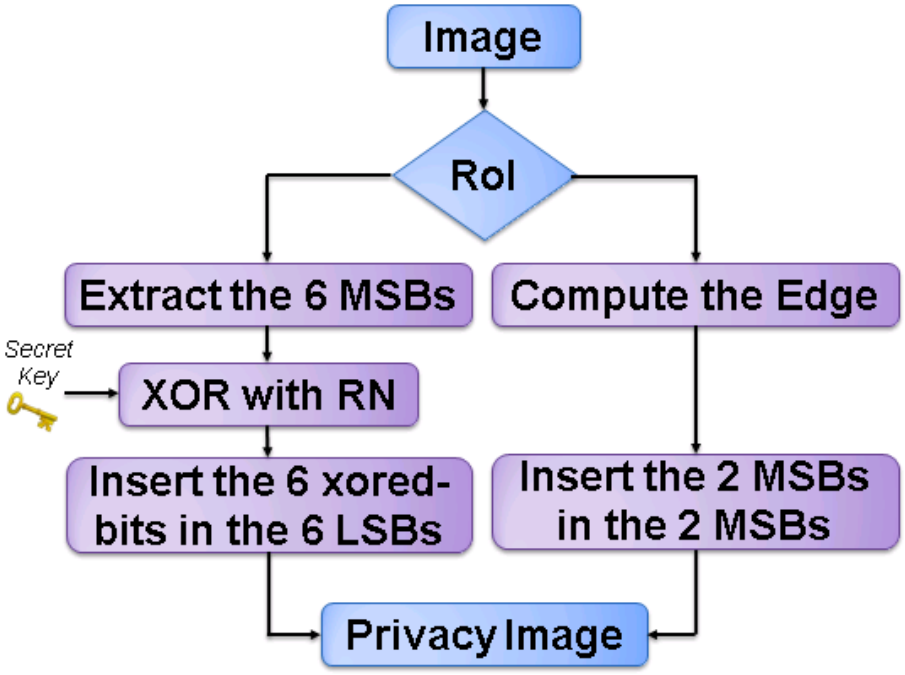


FIGURE 5.1: Workflow of the proposed process

5.2.4 Inverse StegoScrambling

To recover the original RoI, the authorized people have to provide the correct *password* in order to generate the same sequence of random numbers than for the encryption.

We compute an XOR between the 6 LSBs of the privacy image and the random numbers as in the formula 5.3. Finally, to recover the image, we shift the result of the XOR to the MSBs and set to zero the

FIGURE 5.2: Privacy filter applied on a pedestrian

two LSBs. Figure 5.3 illustrates the workflow of the inverse process and the bottom part of the Figure 5.2 shows the recovered image in case of a correct or wrong key.

$$RecoveredImg(x,y,c) = \sum_{i=2}^{7}(PrivacyImg(x,y,c,i-2) \oplus RandNums(x,y,c,i)) * 2^i \tag{5.3}$$

5.2.5 Pixel example

We consider one pixel with 8-bit from MSB to LSB.

Original pixel	$b7$	$b6$	$b5$	$b4$	$b3$	$b2$	$b1$	$b0$

For each pixel of the RoI, we preserve the bit between 2 and 7 only (i.e., the MSBs). We compute an XOR between the MSBs of the original pixel and the bits of a random number. We denote the encrypted pixel, b'.

XORpixel, b'	$b'7$	$b'6$	$b'5$	$b'4$	$b'3$	$b'2$	X	X

We shift the bit of b' into the 6 LSB.

XORpixel, b'	X	X	$b'7$	$b'6$	$b'5$	$b'4$	$b'3$	$b'2$

2 MSBs of an edge, e, is represented by $e_6 = 1$ and $e_7 = 1$ and of a no-edge by $e_6 = 0$ and $e_7 = 0$. Finally, we add the 2 MSBs of e (the edge pixel) with the 6 LSBs of b' and denote this pixel as the protected pixel.

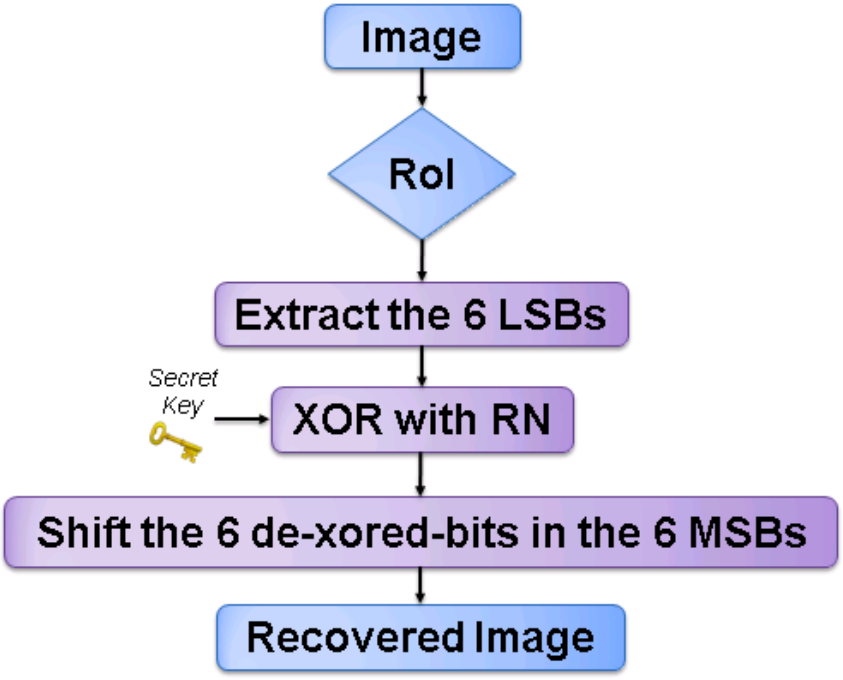


FIGURE 5.3: Workflow of the inverse process

Edge pixel, b'+e	1	1	$b'7$	$b'6$	$b'5$	$b'4$	$b'3$	$b'2$
No-edge pixel, b'+e	0	0	$b'7$	$b'6$	$b'5$	$b'4$	$b'3$	$b'2$

To recover the original pixel, we first compute an XOR between the same random number than previously thanks to the *password* and the 6 LSBs of the protected pixel, and then we shift the LSBs towards the MSBs.

Recovered pixel	$b7$	$b6$	$b5$	$b4$	$b3$	$b2$	X	X

5.2.6 Experimental Results

5.2.6.1 MediaEval Challenge

We follow the guidelines of the MediaEvel 2015 DroneProtect Task [116], and we use different video sequences from DronesProtect dataset [117] to participate in this challenge.

The evaluation is based on the human perception in terms of level of privacy, intelligibility and appropriateness. The questions are similar to the ones designed in [118] (e.g., Can you recognize the gender/race of the people?, Can you recognize what they are doing?). Two human evaluator groups are selected. In the first group, people come from surveillance security domain (R & D), and in the second group they come from any other domain (Naive).

The aim of the challenge is to find the best trade-off between privacy and visual quality of the protected image. Indeed, the higher the privacy protection, the lower the level of information (i.e., utility of visual surveillance and pleasantness).

In the Table 5.1, we report the average results of our approach as well as the average of all other proposed methods in the challenge. We obtained positive feedback from the jury and especially for the privacy protection criterion. Indeed, according to the results, 60 % of privacy is well protected. We later prove using an objective evaluation that the gender information is still visible whereas the identity is well protected when applying our privacy protection method. However, we got 40 % for the utility of visual surveillance and the pleasantness. This shows a lack of recognizing events when we apply our proposed approach probably due to the black and white colors.

TABLE 5.1: Average results (%)

Evaluation	Privacy	Utility	Pleasantness
Category 1 (R&D)	63	37	36
Category 2 (Naive)	57	43	48
Average (%)	60	40	40
Average all methods (%)	48.5	58.5	60

5.2.6.2 Quality of the reconstructed images

Two metrics, the **PSNR** and the **SSIM** (explained in the section 2.5.4), have been selected to measure the quality of the reconstruction of the 2400 still images of the Feret [81] and the ScFaceData [83] databases. We provide more details about these databases in the section 2.5.3.

In the Table 5.2, we represent the mean and the standard deviation (Std) of the **PSNR** and the **SSIM** between the original and the recovered images.

	PSNR	SSIM
Mean	42.46	0.9968
Std	0.2786	0.0013

TABLE 5.2: PSNR and SSIM between the original images and the recovered ones

We lost the two LSBs of each pixel of the original RoI thus, in the worst case, the recovered RoI decreases of three $(2^0 + 2^1)$ for each intensity of pixel color (between 0 and 255) compared to the original RoI. This loss has no impact on human vision and few for machine as it is shown in the Table 5.2.

5.2.6.3 Privacy protection evaluation

We trained four face recognition algorithms, LBPH [75], Eigen [76], HoG [77] and OpenFace CNN [84] to extract features and a linear SVM to train for each, using a subset of Feret and ScfaceData database. We provide more details about these methods and databases in the section 2.5.3,

Results clearly prove the failure of all these face recognition tools when we apply our approach (∼0 % of good recognition) whereas the rate of identity recognition are almost the same for original images (∼95 % of good recognition) and recovered ones (∼90 % of good recognition).

5.2.6.4 Time consuming

The whole system (our privacy protection + RoI detection) takes 8 images per second, on/in average, with image resolution of 640 * 480 pixels. More specifically, on a typical video, people detector takes 0.12 seconds per image and our privacy protection takes 0.0097 seconds per image. Therefore, the amount of time consuming is mainly due to the people detector. This evaluation has been done in C++ with GPU on a GeForce GTX 980M graphic card.

5.2.6.5 Brute force attack

The process applies an XOR between each pixel of the RoI (i.e., values between 0 and 255) and random numbers that are between 0 and 63. Therefore, there are $64^{height*width}$ combinations to test if a user wants to illegally recover the original data from the encrypted pixels (with $height$ and $width$ the size of the RoI).

The number of possibilities greater than 2^{128} already represents a limit impossible to reach with current technology [2]. The minimum RoI size needed to generate more than 2^{2048} combinations is 18*18, as detailed in the equations 5.4. Face or body images lower than 18*18 pixels are infrequent. Therefore, our process is robust against a brute force attack because it produces a high number of combinations.

$$\begin{aligned} 64^{height*width} &\geq 2^{2048} \\ \Leftrightarrow 2^{6*height*width} &\geq 2^{2048} \\ \Leftrightarrow 6 * height * width &\geq 2048 \\ \Leftrightarrow height * width &\geq 341 \\ \text{Assuming that the height is equal to the width} & \\ \Leftrightarrow height = width &\geq \sqrt{341} = 18 \end{aligned} \tag{5.4}$$

5.2.6.6 Gender detection evaluation from body contours

We use a pre-trained convolutional neural network (CNN) that detects the gender from the body. Further details about this architecture are available in the section 2.5.3. We evaluate this tool over 1081 videos from the HID database [94], including 864 male videos and 220 female videos. Each video contains one person only. We provide a description of this database in the section 2.5.3.

[2] http://cri.ensea.fr/en/node/208?destination=node%2F208

For each frame of a video, we get the confidence-rate of the gender predicted. 1 corresponds to a male prediction and 0 to a female one. To obtain the predicted gender in each video, we average all confidence-rates obtained for each frame, denoted by mc (i.e., if $mc > 0.5$, the gender is predicted as a male otherwise as a female). Then, the accuracy of female and male detection are computed when we get the gender predictions of the 1081 videos.

We apply the gender classifier [92] over the original videos and on the videos where the shape of the body is only kept as explained in section 5.3.1. In the Figures 5.10 and 5.11, the gender classification for the original body images (black points) and for the shape ones (pink points) provides high accuracy and their results are close, which means that we still detect the gender information even if we represent people by their body shape only. Therefore, we decide to reshape the body contours. The goal of the next section is to make the gender unrecognizable, we denoted this concept, *"de-genderization"*.

5.3 Our proposed *de-genderization* method by body contours reshaping

The main goal of this section is to make the gender of people no more recognizable while preserving enough information concerning body shape and motion of people for action classification. We denote this processing as *de-genderization*. Regarding the existing privacy protection methods, most of them focus on the de-identification only. These methods do not automatically imply the suppression of visual semantic traits such as gender.

Therefore, we propose two approaches that modify the visual appearance of the body shape in order to *de-genderize* people while keeping the possibility to interpret the video. In both methods, we start by extracting the contour points attached to the body shape of people in each frame of the videos. Then, we either mix the coordinates of the body shape with a predefined model, or we smooth the body shape by successive polygonal approximations based on convexity. The strength of each method is tunable depending on the exact application. Our results demonstrate that both proposed approaches protect the gender information while preserving the global body movement. However, the second approach based on convexity better preserves the visibility of human activities.

5.3.1 Finding the body shape of a person

FIGURE 5.4: Respectively, original and extracted body shape

For videos, we apply a mixture of Gaussian (MOG) [119] (i.e., an adaptive background subtraction method) to find the foreground (i.e., body of people). For images, instead of using MOG (because we do not have the several followed images), we apply GrabCut [120], an approach based on optimization by an iterative graph-cut [121] (time consuming). OpenCV[3] includes these methods. We remove noise and holes using basic morphological operations [122], and we apply Canny edge detection.

These methods provide black and white images where white pixels are associated with the silhouette of the body, as it is shown in the Figure 5.4.

5.3.2 Merging coordinates of a body shape with the ones of a reference model

We merge the coordinates of the body shape with the ones of a model by following the steps below. We choose this model such as it represents as much as possible the database.

- **Step 1**: A bounding box denotes the smallest box containing the set of points from a specific object (i.e., in our case, the body shape). The first step consists in aligning a shape body image with a model. For this purpose, we resize the image of the body shape with the ratios computed in the equation 5.5. In the equation 5.5, we denote: i) the height and the width of the model shape (Mh, Mw) and those of the body shape (Bh, Bw), and ii) the upper-left bounding box coordinates for the model shape (Mx, My) and those for the body shape (Bx, By).

$$ratioHeight = \frac{Mh - Mx}{Bh - Bx},\ ratioWidth = \frac{Mw - My}{Bw - By} \tag{5.5}$$

- **Step 2**: Let B denote the set of points of a body shape and M the set of points from the model shape. $\forall p_i \in B$, we find the point p_j included in M such that the distance is minimum. The formula is given in (5.6) where (x, y) represent the coordinates of a point.

$$\min_{\forall p_j \in M} \sqrt{(x_{p_i} - x_{p_j})^2 + (y_{p_i} - y_{p_j})^2} \tag{5.6}$$

- **Step 3**: For each couple (p_i, p_j) found in step 2, we merge the coordinates of p_i with the ones of p_j following the formula 5.7, where α is a parameter that controls the strength of the merging, and p the new point with (x', y') as coordinates. For α, the closer to 1, the closer to the original shape is the new shape. Conversely, the closer to 0, the closer to the model shape is the new shape. If α is equal to 1, the new shape is equal to the original shape, however, if α is equal to 0, the new points are included in a subset of M because $\forall p_i \in B$ (not $\forall p_j \in M$) we found a couple (p_i, p_j). The model shape used to obtain the results of the Figures 5.5 and 5.6, is the male one shown in

[3] http://docs.opencv.org/

the Figure 5.7(a).

$$\begin{aligned} & \forall(p_i, p_j) \\ p(x', y') = (x_{p_i} * \alpha + x_{p_j} * (1-\alpha), & \\ y_{p_i} * \alpha + y_{p_j} * (1-\alpha)) & \end{aligned} \tag{5.7}$$

- **Step 4**: In order to get the final shape, we link all the border segments of the new coordinates found in the step 3. The Figure 5.5 shows the results of the method in images.

FIGURE 5.5: The purple shape (model), the yellow shape (original) and the blue shape (final) with $\alpha = 1, 0.8, 0.6, 0.4, 0.2, 0$

α	1	0.8	0.6	0.4	0.2	0
80.5	80.6	81.6	79.6	75.7	54	50

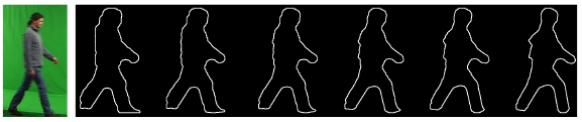


FIGURE 5.6: Original image and merging shape with the associated average accuracy of gender recognition (in the second row) according to the value of the parameter α (in the first row)

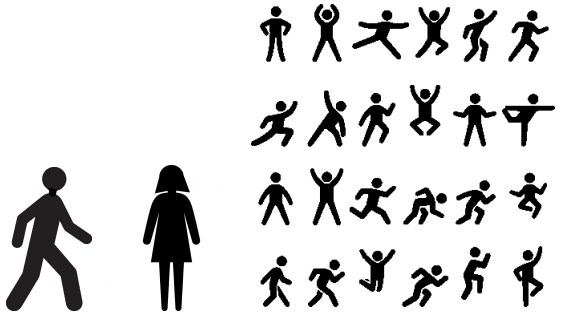


FIGURE 5.7: (a) and (b) Selected models in our experiments, (c) Example of codebook of postures

The average accuracy of gender recognition showing in the Figure 5.6 and the results demonstrated in the section 5.3.4.1, show that the gender of the final shape is not correctly detected anymore. Indeed, it converges systematically towards the gender of the reference model and the gender can be guessed with a probability of 50 %.

In order to better preserve the movement of arms or legs, we could choose an appropriate model, frame by frame, from a codebook such as the one represented in the Figure 5.7(c). Indeed, the visualization of arm and leg movements depend on the selected model. Thus, we have designed a second method based on the convexity of the body shape itself. In other words, it converges towards a polygonal simplification of the silhouette that helps to preserve human activities.

5.3.3 Polygonal approximation of a body shape

A convex set defines a region such as each segment of every pair of points is inside that region. A convex hull defines the minimal convex set containing a set of points.

We replace the coordinates of the body shape (i.e., the coordinates of the white pixels) by the ones from the convex hull of several sets. Each set of points belongs to the points of the body shape and contains n neighbor points.

- **Step 1**: Let S denote the set of points of a body shape. $\forall x \in S$, we find and draw the convex hull of $V \subset S$, where V is a subset of S containing x and its $n-1$ nearest neighbours in terms of distance along the body contour.
- **Step 2**: The step 1 produces several polygons of n points. We only keep the lines on the border and we obtain the new shape.

The Figure 5.8 shows the different steps of the process. The green lines and points in the Figure 5.8(h) define the original body shape, and the blue lines the shape approximated by convexity with $n = 5$.
The higher the value n the more convex is the shape (as illustrated in the Figure 5.9).

The results demonstrated in the section 5.3.4.1 show the suppression of the gender when we apply the approximation by convexity. Indeed, the machine systematically interprets the final shape as female.

5.3.4 Experimental Results

In this part, we prove that our two *de-genderization* methods hamper gender detection like the inpainting [3], JPEG scrambling with high level [6] of protection (i.e., encryption of DC and AC coefficients) and black masking methods. We explain these existing methods in the sections 2.3 and 2.4. In the following, we conclude that the approximation using convexity (the second proposed approach) preserves the human activities contrary to the others.

5.3.4.1 Evaluation of gender detection

We evaluate the performance of gender detection using the same protocol described in the section 5.2.6.6, over the original videos, those generated by the first method (5.3.2) with $\alpha = 0, 0.2, 0.4, 0.6, 0.8, 1$, and those generated by the second method (5.3.3) with $n = 3, 5, 10, 20, 30$.

The Figures 5.10 and 5.11 show the accuracy of male and female detection for original (the black point), shape (the pink point) body images compared to the ones where the body contours are reshaped (the blue points). The random case (the red point) can be considered as the results of the inpainting, scrambling and black masking methods. We can note that the performances decrease for the two approximation methods. For example, when $\alpha = 0.2$ using a male model (the point towards the top left corner

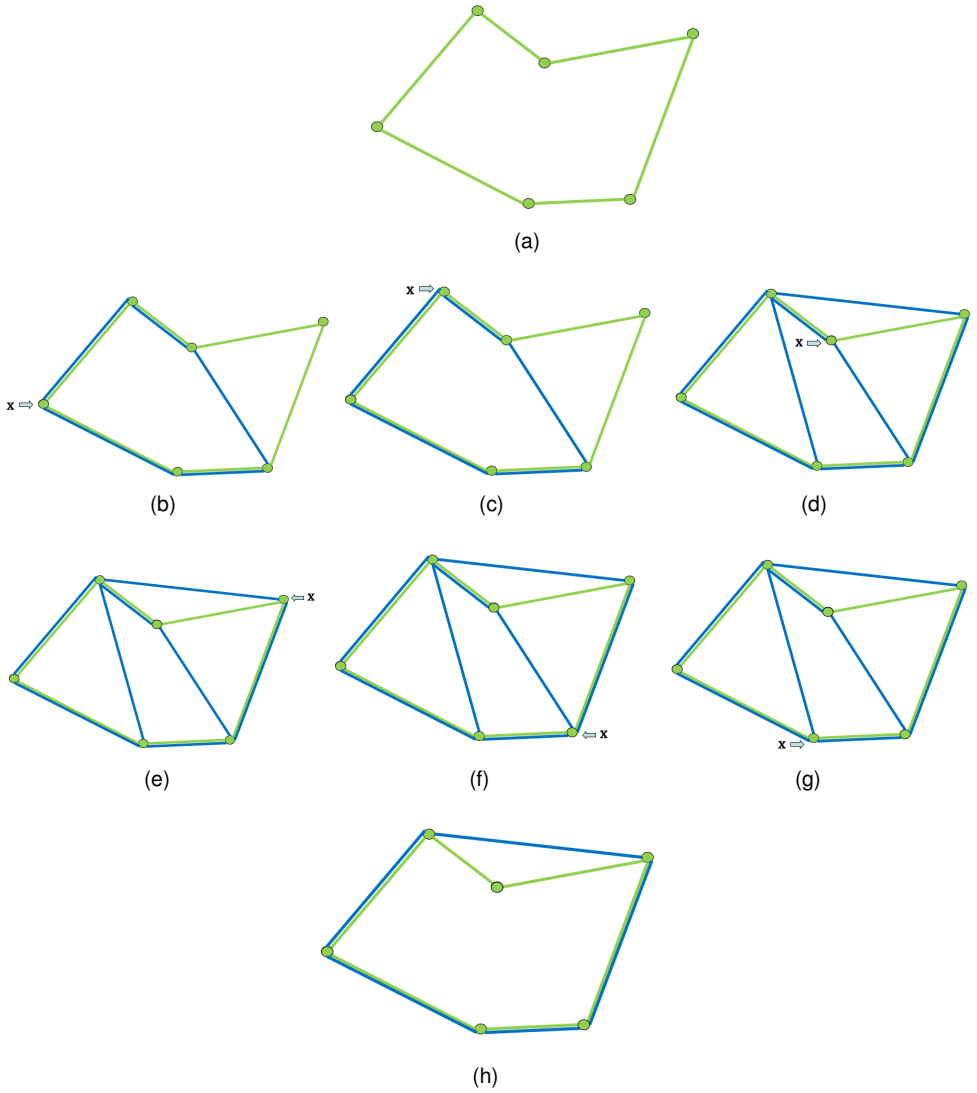


FIGURE 5.8: (a) Original body shape, (b-g) step 1: drawing the convex hull of each point, x, and step 2: its 4 neighbors (n=5), and (h) Keeping the lines on the border only.

in the Figure 5.10), the rate of correct detection for females is about 10 % whereas the one for male is about 95 %.

The Figure 5.10 represents the results of the first method (5.3.2). We can observe that the closer to the model shape is the final shape, the more the tool classifies the gender of the final shape as the gender of the model shape. The Figures 5.7(a) and 5.7(b) show the selected reference models.

The Figure 5.11 represents the results of the second method (5.3.3). We can observe that the higher the value of n the more the classifier tags the body images as female. This is mainly due to the convex approximation which seems to produce more female forms. Indeed, as it is shown in the Figure 5.9 and 5.14 several classical women hairstyle appear, and in the lower body we can guess the presence of a skirt.

n	3	5	10	20	30
80.5	76	74.4	58.8	50	50

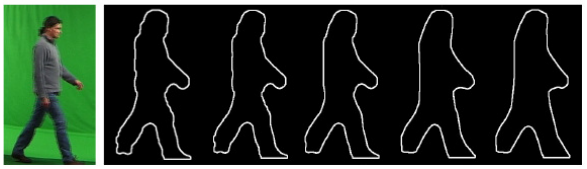


FIGURE 5.9: Original image and shape approximation using convexity with the associated average accuracy of gender recognition (in the second row) according to the value of the parameter n (in the first row).

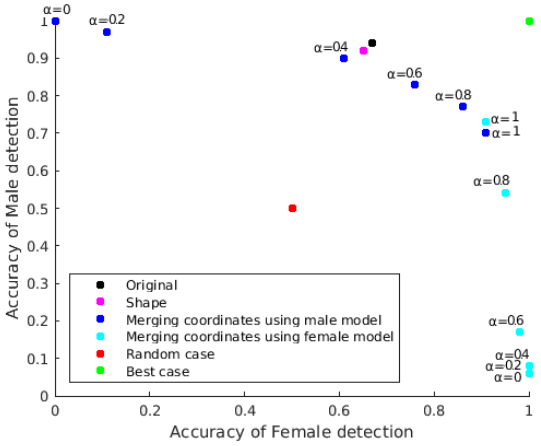


FIGURE 5.10: Results for the merging approach

The approximation using the convexity is a better approach than the other one. As with an intermediate value of n (i.e., 10) the gender detection becomes weak whereas the motion of the arms and legs are much more visible than with the approximation using the merging. This can be observed in the video available online [4]. Moreover, for the first method, given the knowledge of the reference model and α, an attacker might easily reverse the transformation, and then retrieve the original shape.

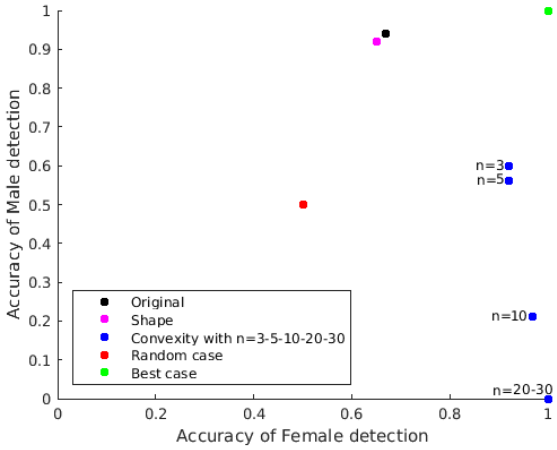


FIGURE 5.11: Results for the body approximation using convexity

[4] https://youtu.be/rpuIDLrHx3g

5.3.4.2 Evaluation of sport events classification

To evaluate the impact on event classification, we chose to test our algorithm on sports categorization. We utilize Deepdetect[5] to classify the sports and the UCF Sports [95] as dataset. We include further details of this tool and this database in the section 2.5.4.

For each selected images, we apply inpainting, JPEG high-Level scrambling, black masking, shape detection (5.3.1) and approximations using convexity (5.3.3) with $n = 3, 5, 10, 20, 30$ (Figure 5.12).

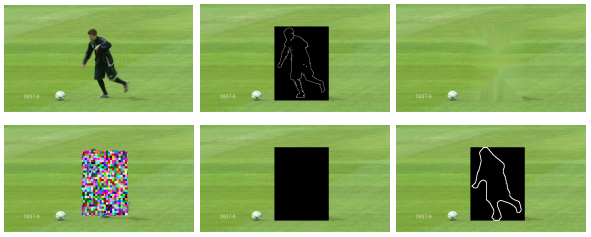


FIGURE 5.12: Respectively: Original image, shape, inpainting, scrambling, black masking and approximated body with $n = 10$

The classification tool outputs an ordered list of classes from the best to the worst one. Therefore, we compute and show in the Figure 5.13, the @k accuracy curve from k=1 to 10 (i.e., if the proper class is among the first k best results in the ordered list).

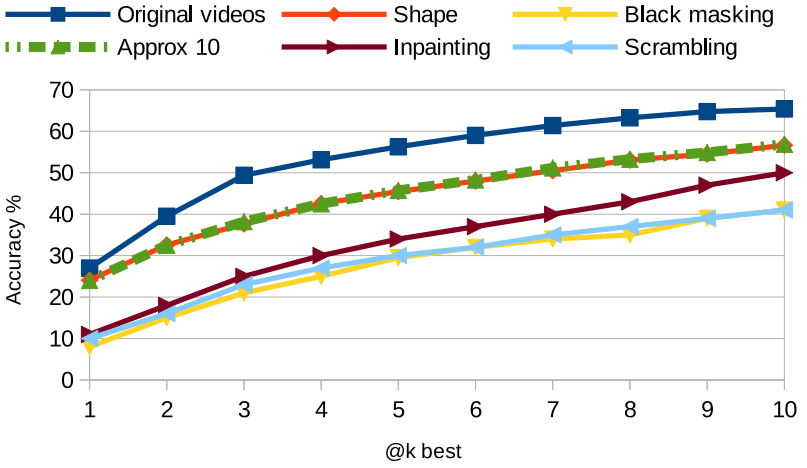
The results denote similar performances for sports classification in terms of shape and approximated body ($\sim 10^{-2}$ % of difference). Thus, removing the gender information by reshaping the shape does not impact the recognition of the sports compared when we use the original shape of the body. Moreover, the body approximation method achieves the best score compared to the black masking, the inpainting and the scrambling methods. However, the performances of sports classification 5.13(a) for the black masking, inpainting or scrambling filters, are closer than expected from the original ones. This is due to the background which helps a lot in the recognition of some sports (e.g., acrobatic gym, horse riding, diving, golf or football). Indeed, in 5.13(b), where the tool recognizes sport from the RoI only, the performances drop more than 10 % compared to the original ones.

Thus, our approximation using convexity preserves the global movement of a person as well as people activities even in the absence of the background contrary to the other methods that in the best cases preserve only the global motion.

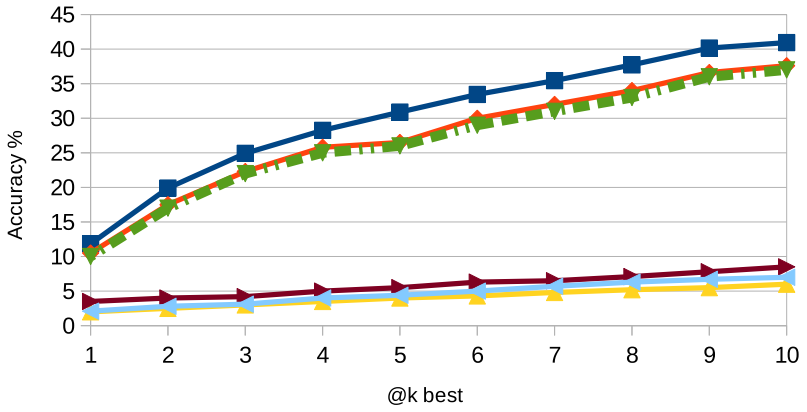
5.3.5 Optimal parameter for the body approximation using convexity

For each n, we average the accuracy of male and female detection, and the accuracy of the @k (from 1 to 10) best sports classification. We illustrate the results in the Table of the Figure 5.14. This Table reveals that both, the performances of gender detection and sports classification are decreasing when applying our body approximation approach. Nevertheless, the performances of sports classification slightly decrease (less than 1 %). Thus, the value of the parameter n does not much impact the sports

[5] http://www.deepdetect.com/



(a)



(b)

FIGURE 5.13: Accuracy@10 of sport events classification (a) using the whole images and (b) using the RoI only

classification. Consequently, the optimal value of n is the smallest value such as the accuracy of gender detection is equal to 50 %. According to our tests this value is $n = 18$.

n	3	5	10	17	**18**	20	30
Gender	76	74.4	58.8	51.5	**50**	50	50
Sports	45	44.93	44.74	44.63	**44.6**	44.27	44

FIGURE 5.14: Average accuracy in % of gender detection and of sports classification according to different values of n. Respectively: Original image and approximated body with respectively $n = 3, 5, 10, 20, 30$

To protect the gender as well as the identity while hiding original information, we should use the body approximated images using convexity with $n = 18$ as the cover image of the *StegoScrambling* approach

as explained in the section 5.2.3.

5.4 Conclusion

We have first presented a new method for privacy protection on videos from smart phones, cameras or sensors of drones by combining scrambling (an XoR is applied between random numbers and original pixels) and steganography (scrambled pixels are hidden inside the LSBs of the image pixels). The cover image is the edge map of the person. We proved that we protect the identity whereas we can still extract the gender information from the body shape.

For this reason, we have also designed two different body contours reshaping methods whose goal is not to only de-identify people, but also to hide gender, denoted as *de-genderization*. The first method transforms a body shape by merging its coordinates with the ones of a predefined model whereas the second one approximates a body shape using convexity. We prove, in the experiments, that both methods, hamper the recognition of the gender. Indeed, the gender resulting from the first method, converges towards the gender associated with the model, whereas the second method feminizes the body shape. Moreover, we demonstrate that the body approximation using convexity does not decrease the performance of sport events classification compared when using the body shape only.

The body contours reshaping methods presented in this Chapter are very dependent on the efficiency of the extraction of the body shape. Indeed, if we poorly detect the body shape, we will also degrade the visualization of actions. To improve the extraction of the body shape we could use a detector of key points of the body. The visualization of the images/actions would probably be better if instead of using a rectangle as a RoI we would use the exact shape of people.

The reverse process of the *StegoScrambling* method is not robust enough against compression. Indeed, it is not possible to compress the values of the scrambled coefficients without major deteriorations on the recovered images. Thus, such an approach is not compliant with the widely used coding standards. De facto, many applications cannot then use that tool. In the method proposed in the next Chapter, we apply the idea of keeping the scene understandable while scrambling and hiding the most significant information towards the least significant information. We operate in the Discrete Cosine Transform (DCT) domain instead of the spatial one to be compliant with classical standards such as JPEG and H.264/AVC and to keep the main colors of the image.

Chapter 6

Transform-domain scrambling, preserving the utility of visual surveillance

6.1 Introduction

This Chapter introduces a new privacy preserving surveillance filter ensuring that people are unrecognizable while keeping the scene understandable in terms of events which yields to detect abnormal behaviour. The algorithm operates in the **Discrete Cosine Transform (DCT)** domain to allow compression of data with the popular JPEG and H.264 standards. A detailed explanation of these standards is given in the **Appendix**. For each sensitive area of the picture (i.e., area where privacy needs to be protected), the proposed algorithm uses the low-frequency coefficients of the DCT to display a privacy preserved image of the region and the high-frequency coefficients to hide the majority of the original information. Moreover, we encrypt this original information. Finally, our process allows authorized users to recover almost the original data thanks to a password that is used to generate a sequence of random numbers (used for encryption).

Our main objective is to provide a good trade-off between privacy and utility of the visual surveillance while fulfilling the other criteria, introduced at the beginning of the section 2.5.1 and reported in the Table 2.3. Hence, the methods described in this Chapter, ensure privacy at any image size while preserving the recognition of sports in the scene. The proposed system is near lossless reversible for authorized people who own the key, and secure against brute force, parrot and replacement attacks. Moreover, the method is compliant with the widely adopted **JPEG** and **H.264** standards.

6.2 Proposed method within the JPEG standard

Our process operates between the quantization and the entropy encoding steps of the JPEG process as it is shown in Figure 6.1. There are two ways to access the quantized DCT coefficients:

1) by computing the first steps of the JPEG process when starting from a raw image: Color transformation / 8*8 blocks / DCT / Quantization.

2) by inverting the last step of lossless compression (entropy encoding) when starting from compressed stream.

The results of an 8*8 DCT block transform are 1 DC[1] coefficient and 63 AC[2] coefficients. We scramble the DC and AC coefficients of each 8*8 block of the region of interest (RoI) and shift them towards the high frequencies leaving the DC value available. We, then, automatically determine the average of some blocks as the value of the DC of each 8*8 block.

6.2.1 Principle of the process

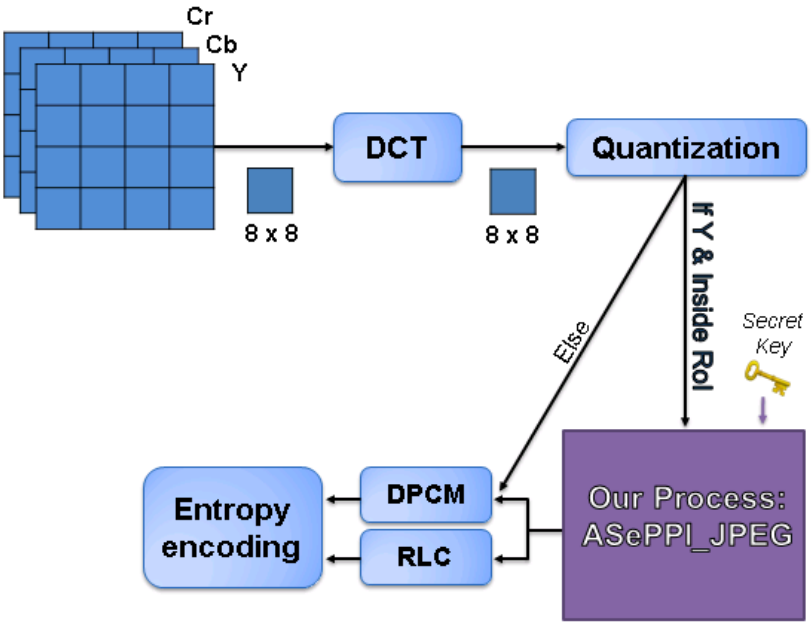
Operating on the chroma channels yields to unpleasant and unnatural colors in the protected RoI (more difficult to monitor). Thus, we operate on the luminance channel (Y) only. The purple square in the Figure 6.1(b) illustrates the steps of our process added to the ones of the classical JPEG framework.

Beforehand, we extract the first 62 coefficients (the DC + 61 AC) according to the zigzag code (Figure 6.2(a)). We intentionally skip the two least significant coefficients (i.e., the high frequency) because we will store later (as explained in the section 6.2.1.3) the encrypted DC into those two coefficients (and not only one in order to reduce the noise that it could create). Note that if the two least significant coefficients are null no information is lost. Therefore, the higher the quantization, the lower the loss created by this process is.

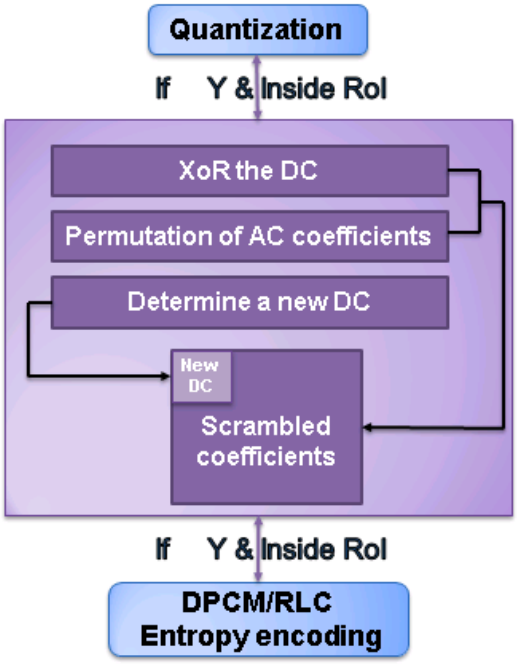
6.2.1.1 DC encryption

To encrypt the DC coefficients, we compute a bitwise XOR operation among each DC coefficient and a random number (between 0 and 127) generated by a pseudo-random number generator (PRNG) controlled by a secret key. This secret key is a password chosen and known by authorized users only (see the section 5.2.2 for the generation of the seed).

[1] The DC coefficient represents the average color of the block.
[2] The AC coefficients represent color variations across the block.



(a)



(b)

FIGURE 6.1: Workflow of the process. YCbCr is a color space representation where Y is the luminance component, and Cb and Cr the chrominance components. (a) The integration in the JPEG scheme (b) Steps added by our approach.

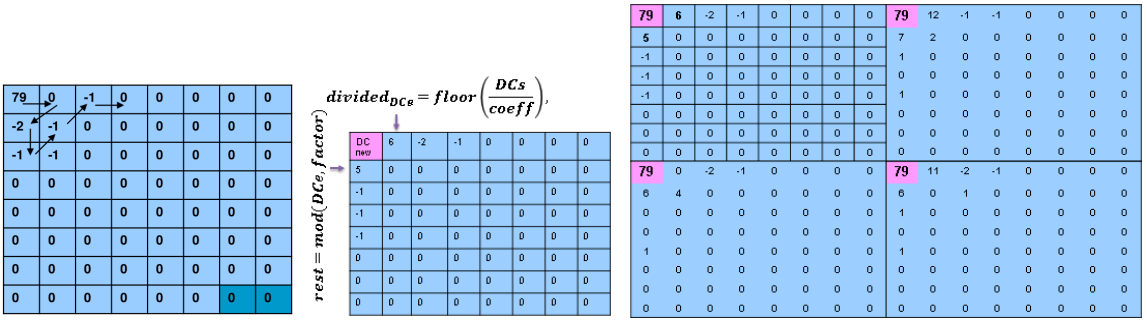


FIGURE 6.2: (a) Extracting coefficients using the zigzag code, (b) Insertion of the scrambled coefficients: [DC_{new}, 6, 5, -1, 0, -2, -1, 0, 0, -1, -1, EOB], (c) Insertion of the new DC coefficients of the corresponding b_{roi} with $S = 16$, (four 8*8 b_e and one 16*16 b_{roi}). Note that we only show the previous steps of the upper left block, in (a) and (b).

6.2.1.2 AC encryption

Note that n is the number of AC coefficients before End-of-Block (EOB). EOB are the remaining coefficients, the ones equal to zero. To scramble them, we randomly permute the $n - 1$ AC coefficients using the Knuth shuffle algorithm [123] that re-arranges their order. The last non-zero coefficient is used to mark the end of the permutation. In other words, the AC coefficients before the last non-zero coefficient are randomly permuted.

6.2.1.3 DC division and AC shifting

The value of the encrypted DC, denoted DC_e, could be high, and then, could produce too much distortion on the decompressed image if inserted as it is. Thus, this value is divided by a factor (set empirically to 17 in our implementation). We shift the 61 scrambled AC coefficients by two positions towards the high frequencies in order to hide the result of the DC division in the first AC coefficient, and the rest in the second one.

6.2.1.4 Example of the process

The extracted coefficients, given in the Figure 6.2 as an example (i.e., [79 (DC), 0, -2, -1, -1, -1, 0, 0, -1, EOB]) become the following scrambled coefficients: [107 (encrypted DC), -1, 0, -2, -1, 0, 0, -1, -1, EOB]. After shifting the coefficients and splitting the encrypted DC, the scrambled coefficients are [DC_{new}, 6 , 5, -1, 0, -2, -1, 0, 0, -1, -1, EOB]. Therefore, we re-insert these scrambled coefficients into the 8*8 block according to the zigzag code (Figure 6.2(b)).

6.2.1.5 Choice of the DC_{new}

We define b_e, the elementary blocks (of 8*8 size) where scrambled coefficients have already been hidden, and b_{roi} the meta blocks (with size to be defined) of the original RoI. In order to preserve the minimum information required by the surveillance (e.g., people activities), we insert the mean of b_{roi} in the DC of its corresponding b_e (Figure 6.2(c)).

The decompressed protected-image looks like a pixelated version of the original RoI with a bit of noise (Figure 6.4(c)). The pixelization is due to the preservation of the mean of b_{roi}, and the noise to the presence of the scrambled AC coefficients.

Keeping the DC of each 8*8 block, leads to a pixelated image of size 8*8. To handle the size of the pixelization (i.e., the strength of the protection), we add the parameter, S, where $S * S$ is the size of the b_{roi}. S is automatically defined in order to protect people privacy for any RoI size. For example, in the Figure 6.2(c), S is equal to 16, thus, the four 8*8 b_e corresponding to the 16*16 b_{roi} have the same DC coefficient being the mean of the 16*16 b_{roi}. S must be a multiple of 8 because the size of each b_e is 8*8. The higher the number of pixels inside the RoI the higher the parameter S should be. The next section explains how to automatically define/set S such that the identity is protected.

6.2.2 Automatically defining the size of the protection (i.e., the S value)

The equation (6.1) represents the relation between the size (S) of the blocks (i.e., the size of b_{roi}) and the number (Nb) of blocks inside the RoI, depending on the number of pixels (h x w) of that RoI.

$$Nb = \frac{h * w}{S * S} \tag{6.1}$$

We re write the relation in function of Nb knowing that S must be a multiple of 8 because the size of each elementary block (i.e., b_e) is 8*8.

$$S = round\left(\frac{\sqrt{\frac{h*w}{Nb}}}{8}\right) * 8 \tag{6.2}$$

The goal is to find the optimal Nb as large as possible (i.e., to keep as much as possible the scene information) such that the privacy is protected (e.g., a face recognition tool fails). Therefore, we apply our process with several values of Nb on different image sizes, and we evaluate the performance of recognition with a face recognition tool: OpenFace [84] available online (a pre-trained deep neural network).

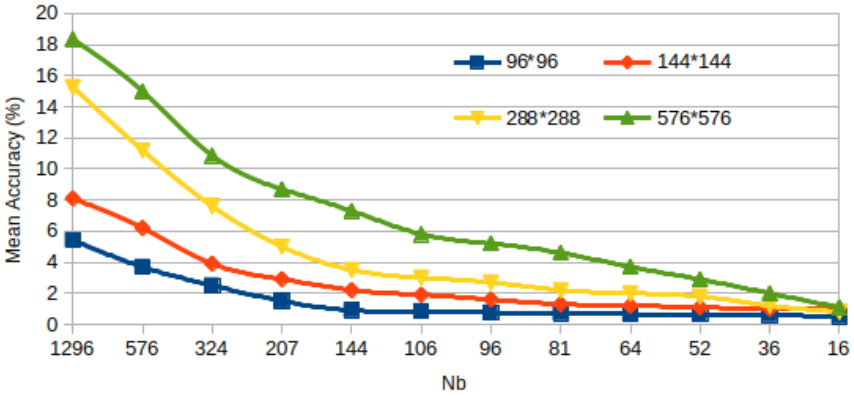


FIGURE 6.3: Accuracy of identity recognition (%) from faces depending on Nb and RoI size.

We randomly split into two parts (i.e., the training 75% and the testing 25% set) ten times the images of 158 people of the LFW Face database [124]. We obtain 84% of mean accuracy for the original images. In the section 2.5.3, we explain in more details OpenFace tool and the LFW Face Database.

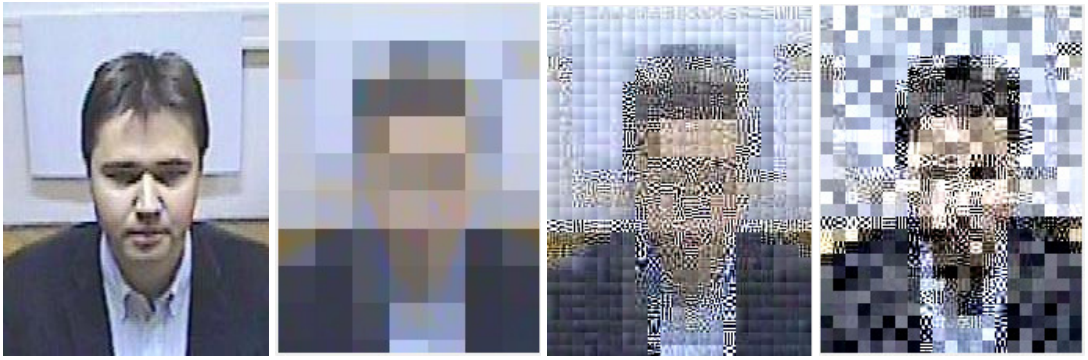
According to the Figure 6.3, we select $Nb = 106$, for our experiments, because the identification rate is under 6%, thus the privacy is well protected. However, we can easily change the value of Nb to have stronger or weaker protection depending on the application or depending on the quality of images recorded. In addition, we can also use this process to hide people attributes (e.g., gender, age, ethnicity). In this case, we need to redo the empirical studies to find the appropriate Nb. For example, if we want to hide the gender information, we apply our approach to several people image with different values of Nb, and we select the appropriate Nb values such as the accuracy of a gender detection tool decrease until 50 %.

6.2.3 Decompression with no secret key, using a basic decoder (Default mode)

The decompression without any secret key leads to visualize the image where the privacy is protected by our process (Figure 6.4(c)). We decode the compressed data with the inverse process of JPEG (a basic JPEG decoder).

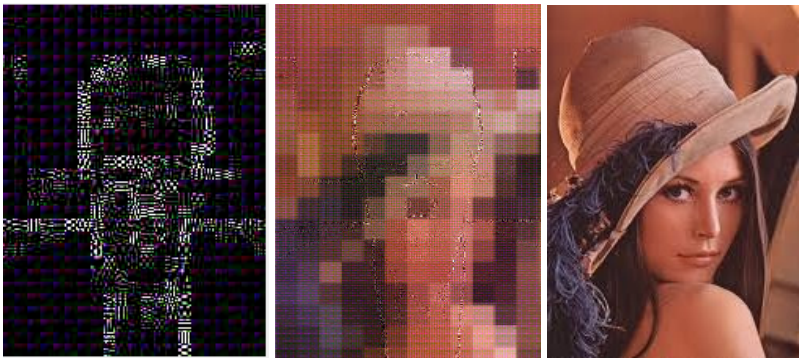
For image understanding, we need to preserve the visualization of events, therefore we keep the mean of some blocks (i.e., DC coefficient) from the original RoI. However, in other applications, replacing the image by a different one is preferable in order to get a stronger protection. Our approach allows this.

Instead of using the DC coefficient of blocks from the original image, other options are possible. For instance, we could simply insert several random DC or the DC of another image. In the first case, we will mainly obtain the shape of the original image (Figure 6.5(a)), and in the second case, we will observe a pixelated version of the other image with a small visibility of the shape of the original image (Figure 6.5(b)). This advantage of flexibility is useful for static privacy areas (e.g., License plate, number of a house, famous brand) where there is no necessity to extract activity inside the image.



(a) (b) (c) (d)

FIGURE 6.4: With S=24 and a JPEG quality of 75: (a) Original RoI, (b) only keeping the DC coefficient of each block for Y channel, (c) the protected RoI , and (d) the decompressed image when using a wrong secret key.



(a) (b) (c)

FIGURE 6.5: (a) The protected version using random DC, and (b) using the DC of the Lena picture (c).

6.2.4 Decompression with the secret key, using a modified version of the decoder

After the entropy decoding, we perform the inverse process of the additional steps created in 6.2.1, over each block of the luminance channel (Y) that are inside the RoI. Firstly, we extract the AC coefficients. To obtain the encrypted DC, the factor, given in 6.2.1.3, multiplies the first AC coefficient, and we add the result to the second AC coefficient (i.e., the rest). To recover the original DC coefficient, we compute a bitwise XOR operation between the encrypted DC and the random number associated. If the secret key is correct, PRNG generates the same random numbers than the ones used in 6.2.1. Using the reverse Knuth shuffle algorithm, we permute the other AC coefficients before the last non-zero coefficient. Then, the process applies the inverse quantization and DCT transformation over the decrypted coefficients. Finally, all blocks are put together, and we convert the recovered image in RGB (Red, Green and Blue channels).

The recovered image is similar to the original one, contrary to the Figure 6.4(d) which is unreadable when a user provides a wrong secret key.

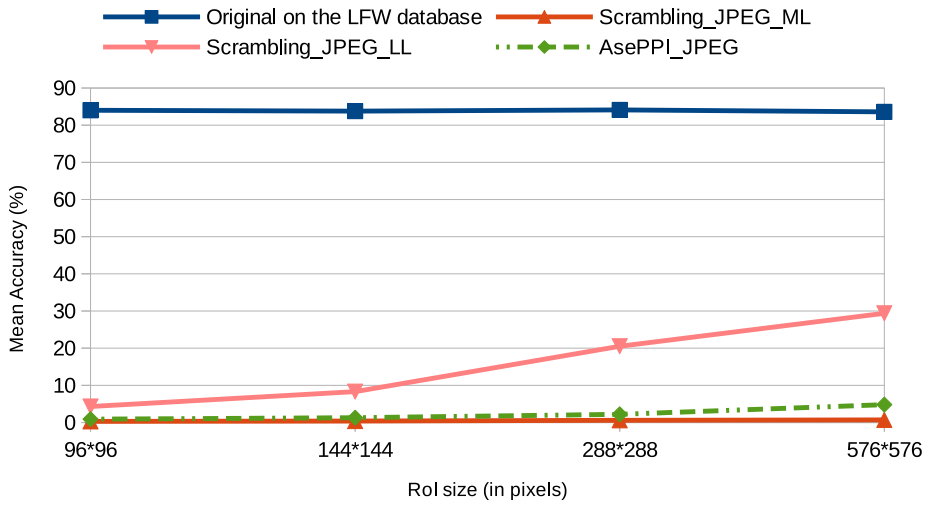


FIGURE 6.6: Accuracy of identity recognition depending on the privacy protection used and the RoI size.

6.2.5 Experimental results

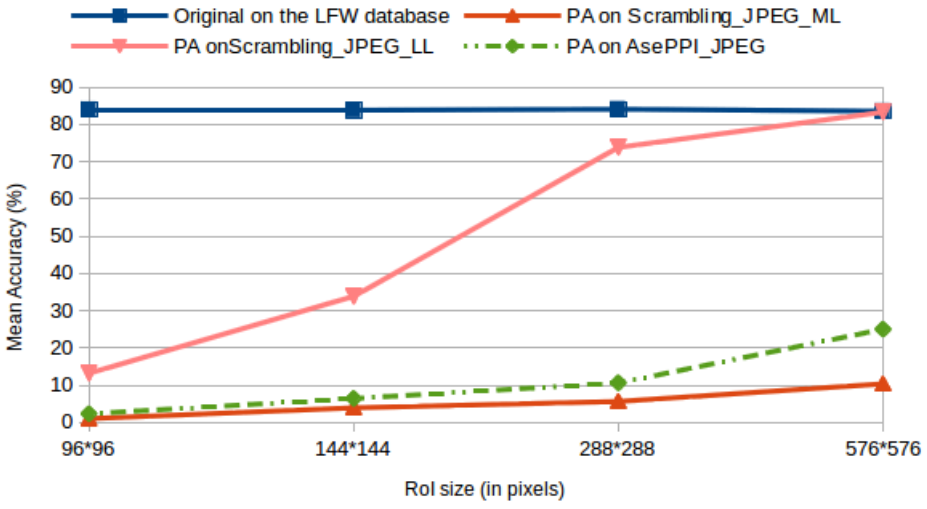
We evaluate and compare our proposed approach, named *ASePPI_JPEG*, over the six criteria, with one similar method: *Scrambling_JPEG* [6] at two different levels. This method randomly flips either the signs of the non-zero AC coefficients denoted *Scrambling_JPEG_LL* as the low level or the signs of the non-zero DC and AC coefficients denoted *Scrambling_JPEG_ML* as the medium level. We interfere only on the Y channel for a fair comparison with our proposed approach. We have presented in Chapter 2, other scrambling/encryption approaches, but they produce images similar to noisy pictures that is why we only compare our method with the *Scrambling_JPEG* one. Moreover, *Scrambling_JPEG* operates at the same position in the JPEG process (i.e., between the quantization and the entropy coding).

6.2.5.1 Evaluation of identity recognition from faces

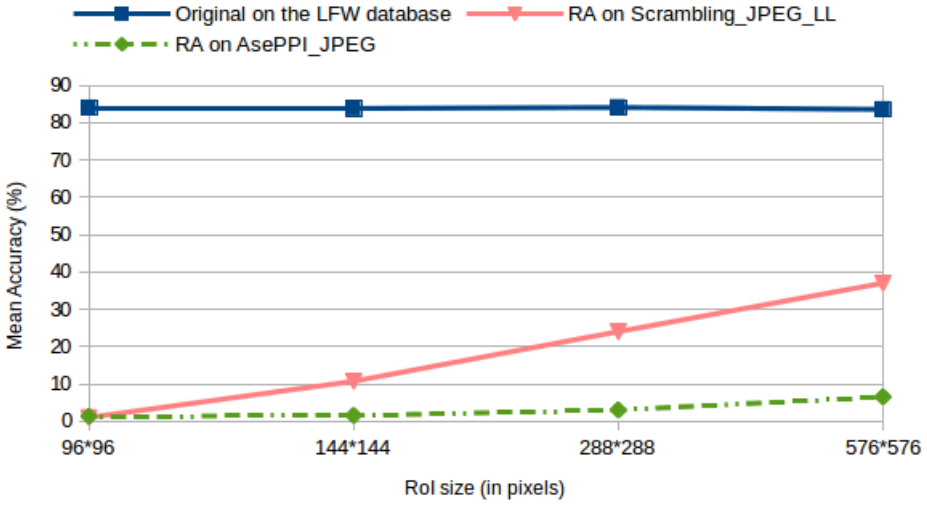
We apply the three different privacy protection methods over the face images of the LFW database with different sizes. We evaluate identity recognition by the face recognition tool (i.e., OpenFace, using the same protocol than the one in the section 6.2.2) over these images.

According to the Figure 6.6, we notice a rise in face identification rate for *Scrambling_JPEG_LL* when the RoI size increases. Indeed, the pixelization size remains the same (blocks of 8*8 pixels) regardless of the RoI size. For instance, for a RoI size of 576*576 we obtain 30% of mean accuracy for face recognition when we apply the *Scrambling_JPEG_LL* method. This is not enough to ensure the protection of privacy. Indeed, the goal is to have the identification rate as low as possible. On the contrary *ASePPI_JPEG*, automatically adapts the size of the pixelization effect (i.e., due to the new DC values), and then makes face recognition very difficult even for a RoI of large size. Indeed, the level of protection is similar to the ones of *Scrambling_JPEG_ML* (when in addition to the ACs scrambling, the DCs are corrupted by flipping their signs).

6.2.5.2 Robustness against Parrot and Replacement attacks



(a)



(b)

FIGURE 6.7: Accuracy of identity recognition with (a) Parrot Attack (PA) and (b) Replacement Attack (RA).

We add in the section 2.5.5, further explanations of the parrot and replacement attacks. A **parrot attack (PA)** [98] consists in performing both training and testing on the images on which the privacy protection has been applied.

A **replacement attack (RA)** [53] implies to set all encrypted values to zero while keeping the unencrypted values. We implement this attack on the *ASePPI_JPEG* and the *Scrambling_JPEG_LL* methods by setting all AC coefficients of the luminance channel to zero while keeping the DC values. Applying RA on these methods produces pixelated images of size 8*8 for the *Scrambling_JPEG_LL* approach and for *ASePPI_JPEG* one this size changes depending on the RoI size.

We evaluate the performance of face recognition (using the same protocol as the one in 6.2.2) after these two attacks, and we report the results in the Figure 6.7.

According to these results, we prove that our approach still protects the privacy for images of large size when applying attacks contrary to the *Scrambling_JPEG_LL* method. This is mainly due to the possibility to control the DC value of each block. Moreover, the level of the robustness is close to the one of *Scrambling_JPEG_ML* (when in addition to the AC scrambling, the DCs are corrupted by flipping their signs).

6.2.5.3 Robustness against brute force attack

We evaluate the security of the proposed technique against a brute-force attack. Assuming that the attacker knows the localization of the RoI and our algorithm, we consider an exhaustive search of all the possible combinations. We compute the average of the AC coefficients before EOB (i.e., End-Of-Block) among all the 8*8 blocks of the luminance channel. We denote this number as nbr_AC. Table 6.1 reports the nbr_AC for different JPEG quality [%] and over 600 face images of size 288*288 taken randomly from the LFW Face database.

JPEG quality [%]	[100-91]	[90-60]	[59-45]	[44-30]
nbr_AC	25	23	15	11

TABLE 6.1: Average number of AC coefficients flipped (ACF).

The number of possibilities per block for the permutation computed in *ASePPI_JPEG*, is $(nbr\_AC - 3)!$ if nbr_AC is greater than 61, otherwise $(nbr\_AC - 1)!$ because the last two AC coefficients are set to zero and the last non-zero AC coefficient remains at the same position. For the flipping computed in *Scrambling_JPEG_LL*, we have 2^{nbr_AC} possibilities and 2^{nbr_AC+1} (AC + DC scrambling) for *Scrambling_JPEG_ML*. We take the worst case for our approach (assuming that the last two AC coefficients are not null so we lose them) and the best case for the others (assuming that all AC before EOB are non-zeros coefficients).

$(nbr\_AC - 3)!$ is greater than 2^{nbr_AC+1} if nbr_AC is greater or equal to 10 which is the case for JPEG quality greater than 45% according to the Table 6.1. JPEG quality lower than 45% is rarely used. Thus, the scrambling designed in *ASePPI_JPEG* is more robust against a brute force attack than the two other approaches. Moreover, for a JPEG quality of 75 % (which is the default value of the JPEG reference encoder), we need to test $23!$ (denoted 2^{51}) combinations to recover an original 8*8 block.

Generally, a minimum size of the image is required for an identification. As an example, to be eligible for proving the identity of a person, the laws in France impose the face region to have at least 90 pixels between the bottom of the chin and the top of the skull or hair, and 60 pixels between the two ears (included) in a video [3]. Thus, 90 x 60 pixels should be the minimum size allowed to identify someone, and an image of this size contains 84 8*8 blocks. For a JPEG quality of 75 %, the number of combinations to test in order to recover an image that is protected by our process is $(23!)^{84} > 2^{4284}$. The number of

[3] http://www.telecoute.re/livre-blanc-conformite-v31.pdf

TABLE 6.2: Degradation comparisons with metrics.

For a JPEG quality of 75%	*JPEG*	*AsePPI_JPEG*	*Scrambling_JPEG_LL*	*Scrambling_JPEG_ML*
PSNR	37 dB	29 dB	32.2 dB	8 dB
SSIM	0.98	0.7	0.89	0.15
ESS	0.84	0.78	0.81	0.52
% of quality compared to JPEG	100%	80.89%	91.42%	32.94%

possibilities greater than 2^{128} already represents a limit impossible to reach with current technology [4]. Therefore, the *ASePPI_JPEG* method provides a presumably good level of security against brute-force attacks.

6.2.5.4 Evaluation of the visual utility preservation (i.e., intelligibility) using metrics

We compare the intelligibility with three different metrics: the **peak signal-to-noise ratio** (PSNR), the **structural similarity** (SSIM) and the **edge similarity score** (ESS) (explained in the section 2.5.4). We apply these metrics between the original images and the protected ones with 600 face images of size 288*288 from the LFW Face database.

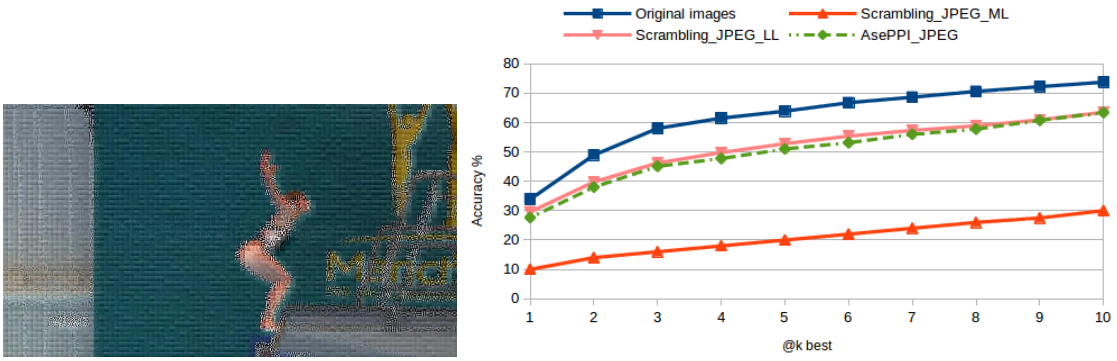
According to the Table 6.2, our approach slightly degrades the image compared to the *Scrambling_JPEG_LL* by 10.53 %, but preserves much more the appearance of the image than the *Scrambling_JPEG_ML* method by 47.95 %. Indeed, encrypting the DC values disturbs a lot the images whereas keeping the exact DC values preserves the global appearance of the pictures. However, in our approach we do not leave the exact DC (but close to the original).

6.2.5.5 Evaluation of the visual utility preservation by sport event classification

To evaluate the visual utility preservation, we choose to test our algorithm on sport event. We use Deepdetect[5] to classify sports and the UCF Sports as dataset [95]. We include further details of this tool and this database in the section 2.5.4. For each selected image, we apply the *ASePPI_JPEG*, *Scrambling_JPEG_LL* and *Scrambling_JPEG_ML* privacy protection methods on the whole images. That way the background will not skew the results. The classification tool outputs an ordered list of classes from the most to the least probable one. Therefore, we compute and show in the Figure 6.8(b), the @k accuracy curve from k=1 to 10 (i.e., if the proper class is among the first k best results in the ordered list).

According to the results plotted in the Figure 6.8(b), the accuracy of sports classification when applying *ASePPI_JPEG* on images decrease (in average, 20% of mean accuracy) compared to the original images. Moreover, they are better than the classification when using *Scrambling_JPEG_ML* (in average, 30% of mean accuracy) whereas slightly decrease when applying *Scrambling_JPEG_LL* (in average, 1% of mean accuracy). Indeed, keeping the mean color of some blocks helps to recognize the actions.

[4] http://cri.ensea.fr/en/node/208?destination=node%2F208
[5] http://www.deepdetect.com/



(a) (b)

FIGURE 6.8: (a) Our privacy protection applied on a sport image (diving) with $Nb = 106$, (b) Accuracy@10 of sport event classification.

TABLE 6.3: Impact on the efficiency of the JPEG process over the RoI parts and % of difference compared to JPEG.

For a JPEG quality of 75%	*JPEG* (100 %)	*AsePPI_JPEG*	*Scrambling_JPEG_LL*	*Scrambling_JPEG_ML*
PSNR	37 dB	37 dB (100 %)	37 dB (100 %)	37 dB (100 %)
SSIM	0.98	0.94 (95.92 %)	0.98 (100 %)	0.98 (100 %)
LSS	0.86	0.76 (88.37 %)	0.86 (100 %)	0.86 (100 %)
ESS	0.84	0.83 (98.81 %)	0.84 (100 %)	0.84 (100 %)
Compression Ratio	18.17	13 (71.54 %)	14.52 (79.86 %)	14.49 (79.75 %)
Time execution	0.2 s	0.205 s	0.202 s	0.202 s

6.2.5.6 Impact on the efficiency of the JPEG standard

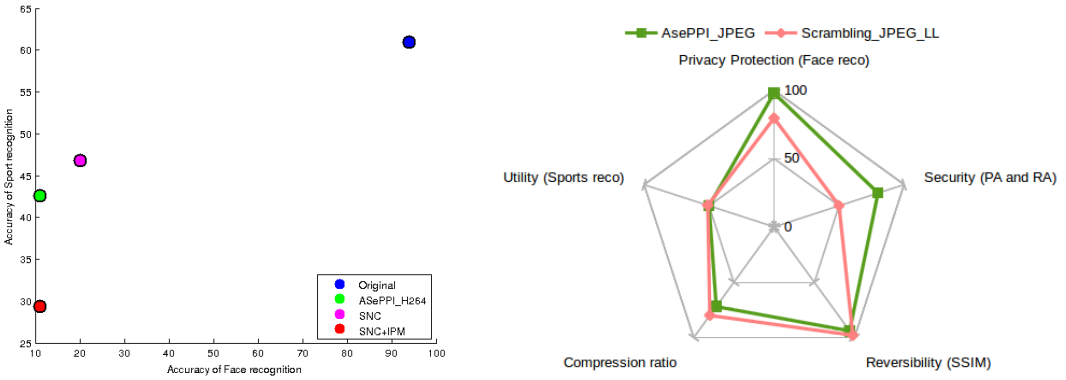
We measure the quality of the reconstructed images with the following metrics: the **peak signal-to-noise ratio** (PSNR), the **structural similarity** (SSIM), the **edge similarity score** (ESS) and the **luminance similarity score** (LSS). We provide more details about these metrics in the section 2.5.4.

For *ASePPI_JPEG*, *Scrambling_JPEG_LL* and *Scrambling_JPEG_ML* methods, the PSNR, the SSIM, the LSS and the ESS evaluate the visual quality of 600 reconstructed face images of size 288*288 taken randomly from the LFW Face database. We also compute the compression ratio with (6.3) and estimate the execution time of the different approaches using Python without GPU. We did these evaluations over the RoI parts (not over the whole image).

$$CompressionRatio = \frac{UncompressedSize}{CompressedSize} \tag{6.3}$$

The Table 6.3 shows the performance of each privacy protection method compared to the JPEG process baseline in terms of quality of the reconstruction, compression ratio and time execution.

Scrambling_JPEG_LL and *Scrambling_JPEG_ML* methods do not impact the quality of the recovered images because they keep all the information contrary to our approach that loses at worst two least significant coefficients for each block (there is no loss when the two least significant coefficients are



(a) (b)

FIGURE 6.9: (a) Face recognition VS Sport event classification, (b) Comparison between all criteria.

equal to 0). However, only the SSIM and the LSS metrics highlight a slight decrease when applying *AsePPI_JPEG* (around 8 % compared to JPEG).

All methods decrease the compression ratio especially when using our approach (around 30 % compared to JPEG). Indeed, we apply an XOR on each DC coefficient which cancels the benefit of its quantization. Moreover, the encrypted DC is split into two coefficients inside the AC coefficients which adds additional coefficients to store.

The impact on the execution time is minimal because the steps added (i.e., computing the new value of the DC from the original image, coefficients scrambling and shifting) are computationally low.

These differences are negligible because we perform our modifications only on the RoI, and usually the RoI is a limited subpart of the whole image.

6.2.5.7 Comparison between the performances of the different criteria

In [125] the authors propose a direct trade-off between privacy protection and data utility that is established through the introduction of the privacy operating characteristic (POC). The POC is a plot similar to the receiver operating characteristic (ROC), which is often used in pattern classifier evaluation. Therefore, we draw in the Figure 6.9(a), the average accuracy of sport event classification in function of the performance of face recognition.

ASePPI_JPEG performs better the trade-off between privacy protection and sport event classification compared to the other methods as shown in the Figure 6.9(a). Moreover, the Figure 6.9(b) highlights the significant improvement of the privacy protection and security (i.e., $100\%-$ average accuracy of face recognition) performances that our system procures compared to the slight decrease in performances of sport event classification, compression ratio and reversibility.

6.2.6 Conclusions

The process, presented in this Chapter, automatically adapts the privacy protection depending on the size of the sensitive regions. This allows the scalability for privacy protection in terms of image size.

We prove that our approach, *ASePPI_JPEG*, is the most appropriate privacy protection method to fulfill the criteria required for surveillance (defined in the section 2.5.1) because:

i) This approach protects privacy even over images of large size (which is not the case of the two other methods).

ii) We prove that the performance of sport classification decreases just a little bit after privacy protection, thus, the actions on the scene are still visible.

iii) The quality of the recovered images by our process is close to the one in JPEG.

iv) The process is sufficiently secure against brute-force attack and is robust against parrot and replacement attacks.

v) The method is compliant with the JPEG standard.

vi) The whole process works with a negligible additional computational time.

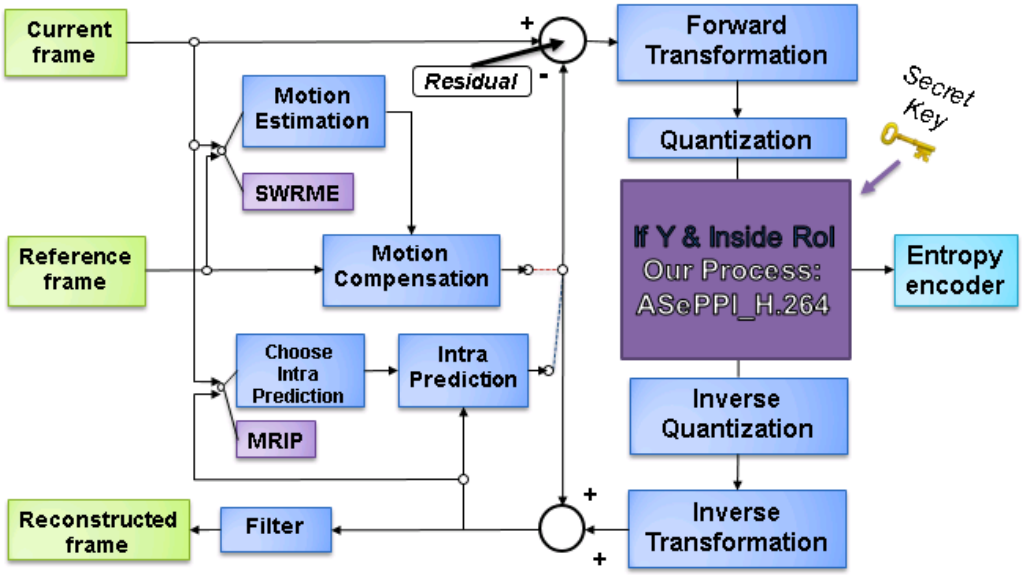
In the work presented in the following of this Chapter, we integrate a similar approach than the one defined in this section, in the H.264/AVC standard to be compliant with video compression.

6.3 Proposed method within the H.264/AVC standard

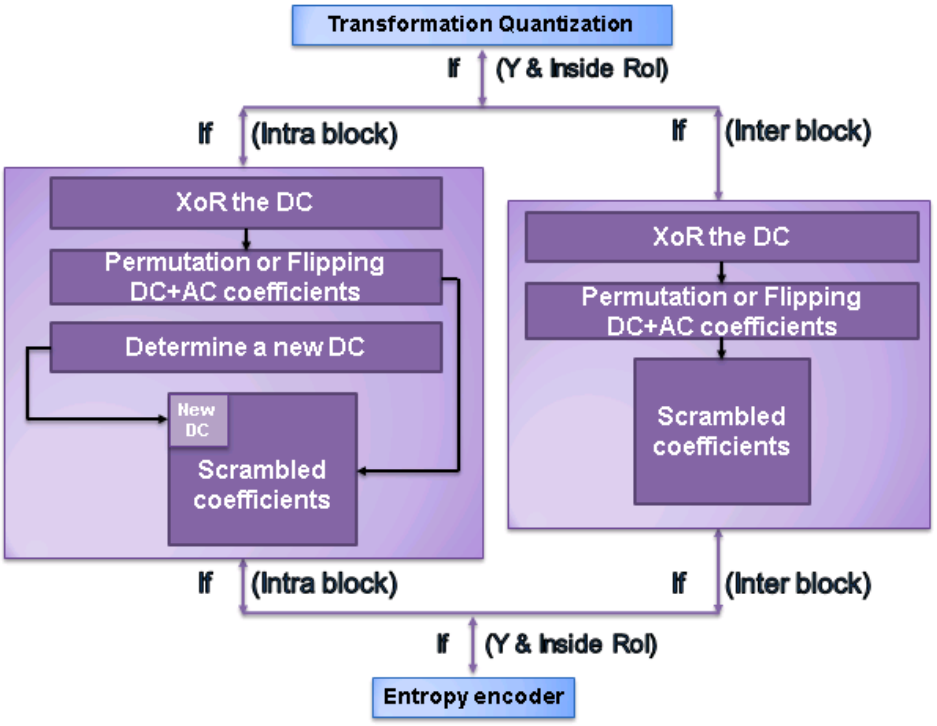
6.3.1 Principle of the process

We add few steps (i.e., see sections 6.3.1.1, 6.3.1.2, 6.3.1.3, 6.3.1.4) after the transformation and the quantization of each 4*4 residual block of H.264/AVC inside the RoI and for the luminance channel only, as we illustrate it in the Figure 6.10. We encrypt the DCT coefficients of the residual blocks for I and P frames (i.e., see sections 6.3.1.1, 6.3.1.2), with a pseudo-random number generator (PRNG) controlled by a secret key in order to protect data information and to be reversible by authorized people only (the section 5.2.2 provides more information about the PRNG). Then, for residual blocks of I frames only, we shift the encrypted coefficients from one position towards the high frequencies (i.e., see section 6.3.1.3) to make available the DC position (i.e., the lowest frequency representing the average of the values in a block). This position will be later used to insert an appropriate DC value (i.e., see section 6.3.1.4). The shifting step leads to lose the least significant coefficient of each block if it is a non-zero value.

We could have applied the extra steps only on I frame blocks. However, residual blocks of P frame needed to be encrypted because no information should be deduced from them. Indeed, we can predict a block of P frame from protected blocks very similar to the original one, thus, the higher the number of



(a)



(b)

FIGURE 6.10: Workflow of the process. (a) The integration in the H.264 scheme (b) Steps added by our approach for the residual Intra and Inter blocks.

successive P frames the closer to the real blocks are the protected blocks. That is why, we also encrypt all the DCT coefficients of the residual blocks of P frames. Nevertheless, we do not insert a new value of the DC for P blocks because they are predicted from I blocks on which the minimum of information required by surveillance are already kept/preserved by our proposed approach.

6.3.1.1 DC encryption

Algorithm 7: DC encryption.

if *(*$|DC| < 16$*)* **then**
  $X = 16$;
else
  $X = 2^n$;
end
if *(*$DC \neq 0$*)* & *((*$|DC| \neq (RN \mod X)$*))* **then**
  DC_e = ($|DC| \oplus (RN \mod X)$)*sign(DC);
else
  DC_e = DC;
end
with $n = \lfloor \log_2 |DC| \rfloor$ an integer

Doing an XOR between the DC and a random number generated from an infinite range may lead to bigger encrypted DC than the original one. This value difference produces noise in the decompressed privacy-protected images. Therefore, to minimize this noise, we design an encryption algorithm where the encrypted values will remain in the same range than their original one. According to the algorithm (7), a $|DC| \in [2^n, 2^{n+1}]$ produces $(RN \mod 2^n) \in [0, 2^n[$ (i.e., $[0, 2^{n-1}]$), thus, $DC_e \in [2^n, 2^{n+1}]$ with RN a random number generated by a PRNG, $sign(DC)$ equal to -1 if the DC sign is negative and +1 otherwise, and DC_e the encrypted DC. For values lower than 16, we increase the range of values (i.e., $[0, 15]$) to create more possibilities. For instance, without adding this condition, all DC equal to 1 will remain at the same value, unlike when using the proposed algorithm (7) that creates 16 possibilities in this case.

This encryption algorithm (7) leaves the DC as it is in two cases: (i) if the DC is null and, (ii) if the DC is equal to $(RN \mod X)$ (giving 0) in order to (i) avoid too much degradation in the decompressed privacy-protected images and (ii) to obtain the DC_e in the same range than the DC.

6.3.1.2 Scrambling the coefficients (the encrypted DC + the original AC)

For each block, we select the scrambling method (between RP and SNC, explained in the following) that creates the highest number of combinations to recover the original data. The encrypted DC is included in the scrambling. We use the PRNG to generate a sequence of random numbers.

RP: Let p_1 be the number of coefficients before EOB (End-of-Block, the remaining coefficients are zero). To scramble them, we randomly permute the $p_1 - 1$ coefficients using the Knuth shuffle algorithm [123]

that re-arranges their order. The last non-zero coefficient is used to mark the end of the permutation (i.e., the coefficients before the last non-zero coefficient are randomly permuted). Thus, there are $(p_1 - 1)!$ combinations.

SNC: Let p_2 be the number of non-zeros coefficients. We flip randomly the sign of these p_2 coefficients. Therefore, there are 2^{p_2} combinations.

Both, RP and SNC are reversible methods. For I blocks only, before applying this step, the last AC coefficient is voluntary lost because of the shifting step, so we set it to 0.

6.3.1.3 Shifting the scrambled coefficients to the AC ones (for I blocks only)

As an example, we suppose that the original extracted coefficients are [31 (DC), 0, -2, -1, -1, -1, 0, 0, -1, EOB]. We encrypt the DC which becomes 24: [24 (DC_e), 0, -2, -1, -1, -1, 0, 0, -1, EOB]. There are $8! = 40320$ combinations with the RP method and $2^6 = 64$ with the SNC one. Thus, we select the RP method to scramble the coefficients which becomes [-1, 0, 24 (DC_e), -2, -1, 0, 0, -1, -1, EOB], and we shift them by one position towards the high frequencies which leads to [DC_{new}, -1, 0, 24 (DC_e), -2, -1, 0, 0, -1, -1, EOB]. Then, we re-insert the scrambled coefficients into a block according to the zigzag code and choose the new DC value with the formula defined in section 6.3.1.4.

This step leads to lose in the worst case one coefficient per block, but as we report in the Table 6.4 there are few blocks with the last AC coefficient not equal to zero.

6.3.1.4 Choice of the DC_{new} value (for I blocks only)

We dedicate the DC_{new} value to restitute some of the original information (e.g., the average luminance of a residual block).

Keeping the original DC (i.e., the mean) of each residual block of the luminance channel leads to a slight smoothing of the face. To get stronger privacy protection, we keep the DC of a bigger block and insert it in the DCs of its 4*4 sub-blocks (following the same process than in the section 6.2.1.5). For example, in the right picture in the Figure 6.11, we kept the original DC of each 4*4 residual block, and in the left one, we inserted the DC of each block of size 24*24 inside the DCs of its corresponding 4*4 sub-blocks.

The equation (6.2) already defined in the section 6.2.2, represents the relation between the size of the blocks (i.e., b_{roi}), denoted by S, and the number of blocks inside the RoI, denoted by Nb, depending on the number of pixels (h x w) of that RoI. For instance, if S is equal to 24, the residual blocks inside the 24*24 block have the same DC coefficient, which is the DC of the 24*24 block (i.e., the mean of the 24*24 block).

The higher Nb (i.e., the higher the image quality is), the better the recognition is. Our goal is to find the maximum Nb to preserve as much intelligibility (i.e., utility of the video surveillance) as possible while minimizing the performance of face recognition. Therefore, in order to fulfill this purpose, we did, in the section 6.2.2, an empirical study by fixing several values, and we find in our case that Nb should be

FIGURE 6.11: Keeping only the DC of each block of the luminance channel with $h = 204$ and $w = 220$ (on the RoI). Blocks size: 4*4 for the right image and 24*24 for the left one.

equal to 106. Thus, we automatically compute S with the equation 6.4 taking care that the size of each residual block is 4*4. However, we can change the value of Nb to have stronger or weaker protection.

$$S = \left\lceil \frac{\sqrt{\frac{h*w}{106}}}{4} \right\rceil * 4 \tag{6.4}$$

6.3.2 Decompression with/without secret key

The decompression without any secret key leads to visualize the decompressed protected-video (illustrated in [6]). We decode the compressed data with a basic H.264 decoder.

To recover the initial video, we apply the reverse process using the correct secret key. Since we predict each block from unencrypted blocks during the encoding, we only need to decrypt the scrambled residual blocks (i.e., the ones inside the RoI). The correct secret key generates the same random numbers than the ones in the encoding part which allows recovering the original data. For I frames, we extract the AC coefficients (not the DC) of the residual blocks inside the RoI, and for P frames all the coefficients (DC+AC) of these blocks. As in section 6.3.1.2, we select the scrambling method (RP or SNC) that produces the highest number of combinations to recover the original data and, then, apply its reverse process. Finally, we decrypt the DC (i.e., the first decrypted coefficient) by applying the algorithm 7.

6.3.3 Experimental Results

We compare the proposed method, denoted *ASePPI_H.264*, with the encryption of the signs of the non-zero coefficients *SNC* and with the addition of the encryption of the intra prediction modes *SNC+IPM* within the privacy area. We apply all methods only on the luminance channel for a fair comparison with our proposed approach. Three video examples of the application of these methods are available online[6].

[6] www.dropbox.com/s/r0hbc8n48ocu4uk/Video_Examples.zip?dl=0

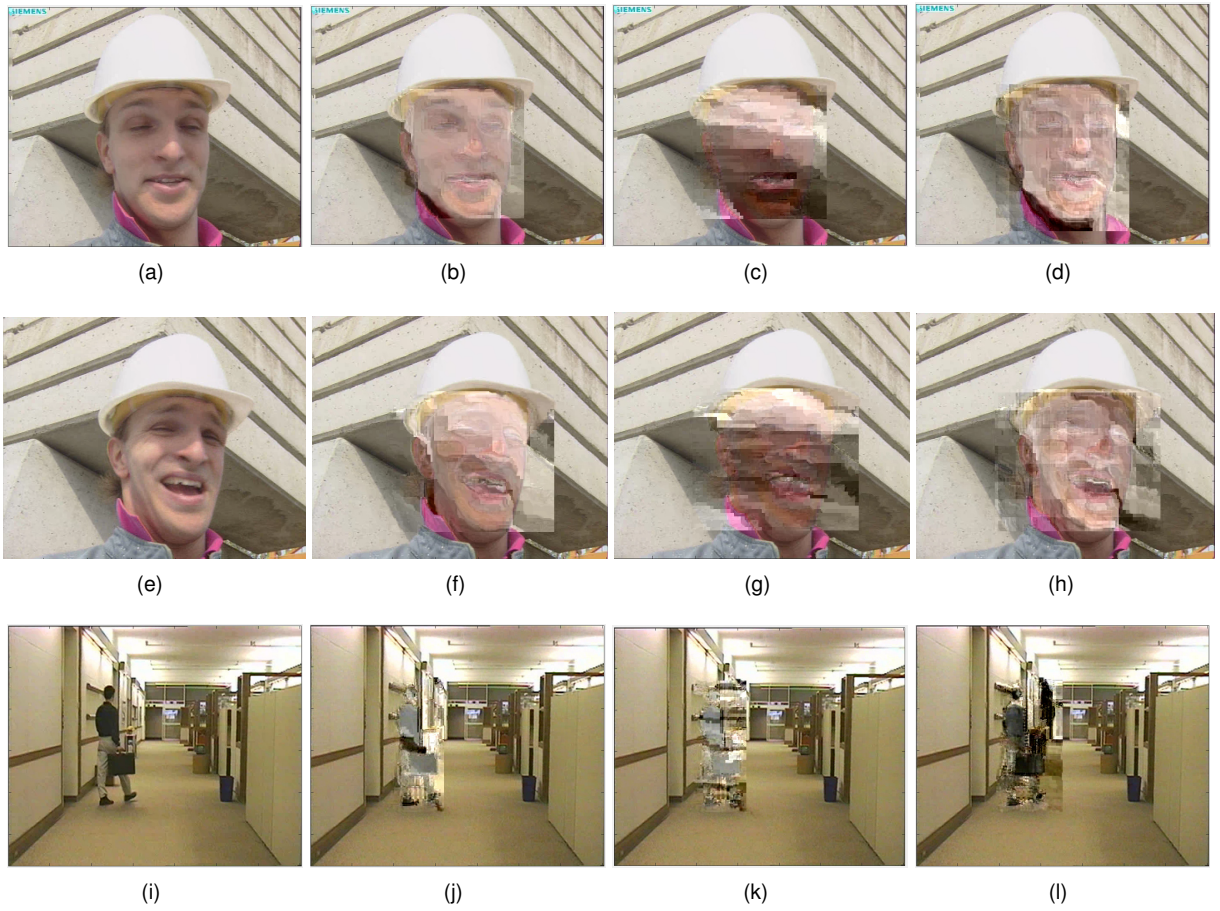


FIGURE 6.12: With CIF size, QP= 24 and IP= 5: (a) The 1st original frame of the 'foreman' sequence (I), (e) the 15th original frame of the 'foreman' sequence (P), (i) the 39th original frame of the sequence 'hall' (P), (b), (f) and (j) encrypted by SNC, (c), (g) and (k) encrypted by SNC+IPM, (d), (h) and (l) encrypted by *ASePPI_H.264*.

For the evaluation, we have selected the following sequences: 'hall', 'foreman', 'suzie', 'akiyo', 'carphone', 'claire', and 'miss-america' all available on the web[7], and 3 videos for each of the 9 persons that have the most data in the YouTube Face database to evaluate the identity recognition. We provide more information of this database in the section 2.5.3. We use different values of QP and IP in our evaluations. QP is the quantization parameter and IP the intra period that defines the frames number between two I frames.

6.3.3.1 Evaluation of identity recognition from faces

We train the OpenFace tool [84] with two videos of 9 people from the YouTube database and test on the remaining video of each people (different from the ones that have been used in the training part). For original faces, we get 93.95% of mean accuracy (i.e., correct classifications). We notice in the Figure 6.13 that the larger the RoI size the higher is the accuracy (more than 30 % for 576*576) for faces

[7] http://trace.eas.asu.edu/yuv/

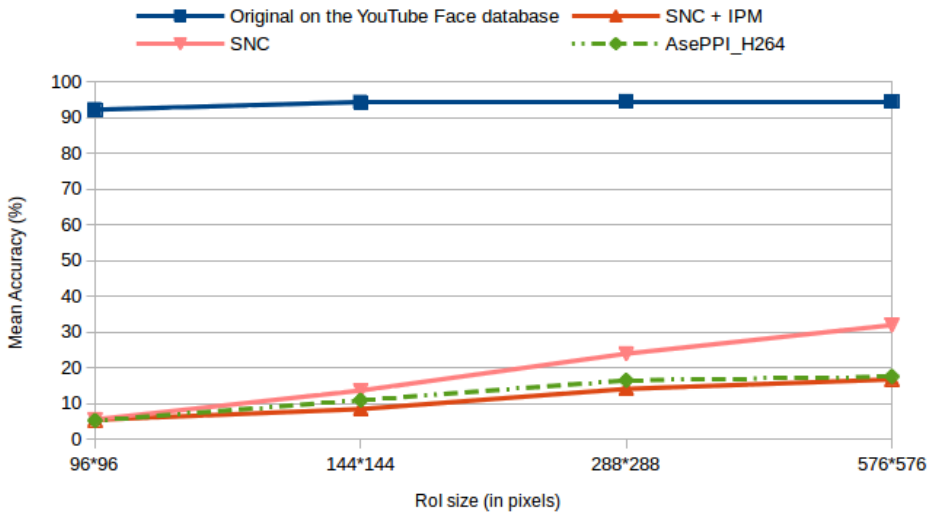


FIGURE 6.13: Accuracy of identity recognition depending on the privacy protection used and the RoI size.

protected by SNC (see the Figures 6.12(b), 6.12(f), 6.12(j)). With the two other methods, SNC+IPM (see the Figures 6.12(c), 6.12(g), 6.12(k)) and *ASePPI_H.264* (see the Figures 6.12(d), 6.12(h), 6.12(l)), less than 17% of faces are well identified. Therefore, SNC method has issues in protecting the privacy at any resolution contrary to the two other approaches.

6.3.3.2 Robustness against a parrot attack (PA)

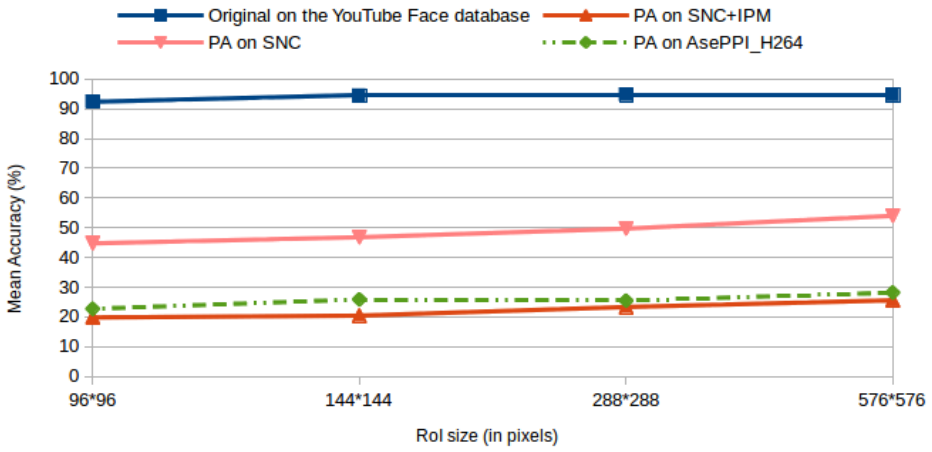
Using the same protocol than the one presented in the previous section 6.3.3.1 (with the YouTube faces database [126] and OpenFace tool), we train and test on the images on which the privacy protection methods have been applied.

We report, in the Figure 6.14(a), the accuracy of identity recognition when applying the parrot attack on each privacy protection methods. We notice that we can re-identify people with more than 50 % when PA attacks the SNC approach. The results of PA attack on the *ASePPI_H.264* and *SNC+IPM* methods are almost equal (around 24 % of re-identification).

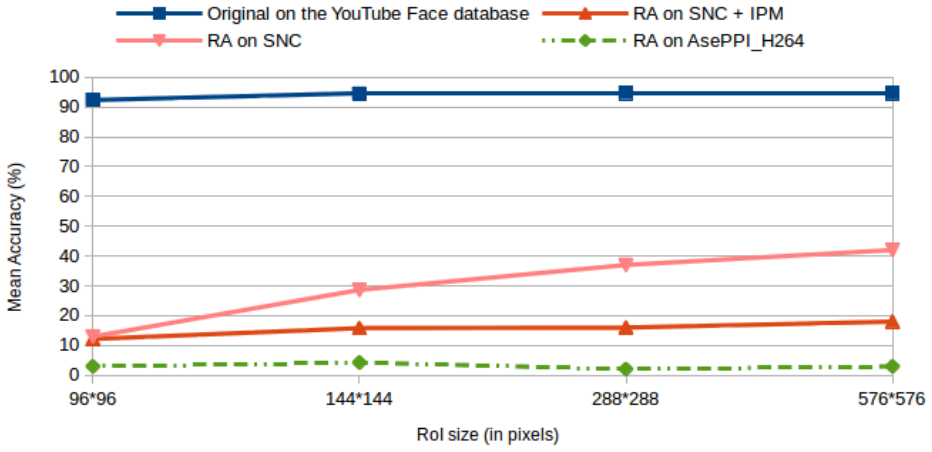
In privacy protection, the lower the value of the accuracy of identification, the better the performance of the method. We cannot claim that we totally protect the privacy because we identify people with more than 20 % of correct recognition. Indeed, it means that we correctly identify more than one person out of five. However, the experiments have been performed using only 9 persons that makes the training easier to learn.

6.3.3.3 Robustness against replacement attack (RA)

An attacker may try to suppress encrypted data (i.e., set to zero all the encrypted coefficients). This attack consists in extrapolating the scrambled data by motion compensation from the previous frame



(a)



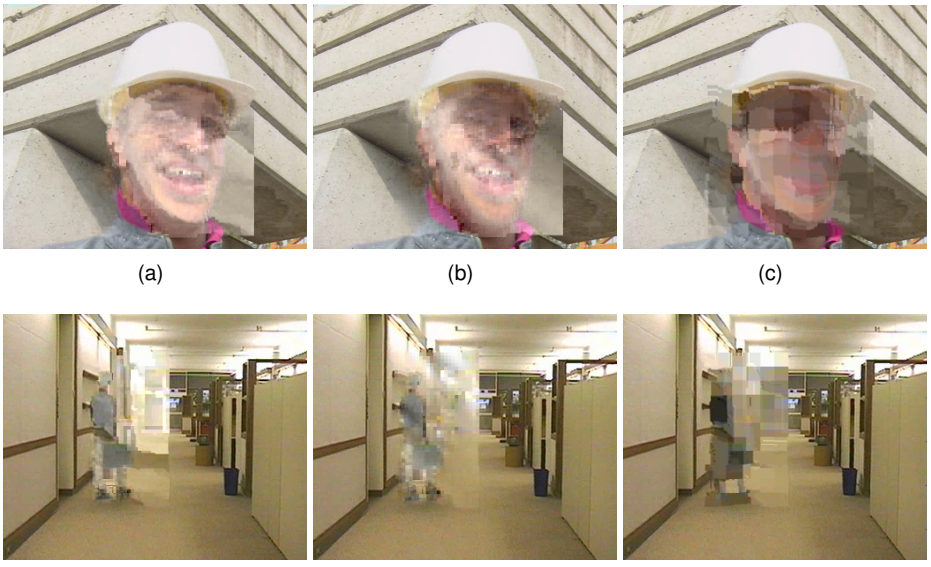
(b)

FIGURE 6.14: Accuracy of identity recognition with (a) Parrot Attack (PA) and (b) Replacement Attack (RA).

using the motion vectors which are available to the attacker. Thus, to implement this attack in our method, all AC coefficients of the RoI for the I frames and all DCT coefficients (DC+AC) of the RoI for the P frames are set to 0. For SNC method, we simply set to 0 all AC coefficients of the RoI, and for SNC+IPM we set, in addition, all intra prediction modes to 2 (i.e., the mean). We produced video examples [8], with 'foreman' and 'carphone' sequences, on which the replacement attack is applied to the different methods.

We report, in the Figure 6.14(b) , the accuracy of identity recognition with OpenFace CNN tool (using the same protocol than the previous section 6.3.3.1) when the replacement attack is applied to the different methods. According to the results, the SNC method becomes weaker in terms of privacy protection when the RoI size increases because the algorithm re-identifies people with more than 40% of correct recognition, whereas with our method the identity recognition rate is still under 5%.

[8] www.dropbox.com/s/39ke5wy6mgezq4k/Video_Examples_SA.zip?dl=0



(a) (b) (c)

(d) (e) (f)

FIGURE 6.15: With CIF size, QP= 24 and IP= 5: After the replacement attack on the SNC privacy protection (a) and (d), on the SNC+IPM privacy protection (b) and (e), on the *ASePPI_H.264* one (c) and (f).

The replacement attack on the SNC method sets the AC coefficients to 0, whereas the DC of each 4*4 residual block are still available and retain too much information. Thus, the privacy is not enough protected especially on high-resolution images. For *ASePPI_H.264* method, we automatically adapt the DC values by using the same DC for multiple blocks in order to protect the privacy at any resolution which makes it stronger/robust against the replacement attack.

The accuracy of re-identification using the replacement attack on the *SNC+IPM* method stays around 15.5 %. However, from a human point of view, the details of the face or the shape of the body can be much more visible than in the case of *ASePPI_H.264* as it is illustrated in Figure 6.15 and in the video examples [13].

6.3.3.4 Robustness against brute force attack

We consider an exhaustive search of all combinations. The number of combinations per block for *ASePPI_H.264* method is always greater than or equal to the one of SNC. Indeed, as explained in section 6.3.1.2, we select the method (between SNC or RP) that performs the highest number of combinations. Therefore, we only compare the *SNC+IPM* method with *ASePPI_H.264* on I frames.

We compute the average number of the AC coefficients before End-Of-Block (EOB) as well as the average number of the non-zeros AC coefficients before EOB among all the 4*4 blocks of I frames of the luminance channel. We, respectively, denote these numbers as nbr_AC and nbr_NAC and we report them in the Table 6.4 for different QP values over the I frames of the 7 sequences.

QP	12	18	24	30
nbr_AC	11.43	8.32	4.94	2.35
nbr_NAC	8.04	4.96	2.51	1

TABLE 6.4: Average number of AC and non-zeros AC coefficients before EOB.

The number of possibilities per block of I frames for the *ASePPI_H.264* is $max(2^{nbr\_NAC}, nbr\_AC!)$, and for *SNC+IPM* $(2^{nbr\_NAC}) * 9$ (9 intra modes). We do not count the number of combinations to decrypt the DC (for our process) because we assume that the DC can be deduced from the DC_{new}. Thus, we compute and report the results in the Table 6.5. According to this Table, we deduce that our approach is more secure in terms of brute force attack than *SNC+IPM* for QP lower or equal to 24.

QP	12	18	24	30
ASePPI_H.264	3.99*10^7	40320	120	5.1
SNC+IPM	2.37*10^3	280	51	18

TABLE 6.5: Average number of combinations to recover one encrypted block of I frames.

As already stated in the section 6.2.5.3, a minimum size of the image is required to identify someone (e.g., 90 x 60 pixels for face, in France [9]). An image of this size contains 337.5 4*4 blocks. To recover an I frame, with QP = 30 and QP = 24, the number of combinations is, respectively, $5.1^{337.5} = 10^{238} > 2^{512}$ and more than 2^{2048}. The number of possibilities greater than 2^{128} already represents a limit impossible to reach with current technology [10]. Therefore, the method provides a good level of security.

6.3.3.5 Evaluation of the visual utility preservation (i.e., intelligibility) using metrics

Using both SNC and IPM hampers the global understanding of the scene or human actions. For example, in the Figure 6.12(k) it is not obvious that the protected area contains a person carrying her bag whereas in the Figure 6.12(l) the shape of the head and feet are clearly distinguishable as well as the bag.

We evaluate the utility preservation of the video surveillance with three different metrics: the **peak signal-to-noise ratio** (PSNR), the **structural similarity** (SSIM), and the **edge similarity score** (ESS) (all explained in the section 2.5.4). We apply these metrics on the original RoI (size of around 288*288 pixels) and the protected one of each frame of the seven sequences (with IP= 1, 5, 10, 30 and QP= 24).

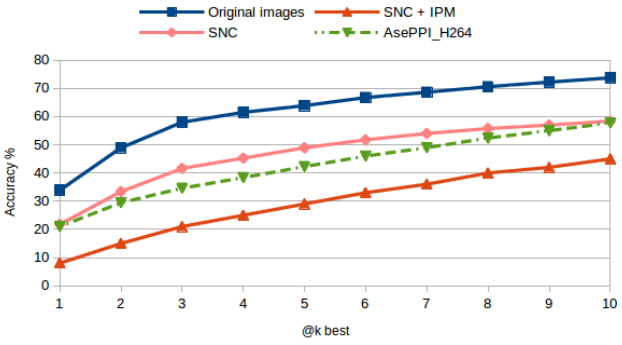
According to the results of the Table 6.6, we conclude that the *SNC + IPM* method degrades the visibility of the scene more than *ASePPI_H.264* (around 24 % more). Indeed, the encryption of the Intra Prediction Mode (IPM) leads to predict blocks from wrong ones (not the same ones as in the encoding). This produces disturbances. Moreover, in *ASePPI_H.264*, we design the encryption of the DC so that it produces limited noise (less than 14 % of quality degradation compared to JPEG and less than 2 % compared to *SNC*).

[9] http://www.telecoute.re/livre-blanc-conformite-v31.pdf
[10] http://cri.ensea.fr/en/node/208?destination=node%2F208

TABLE 6.6: Degradation comparisons with metrics.

For QP=24%	*H.264/AVC*	*ASePPI_H.264*	*SNC*	*SNC+IPM*
PSNR	42.78 dB	32.78 dB	34 dB	27.48 dB
SSIM	0.98	0.87	0.89	0.45
ESS	0.86	0.81	0.82	0.66
% of quality compared to H.264	100 %	86.53 %	88.55 %	62.3 %

(a)



(b)

FIGURE 6.16: (a) Our protection applied on a sport image (football) with $Nb = 106$, (b) Accuracy@10 of sport event classification.

6.3.3.6 Evaluation of the visual utility preservation by sport event classification

To evaluate the utility preservation of video surveillance, we choose to test our algorithm on sport event. We use Deepdetect[11] to classify sports and the UCF Sports as dataset [95]. We include further details of this tool and this database in the section 2.5.4. For each selected image, we apply the *ASePPI_H264*, *SNC* and *SNC+IPM* privacy protection methods on the whole image. That way the background will not skew the results. The classification tool outputs an ordered list of classes from the most to the least probable one. Therefore, we compute and show in the Figure 6.16(b), the @k accuracy curve from k=1 to 10 (i.e., if the proper class is among the first k best results in the ordered list).

According to the results plotted in the Figure 6.16(b), the accuracy of sport classification when applying *ASePPI_H.264* on images decrease (in average, 20% of mean accuracy) compared to the original images. Moreover, their performances (with *ASePPI_H.264*) are better than the ones when applying *SNC+IPM* (in average, 13% of mean accuracy) whereas slightly decrease when applying *SNC* (in average, 4% of mean accuracy). Indeed, keeping the mean color of some blocks helps to recognize the actions.

[11] http://www.deepdetect.com/

TABLE 6.7: Impact on the efficiency of the H.264/AVC process over the RoI parts.

For QP=24 and different IP %	*H.264/AVC* (100 %)	*AsePPI_H.264*	*SNC*	*SNC_IPM*
PSNR	42.78 dB	39 dB (91.16%)	42.78 dB (100 %)	42.78dB (100 %)
SSIM	0.98	0.95 (96.94 %)	0.98 (100 %)	0.98 (100 %)
LSS	0.87	0.86 (98.85 %)	0.87 (100 %)	0.87 (100 %)
ESS	0.86	0.84 (97.67 %)	0.86 (100 %)	0.86 (100 %)
Bit overhead (%)	0	11 %	2.97 %	5 %

6.3.3.7 Impact on the efficiency of the H.264/AVC standard

We measure the quality of the reconstructed images with the following metrics: the **peak signal-to-noise ratio** (PSNR), the **structural similarity** (SSIM), the **edge similarity score** (ESS) and the **luminance similarity score** (LSS) (more details are provided in the section 2.5.4). We apply these metrics on the original RoI (size of around 288*288 pixels) and the reconstructed one of each frame of the seven sequences (with IP= 1, 5, 10, 30 and QP= 24).

We lost at worst one coefficient per block only for the I frames, but since blocks in P frames are predicted from blocks of I frames we also lost some information for P frames. However, according to the results reported in the Table 6.7, the quality of the reconstructed images using our approach is close to the one of H.264/AVC (less than 4 % of the drop of performance).

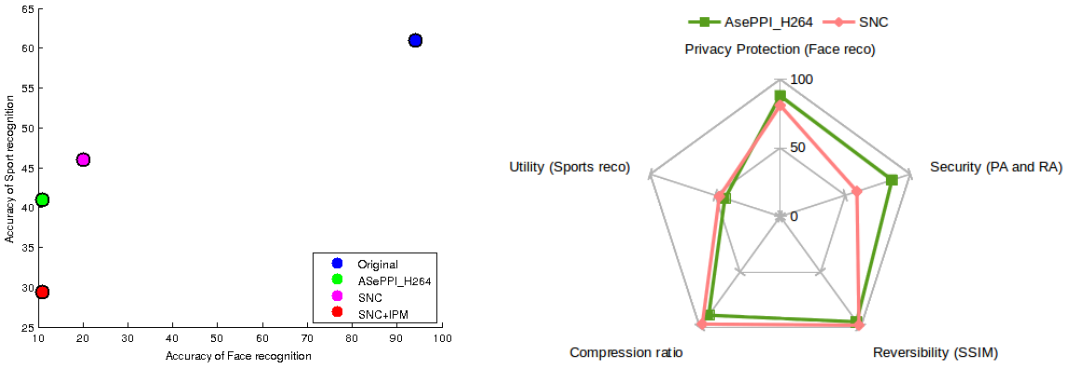
The bit overhead is the percentage of bits added by our process compared to the baseline profile (H.264/AVC without privacy-protection). For example, for the 'foreman' sequence, with QP = 24 and IP = 10, the number of bits are 83289 for the baseline profile and 85600 with the integration of our process which produces $100 - 100 * 83289/85600\%$ of bit overhead, i.e., 2.7%. We generate bit overhead (11 % compared to H.264/AVC) because the DC encryption loses some efficiency in quantification and, moreover, we insert a new DC coefficient.

According to the results, the impact on the efficiency of the H.264/AVC standard is negligible. Moreover, we evaluate the performances on the RoI only, thus the impact on the whole image should be even less significant.

6.3.3.8 Comparison between the performances of the different criteria

As presented in the section 6.2.5.7, a POC is a curve representing the level of privacy protection with the amount of data utility. We draw in the Figure 6.17(a), the average accuracy of sport event classification in function of the performance of face recognition.

ASePPI_H264 provides a better the trade-off between privacy protection and sport event classification compared to the other methods as you can remark on the Figure 6.17(a). Moreover, the Figure 6.17(b) highlights the significant improvement of privacy protection and security (i.e., $100\%-$ average accuracy of face recognition) performances that our system procures compared to the slight decrease in performances of sport event classification, compression ratio, and reversibility.



(a) (b)

FIGURE 6.17: (a) Privacy protection VS Sport event classification, (b) Comparison between all criteria.

6.3.4 Conclusions

Unlike the existing methods (mentionned in the sections 6.2.5 and 6.3.3), the application of *ASePPI* provides the best trade-off between the protection of privacy and the utility preservation of visual surveillance. Moreover, we prove that our system is robust against common de-anonymization attacks. Indeed, we design the algorithm of the *ASePPI_H264* for that purpose. We select the scrambling method that produces the highest number of combinations to recover the original data. *ASePPI* automatically adapts the level of the pixelization effect to be optimal in terms of privacy protection at any resolution. Furthermore, we evaluate the impact on the efficiency of the standards when integrating our process. We conclude that the quality of the reconstructed videos is close to the original ones, and the process produces a small percentage of bit overhead.

6.3.5 Perspective/Discussion

It would have been interesting to show the impact on the rate-distortion curve and also to evaluate the impact on these performances using the entire image and not only the region of interest.

We could have shown the impact on the face recognition with the temporal correlation according to the GOP (i.e., Group of picture). Indeed, the temporal correlation could restore some of the images. Moreover, the parameter of protection level could depend on the motion speed.

With the rapid evolution and success of deep learning algorithms, the parrot attack will become more and more efficient, thus, we should take the matter seriously. For example, we could set the DC such that the method produces more random blocks without impacting too much the quality of the image.

As an extension of the evaluation provided in this Chapter, we should evaluate the rate of correct recognition of people attributes (e.g., gender, age, ethnicity). In addition, we also could subjectively evaluate the efficiency of the privacy protection and the intelligibility by doing a survey using images that are protected by our method. For example, by asking to people, the identity of famous people, the level of pleasantness of the image and the events of the scene, and proposing 'multiple choice' to answer.

Chapter 7

Conclusion & Future work

Surveillance has become ubiquitous and a part of our everyday life in public space and even in office buildings. While technology in this area has seen constant advancement, for example the automatic identification of individuals under surveillance, little has been done to prevent a further erosion of our personal privacy due to these developments. However, this issue is important, since it impacts the very foundations of our freedom and should not be underestimated. Consequently, we should find solutions to protect our freedom and privacy against the ever-increasing pervasiveness of technology. The purpose of this Thesis was to provide a contribution to this issue and to solve the main research question, which was formulated as follows:

How can we protect the privacy of individuals while maintaining the utility in surveillance applications?

After defining the main research question, we investigated prior studies in the area of interest. We discovered that privacy protection is an emerging field with different proposed methods for hiding privacy information. However, we have noticed that several criteria, essential for effective surveillance, have not been taken into due consideration.

In **Chapter 3**, we demonstrated that an objective and a subjective evaluation perform almost equally in the task of gender recognition on body images protected by privacy protection filters. As an extension of this work, we could redo this study for the identity recognition task.

The purpose of the workflow presented in **Chapter 4** was to prove the weakness of the common privacy methods (i.e., pixelization, blur, black masking). The algorithms used in the proposed architecture can be further improved. For instance, instead of using handcrafted features (i.e., LBPH, HOG, PCA), we could build a deep neural network to automatically extract the features. We could also use the newest methods of super-resolution, de-blurring and de-noising based on deep neural network.

Our solutions, provided in **Chapter 5 and 6**, protect the privacy of innocent persons under surveillance. The main idea is to encrypt the original information and hide it in the least important part of the data while keeping the main information required for surveillance in the most important part of the data.

We first designed, in **Chapter 5**, a privacy filter applied to the whole body where we retain the edges of a person only (in the most significant bits) while the original information is hidden inside the image itself (in the least significant bits). This approach hides the identity, but retains other sensible information, that is the gender and other attributes such as the height, the weight, ect. For this purpose, we have proposed a new concept, denoted as “de-genderization" which means that the gender of people must be unrecognizable. Therefore, we have created a “de-genderization" method using the convexity of the body contours which preserves the recognition of the movements of body parts. The efficiency of the method mainly depends on the body shape/silhouette extraction. Thus, this crucial step needs to be improved.

The images generated by the previous method can not be compressed because the approach operates in the spatial domain. Therefore, in **Chapter 6**, we have designed a process operating in the DCT domain and which is compliant with the widely adopted JPEG and H.264/AVC standards. Following the same idea as the one of **Chapter 5**, we encrypt and hide the original DCT coefficients in the AC coefficients. We retain the DC (i.e., the most important DCT coefficient), representing the mean of the block to keep a minimum information required for surveillance, and set it such as the privacy is protected. In our experiments, we proved that this approach well protects the identity and preserves the utility of the surveillance. In addition, the performance of the overall system is better than previous privacy preserving surveillance approaches. Moreover, our process is flexible because the main values can be set according to what we want to hide (e.g., some attributes of people such as gender, ethnicity, weight, height, colors of clothes), but we did not prove yet that the utility is still protected in all cases. This flexibility is an important advantage. Indeed, depending on the country there are different legal requirements, thus, according to these we could tune the system (e.g., Europeans might deal differently with privacy than Asians). What needs to be protected and what is private is a parameter that may vary according to culture and time. For example, a person living in a big city might share information differently than someone from a village.

However, all the works presented in this Thesis leaves room for improvement:

- **Detection of the regions of interest:** The regions of interest in the image, where identity information should be hidden, have to be identified. Face and people detection are an on-going research field, where significant contributions will be made in the future, especially, in scenes with different luminance or low resolution. This step is crucial in privacy protection because if the detection fails on only one image, privacy of people in that image cannot be protected. A possible way to improve the accuracy of people detection would be by using multiple cameras (e.g., thermal cameras). This would allow the detection of individuals over multiple cameras, reducing the amount of non-detected people.

  In addition, the pleasantness of the image will increase if we apply the protection only on blocks that are considered to be protected instead of the entire rectangular area (i.e., bounding box) of the region of interest. Furthermore, we could select people that do not need or want the protection of their identity (e.g., policeman, politician, celebrity, ...).

- **Part of the body containing privacy:** We could think that persons are most likely identified by their face and their lower body (i.e., legs, by motion for gait recognition) whereas the information needed for the surveillance is most likely extracted from the upper body (i.e., arms, chest). Thus, future researchers in this area should investigate and prove this statement to then apply the protection only on the interested part of the body (e.g., on the face and on the lower body). This will improve the utility of surveillance.

- **Attacks:** We evaluated the robustness of our approaches against several attacks (i.e. the brute force, parrot and replacement attacks). However, there exists more advanced attacks [127] that worth being tested. We could also design a specific method to attack our proposed system.

- **Subjective evaluation:** We proved in **Chapter 3** and many authors claim that human vision and recognition algorithms perform equally. Despite these claims, it should be interesting to evaluate the approaches presented in this thesis with a survey (e.g., using Crowdsourcing). Indeed, video surveillance is usually still supervised by humans.

- **Real case study:** The next step is to integrate and evaluate the proposed approaches of this Thesis (the ones designed in **Chapter 5 and 6**), in a case study at a customer's site. For example, in the video surveillance system of an airport.

- **Laws:** Finally, certifications and standards should be defined for privacy preserving video surveillance. With EuroPriSe [128] there is already a privacy certificate present in the EU, however, it is a general certificate for IT products and not focused specifically on video surveillance. In this context, legal issues will be important for future investigations and standards should be defined on how private information should be stored in a secure system.

Publications and Other Scientific Activities

All along this Thesis, I helped my supervisor for all teaching activities (assisting laboratory sessions, supervising student projects, correcting exams).

I also collaborated several times with other PhD students.

During the first year of this Thesis I was involved in an European Project: VideoSense Network of Excellence[1]. Therefore, I participated in weekly Skype calls and review meetings.

Papers in international conferences

February 2018 **Privacy protection and image data utility preservation within JPEG**, *Natacha Ruchaud, Jean Luc Dugelay*, IET Signal processing

09/2017 **ASePPI : protéger la vie privée tout en préservant l'utilité de la vidéo-surveillance**, Natacha Ruchaud, Jean Luc Dugelay, Gretsi, Juan les pins, France

08/2017 **ASePPI, an Adaptive Scrambling enabling Privacy Protection and Intelligibility in H.264/AVC**, Natacha Ruchaud, Jean Luc Dugelay, European signal processing conference, Kos island, Greece

07/2017 **ASePPI: Robust privacy protection against de-anonymization attacks**, Natacha Ruchaud, Jean Luc Dugelay, IEEE Conference on Computer Vision and Pattern Recognition Workshops, Honolulu, Hawaii, United States

02/2017 **De-genderization by body contours reshaping**, Natacha Ruchaud, Jean Luc Dugelay, IEEE International Conference on Identity, Security and Behavior Analysis, New Delhi, India

[1] http://www.videosense.eu/

07/2016 **Privacy protecting intelligibility preserving video surveillance**, Natacha Ruchaud, Jean Luc Dugelay, IEEE International Conference on Multimedia and Expo, Seattle, Washington, United States

02/2016 **Automatic Face Anonymization in Visual Data: Are we really well protected?**, Natacha Ruchaud, Jean Luc Dugelay, Electronic Imaging, San Francisco, California, United States

11/2015 **Efficient Privacy Protection in Video Surveillance by StegoScrambling**, Natacha Ruchaud, Jean Luc Dugelay, IEEE International Workshop on Information Forensics and Security (WIFS), Rome, Italy

10/2015 **Privacy Protection Filter Using StegoScrambling in Video Surveillance**, Natacha Ruchaud, Jean Luc Dugelay, MediaEval workshop, Wurzen, Germany

10/2015 **Overview of the MediaEval 2015 Drone Protect Task**, Atta Badi, Pavel Korshunov, Hamid Oudi, Touradj Ebrahimi, Tomas Piatrik, Volker Eiselein, Natacha Ruchaud, Christian Fedorczak, Jean-Luc Dugelay, Diego Fernandez Vazquez, MediaEval workshop, Wurzen, Germany

10/2015 **Learned vs. hand-crafted features for pedestrian gender recognition**, Grigory Antipov, Sid-Ahmed Berrani, Natacha Ruchaud, Jean Luc Dugelay, ACM Conference on Multimedia Conference, Brisbane, Australia

08/2015 **The impact of privacy protection filters on gender recognition**, Natacha Ruchaud, Grigory Antipov, Pavel Korshunov, Jean-Luc Dugelay, Touradj Ebrahimi, Sid-Ahmed Berrani, SPIE Optical Engineering+ Applications, San Diego, California, United States

---

10/2014 **Stereo reconstruction of semiregular meshes, and multiresolution analysis for automatic detection of dents on surfaces** Jean-Luc Peyrot, Frédéric PAYAN, Natacha Ruchaud and Marc Antonini IEEE International Conference in Image Processing (ICIP)

Appendix

Storing the bounding box of each RoI

Algorithm 8: Bounding box insertion in the image itself

```
% nbr is the number of detected RoI ;
% Img is the original image ;
Img(height, width, blue) = nbr;
for i=1:nbr do
    if x then
        n = 1, m = 1, X = x;
    else if y then
        n = 1, m = 2, X = y;
    else if height then
        n = 2, m = 1, X = height;
    else if width then
        n = 2, m = 2, X = width;
    end
    un = (⌊X/255⌋ == 0);
    deux = (⌊X/255⌋ == 1);
    Img(n,m+2*i,red) = mod(X, 255)*un + 255*(1-un);
    Img(n,m+2*i,green) = mod(X, 255)*deux + 255*(1-un)*(1-deux);
    Img(n,m+2*i,blue) = (X-255*2)*(1-un)*(1-deux);
end
```

Algorithm 9: Bounding box extraction

```
% nbr is the number of detected RoI ;
% Img is the original image ;
nbr = Img(height, width, blue);
for i=1:nbr do
    x = Img(1,1+2*i,red) + Img(1,1+4*i,green) + Img(1,1+4*i,blue);
    y = Img(1,2+2*i,red) + Img(1,2+4*i,green) + Img(1,2+4*i,blue);
    height = Img(2,1+2*i,red) + Img(2,1+2*i,green) + Img(2,1+2*i,blue);
    width = Img(2,2+2*i,red) + Img(2,2+2*i,green) + Img(2,2+2*i,blue);
end
```

Joint Photographic Experts Group (JPEG)

JPEG is a commonly used method of lossy compression for images, mainly employed by digital cameras and other photographic image capture devices. The process is based on the discrete cosine transform (DCT) that converts each frame of the video source from the spatial domain into the frequency domain.

Quantization reduces information (i.e., shrinks a large number scale into a smaller one). The frequency domain is an efficient representation of the image because the high-frequency coefficients are characteristically small-values with high compressibility.

The compression method is usually lossy. Indeed, some original image information is lost and cannot be restored. This may affect image quality. There exists an optional lossless mode, but not widely supported in products.

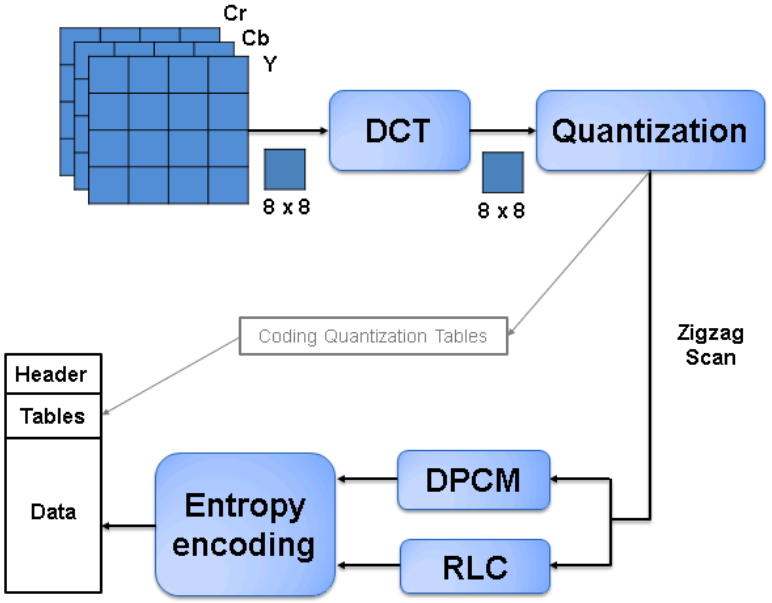


FIGURE 7.1: Scheme of JPEG coding.

The following describes the several steps of the encoding. The decoding is performed using the inverse steps.

- **Color space transformation**

  First, the image is converted from RGB (i.e., Red, Green, Blue channels) into the YCbCr color space. The Y component represents the brightness of a pixel, and the Cb and Cr components represent the chrominance.

  The brightness information is more important to the eventual perceptual quality of the image and more closely corresponds to the perception of color in the human visual system. This information is confined to a single channel that makes the compression much more efficient. Indeed, the YCbCr color space conversion allows greater compression without a significant effect on perceptual image quality (or greater perceptual image quality for the same compression).

- **Downsampling**

  The transformation into the YCBCR color model allows to reduce the spatial resolution of the Cb and Cr components because humans are more sensitive to the fine detail in the brightness of an image (the Y component) than in the color saturation of an image (the Cb and Cr components). The ratios of the downsampling are either 4:2:2 (reduction by a factor of 2 in the horizontal direction), or (most commonly) 4:2:0 (reduction by a factor of 2 in both the horizontal and vertical directions). Note that when there is no downsampling ratios are 4:4:4.

- **Block splitting**

  Images for each channel must be split into 8x8 blocks. Sophisticated border filling techniques fill the remaining area (borders).

- **Discrete cosine transform (DCT)**

  Next, each 8x8 block of each component (Y, Cb, Cr) is converted into a frequency domain, using the two-dimensional discrete cosine transform (DCT), which was introduced by N. Ahmed et al. [129] in 1974.

  Before applying the DCT, the values of the 8x8 block are shifted from a positive range to one centered on zero. For an 8 bit image, the value of each pixel in the original block is between 0 and 255. The midpoint of this range is subtracted from each pixel to generate a range that is centered on zero (in this case, the range becomes [-128, 127]).

  The formula of the two-dimensional DCT is given by:

$$G_{u,v} = \frac{1}{4}\alpha(u)\alpha(v)\sum_{x=0}^{7}\sum_{y=0}^{7} g_{x,y}\cos\frac{(2x+1)u\pi}{16}\cos\frac{(2y+1)v\pi}{16} \tag{7.1}$$

  with u and v respectively the horizontal/vertical spatial frequency, $G_{u,v}$ is the DCT coefficient at coordinates (u, v), $g_{x,y}$ is the pixel value at coordinates (x, y) and $\alpha(u)$ a function that makes the transformation orthonormal by normalizing scale factor (equal to $\frac{1}{\sqrt{2}}$ if $u = 0$ otherwise to 1).

  The top-left corner coefficient obtains the rather large magnitude. This is the DC coefficient (also called the constant component). The remaining 63 coefficients are the AC coefficients (also called the alternating components). The DCT aggregates most of the signal in one corner of the result (the top-left) represented by the low-frequencies. The high frequencies (the bottom-right corner) contains details of the signal.

- **Quantization**

  The human eye is not so sensitive to a high frequency brightness variation. This allows to greatly reduce the amount of information in the high frequency by simply dividing each component in the frequency domain by a constant for that component (the higher the frequency, the higher the constant is). Then, the divided coefficients are rounded to the nearest integer that produces many of the higher frequency components to zero and many of the rest to small numbers. The rounding operation is the only lossy operation in the whole process.

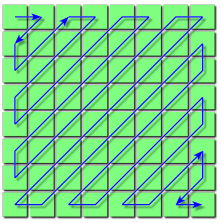


FIGURE 7.2: Zigzag ordering of a JPEG block.

- **Entropy coding**

  Entropy coding, first, arranges the blocks in a zigzag order to exploit redundancy (see Figure 7.2). This is a lossless data compression employing run-length encoding (RLE) algorithm that groups similar frequencies together and then apply Huffman coding.

  Instead of using the Huffman coding, JPEG standard also allows arithmetic coding, which is mathematically superior to Huffman coding. However, this method has rarely been used, mainly, because it is slower to encode and decode compared to Huffman coding.

  The current quantized DC coefficient is predicted from the previous quantized DC coefficient, called Differential Pulse Code Modulation (DPCM). For the DC coefficient only, the difference between the two is encoded, rather than the actual value.

H.264/MPEG[2]-4 AVC

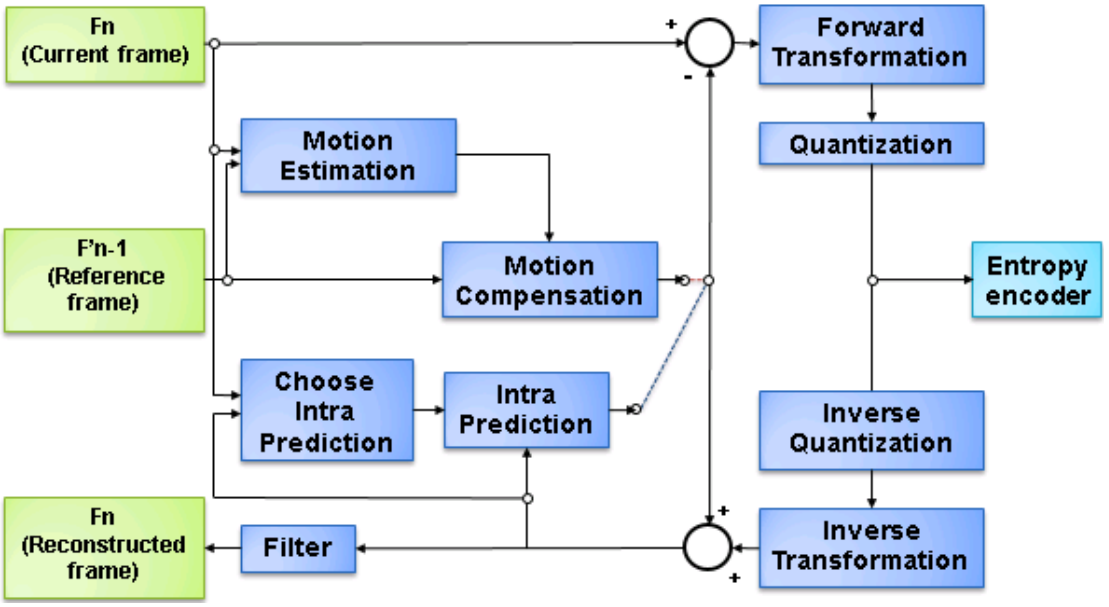


FIGURE 7.3: Scheme of H.264 coding.

[2]Moving Picture Experts Group

H.264 or MPEG-4 Part 10, Advanced Video Coding (MPEG-4 AVC) is a block-oriented motion-compensation-based video compression standard. This standard takes advantage of the redundancy of the blocks in the spatial and temporal domain.

A video picture is coded as one or more slices, each containing macroblocks (MB) of 16x16 luma samples and 8x8 chroma samples. Each macroblock is encoded in intra or inter mode.

- **Intra prediction**

  The intra prediction is often called as the spatial prediction because the current block is predicted using previously encoded and reconstructed blocks from the same frame. There are nine prediction modes for each 4*4 luma block (illustrated in 7.4(a)), four for the 16*16 luma block and four for the chroma (illustrated in 7.4(b)).

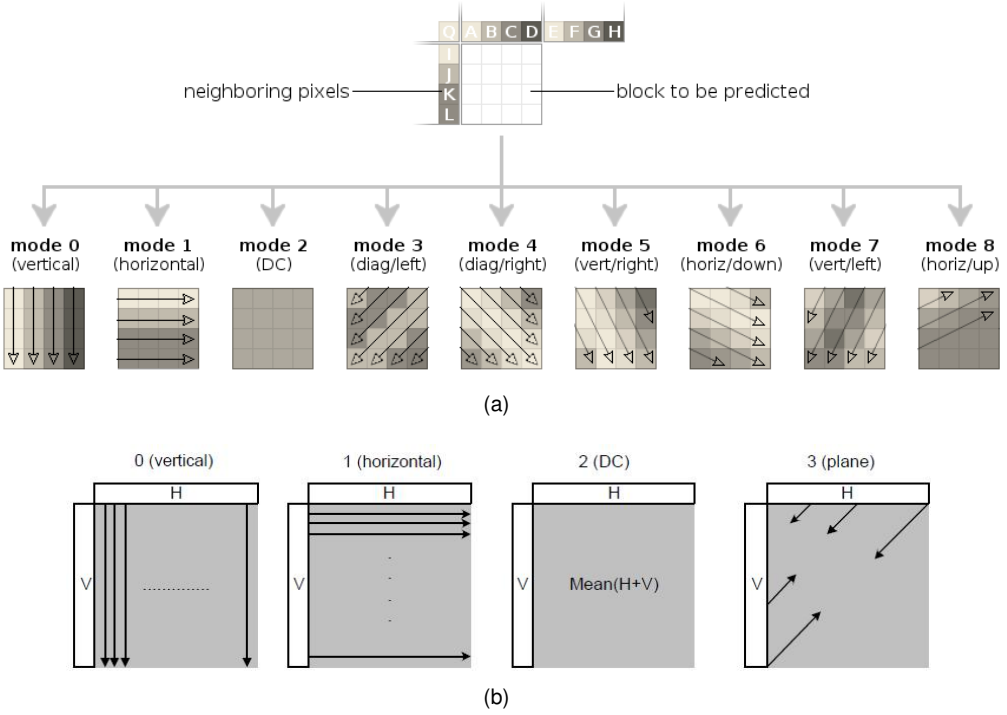
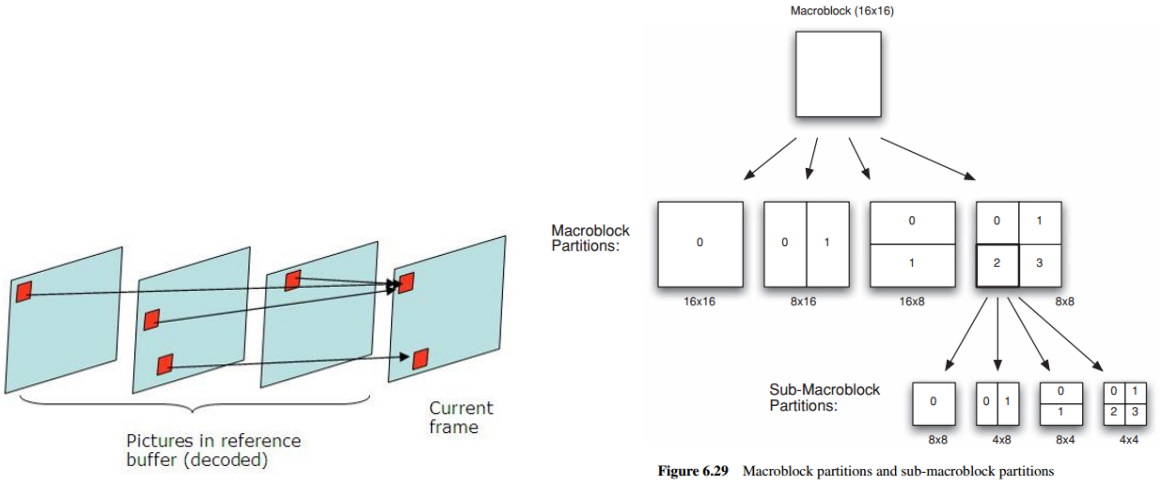


FIGURE 7.4: H.264/AVC intra prediction modes. (a) For luma 4*4 and (b) luma 16*16 and chroma. The pictures are reprinted from [3] and from the book [10].

- **Inter prediction** The inter prediction is often called as the temporal prediction because the current block is predicted using previously encoded and reconstructed blocks from one or more previously encoded video frames. The macroblock is split into several partitions (shown in the Figure 7.5) and each partition is predicted from areas of the same size in reference pictures.

The baseline profile supports Intra (I), Predicted frames (P) and entropy encoding with context-adaptive variable-length codes (CAVLC). I frames contain only intra prediction, intra blocks are predicted from

[3] https://people.xiph.org/~xiphmont/demo/daala/demo2.shtml
[3] http://www.cnitblog.com/luofuchong/archive/2015/06/01/90112.html
[4] https://en.wikipedia.org/wiki/Inter_frame



(a) (b)

FIGURE 7.5: (a) Temporal prediction, and (b) Macroblock/sub-macroblock partitions. The pictures are reprinted from [3] and from [4].

previously coded data within the same frame. P frames contain intra prediction, but also inter prediction, inter blocks are predicted from blocks of a previous reference frame. The residual blocks are the differences between the predicted blocks and the real/original ones. The residual block is then transformed (DCT), quantized before being encoded (i.e., before the entropy encoding).

More explanation is provided in the "H. 264 and MPEG-4 video compression: video coding for next-generation multimedia" book [10].

Chapter 8

Résumé en Français

1 Introduction

1.1 Contexte et motivation

En raison d'une série d'attaques terroristes au début du XXIe siècle et d'activités criminelles croissantes au cours des dernières années, les systèmes de vidéosurveillance sont de plus en plus nombreux et performants. Ainsi, ils constituent une composante majeure de la sécurité. Cependant, la surveillance accrue des citoyens dans l'espace public affecte les droits individuels de la vie privée.

Alors qu'il y a de plus en plus de caméras, la résolution des capteurs visuels (par exemple, 4k, HD) et la performance des algorithmes de traitement vidéo (tel que, la reconnaissance d'identité) sont continuellement en évolution.

D'une part, la vidéosurveillance peut contribuer à accroître notre sécurité (pour dissuader les criminels, par exemple, et fournir des preuves pour enquêter et résoudre les crimes). D'autre part, la vie privée des citoyens est constamment envahie, et il existe une crainte que les informations recueillies soient mal utilisées.

Des solutions pour atténuer les problèmes de confidentialité liés à l'utilisation des caméras de surveillance existent déjà. En effet, les caméras de sécurité ATM ainsi que les caméras de sécurité extérieures protègent l'entrée d'un code PIN ainsi que les propriétés privées en appliquant un masque noir. Cependant, la protection de la vie privée des personnes est plus complexe étant donné que le suivi de leurs actions ne doit pas être entravé. Les méthodes actuelles de protection de la vie privée, souvent employées par les médias, sont le flou ou la pixellisation (par exemple, Google Street View, FacePixelizer [1], ObscuraCam [2] sur Android) mais ils ne sont pas réversibles (c.-à-d., que les utilisateurs ne peuvent pas récupérer la vidéo originale).

[1] http://www.facepixelizer.com/
[2] https://guardianproject.info/apps/obscuracam/

D'autres solutions technologiques ont été proposées, mais aucune d'entre elles n'offre une protection de la vie privée tout en respectant les critères nécessaires pour à la surveillance. Plus précisément, les travaux récents contournent le plus souvent les tâches/critères de compression et de réversibilité ou ont des difficultés à gérer le compromis entre la protection de la vie privée et la préservation de la surveillance visuelle.

Cette thèse traite la problématique suivante : Comment pouvons-nous protéger la vie privée des personnes tout en maintenant l'utilité de la surveillance?

Dans un premier temps, nous avons étudié les méthodes de protection de la vie privée existantes, et leurs critères qui définissent le niveau de protection de la vie privée ainsi que celui de l'utilité de la surveillance visuelle. Nous démontrons que les approches existantes ne sont pas assez efficaces. Nous proposons donc une première méthode réversible qui garde le mouvement et la silhouette du corps tout en protégeant l'identité. Enfin, pour être compatible avec la compression nous proposons d'intégrer notre système dans les standards JPEG et H.264/AVC.

1.2 Contributions

Le défi soulevé dans cette thèse est de proposer une méthode de protection de la vie privée qui préserve l'utilité dans les données visuelles sans ignorer la robustesse contre les attaques potentielles (c.-à-d., que quelqu'un tente d'inverser le processus sans autorisation).

De nos jours, la surveillance est principalement effectuée par des humains, mais, de plus en plus, il existe des outils pour détecter et classifier automatiquement divers objets / tâches. Ainsi, nous avons étudié les différences entre une évaluation objective et subjective lors de l'application des filtres de protection de la vie privée. Nous avons sélectionné PETA comme base de données (piétons) et nous évaluons la détection du genre sur les images originales et sur celles qui sont protégées par un filtre de confidentialité. Pour évaluer objectivement la détection du genre, nous appliquons un réseau de neurones convolutif (CNN), puis nous faisons un Crowdsourcing pour l'évaluation subjective (c.-à-d., un sondage réalisé par une vision humaine). Notez que nous utilisons les mêmes données pour l'ensemble de tests dans les deux évaluations. Nous aurions pu penser que l'humain aurait de meilleurs résultats que la machine, mais, étonnamment, les deux obtiennent des résultats similaires. Ce travail a été publié à ADIP 2015.

Dans une autre étude, nous avons prouvé que notre vie privée n'est pas assez protégée par les méthodes actuelles de dé-identification telles que la pixellisation, le flou Gaussien ou même le masquage noir (noircisseur). Nous proposons une architecture complète qui permet de récupérer l'identité d'un visage anonymisé. Il consiste, d'une part, à détecter la présence d'une protection d'identité visuelle ou non, puis à détecter le type ainsi que le niveau de protection (par exemple, pixellisation de taille 4) et enfin appliquer une méthode de restauration associée au type identifié ainsi qu'au niveau de protection détecté. Cette architecture a été publiée à Electronic Imaging 2016.

Après avoir montré les faiblesses des méthodes de dé-identification actuelles, nous proposons une première méthode réversible (uniquement pour les personnes autorisées) qui protège la vie privée tout

en gardant la silhouette du corps. Dans le domaine spatial, nous cryptons et déplaçons les bits les plus significatifs (MSBs) des pixels originaux d'une région d'intérêt (RoI) vers les bits les moins significatifs (LSBs). Ensuite, nous insérons les bits associés au contour du RoI dans les MSBs afin de garder la scène compréhensible. Cette méthode a été publiée dans le cadre du challenge *Mini-drone Video Privacy* à MediaEval Benchmark 2015, et une version améliorée a été présentée à WIFS 2015. Nous proposons également une version déformée de la silhouette du corps afin de cacher l'information sur le genre. Ce travail a été publié à ISBA 2017. La méthode, présentée à MediaEval et WIFS, est réversible, mais n'est pas robuste à la compression avec perte. De nos jours presque toutes les images et vidéos sont compressées, par conséquent, les algorithmes de traitement d'images doivent être conformes aux normes les plus répandues telles que JPEG et H.264/AVC.

Nous proposons donc d'intégrer notre approche dans ces standards de compression. Nous intégrons notre algorithme dans le domaine de la transformée en cosinus discrète (DCT) pour permettre la compression avec les normes JPEG et H.264/AVC. Pour chaque zone sensible de l'image (zone où la confidentialité doit être protégée), l'algorithme proposé utilise les coefficients basse fréquence de la DCT pour préserver les évènements de l'image et les coefficients haute fréquence pour cacher la majorité des informations originales (qui est également cryptée). Notre approche permet aux utilisateurs munis d'un mot de passe de restaurer l'image originale (avec très peu de pertes) à partir de ces informations cachées et cryptées. Ainsi, cette méthode assure la protection de la vie privée à toute taille d'image (ce qui n'est pas le cas pour d'autres méthodes similaires existantes) tout en préservant un minimum d'information requis par la surveillance (tel que, la reconnaissance des sports dans la scène). Notre système est réversible seulement pour les personnes autorisées (ceux qui possèdent le mot de passe), et sécurisé contre les attaques par force brute, perroquet ou encore les attaques par remplacement. De plus, la méthode est conforme aux standards JPEG et H.264/AVC. L'intégration dans le standard JPEG a été publiée à ICME 2016 et celle sur le standard H.264/AVC à EUSIPCO 2017. Une version étendue du second a été publiée à CVPR 2017 et a été présentée sur un poster au GRETSI 2017.

1.3 Plan

Dans la première phase de cette thèse, nous étudions principalement les motivations à protéger la vie privée, les critères associés à la vidéosurveillance et les méthodes existantes.

Tout d'abord, dans la section 2, nous analysons les différences entre une évaluation objective et une évaluation subjective de la reconnaissance du genre dans les images où l'identité a été protégée par des méthodes existantes. Les résultats montrent qu'ils fonctionnent de manière égale.

Dans la section 3, nous proposons une architecture qui améliore la restauration des images où l'identité a été protégée par des méthodes existantes (telles que, la pixellisation, le flou gaussien, le masquage noir) et nous prouvons donc la faiblesse de ces filtres. Pour construire cette architecture, nous avons combiné plusieurs représentations de caractéristiques populaires (à savoir, HoG, LBPH, PCA) avec un classificateur SVM linéaire.

L'idée développée dans les sections 4 et 5, est de garder les informations principales requises par la surveillance dans les coefficients les plus importants tout en cryptant et cachant les coefficients originaux dans les coefficients les moins importants.

Plus précisément, dans la section 4, nous concevons une nouvelle méthode de protection de la vie privée dans le domaine spatial (c.-à-d., dans le domaine des pixels) qui conserve la silhouette du corps. Le processus est réversible uniquement pour les personnes autorisées. Nous cryptons les bits d'origine et les cachons dans les bits les moins significatifs. Pour préserver le mouvement du corps, nous insérons les informations de contours qui peuvent être modifiées ou non (selon si nous voulons ou non cacher l'information du genre), dans les bits les plus significatifs.

En suivant la même idée, dans la section , nous proposons de travailler avec les coefficients DCT afin d'intégrer le processus dans les normes JPEG et H.264/AVC. Nous cryptons et cachons les coefficients DCT d'origine dans les coefficients AC (c.-à-d., les variations de couleur). Pour conserver une information minimale requise par la surveillance, nous modifions le coefficient DCT le plus important, le DC (c.-à-d., la couleur moyenne).

Dans la dernière section, nous concluons sur les travaux présentés, soulignons ses limites et suggérons de nouvelles directions de recherche.

2 Évaluation objective VS subjective de la reconnaissance du genre sur des images où l'identité est protégée

Les algorithmes basés sur l'apprentissage profond sont devenus de plus en plus efficaces dans les tâches de reconnaissance et de détection, notamment quand ils sont entrainés sur des jeux de données à grande échelle. Ce tel succès peut nous induire à penser que les méthodes d'apprentissage en profondeur sont comparables ou même supérieures au système visuel humain dans sa capacité à détecter et à reconnaître des objets et leurs caractéristiques. Dans cette section, nous nous concentrons sur la tâche spécifique de la reconnaissance du genre dans les images où l'identité est protégée par des méthodes existantes. En supposant un scénario de protection de la vie privée, nous comparons la performance d'un algorithme d'apprentissage en profondeur avec une évaluation subjective obtenue par crowdsourcing pour comprendre comment les filtres de protection de la vie privée affectent la vision humaine et la machine.

Nous utilisons la collection de bases de données PETA (une description complète de cette base de données est faite dans la section 2.5.3).

2.1 Reconnaissance du genre avec un CNN

Pour l'évaluation objective, nous avons sélectionné un réseau de neurones convolutionnels (CNN). Nous utilisons l'architecture proposée par Krizhevsky et al. [68], souvent dénommé AlexNet. Cette architecture est présentée dans la Figure 8.1. Elle se compose de cinq couches convolutives et de trois couches

entièrement connectées. Nous appliquons pratiquement la même architecture sur l'ensemble de données PETA. La seule différence est que dans la dernière couche entièrement connectée, nous utilisons 2 neurones au lieu de 1000, puisque nous n'avons que deux classes (hommes et femmes).

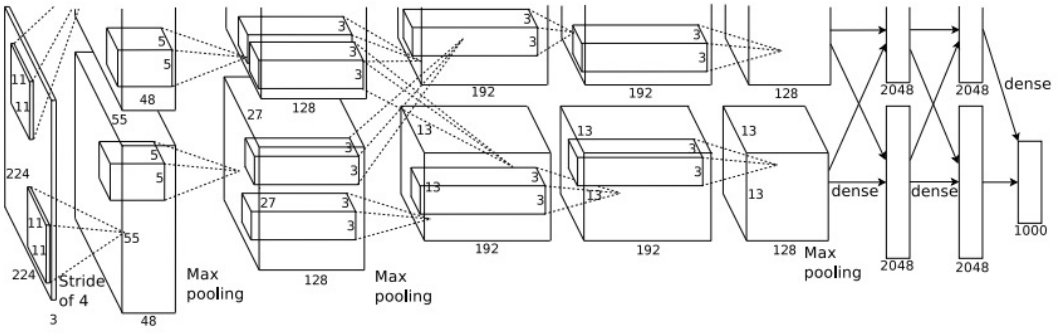


FIGURE 8.1: CNN.

Afin de généraliser notre CNN sur des ensembles de données hétérogènes, nous entrainons d'abord le CNN sur 90 % de l'ensemble de données de CUHK, PRID, GRID, MIT et VIPeR (soit 4488 images d'hommes et 2744 de femmes). Ainsi, notre modèle est testé sur un seul grand ensemble de données composé d'images provenant des 10 ensembles de données et qui n'ont pas été utilisés dans l'apprentissage (160 hommes et 161 femmes). Le Tableau 8.1 résume le nombre exact d'images utilisées en apprentissage et en test pour chaque jeu de données.

Dataset	Train size (male + female)	Test size (male + female)
CUHK	$3432 = (2420 + 1012)$	$73 = (34 + 39)$
PRID	$942 = (449 + 493)$	$36 = (16 + 20)$
GRID	$928 = (531 + 397)$	$23 = (10 + 13)$
MIT	$792 = (532 + 260)$	$69 = (41 + 28)$
VIPeR	$1138 = (556 + 582)$	$48 = (24 + 24)$
3DPeS	0	$15 = (10 + 5)$
CAVIAR	0	$9 = (3 + 6)$
i-LIDS	0	$6 = (3 + 3)$
SARC3D	0	$5 = (3 + 2)$
TownCentre	0	$37 = (16 + 21)$
Total	$7232 = (4488 + 2744)$	$321 = (160 + 161)$

TABLE 8.1: Images utilisées en apprentissage et en test.

Nous appliquons cinq méthodes de protection d'identité avec trois niveaux de protection différents pour chacun, énoncés et illustrés dans le Tableau 8.2 et dans la Figure 8.2. Nous expliquons plus en détail chacune de ces méthodes, dans les sections 2.3 et 2.4.

2.2 Évaluation par crowdsourcing

Pour l'évaluation subjective, nous avons effectué un crowdsourcing (une description plus précise est donnée dans la section 2.5.2). Cette évaluation vise à vérifier si une personne dans une image donnée

Filter	Parameter	Niveau de protection
Black Masking	opacité	0.5, 0.7, 0.9
Morphing	opacité	0.4, 0.7, 0.9
Pixellisation	taille des carrés	3, 5, 7
Gaussian Blur	écart type	2, 4, 6
Kmeans	nombre de groupes	6, 4, 2

TABLE 8.2: Méthode de protection avec le niveau de protection que nous utilisons.

FIGURE 8.2: De gauche à droite: image originale, masque noir d'opacité 0.5, 0.7 et 0.9, morphing d'opacité 0.4, 0.7 et 0.9, pixellisation de taille 3, 5 et 7, flou Gaussian d'écart type 2, 4 et 6 et Kmeans avec un nombre de groupe de 6, 4 et 2.

peut être correctement identifiée comme féminine ou masculine par un individu, même avec l'application d'un filtre de protection de la vie privée. Nous avons donc demandé à plusieurs personnes de répondre à la question "Quel est le sexe de la personne?" dans un formulaire comprenant plusieurs images de piétons.

Au total, 300 images aléatoires appartenant à l'ensemble de données PETA ont été utilisées dans l'évaluation. Ces images ont été protégées par les cinq méthodes à trois niveaux de protection différents, ce qui a permis d'évaluer $300 \times 5 \times 3 = 4500$ images dans cette étude de crowdsourcing.

Pour assurer un nombre statistiquement significatif d'évaluations pour chaque image, en tenant également compte de la présence de sujets peu fiables (environ 50% dans une évaluation de crowdsourcing typique), 40 sujets ont été assignés à chaque image, au total 2652 personnes ont participé à cette évaluation.

Toutes les versions des images ont été distribuées au hasard avec la garantie que chaque sujet n'évaluait qu'une seule version d'un contenu donné. Chaque formulaire commence par une consigne, suivie d'une séance de test décrivant la procédure d'évaluation. Un test de luminosité d'affichage est effectué en utilisant une méthode similaire à celle décrite dans [99] et permet d'estimer les paramètres d'affichage des sujets. Les sujets ne sont pas autorisés à sauter une séquence ou à éviter de répondre à une question.

Contrairement aux expériences subjectives en laboratoire où tous les sujets peuvent être observés et où l'environnement de test peut également être contrôlé, le principal défaut de l'évaluation subjective basée sur le crowdsourcing est l'incapacité à superviser le comportement des participants et à restreindre leurs conditions de tests. En utilisant le crowdsourcing pour l'évaluation, il existe un risque d'inclure des données non fiables dans l'analyse en raison de mauvaises conditions de tests ou de comportements peu fiables de certaines personnes qui tentent de soumettre des réponses de mauvaise qualité afin de réduire leur effort. Pour cette raison, la détection des personnes non fiables est un processus inévitable dans l'évaluation subjective basée sur le crowdsourcing. Pour identifier une personne comme «digne de confiance», les quatre facteurs suivants ont été utilisés dans notre expérience: *i)* Temps d'achèvement de la tâche; *ii)* Temps moyen d'observation par question; *iii)* déviation de la durée d'observation; *iv)* Nombre de réponses minoritaires.

L'objectif des trois premiers facteurs est de filtrer les personnes qui ont des comportements étranges au cours de leur tâche, car ils ne sont pas sérieux ou ont une faible concentration. Le temps d'observation par question est mesuré comme le temps écoulé entre l'affichage de la question et le moment où la réponse est donnée par l'individu. Le temps d'achèvement de la tâche, le temps d'observation moyen et l'écart de durée d'observation peuvent être calculés en utilisant ces données. Si le temps d'achèvement de la tâche ou le temps d'observation moyen par question est trop long par rapport à la moyenne de tous les autres, on peut en déduire qu'ils n'ont pas pris le test au sérieux ou ont été distraits pendant leurs tâches.

Pour renforcer notre décision, les personnes non fiables ont également été identifiées en utilisant une approche similaire à une méthode de détection de valeurs aberrantes, couramment utilisée dans la plupart des évaluations de qualité subjectives. Par conséquent, le nombre de réponses minoritaires a été utilisé pour la détection des valeurs aberrantes. L'hypothèse est qu'un participant qui a beaucoup de réponses différentes par rapport à la majorité des autres n'est pas fiable.

Nous avons donc identifié 198 personnes non fiables sur un total de 2652.

2.3 Résultats et conclusion

Une sortie du CNN est binaire ("male" ou "femelle") alors que la sortie du crowdsourcing est ternaire ("male", "female" et "je ne sais pas"). Par conséquent, afin de comparer équitablement les résultats de la reconnaissance du genre basée sur le CNN et le crowdsourcing, nous supposons que 50% des réponses " je ne sais pas " enregistrées lors du crowdsourcing auraient été correctes si la réponse avait été sélectionnée au hasard.

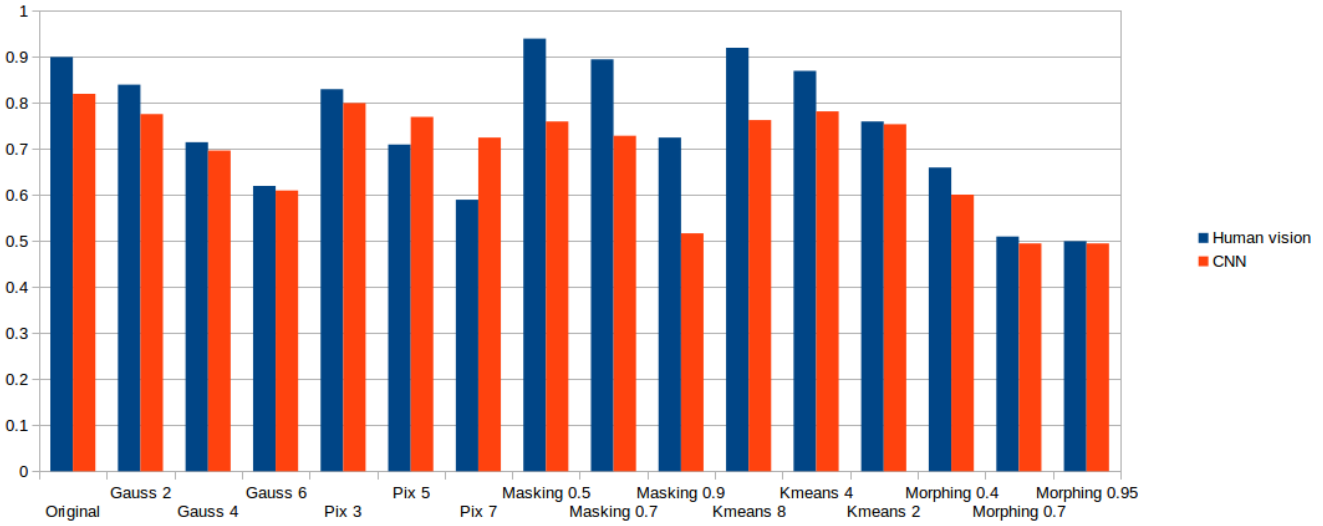


FIGURE 8.3: Taux de bonne reconnaissance du genre par la vision humaine et par CNN.

La Figure 8.3 illustre les résultats de la reconnaissance du genre en utilisant le CNN ainsi que ceux du crowdsourcing pour des images originales (c.-à-d., sans filtre) et altérées (application d'une méthode de protection). Le CNN montre des résultats proches de ceux du crowdsourcing pour le flou gaussien, le K-means et le morphing (moins de 10 % environ de différences sauf pour Kmeans 8). Le masquage noir dépend vraiment de la luminosité de l'écran et de son environnement. En effet, certains participants au crowdsourcing pourraient avoir augmenté la luminosité de leurs écrans d'ordinateur rendant ainsi les images de tests plus visibles, alors que le CNN fonctionne avec des valeurs de pixels quelle que soit la méthode d'affichage. De plus, la vision humaine s'adapte au changement de luminosité.

Dans cette étude, nous avons montré l'impact des méthodes de protection de la vie privée sur la reconnaissance du genre par des algorithmes de vision artificielle (en utilisant un CNN) et par la vision humaine (en utilisant une approche de crowdsourcing). On pourrait s'attendre à ce que la vision humaine soit plus robuste aux méthodes de protection d'identité que la vision par ordinateur. Néanmoins, nos résultats montrent que les humains et les systèmes de reconnaissance automatique ont des performances sensiblement égales pour détecter le genre.

3 Protection de l'identité visuelle: les méthodes actuelles protègent-elles vraiment notre vie privée ?

Les médias et certains systèmes de surveillance appliquent des méthodes de protection d'identité standard telles qu'un masquage par un rectangle de couleur unie, un flou [3], une pixellisation [4] ou encore

[3] lissage de l'image avec, par exemple, un filtre gaussien avec une grande variance
[4] sous-échantillonnage d'image

une addition de bruit. Une description détaillée de ces approches populaires est donnée dans la section 2.3.

Nous illustrons, dans la Figure 8.4, les méthodes que nous avons choisies pour cette étude, et nous résumons les noms de leurs paramètres associés dans le Tableau 8.3.

FIGURE 8.4: De gauche à droite, visages originaux, masque noir, pixellisation, flou Gaussien, bruit Gaussien.

Filtre de protection	Paramètre	Strength
Masque noir	opacité	0.1, 0.2, 0.3
Pixellisation	taille de ces carrés	3-10
Flou Gaussien	écart type	2-5, 8
Bruit Gaussien	écart type	0.001, 0.005, 0.01-0.1, 0.3

TABLE 8.3: Filtres de confidentialité avec la force utilisée et le nom de leur paramètre associé.

Le domaine de la restauration d'images permet de restaurer partiellement des images, que l'on nomme visage dé-masqué. En effet, la restauration d'images améliore la qualité des images ou reconstruit des images corrompues. En conséquence, l'impact de l'application d'une méthode de protection d'identité peut être annulé par ces approches, notamment, dès que la catégorie et la force de la protection sont identifiées. Plusieurs méthodes de restauration d'images efficaces existent telles que le défloutage [102], le débruitage [103–105], la super-résolution [22, 23], mais elles nécessitent une connaissance du type et de la force de corruption.

Au meilleur de notre connaissance, il n'existe aucune méthode qui détecte le type et le niveau de protection d'identité appliquée à une image. Par conséquent, l'étape clé consiste à identifier automatiquement la catégorie (type) et le niveau associé de la protection qui a été utilisée pour masquer un visage. Tout d'abord, nous différencions les visages masqués des non masqués. Ensuite, une seconde approche classifie le type de protection d'identité utilisée. Dès que la catégorie est identifiée, une dernière étape consiste à définir le niveau de la protection. Enfin, nous démontrons qu'en appliquant les méthodes de restauration d'images avec la connaissance au préalable du type et du niveau de la protection, l'identité redevient visible.

3.1 Présentation du système

Pour distinguer les visages masqués de ceux qui ne le sont pas, nous utilisons l'histogramme de gradient orienté (HOG) pour extraire les caractéristiques des images et nous entrainons un modèle qui différencie

les visages originaux des visages masqués avec le classificateur SVM linéaire.

La seconde étape consiste à classifier le type de protection d'identité utilisé pour masquer les visages. L'analyse en composantes principales (PCA ou Eigen) est plus sensible aux variations de lumière, d'échelle et de translation, ce qui conduit à avoir moins de robustesse face à une protection de type masquage noir (noircisseur). Les motifs binaires locaux (LBPH), sont utilisés pour classifier la texture ce qui est approprié dans le cas présent, car les méthodes de protection créent des motifs de textures spécifiques à l'exception du masquage noir qui les supprime. Ainsi, nous utilisons les caractéristiques Eigen des images pour extraire leurs caractéristiques puis nous différencions le masquage noir des autres protections avec le classificateur SVM linéaire, puis nous classifions les trois autres filtres entre eux en utilisant les caractéristiques LBPH et le SVM linéaire.

Une fois la catégorie de la protection d'identité connue, nous estimons automatiquement le niveau de cette protection par les approches listées suivantes:

- Nous estimons le niveau de protection du **masque obscurcissant/noircissant** en calculant la couleur moyenne de l'image. Nous supposons que la moyenne de l'image originale devrait être d'environ 127 car, dans notre cas, les valeurs des pixels d'une image sont comprises entre 0 et 255. Notez que α représente l'opacité.

$$\begin{aligned} IblackMask(x,y) &= I(x,y) * (1-\alpha) + color * \alpha \\ \text{if } color \text{ is equal to zero, } &\Rightarrow IblackMask(x,y) = I(x,y) * (1-\alpha) \\ &\Leftrightarrow \alpha = 1 - \frac{IblackMask(x,y)}{I(x,y)} \\ \alpha &= 1 - \frac{mean(IblackMask(x,y))}{mean(I(x,y))} \\ \alpha &= 1 - \frac{mean(IblackMask(x,y))}{127} \end{aligned} \tag{8.1}$$

- Nous estimons le niveau de protection de la **pixellisation** en comptant le nombre de pixels entre chaque changement de couleur horizontalement et verticalement, noté $change\_size$. Nous faisons la moyenne de tous les $change\_size$. La relation entre la taille des carrés $squares\_size$ et la moyenne des $change\_size$ est notée dans la formule 8.2.

$$squares\_size = -6.642 + 1.43 * mean(change\_size) \tag{8.2}$$

- Nous estimons le niveau de protection du **floutage** en fonction du pourcentage des contours, noté $edges$. Nous détectons les contours avec Canny. La relation entre le pourcentage des contours $edges$ et l'écart-type, $standard\_deviation$ est notée dans la formule 8.3.

$$standard\_deviation = 16.17 * \exp(-37.8 * edges) \tag{8.3}$$

- Nous estimons le niveau de protection du **bruit** en fonction de la quantité de bruit (en pourcentage), notée $noise$, que nous extrayons des coefficients de la transformée discrète en ondelettes (DWT) [103, 105]. Selon ce pourcentage, nous estimons l'écart type, $standard\_deviation$ avec la formule 8.4.

$$standard\_deviation = 0.003 + \exp(94.35 * noise - 8.75) \tag{8.4}$$

Les méthodes de restauration d'images que nous avons sélectionnées pour nos expériences sont:

- **De-noicisseur**: La formule 8.1 permet d'assombrir des images. $color$ est égal à zéro car on utilise une couleur noire. Ainsi nous obtenons l'équation inverse suivante:

$$\begin{aligned} color &= 0 \\ I'(x,y) &\sim \frac{mean(IblackMask(x,y))}{(1-\alpha)} \end{aligned} \tag{8.5}$$

- **Super-résolution 1**: On réduit la taille de l'image en fonction de l'estimation du paramètre $squares\_size$, puis on ré-agrandit en utilisant la méthode d'interpolation bicubique [106] très populaire dans le domaine de la super-résolution.
- **Super-résolution 2**: En fonction de l'estimation du paramètre $squares\_size$, nous en déduisant la force de la Gaussienne ainsi que l'écart type associé, puis nous appliquons la méthode décrite dans [109].
- **Dé-floutage 1**: On utilise la méthode unsharp. Le principe de cette méthode est de calculer une image contour à partir d'une image d'entrée et une version lissée de celle-ci et d'ajouter cette image contour à l'image d'entrée afin de réduire le flou.
- **Dé-floutage 2**: En fonction de l'estimation du paramètre $standard\_deviation$, nous déduisons le PSF (point spread function) et nous appliquons la méthode décrite dans [110].
- **Dé-bruitage 1**: On utilise l'algorithme de Wiener [104] que nous re itérons en fonction de l'estimation du paramètre $standard\_deviation$.
- **Dé-bruitage 2**: On utilise une méthode basée sur la décomposition en ondelette. Le nombre de décompositions en ondelettes dépend de l'estimation du paramètre $standard\_deviation$.

La Figure 8.5 illustre le processus proposé.

3.2 Résultats expérimentaux

Nous évaluons le taux de pourcentage de classification correcte (c.-à-d., la précision) et nous avons sélectionné trois bases de données de visages populaires: Feret [81], ScFace [83] et AT&T [82].

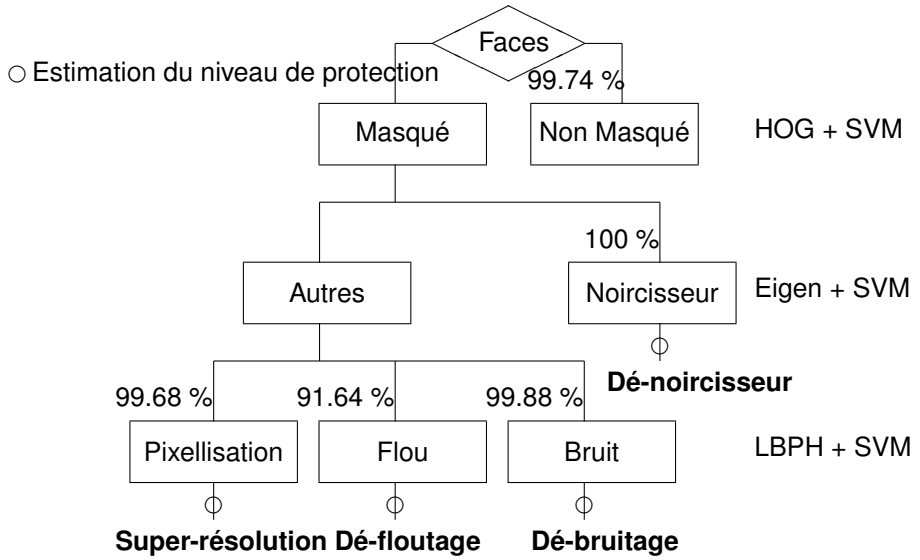


FIGURE 8.5: Méthode proposée.

		Prediction				
		Orig	Black	Pix	Blur	Noise
Ground Truth	Orig	99.7%	0.3%			
	Black	0%	100%	0%	0%	0%
	Pix	0%	0%	99.7%	0.3%	0%
	Blur	8.4%	0%	0%	91.6%	0%
	Noise	0%	0.1%	0%	0%	99.9%

TABLE 8.4: Matrice de confusion

Notre méthode de classification du type de protection a été testée sur 13 810 visages, plus précisément, sur 1149 visages originaux, 2302 visages assombris, 3138 pixelisés, 3768 floutés et 3453 bruités avec différents niveaux de protection.

La matrice de confusion présentée dans le Tableau 8.4 représente les taux de bonne classification. Selon ces résultats, la catégorie de certains visages masqués est légèrement classée à tort comme une autre catégorie. Par exemple, les visages pixelisés (0.3 %) comme visages flous, les visages flous (8.4 %) comme visages originaux et les visages avec du bruit (0.1 %) comme visages assombris. En effet, les visages pixélisés avec un faible niveau de protection ressemblent à des visages flous. Les visages flous de faible protection ressemblent à des visages originaux. Les visages bruités de forte protection sont plus proches des visages assombris, car le bruit affecte la luminance des images, perturbant ainsi l'algorithme Eigen.

Nous évaluons le taux d'identification faciale avec trois algorithmes souvent utilisés dans ce domaine: l'extraction des caractéristiques LBPH, HOG et Eigen avec le classifieur linéaire SVM.

Nous illustrons dans les Figures 8.6, 8.7 et 8.9, la différence entre le taux de bonne identification sur des visages sans protection et ce taux sur des visages masqués avant la restauration de l'image (en bleu), après restauration de l'image sans estimation de la protection (en jaune et brun) et après restauration

de l'image avec estimation de la protection (en rouge et vert). Plus la courbe est basse, plus les performances sont proches de celles des visages sans protection, et donc meilleure est la reconnaissance. D'après les Figures 8.7 et 8.9, les performances après restauration d'images avec l'estimation de la protection (en rouge et vert) sont les meilleures.

Par conséquent, la catégorisation de la protection d'identité et l'estimation du niveau de cette protection améliorent la performance des méthodes de restauration d'images. Nous avons prouvé que les protections basiques de la vie privée (masque noir, pixellisation, floutage ou bruit) ne sont plus efficaces avec l'approche proposée, car nous pouvons à nouveau identifier les visages.

3.3 Conclusion

Nous avons conçu un processus qui permet, tout d'abord, de détecter la présence d'une protection d'identité et dans un second temps, de classer le type de protection (masquage noir, pixellisation, flou et bruitage) et d'estimer son niveau. En utilisant une méthode de restauration appropriée à chaque type de protection, les résultats montrent que les performances de reconnaissance faciale sur les images de visage restaurées sont proches de celles obtenues pour les visages sans protection. Par conséquent, la vie privée des personnes peut être révélée et n'est plus protégée.

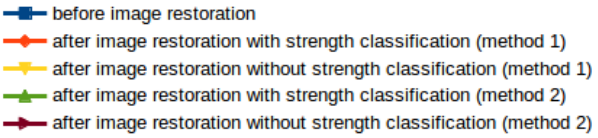


FIGURE 8.6: Légende pour les Figures 8.7 et 8.9.

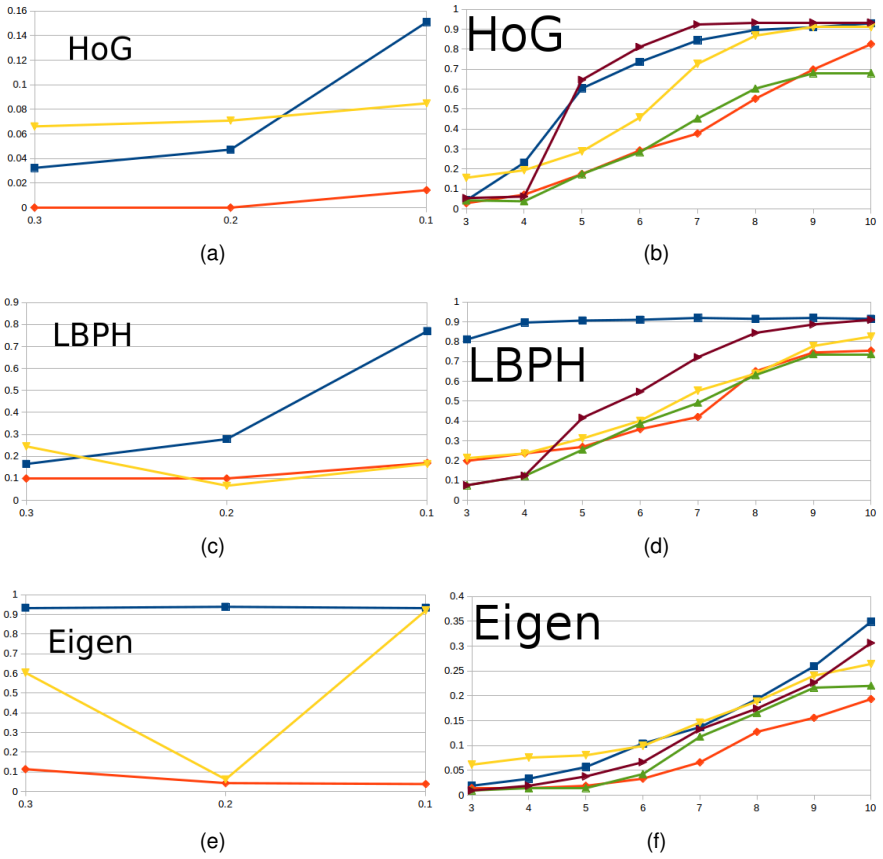


FIGURE 8.7: (a), (c) et (e) Impact du dé-noircisseur en fonction de l'opacité, (b), (d) et (a) Impact de la super-résolution en fonction de la taille des carrés.

FIGURE 8.8: Respectivement, l'image originale (a), masquage noir avec α = 0.9 (b), dé-noircisseur sans (c) et avec (d) l'estimation du niveau de protection. La pixellisation avec une taille des carrés = 5 (e), super-résolution sans (f) et avec (g) l'estimation du niveau de protection pour la première méthode, super-résolution sans (h) et avec (i) l'estimation du niveau de protection pour la seconde méthode.

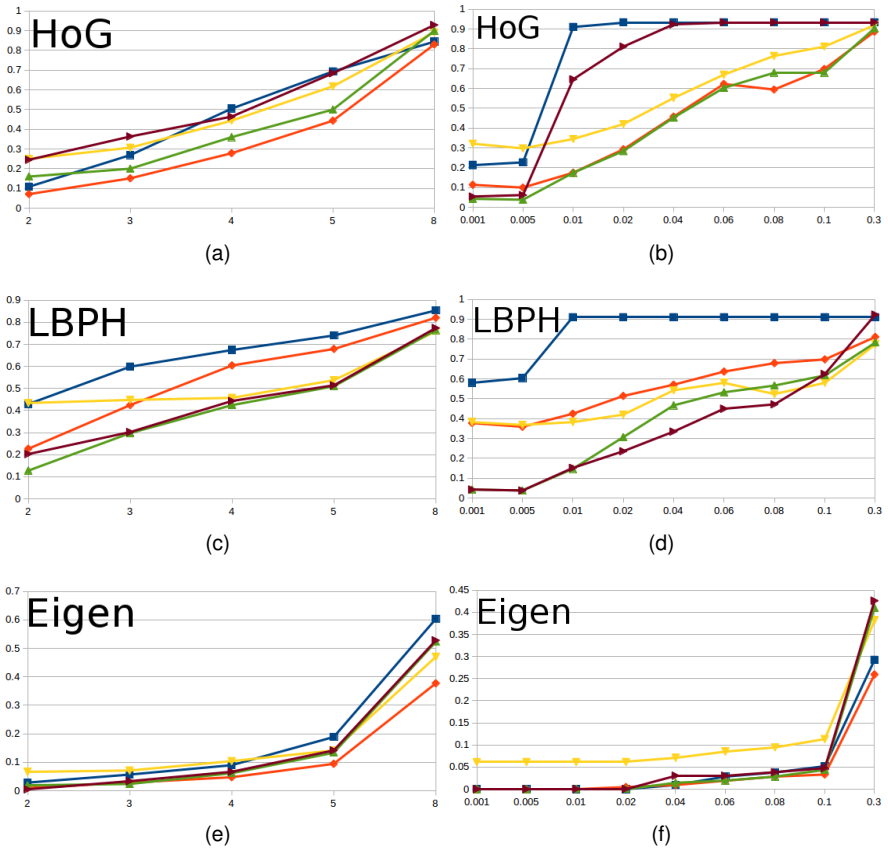


FIGURE 8.9: (a), (c) et (e) Impact du dé-floutage en fonction de l'écart type, (b), (d) et (a)Impact du dé-bruitage en fonction de l'écart type.

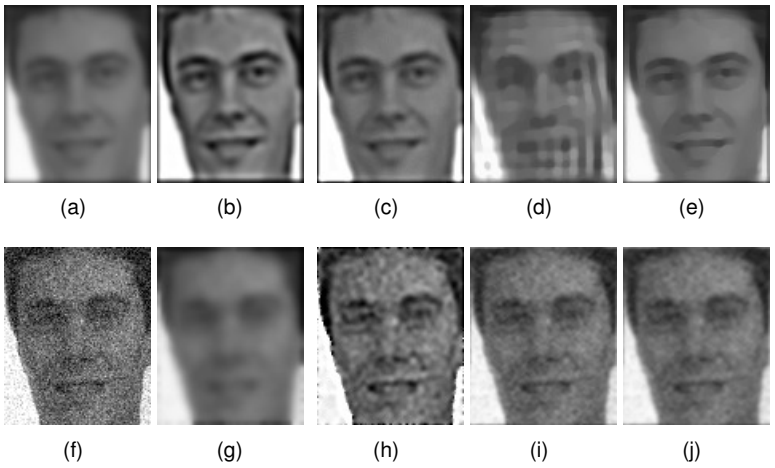


FIGURE 8.10: Respectivement, le floutage avec $\sigma = 2$ (a), dé-floutage sans (b) et avec (c) l'estimation du niveau de protection pour la première méthode, dé-floutage sans (d) et avec (e) l'estimation du niveau de protection pour la seconde méthode. L'image bruitée avec $\sigma = 0.01$ (f), dé-bruitage sans (g) et avec (h) l'estimation du niveau de protection pour la première méthode, dé-bruitage sans (i) et avec (j) l'estimation du niveau de protection pour la seconde méthode.

4 Protection de la vie privée préservant l'utilité de la surveillance visuelle dans le domaine spatial

Aucune des méthodes existantes de protection d'identité ne satisfait tous les critères requis par la surveillance (voir 2.3). Nous avons donc conçu une nouvelle méthode, appelée *StegoScrambling* qui protège la vie privée (aucune possibilité d'identifier les personnes) tout en préservant l'utilité de la surveillance visuelle (maintenir la forme et le mouvement des personnes pour reconnaître les événements). De plus, il satisfait également les trois critères suivants: réversibilité quasi sans perte (possibilité de récupérer les données originales), calculs rapides (exécution en temps réel) et sécurité (seules les personnes autorisées peuvent inverser le processus).

En images, nous définissons le nombre de bits par pixel (bpp) par le nombre de couleurs différentes qui est habituellement de 256, nous avons donc 8 bpp ($2^0 + 2^1 + 2^2 + 2^3 + 2^4 + 2^5 + 2^6 + 2^7 = 255$). L'approche proposée combine une méthode de cryptage et de stéganographie modifiant les 8 bpp. Nous cryptons et déplaçons les bits les plus significatifs (MSB) (c.-à-d., l'information la plus significative) des pixels d'une région d'intérêt (RoI) dans les bits les moins significatifs (LSB) (c.-à-d., l'information la moins significative). Ensuite, nous détectons les contours dans le RoI et les bits de ses pixels remplacent les MSBs de l'image résultante afin de garder la scène compréhensible.

4.1 Description de la méthode

Nous effectuons un XOR entre les six bits les plus significatifs (MSB) du RoI et des nombres aléatoires (RNs) (formule 8.6). Cette étape protège l'information originale et permet la réversibilité.

$$XORImg(x, y, c, i) = RoI(x, y, c, i) \oplus RandNums(x, y, c, i), \forall i \tag{8.6}$$

avec *(x, y)* les coordonnées des pixels, *c* le canal, et *i* la position du bit et chaque bit $\in \{0, 1\}$.

En parallèle, nous appliquons un détecteur de contour du RoI qui renvoie une image binaire (les contours en blanc et le fond en noir). Nous insérons les deux MSBs de l'image contour (l'intensité du pixel est soit 192 ou 0) dans les deux MSBs de l'image finale tandis que les six bits cryptés (intensité des pixels entre 0 et 63) sont intégrés dans les LSBs de l'image finale (formule **??**).

$$PrivacyImg(x, y, c) = \sum_{i=0}^{5} XORImg(x, y, c, i) * 2^i + \sum_{i=6}^{7} EdgeImg(x, y, i) * 2^i, \tag{8.7}$$

La Figure 8.11 illustre les étapes de la méthode proposée et l'image en haut à droite de la Figure 8.12 montre un exemple sur laquelle nous appliquons notre approche.

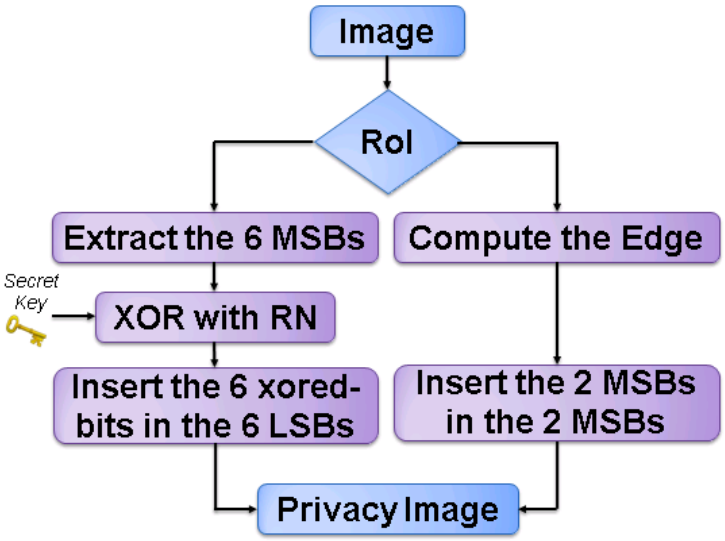


FIGURE 8.11: Présentation de l'approche.

FIGURE 8.12: Exemple de l'application de notre méthode de protection d'identité.

4.2 Description du processus inverse

Seules les personnes connaissant le mot de passe peuvent récupérer l'image d'origine. Ce mot de passe, s'il y est correct, génère la même séquence de nombres aléatoires que pour le cryptage. Nous effectuons un XOR entre les 6 LSBs de l'image protégée et des nombres aléatoires (formule 8.8). Enfin, pour récupérer l'image originale, on déplace le résultat du XOR vers les MSBs et on met à zéro les deux LSBs. Les deux dernières sous images de la Figure 8.12 montrent l'image récupérée dans le cas d'une clé correcte ou fausse.

$$RecoveredImg(x,y,c) = \sum_{i=2}^{7}(PrivacyImg(x,y,c,i-2) \oplus RandNums(x,y,c,i)) * 2^i \quad (8.8)$$

4.3 Exemple pour un pixel

Nous considérons un pixel codé avec 8 bits de MSB à LSB.

Pixel d'origine	$b7$	$b6$	$b5$	$b4$	$b3$	$b2$	$b1$	$b0$

Pour chaque pixel du RoI, nous conservons les bits entre 2 et 7 seulement (c.-à-d., les MSBs). Nous calculons un XOR entre les MSBs du pixel d'origine et les bits d'un nombre aléatoire. Nous dénotons le pixel crypté, b'.

XORpixel, b'	$b'7$	$b'6$	$b'5$	$b'4$	$b'3$	$b'2$	X	X

Nous déplaçons les bits de b' vers les LSBs.

XORpixel, b'	X	X	$b'7$	$b'6$	$b'5$	$b'4$	$b'3$	$b'2$

Les 2 MSBs d'un contour, e, sont représentées par $e_6 = 1$ et $e_7 = 1$ et d'un non-contour par $e_6 = 0$ et $e_7 = 0$. Enfin, nous ajoutons les 2 MSBs de e (le pixel représentant un contour ou non) avec les 6 LSBs de b' et désignons ce pixel comme le pixel protégé.

Pixel contour, b'+e	1	1	$b'7$	$b'6$	$b'5$	$b'4$	$b'3$	$b'2$
Pixel non-contour, b'+e	0	0	$b'7$	$b'6$	$b'5$	$b'4$	$b'3$	$b'2$

Pour récupérer le pixel d'origine, nous effectuons d'abord un XOR entre les bits du même nombre aléatoire que précédemment (grâce au mot de passe) et aux 6 LSBs du pixel protégé, puis nous décalons les LSBs vers les MSBs.

Pixel reconstruit	$b7$	$b6$	$b5$	$b4$	$b3$	$b2$	X	X

4.4 Résultats expérimentaux

4.4.1 Qualité des images reconstruites

Deux métriques, le **PSNR** et le **SSIM** (expliqué dans la section 2.5.4) ont été sélectionnées pour mesurer la qualité de la reconstruction des images de la base de données Feret [81] ainsi que sur celle de ScFaceData [83].

Dans le Tableau 8.5, nous représentons la moyenne et l'écart type (Std) du **PSNR** et du **SSIM** entre les images originales et celles reconstruites avec l'application inverse.

Nous perdons, dans le pire des cas, les deux LSBs de chaque pixel du RoI original. Par conséquent, l'intensité de couleur de l'image reconstruite diminue de trois (2^0+2^1), au maximum, par rapport à l'image

	PSNR	SSIM
Mean	42.46	0.9968
Std	0.2786	0.0013

TABLE 8.5: PSNR et SSIM entre les images originales et celles reconstruites.

originale. Cette perte n'a aucun impact sur la vision humaine et très peu pour la machine comme cela est montré dans le Tableau 8.5.

4.4.2 Évaluation de la protection de la vie privée

En utilisant un sous-ensemble de la base de données Feret et ScfaceData, nous avons entrainé quatre algorithmes de reconnaissance faciale: LBPH [75], Eigen [76], HoG [77] et OpenFace CNN [84] pour extraire les caractéristiques de l'image avec le classifieur SVM linéaire.

Les résultats prouvent clairement l'échec de tous ces outils de reconnaissance faciale lorsque nous appliquons notre approche ($\sim$ 0 % de bonne reconnaissance) alors que le taux de reconnaissance d'identité est presque le même pour les images originales ($\sim$ 95 % de bonne reconnaissance) et reconstruites ($\sim$ 90 % de bonne reconnaissance).

4.4.3 Attaque par force brute

Pour crypter, nous effectuons un XOR entre chaque pixel du RoI et des nombres aléatoires compris entre 0 et 63. Par conséquent, il existe $63^{height*width}$ combinaisons à tester si un utilisateur veut récupérer illégalement les données d'origine à partir des pixels cryptés (avec $height$ et $width$ la taille du RoI).

Le nombre de possibilités supérieur à 2^{128} représente déjà une limite impossible à atteindre avec la technologie actuelle [5]. La taille minimum du RoI nécessaire pour générer plus de 2^{2048} combinaisons est 18*18, comme nous le détaillons dans les équations 8.9. Les images de visage ou de corps inférieures à 18*18 pixels sont peu fréquentes. Par conséquent, notre processus est robuste contre une attaque par force brute, car il produit un nombre élevé de combinaisons.

$$\begin{aligned} 64^{height*width} &\geq 2^{2048} \\ \Leftrightarrow 2^{6*height*width} &\geq 2^{2048} \\ \Leftrightarrow 6*height*width &\geq 2048 \\ \Leftrightarrow height*width &\geq 341 \\ \text{En supposant que la hauteur est égale à la largeur} & \\ \Leftrightarrow height = width &\geq \sqrt{341} = 18 \end{aligned} \tag{8.9}$$

[5] http://cri.ensea.fr/en/node/208?destination=node%2F208

4.5 Conclusion

Nous avons présenté une nouvelle méthode de protection de la vie privée qui combine le cryptage (un XoR est appliqué entre des nombres aléatoires et des pixels originaux) et la stéganographie (les pixels cryptés sont cachés dans les LSBs des pixels de l'image).

Le processus inverse de cette méthode, nommé *StegoScrambling*, n'est pas assez robuste à la compression. En effet, il n'est pas possible de compresser les valeurs cryptées sans détériorations majeures sur les images reconstruites.

Nous proposons donc, dans la section suivante, une seconde méthode inspirée de celle-ci. En effet, nous préservons la compréhension de la scène tout en cryptant et cachant les informations les plus significatives vers les informations les moins significatives. Nous opérons dans le domaine de la transformée en cosinus discrète (DCT) au lieu du domaine spatial pour être conforme aux normes de compression classiques telles que JPEG et H.264/AVC.

5 Protection de la vie privée préservant l'utilité de la surveillance visuelle dans le domaine DCT

Cette section présente une nouvelle méthode de protection de la vie privée préservant la surveillance visuelle. L'algorithme fonctionne dans le domaine **Transformée en cosinus discrète (DCT)** pour permettre la compression de données avec les standards populaires, **JPEG** et **H.264**. Pour chaque zone sensible de l'image (zone où l'identité doit être protégée), l'algorithme proposé utilise les coefficients basses fréquences de la DCT pour préserver les actions de cette région et les coefficients hautes fréquences pour crypter et cacher la majorité de l'information originale. Enfin, notre processus permet aux utilisateurs, seulement autorisés, de récupérer les données d'origine grâce à un mot de passe qui est utilisé pour générer une séquence de nombres aléatoires.

Notre but principal est de fournir un bon compromis entre la vie privée et l'utilité de la surveillance visuelle tout en remplissant les autres critères introduits au début de la section 2.5.1. Par conséquent, notre méthode assure la protection de la vie privée à toute taille d'image tout en préservant la reconnaissance des sports dans la scène. Le système proposé est réversible presque sans perte (seulement pour les personnes possédant le mot de passe), et sécurisé contre les attaques par force brute, perroquet ou encore les attaques par remplacement. De plus, la méthode est conforme aux standards **JPEG** et **H.264**.

5.1 Méthode proposée compatible avec la norme JPEG

Notre processus opère entre les étapes de quantification et de codage entropique du processus JPEG tel qu'il est montré dans la Figure 8.13.

Appliquer une DCT sur un bloc 8*8 donne 64 coefficients : 1 DC [6] et 63 AC [7]. Pour chaque bloc 8*8 de la région d'intérêt (RoI), nous cryptons les coefficients DC et AC et les décalons vers les hautes fréquences ce qui laisse la valeur DC disponible. Nous déterminons alors automatiquement la moyenne de certains blocs comme la valeur du DC de chaque bloc 8*8.

Nous opérons uniquement sur le canal de luminance (Y). Le carré violet de la Figure 8.13(b) illustre les étapes de notre processus ajoutées à celles du framework JPEG classique.

5.1.1 Cryptage du DC

Pour crypter les coefficients DC, nous effectuons l'opération XOR bit à bit parmi chaque coefficient DC et un nombre aléatoire (entre 0 et 127). Ce nombre aléatoire est produit par un générateur de nombres pseudo-aléatoires (PRNG) contrôlé par une clé secrète. Cette clé secrète est un mot de passe choisi et connu uniquement des utilisateurs autorisés.

5.1.2 Cryptage des AC

Notons n, le nombre de coefficients AC avant la fin du bloc (EOB). EOB représente les coefficients restants, ceux qui sont égaux à zéro. Pour les crypter, nous permutons aléatoirement les $n-1$ coefficients AC en utilisant l'algorithme de mélange de Knuth [123] qui réorganise leur ordre. Le dernier coefficient non nul est utilisé pour marquer la fin de la permutation. En d'autres termes, les coefficients AC avant le dernier coefficient non nul sont permutés de manière aléatoire.

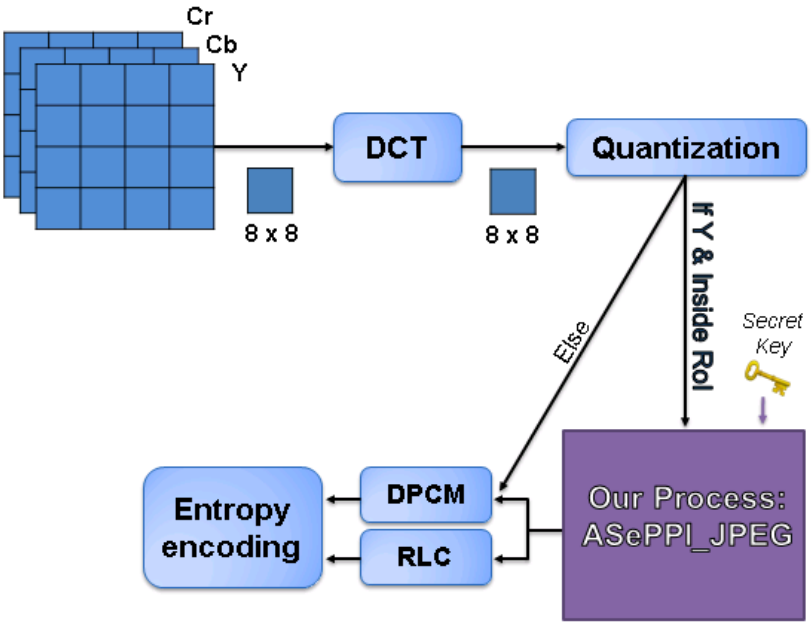
5.1.3 Division du DC et décalage des AC

La valeur du DC crypté, notée DC_e, pourrait être élevée et produire trop de distorsion sur l'image décompressée si elle est insérée telle quelle. Ainsi, cette valeur est divisée par un facteur (fixé à 17 dans notre implémentation). Nous décalons les 61 coefficients AC cryptés de deux positions vers les hautes fréquences afin de cacher le résultat de la division du DC dans le premier coefficient AC, et le reste dans le second.

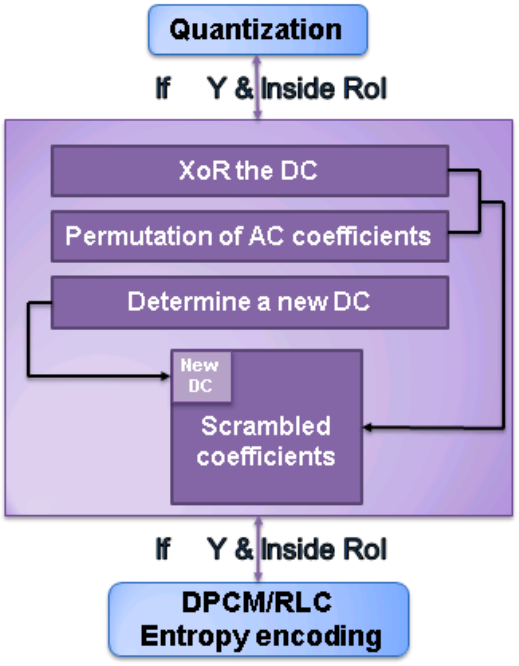
5.1.4 Exemple de la méthode

Soit les coefficients extraits [79 (DC), 0, -2, -1, -1, -1, 0, 0, -1, EOB], nous les encryptons : [107 (DC crypté), -1, 0, -2, -1, 0, 0, -1, -1, EOB]. Après avoir décalé les coefficients AC et divisé le DC crypté, les coefficients deviennent [$DC_{nouveau}$, 6, 5, -1, 0, -2, -1, 0, 0, -1, -1, EOB]. Nous réinsérons ces coefficients dans le bloc 8*8 en fonction du code en zigzag.

[6] Le coefficient DC représente la couleur moyenne du bloc.
[7] Les coefficients AC représentent les variations de couleur à travers le bloc.



(a)



(b)

FIGURE 8.13: Présentation du processus. YCbCr est une représentation de l'espace de couleur avec Y la luminance et Cb, Cr les composants de chrominance. (a) L'intégration dans le schéma JPEG (b) Les étapes ajoutées par notre approche.

5.1.5 Choix de la nouvelle valeur du DC

Tandis que les coefficients cryptés apparaissent sous forme de bruit dans la région protégée, la valeur du DC est dédiée à restituer une partie des informations d'origine.

Si nous gardons uniquement le DC (la luminance moyenne) de chaque bloc, l'image devient pixelisée de la taille de ces blocs (8*8). Nous allons insérer le même DC pour plusieurs blocs 8*8. Par exemple, si on veut une image pixélisée avec une taille de blocs de 16*16, les 4 sous-blocs 8*8 auront le même coefficient DC.

L'équation (8.10) représente la relation entre la taille des blocs voulus, notée S, et le nombre de blocs voulus, noté Nb, en fonction du nombre de pixels (hxw) du RoI.

$$Nb = \frac{h * w}{S * S} \tag{8.10}$$

Nous re écrivons l'équation (8.10) en fonction de Nb et sachant que S est un multiple de 8 car la taille des blocs (dans le processus de JPEG) est de 8*8.

$$S = round\left(\frac{\sqrt{\frac{h*w}{Nb}}}{8}\right) * 8 \tag{8.11}$$

Plus il y a de blocs, plus la qualité de l'image sera élevée et la visibilité de la scène meilleure. L'objectif est de trouver le nombre de blocs, Nb, qui préserve la visibilité des évènements tout en minimisant les performances de reconnaissance faciale. Pour cela, nous évaluons les performances de reconnaissance de visage en faisant varier Nb et la taille du RoI (h et w).

Nous avons utilisé l'algorithme de reconnaissance de visage OpenFace [84] disponible en ligne et les visages des 158 personnes qui avaient le plus d'images dans la base de données LFW Face [124]. Nous avons séparé aléatoirement dix fois la sous-base de données utilisée, pour obtenir 75 % des images de cette base en entrainement et 25 % en test. Le taux d'identification moyen pour les images originales est de 84 %.

Nous avons sélectionné $Nb = 106$ parce que le taux d'identification est inférieur à 6% d'après la Figure 8.14. Cependant, nous pouvons changer la valeur de Nb selon l'application, pour avoir plus ou moins de protection. Par exemple, pour cacher l'âge d'une personne, la valeur Nb pourrait être plus élevée.

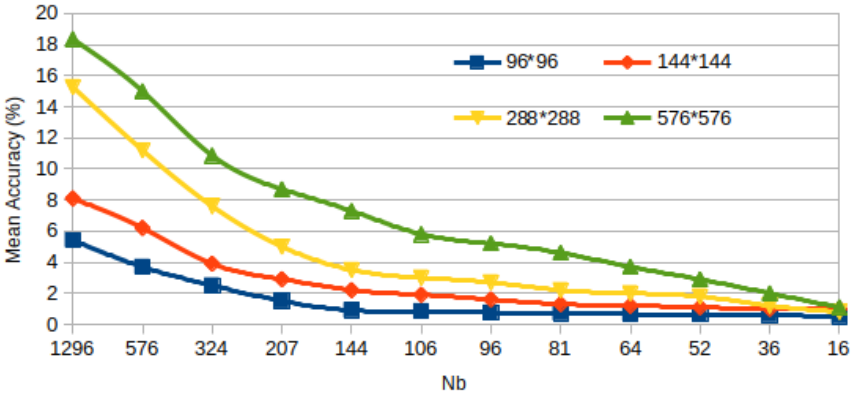


FIGURE 8.14: Précision de la reconnaissance d'identité (%).

5.2 Décompression avec ou sans mot de passe

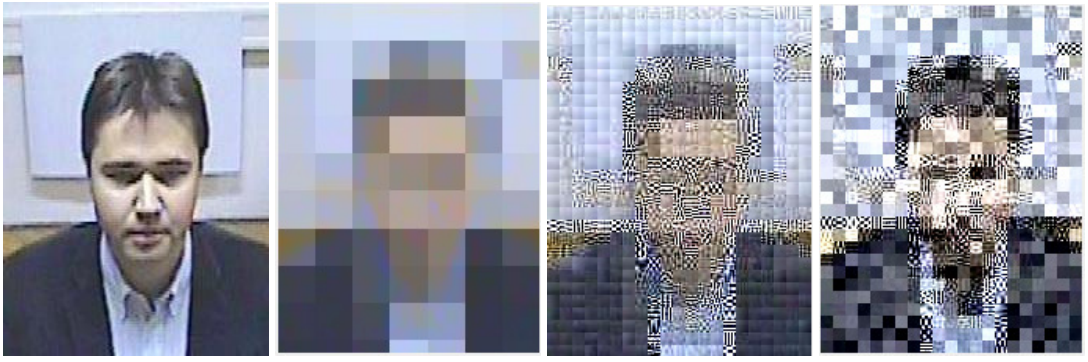
La décompression sans aucun mot de passe conduit à visualiser l'image protégée (Figure 8.15(c)). Nous décodons les données compressées avec le processus inverse de JPEG.

Lorsqu'un mot de passe est fourni, nous effectuons le processus inverse de nos étapes ajoutées, sur chaque bloc du canal de luminance (Y) situé à l'intérieur du RoI. Premièrement, nous extrayons les coefficients AC. Pour obtenir le DC crypté, nous multiplions le premier coefficient AC avec le facteur utilisé, et nous ajoutons le résultat au second coefficient AC. Pour récupérer le coefficient DC d'origine, nous effectuons l'opération XOR bit à bit entre le DC crypté et le nombre aléatoire associé. Si le mot de passe est correct, le PRNG génère les mêmes nombres aléatoires que ceux utilisés pour crypter. En utilisant l'algorithme inverse de Knuth shuffle, nous permutons les autres coefficients AC avant le dernier coefficient non nul. Ensuite, le processus applique la quantification inverse et la transformation DCT inverse sur les coefficients décryptés. Enfin, tous les blocs sont assemblés et nous convertissons l'image récupérée en RGB (rouge, vert et bleu).

L'image récupérée est similaire à l'originale, contrairement à la Figure 8.15(d) qui est illisible lorsqu'un utilisateur fournit un mauvais mot de passe.

5.3 Résultats expérimentaux

Nous évaluons et comparons notre approche proposée, appelée *ASePPI_JPEG*, avec une méthode similaire: *Scrambling_JPEG* [6] à deux niveaux de protection différents. Cette méthode crypte aléatoirement soit les signes des coefficients AC non nuls (dénoté *Scrambling_JPEG_LL* le niveau de protection faible) ou les signes des coefficients DC et AC non nuls (noté *Scrambling_JPEG_ML* le niveau de protection moyen). Nous opérons uniquement sur le canal Y pour une comparaison équitable avec notre approche.



(a) (b) (c) (d)

FIGURE 8.15: Avec S=24 et une qualité de compression JPEG de 75: (a) RoI original, (b) seuls les coefficients DC sont préservés pour chaque bloc du canal Y, (c) le RoI protégé , et (d) l'image décompréssée quand un mauvais mot de passe est utilisé.

5.3.1 Évaluation de la reconnaissance d'identité à partir des visages

Nous appliquons les trois différentes méthodes de protection de la vie privée sur les visages de la base de données LFW à différentes tailles. Nous évaluons la reconnaissance d'identité avec l'outil de reconnaissance de visage, OpenFace, en utilisant le même protocole que celui de la section précédente.

D'après la Figure 8.16, nous remarquons une augmentation du taux d'identification avec la méthode *Scrambling_JPEG_LL* lorsque la taille du RoI augmente. En effet, la taille de la pixellisation reste la même (blocs de 8*8 pixels) quelle que soit la taille du RoI. Par exemple, pour une taille RoI de 576*576, nous obtenons 30% en moyenne de bonne reconnaissance faciale lorsque nous appliquons la méthode *Scrambling_JPEG_LL*. Ce n'est pas suffisant pour assurer la protection de la vie privée. En effet, l'objectif est d'avoir le taux d'identification le plus bas possible. Au contraire, la méthode *ASePPI_JPEG* adapte automatiquement la taille de l'effet de pixellisation (en raison du choix des valeurs du DC), et rend ainsi la reconnaissance de visage très difficile même pour un RoI de taille élevée. En effet, le niveau de protection est similaire à celui de *Scrambling_JPEG_ML* (lorsqu'en plus de crypter les AC, les DC sont corrompus en changeant leurs signes).

5.3.2 Robustesse contre les attaques de perroquet et de remplacement

Une **attaque perroquet (PA)** [98] consiste à entrainer et à tester sur les images cryptées.

Une **attaque par remplacement (RA)** [53] implique de mettre à zéro toutes les valeurs cryptées tout en conservant les valeurs non cryptées. Nous implémentons cette attaque sur les méthodes *ASePPI_JPEG* et *Scrambling_JPEG_LL* en mettant tous les coefficients AC du canal de luminance à zéro tout en conservant les valeurs DC. Appliquer RA sur ces méthodes produit des images pixelisées de taille 8*8 pour l'approche *Scrambling_JPEG_LL* et pour *ASePPI_JPEG* cette taille change en fonction de la taille du RoI.

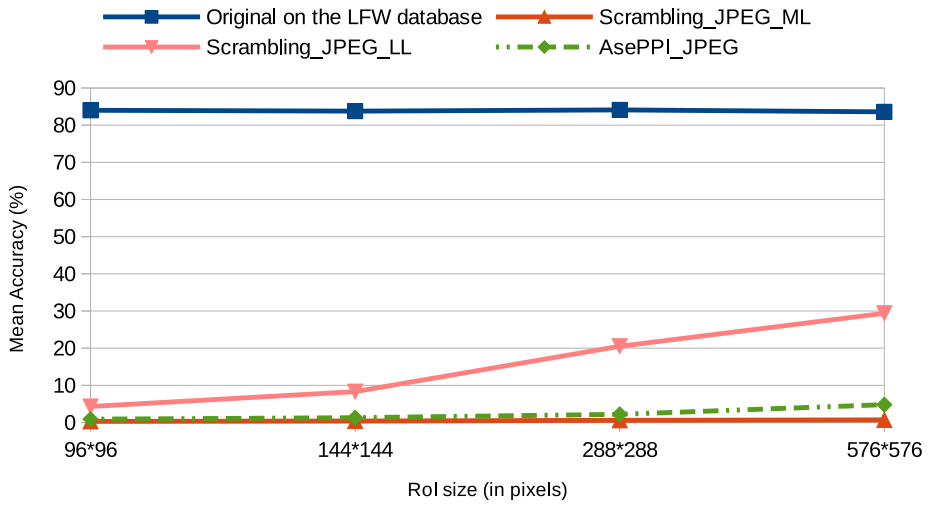


FIGURE 8.16: Le taux de bonne reconnaissance d'identité en fonction de la protection de la vie privée utilisée et de la taille RoI.

Nous évaluons les performances de la reconnaissance faciale avec ces deux attaques, et nous rapportons les résultats dans la Figure 8.17.

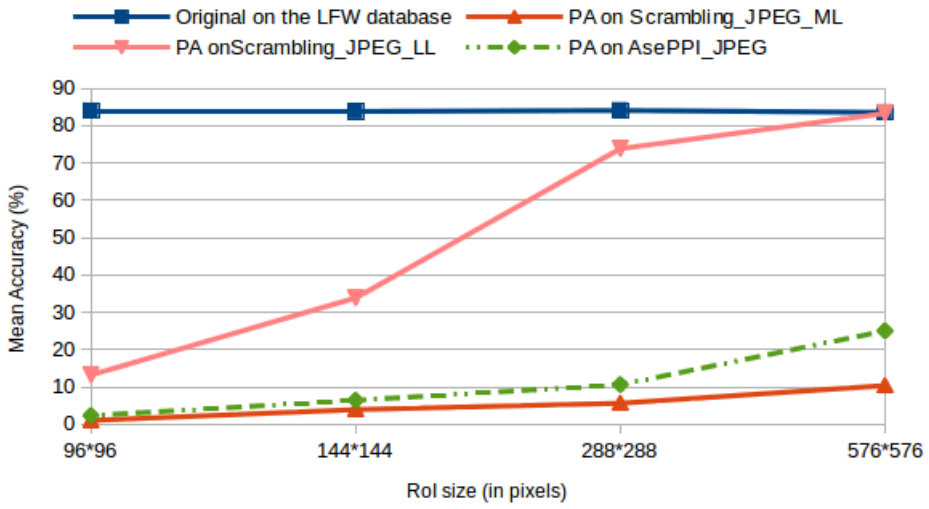
D'après ces résultats, nous prouvons que notre approche protège encore l'identité même en cas d'attaques contrairement à la méthode *Scrambling_JPEG_LL*. Ceci est principalement dû à la possibilité de contrôler la valeur du DC de chaque bloc. De plus, le niveau de robustesse de notre méthode est proche de celui de *Scrambling_JPEG_ML* (lorsqu'en plus de crypter les AC, les DC sont corrompus en changeant leurs signes).

5.3.3 Robustesse contre les attaques par force brute

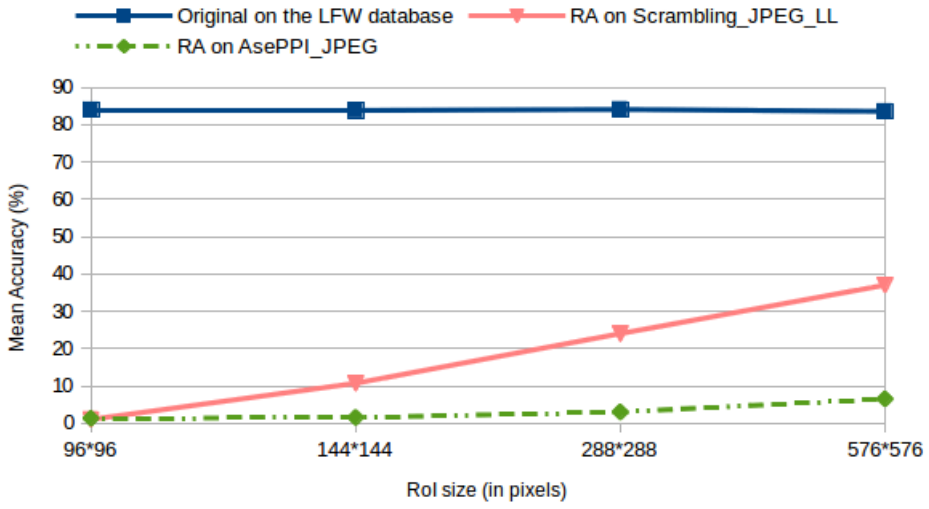
Nous évaluons la sécurité de la technique proposée contre une attaque par force brute. En supposant que l'attaquant connaisse la localisation du RoI et les étapes de notre algorithme, nous considérons une recherche de toutes les combinaisons possibles. Pour une qualité JPEG de 75% (qui est la valeur par défaut de l'encodeur de référence JPEG), le nombre moyen de coefficients AC par bloc est de 23, nous devons donc tester environ $23!$ (soit 2^{51}) combinaisons pour retrouver un bloc original 8*8.

Généralement, une taille minimale de l'image est requise pour une identification. À titre d'exemple, pour être éligible à prouver l'identité d'une personne, les lois en France imposent d'avoir au moins 90 pixels entre le bas du menton et le haut du crâne ou des cheveux, et 60 pixels entre les deux oreilles (incluses) [8]. Ainsi, 90 x 60 pixels est la taille minimale autorisée pour identifier quelqu'un, et une image de cette taille contient 84 blocs 8*8. Pour une qualité JPEG de 75 %, le nombre de combinaisons à tester pour reconstruire une image de taille 90 x 60 pixels protégée par notre processus est de $(23!)^{84} > 2^{4284}$. Le nombre de possibilités supérieur à 2^{128} représente déjà une limite impossible

[8] http://www.telecoute.re/livre-blanc-conformite-v31.pdf



(a)



(b)

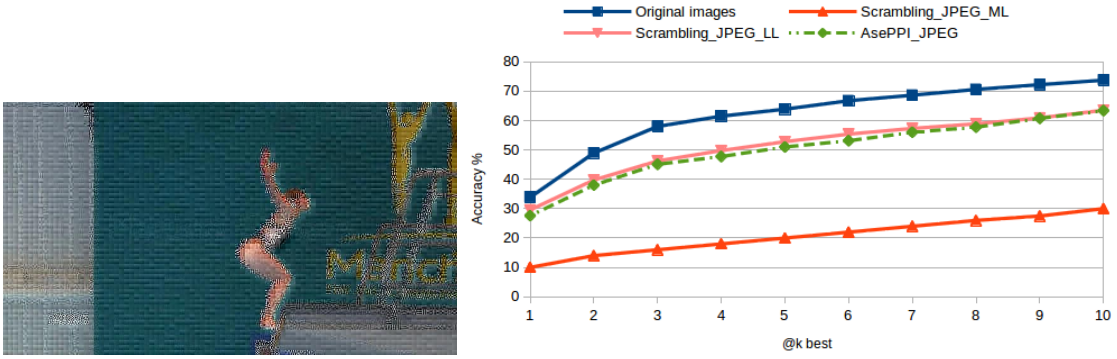
FIGURE 8.17: Le taux de bonne reconnaissance d'identité avec une attaque de (a) Perroquet (PA) et (b) par remplacement (RA).

à atteindre avec la technologie actuelle [9]. Par conséquent, la méthode *ASePPI_JPEG* fournit un niveau de sécurité probablement suffisant contre les attaques par force brute.

5.3.4 Évaluation de la préservation de l'utilité visuelle par classification des événements sportifs

Pour évaluer la préservation de l'utilité visuelle, nous avons choisi de tester notre algorithme sur les événements sportifs. Nous utilisons Deepdetect [10] pour classifier les sports et UCF Sports comme

[9] http://cri.ensea.fr/en/node/208?destination=node%2F208
[10] http://www.deepdetect.com/



(a) (b)

FIGURE 8.18: (a) Notre protection de la vie privée appliquée sur une image de sport (plongée) avec $Nb = 106$, (b) Taux de bonnes classifications des sports.

jeu de données [95]. Nous incluons plus de détails sur cet outil et cette base de données dans la section 2.5.4. Pour chaque image sélectionnée, nous appliquons les méthodes de protection de la vie privée *ASePPI_JPEG*, *Scrambling_JPEG_LL* et *Scrambling_JPEG_ML* sur l'image entière. De cette façon, l'arrière-plan ne faussera pas les résultats. L'outil de classification génère une liste ordonnée de prédiction des classes du meilleur au pire. Par conséquent, la Figure 8.18(b) représente la courbe de précision de k = 1 jusqu'à 10 (c.-à-d. si la classe appropriée est parmi les k premiers meilleurs résultats dans la liste ordonnée).

Selon les résultats de la Figure 8.18(b), le taux de bonne classification sur des images protégées avec *ASePPI_JPEG* diminue (en moyenne de 20 $\%$) par rapport au taux sur des images originales. Cependant, ils sont meilleurs qu'avec *Scrambling_JPEG_ML* (en moyenne de 30 $\%$) et sont proches des résultats de la méthode *Scrambling_JPEG_LL* (en moyenne 1 $\%$). En effet, garder la couleur moyenne de certains blocs aide à reconnaître les actions.

5.3.5 Impact sur l'efficacité de la norme JPEG

Nous mesurons la qualité des images reconstruites avec les métriques suivantes: le **PSNR**, le **SSIM**, le **ESS** et le **LSS**. Nous fournissons plus de détails sur ces mesures dans la section 2.5.4.

Pour les méthodes *ASePPI_JPEG*, *Scrambling_JPEG_LL* et *Scrambling_JPEG_ML*, le PSNR, le SSIM, le LSS et le ESS évaluent la qualité visuelle sur des images de visage reconstruites et prises au hasard dans la base de données LFW Face. Nous calculons également le taux de compression et le temps d'exécution des différentes approches en utilisant le langage Python sans GPU. Nous avons fait ces évaluations sur les parties du RoI (non sur l'image entière).

Le Tableau 8.6 montre les performances de chaque méthode de protection de la vie privée par rapport celle du processus JPEG en termes de qualité de la reconstruction, du taux de compression et du temps d'exécution.

TABLE 8.6: Impact sur l'efficacité du processus JPEG sur les parties du RoI.

For a JPEG quality of 75%	*JPEG* (100 %)	*AsePPI_JPEG*	*Scrambling_JPEG_LL*	*Scrambling_JPEG_ML*
PSNR	37 dB	37 dB (100 %)	37 dB (100 %)	37 dB (100 %)
SSIM	0.98	0.94 (95.92 %)	0.98 (100 %)	0.98 (100 %)
LSS	0.86	0.76 (88.37 %)	0.86 (100 %)	0.86 (100 %)
ESS	0.84	0.83 (98.81 %)	0.84 (100 %)	0.84 (100 %)
Compression Ratio	18.17	13 (71.54 %)	14.52 (79.86 %)	14.49 (79.75 %)
Time execution	0.2 s	0.205 s	0.202 s	0.202 s

Les méthodes *Scrambling_JPEG_LL* et *Scrambling_JPEG_ML* n'affectent pas la qualité des images récupérées, car elles conservent toutes les informations contrairement à notre approche qui perd, dans le pire cas, deux coefficients les moins significatifs pour chaque bloc (il n'y a pas de perte lorsque les deux coefficients les moins significatifs sont nuls). Cependant, nous observons une légère diminution de qualité (environ 8 % par rapport à JPEG).

Toutes les méthodes diminuent le taux de compression surtout en utilisant notre approche (environ 30 % par rapport à JPEG). En effet, nous appliquons un XOR sur chaque coefficient DC qui annule le bénéfice de sa quantification. De plus, le DC crypté est divisé en deux coefficients à l'intérieur des coefficients AC ce qui ajoute des coefficients supplémentaires à mémoriser.

L'impact sur le temps d'exécution est minime, car les étapes ajoutées (c.-à-d., le calcul de la nouvelle valeur du DC, le cryptage et le décalage des coefficients) s'exécutent rapidement.

Ces différences sont négligeables, car nous effectuons nos modifications uniquement sur le RoI, et habituellement le RoI est une sous-partie de l'image entière.

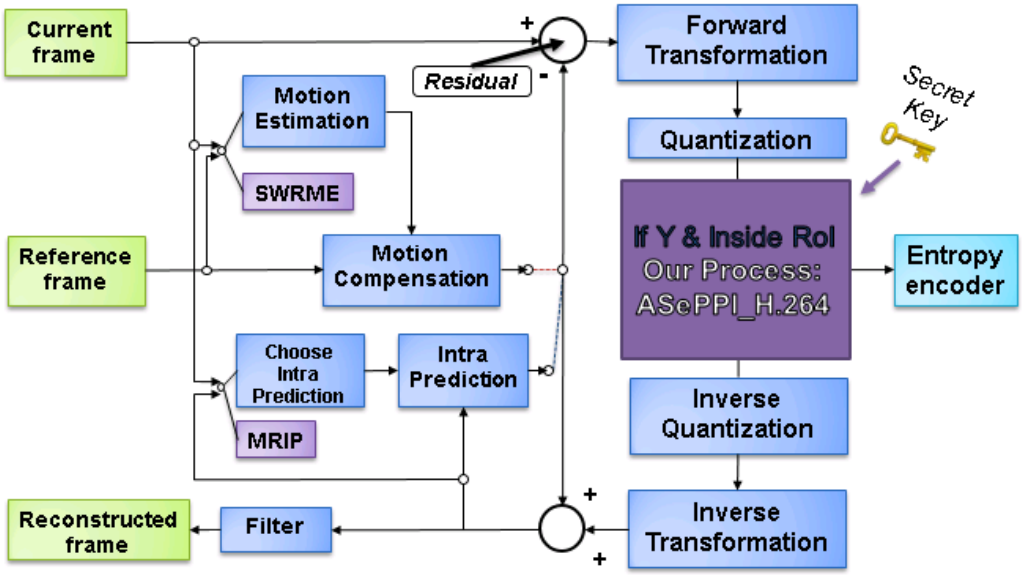
5.4 Conclusion

Le processus présenté dans cette section adapte automatiquement la protection de la vie privée en fonction de la taille des régions à protéger. Nous avons prouvé que notre méthode est la plus appropriée à protéger la vie privée tout en préservant la surveillance visuelle. Elle satisfait également d'autres critères très importants pour la surveillance tels que la réversibilité (seulement pour les personnes connaissant le mot de passe), la sécurité (robuste aux attaques) et la compatibilité avec la compression (JPEG).

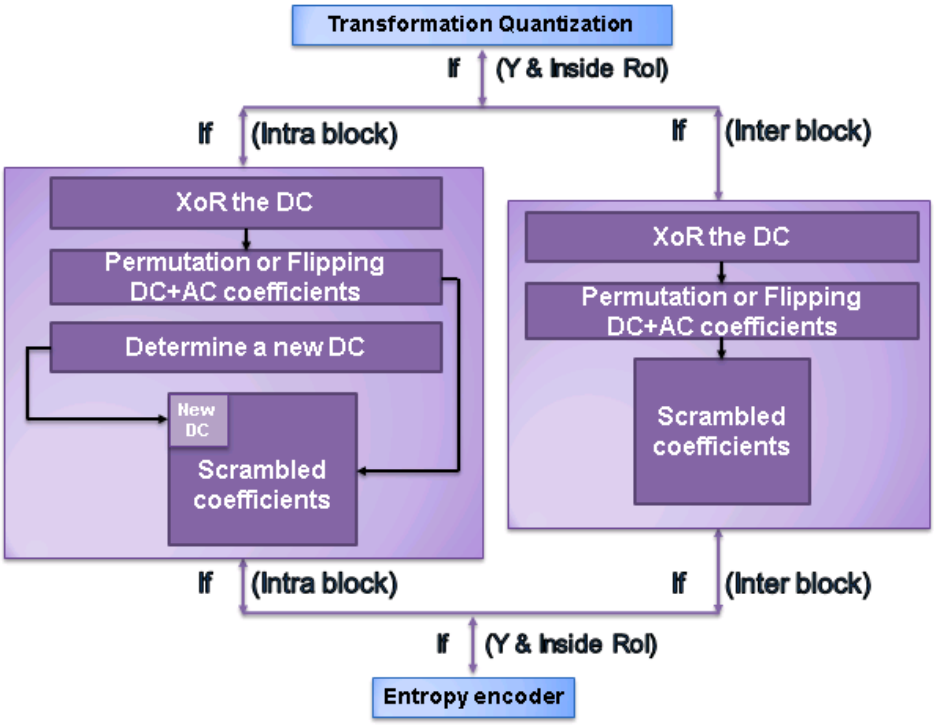
Dans la prochaine section, nous intégrons une approche similaire dans la norme H.264/AVC pour être compatible avec la compression vidéo.

5.5 Méthode proposée compatible avec la norme H.264/AVC

Nous ajoutons quelques étapes (5.5.1, 5.5.2, 5.5.3, 5.5.4) après la transformation et la quantification de chaque bloc résiduel 4*4 qui se trouve à l'intérieur du RoI et pour le canal de luminance seulement, comme nous l'illustrons dans la Figure 8.19.



(a)



(b)

FIGURE 8.19: Présentation du processus. (a) L'intégration dans le schéma H.264/AVC (b) Les étapes ajoutées par notre approche.

Nous cryptons les coefficients DCT des intra et inter blocs résiduels (5.5.1, 5.5.2), avec un générateur de nombres pseudo-aléatoires (PRNG) contrôlé par un mot de passe afin de protéger les données et être réversibles uniquement par les personnes autorisées. Ensuite, pour les intra blocs résiduels seulement, nous décalons les coefficients cryptés d'une position vers les hautes fréquences (5.5.3) pour rendre disponible la position DC (c.-à-d., la fréquence la plus basse représentant la moyenne des valeurs d'un bloc). Cette position sera utilisée plus tard pour insérer une valeur DC appropriée (5.5.4). L'étape de décalage conduit à perdre le coefficient le moins significatif de chaque bloc s'il s'agit d'une valeur non nulle.

5.5.1 Cryptage du DC

Algorithm 10: Cryptage du DC.

if $(|DC| < 16)$ **then**
  $X = 16$;
else
  $X = 2^n$;
end
if $(DC \neq 0)$ & $((|DC| \neq (RN \mod X)))$ **then**
  DC_e = ($|DC| \oplus (RN \mod X)$)*sign(DC);
else
  DC_e = DC;
end
with $n = \lfloor \log_2 |DC| \rfloor$ an integer

Effectuer un XOR entre le DC et un nombre aléatoire généré à partir d'une plage infinie peut conduire à un DC crypté plus grand que celui d'origine. Cette différence de valeur produit beaucoup de bruit dans les images protégées. Par conséquent, pour minimiser ce bruit, nous concevons un algorithme de cryptage (voir algorithme 10), dans lequel les valeurs cryptées resteront dans la même plage que celle d'origine.

5.5.2 Brouillage des coefficients (le DC crypté + les coefficients AC)

Pour chaque bloc, nous sélectionnons la méthode de brouillage (entre RP et SNC, expliquée ci-dessous) qui a le plus grand nombre de combinaisons pour récupérer les coefficients d'origine. Le DC crypté est inclus dans le brouillage. Nous utilisons le PRNG pour générer une séquence de nombres aléatoires.

RP: Notez p_1, le nombre de coefficients avant EOB (End-of-Block, les coefficients restants sont nuls). Pour les brouiller, nous permutons aléatoirement les coefficients $p_1 - 1$ en utilisant l'algorithme de mélange de Knuth [123] qui réorganise leur ordre. Le dernier coefficient non nul est utilisé pour marquer la fin de la permutation (c.-à-d., les coefficients avant le dernier non nul sont permutés de manière aléatoire). Ainsi, il y a $(p_1 - 1)!$ combinaisons.

SNC: Notez p_2, le nombre de coefficients non-zéros. Nous permutons aléatoirement le signe de ces coefficients. Par conséquent, il y a 2^{p_2} combinaisons.

Les deux méthodes, RP et SNC, sont réversibles.

5.5.3 Déplacement des coefficients brouillés vers les coefficients AC (pour les intra blocs seulement)

À titre d'exemple, nous supposons que les coefficients extraits d'origine sont [31 (DC), 0, -2, -1, -1, -1, 0, 0, -1, EOB]. Nous cryptons le DC qui devient 24: [24 (DC_e), 0, -2, -1, -1, -1, 0, 0, -1, EOB]. Il y a $8! = 40320$ combinaisons avec la méthode RP et $2^6 = 64$ avec SNC. Ainsi, nous sélectionnons la méthode RP pour brouiller les coefficients qui deviennent [-1, 0, 24 (DC_e), -2, -1, 0, 0, -1, -1, EOB], et nous les décalons d'une position vers les hautes fréquences ce qui conduit à [DC_{new}, -1, 0, 24 (DC_e), -2, -1, 0, 0, -1, -1, EOB]. Ensuite, nous réinsérons les coefficients cryptés dans le bloc 4*4 en fonction du code zigzag et choisissons la nouvelle valeur DC avec la formule définie dans la section 5.5.4.

Cette étape conduit à perdre dans le pire des cas un coefficient par bloc quand le dernier coefficient AC est différent de zéro.

5.5.4 Choix de la nouvelle valeur du DC (pour les intra blocs seulement)

Nous dédions la valeur DC_{new} pour préserver un minimum d'information requis par la surveillance (par exemple, la luminance moyenne des blocs résiduels).

Le maintien du DC d'origine (c.-à-d., la moyenne) de chaque bloc résiduel du canal de luminance conduit à un léger floutage du visage. Pour obtenir une meilleure protection de la vie privée, nous conservons le DC d'un bloc plus grand et l'insérons dans les DC de ses sous-blocs 4*4 (en suivant le même processus que dans la section 5.1.5).

L'équation (8.10) déjà définie dans la section 5.1.5, représente la relation entre la taille des blocs b_{roi}, notée S, et le nombre de blocs à l'intérieur du RoI, notés Nb, en fonction du nombre de pixels (h x w) de ce RoI. Par exemple, si S est égal à 24, les blocs résiduels à l'intérieur du bloc 24*24 ont le même coefficient DC, qui est le DC du bloc 24*24 (autrement dit, la moyenne du bloc 24*24).

Plus Nb est grand, plus la qualité de l'image sera élevée et donc meilleure est la reconnaissance. Notre objectif est de maximiser Nb pour préserver l'intelligibilité (c.-à-d., l'utilité de la vidéosurveillance) autant que possible tout en minimisant les performances de la reconnaissance faciale. Par conséquent, pour atteindre ce but, nous avons fait dans la section 5.1.5, une étude empirique en fixant plusieurs valeurs, et nous trouvons dans notre cas que Nb devrait être égal à 106. Ainsi, nous calculons automatiquement S avec l'équation 8.12 en prenant en compte que la taille de chaque bloc résiduel est de 4*4. Cependant, nous pouvons changer la valeur de Nb pour avoir une protection plus ou moins forte.

$$S = \left\lceil \frac{\sqrt{\frac{h*w}{106}}}{4} \right\rceil * 4 \tag{8.12}$$

5.6 Décompression avec ou sans mot de passe

La décompression sans aucun mot de passe conduit à visualiser la vidéo protégée décompressée en utilisant un décodeur H.264/AVC.

Pour récupérer la vidéo initiale, nous appliquons le processus inverse en utilisant le mot de passe. Si le mot de passe est correct, les mêmes nombres aléatoires que ceux utilisés lors du cryptage sont générés, ce qui permet de décrypter correctement. Pour les intra blocs résiduels à l'intérieur du RoI, nous extrayons les coefficients AC (pas les DC), et pour les inter blocs tous les coefficients (DC + AC) de ces blocs. Comme dans la section 5.5.2, nous sélectionnons la méthode de brouillage (RP ou SNC) qui produit le plus grand nombre de combinaisons pour récupérer les données d'origine, puis nous appliquons son processus inverse. Finalement, nous décryptons le DC (c.-à-d., le premier coefficient décrypté) en appliquant l'algorithme 10.

5.7 Résultats expérimentaux

Nous comparons la méthode proposée, notée *ASePPI_H.264*, avec le cryptage des signes des coefficients non nuls *SNC* et avec celle qui en plus crypte les modes d'intra-prédiction *SNC+IPM*. Nous appliquons toutes les méthodes uniquement sur le canal de luminance à l'intérieur du RoI pour une comparaison équitable avec notre approche proposée. Trois exemples vidéo de l'application de ces méthodes sont disponibles [11].

Pour l'évaluation, nous avons sélectionné les séquences suivantes: 'hall', 'foreman', 'suzie', 'akiyo', 'carphone', 'claire' et 'miss-america' toutes disponibles sur le web [12], et 3 vidéos de 9 personnes qui ont le plus d'images dans la base de données YouTube pour évaluer la reconnaissance d'identité. Nous utilisons différentes valeurs de QP et IP dans nos évaluations. QP est le paramètre de quantification et IP la période intra qui définit le nombre d'images entre deux images Intra.

5.7.1 Évaluation de la reconnaissance d'identité à partir des visages

Nous entrainons l'outil OpenFace [84] avec deux vidéos pour chacune des 9 personnes de la base de données YouTube et testons le taux d'identification sur la vidéo restante de chaque personne (différente de celles qui ont été utilisées pour l'entrainement). Pour les visages originaux, nous obtenons 93,95 % de précision moyenne (c.-à-d., de classifications correctes). On remarque sur la Figure 8.21 que plus la taille du RoI est grande, plus la précision est élevée pour les visages protégés par SNC (plus de 30 % de

[11] www.dropbox.com/s/r0hbc8n48ocu4uk/Video_Examples.zip?dl=0
[12] http://trace.eas.asu.edu/yuv/

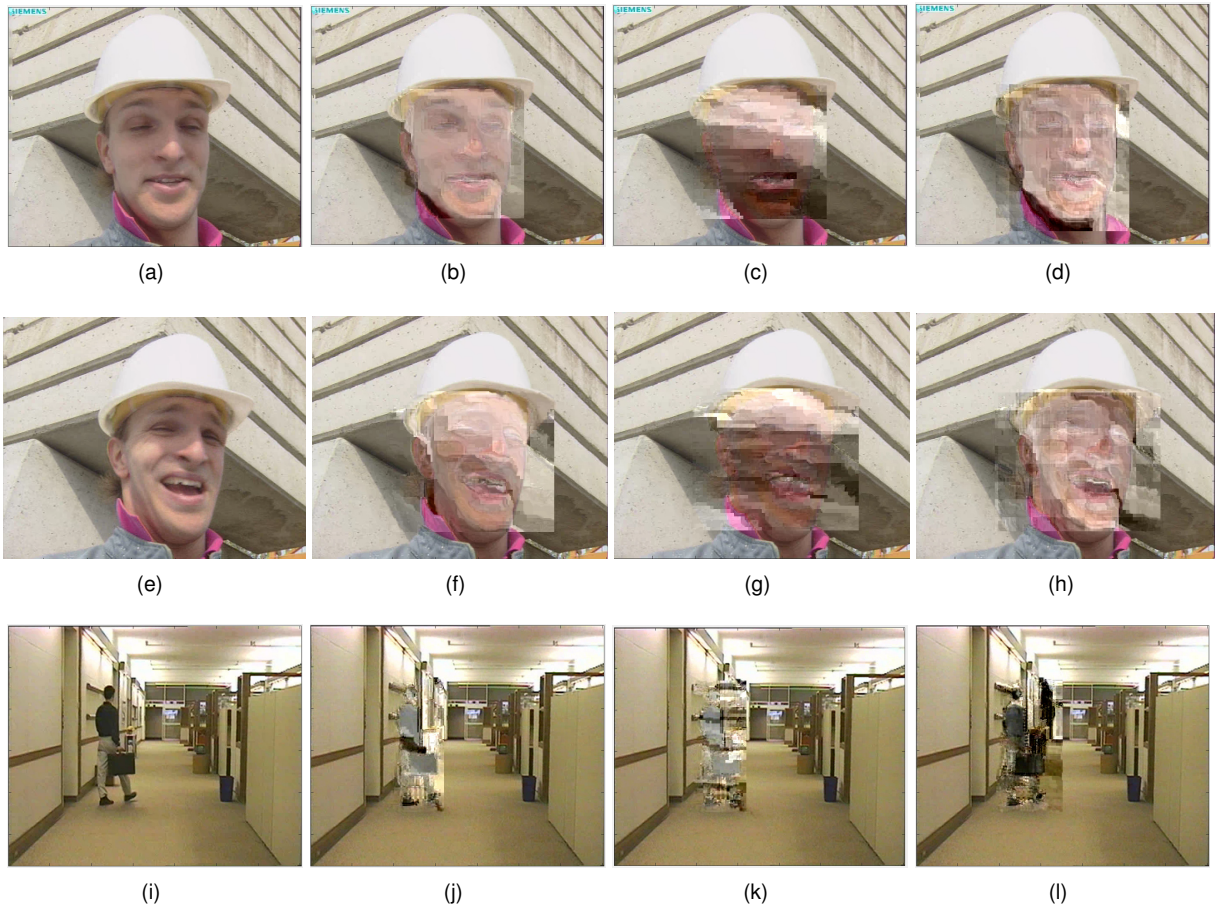


FIGURE 8.20: Avec QP= 24 et IP= 5: (a) La première image originale de la séquence 'foreman' (I), (e) la 15e image originale de la séquence 'foreman' (P), (i) la 39e image originale de la séquence 'hall' (P), (b), (f) et (j) cryptées avec SNC, (c), (g) et (k) cryptées avec SNC+IPM, (d), (h) et (l) cryptées avec *ASePPI_H.264*.

bonnes reconnaissances avec les images de taille 576*576). Avec les deux autres méthodes, SNC+IPM et *ASePPI_H.264*, moins de 17 % des visages sont bien identifiés. Par conséquent, la méthode SNC ne protège pas la vie privée à toute résolution contrairement aux deux autres approches.

5.7.2 Robustesse contre les attaques de perroquet et de remplacement

En utilisant le même protocole que la section précédente (avec la même base de données de visages YouTube et l'outil OpenFace), nous entrainons et testons les images sur lesquelles les méthodes de protection d'identité ont été appliquées.

La Figure 8.22(a) montre le taux de bonnes identifications lors de l'application de l'attaque de perroquet sur chaque méthode de protection de la vie privée. Nous remarquons que nous pouvons ré-identifier les personnes à plus de 50 % avec cette attaque sur l'approche SNC alors que les impacts de celle-ci sur les méthodes *ASePPI_H.264* et *SNC+IPM* sont presque équivalents (environ 24 % de ré-identification).

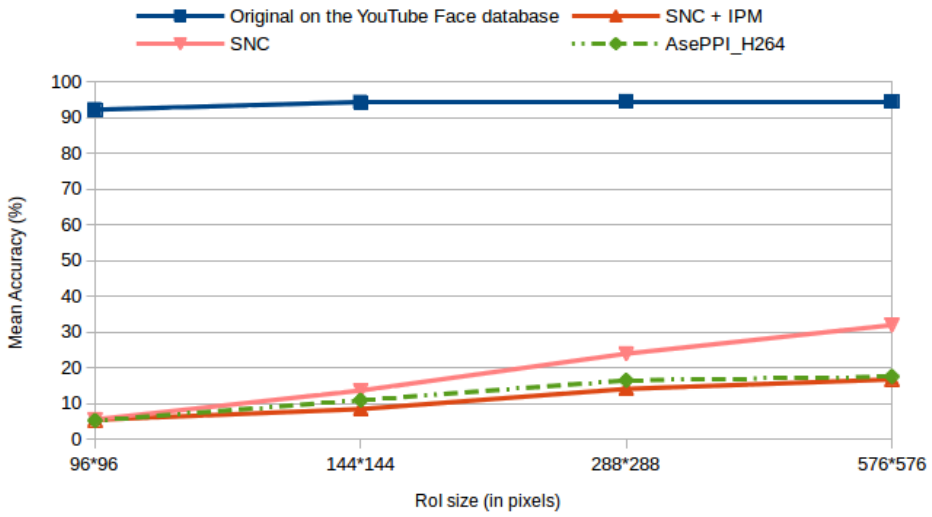


FIGURE 8.21: Le taux de bonne reconnaissance d'identité en fonction de la protection de la vie privée utilisée et de la taille RoI.

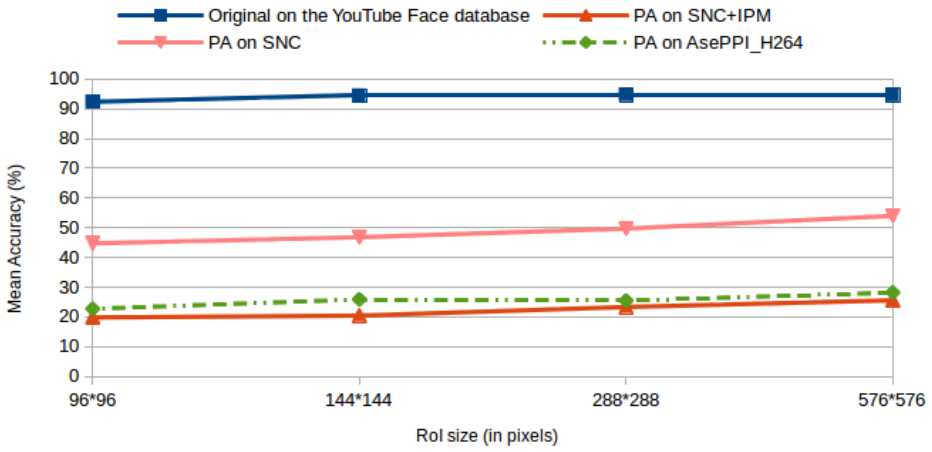
Un attaquant peut essayer de supprimer des données cryptées (c.-à-d., mettre à zéro tous les coefficients cryptés). Ainsi, pour notre méthode, tous les coefficients AC du RoI pour les intra blocs et tous les coefficients DCT (DC + AC) du RoI pour les inter blocs sont mis à 0. Pour la méthode SNC, nous avons simplement mis à 0 tous les coefficients AC du RoI, et pour SNC + IPM, nous avons mis en plus tous les modes intra-prédiction à 2 (c.-à-d., la moyenne). Des exemples vidéo de cette attaque sont disponibles en ligne [13].

Nous reportons, dans la Figure 8.22(b), le taux de bonne identification avec l'outil OpenFace (en utilisant le même protocole que la section précédente) lorsque l'attaque de remplacement est appliquée aux différentes méthodes. Selon les résultats, la méthode SNC devient plus faible en termes de protection de la vie privée lorsque la taille du RoI augmente, car l'algorithme ré-identifie les personnes avec plus de 40% de bonne reconnaissance, alors qu'avec notre méthode le taux de reconnaissance d'identité est toujours inférieur à 5%.

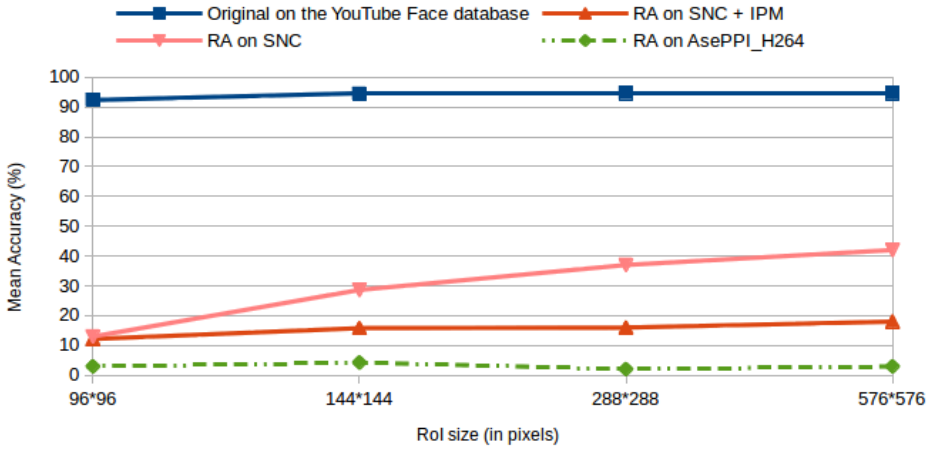
L'attaque de remplacement sur la méthode SNC fixe les coefficients AC à 0, tandis que les DC de chaque bloc résiduel 4*4 sont toujours disponibles et contiennent trop d'informations. Ainsi, l'identité n'est pas suffisamment protégée en particulier sur les images de haute résolution. Pour la méthode *ASePPI_H.264*, nous adaptons automatiquement les valeurs DC en utilisant le même DC pour plusieurs blocs afin de protéger l'identité à toute résolution ce qui la rend plus robuste à l'attaque par remplacement.

Le taux de bonnes ré-identifications en utilisant l'attaque de remplacement sur la méthode *SNC+IPM* reste autour de 15.5 %. Cependant, d'un point de vue humain, les détails du visage ou de la forme du corps sont beaucoup plus visibles que dans le cas de *ASePPI_H.264* comme l'illustre les Figures 8.23 et les exemples vidéo [13].

[13] www.dropbox.com/s/39ke5wy6mgezq4k/Video_Examples_SA.zip?dl=0



(a)



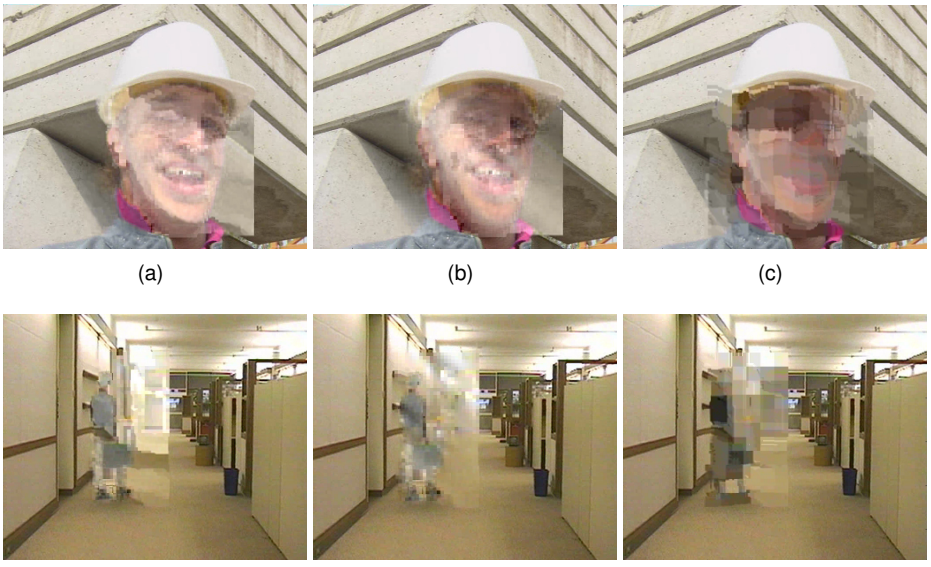
(b)

FIGURE 8.22: Le taux de bonne reconnaissance d'identité avec une attaque de (a) Perroquet (PA) et (b) par remplacement (RA).

5.7.3 Robustesse contre les attaques par force brute

Nous considérons une recherche exhaustive de toutes les combinaisons. Le nombre de combinaisons par bloc pour la méthode *ASePPI_H.264* est toujours supérieur ou égal à celui du SNC. En effet, comme expliqué dans la section 5.5.2, nous sélectionnons la méthode (entre SNC ou RP) qui effectue le plus grand nombre de combinaisons. Par conséquent, nous comparons seulement la méthode *SNC+IPM* avec *ASePPI_H.264* sur les intra blocs.

Nous calculons la moyenne du nombre de coefficients AC avant la fin du bloc (EOB) ainsi que la moyenne du nombre des coefficients non-zéros AC avant EOB parmi tous les 4*4 intra blocs du canal de luminance. Nous notons respectivement ces nombres, nbr_AC et nbr_NAC.



(a) (b) (c)

(d) (e) (f)

FIGURE 8.23: Avec QP= 24 et IP= 5: Après l'attaque par replacement sur la méthode SNC (a) et (d), sur SNC+IPM (b) et (e) sur *ASePPI_H.264* (c) et (f).

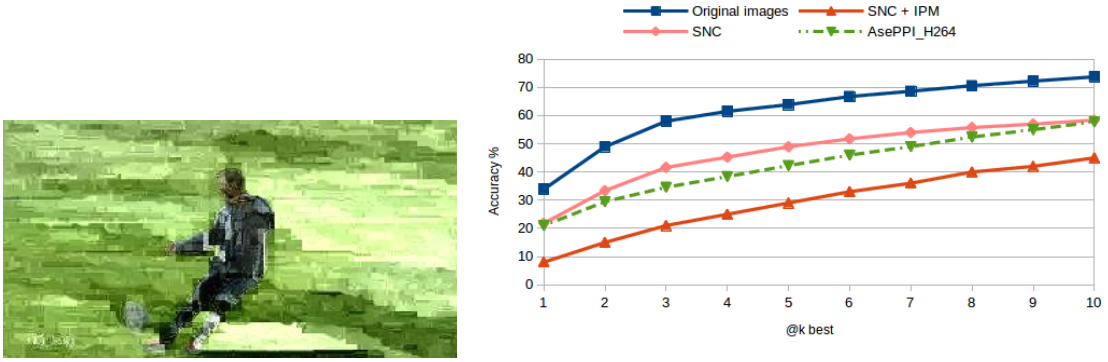
Le nombre de possibilités par intra bloc pour la méthode *ASePPI_H.264* est de $max(2^{nbr\_NAC}, nbr\_AC!)$, et pour *SNC+IPM* de $(2^{nbr\_NAC}) * 9$ (9 intra modes). Nous ne comptons pas le nombre de combinaisons pour décrypter le DC (pour notre processus), car nous supposons que le DC peut être déduit du DC_{new}. Ainsi, nous rapportons les résultats dans le Tableau 8.7. D'après ce Tableau, nous déduisons que notre approche est plus robuste aux attaques par force brute que *SNC+IPM* pour QP inférieur ou égal à 24.

QP	**12**	**18**	**24**	**30**
ASePPI_H.264	3.99*10^7	40320	120	5.1
SNC+IPM	2.37*10^3	280	51	18

TABLE 8.7: Nombre moyen de combinaisons pour décrypter un intra bloc.

Comme déjà indiqué dans la section 5.3.3, une taille minimale de l'image est nécessaire pour identifier quelqu'un (par exemple, 90 x 60 pixels pour identifier le visage, en France [14]). Une image de cette taille contient 337.5 4*4 blocs. Pour reconstruire une image contenant des intra blocs, avec QP = 30 et QP = 24, le nombre de combinaisons est respectivement de $5.1^{337.5} = 10^{238} > 2^{512}$ et plus de 2^{2048}. Avec nos capacités de calcul actuelles, 2^{128} possibilités ou plus représente déjà une limite très difficile à atteindre [15]. Par conséquent, la méthode offre un niveau de sécurité suffisant.

[14] http://www.telecoute.re/livre-blanc-conformite-v31.pdf
[15] http://cri.ensea.fr/en/node/208?destination=node%2F208



(a) (b)

FIGURE 8.24: (a) Notre protection appliquée sur une image sportive (football) avec $Nb = 106$, (b) Précision de la classification des sports.

5.7.4 Évaluation de la préservation de l'utilité visuelle par classification des événements sportifs

Nous avons choisi Deepdetect [16] ainsi que la base de données UCF Sports [95] pour classer les événements sportifs. Nous appliquons les méthodes *ASePPI_H264*, *SNC* et *SNC+IPM* sur toute l'image.

L'algorithme de classification prédit des catégories du meilleur au moins bon résultat. Par conséquent, nous calculons et montrons sur la Figure 8.24(b), la courbe de précision de k = 1 jusqu'à 10 (c.-à-d., si la classe voulue est parmi les k premiers résultats dans la liste ordonnée).

D'après les résultats de la Figure 8.24(b), la précision de la classification sportive en appliquant *ASePPI_H.264* sur les images diminue (en moyenne de 20 %) par rapport aux images originales. De plus, les résultats sur notre méthode sont meilleurs qu'en appliquant *SNC+IPM* (en moyenne de 13 %) et diminuent légèrement en appliquant *SNC* (en moyenne de 4 %). En effet, garder la couleur moyenne de certains blocs aide à reconnaître les actions.

5.7.5 Impact sur l'efficacité de la norme H.264/AVC

Nous mesurons la qualité des images reconstruites avec les métriques suivantes: le **PSNR**, le **SSIM**, le **ESS** et le **LSS** (plus de détails sont fournis dans la section 2.5.4). Nous appliquons ces métriques entre le RoI original et celui reconstruit de chaque image des sept séquences (avec IP = 1, 5, 10, 30 et QP = 24).

Nous avons perdu au pire un coefficient pour chaque intra bloc uniquement, mais comme les inter blocs sont prédits à partir des intra blocs, nous avons également perdu certaines informations pour ces blocs. Cependant, selon les résultats du Tableau 8.8, la qualité des images reconstruites utilisant notre approche est proche de celle de H.264/AVC (moins de 4 % de baisse de performance).

[16] url http://www.deepdetect.com/

TABLE 8.8: Impact sur l'efficacité du processus H.264/AVC sur les parties du RoI.

For QP=24 and different IP %	*H.264/AVC* (100 %)	*AsePPI_H.264*	*SNC*	*SNC_IPM*
PSNR	42.78 dB	39 dB (91.16%)	42.78 dB (100 %)	42.78dB (100 %)
SSIM	0.98	0.95 (96.94 %)	0.98 (100 %)	0.98 (100 %)
LSS	0.87	0.86 (98.85 %)	0.87 (100 %)	0.87 (100 %)
ESS	0.86	0.84 (97.67 %)	0.86 (100 %)	0.86 (100 %)
Bit overhead (%)	0	11 %	2.97 %	5 %

Le pourcentage de bits ajoutés par notre processus par rapport au profil de référence (H.264/AVC sans protection de la confidentialité) est de 11 %. En effet, le cryptage DC perd une certaine efficacité dans la quantification, de plus, nous insérons un nouveau coefficient DC.

Selon les résultats, l'impact sur le processus H.264/AVC est négligeable. De plus, nous évaluons les performances que sur le RoI, donc l'impact sur l'ensemble de l'image devrait être encore moins significatif.

5.8 Conclusion

Contrairement aux méthodes existantes, l'application de notre approche *ASePPI* offre le meilleur compromis entre la protection de la vie privée et la préservation de la surveillance visuelle. De plus, nous avons prouvé que notre système est robuste contre les attaques de désanonymisation courantes. *ASePPI* adapte automatiquement son niveau de protection d'identité (c.-à-d., l'effet de pixellisation) pour qu'il soit optimal à toute résolution. De plus, nous évaluons l'impact sur l'efficacité des normes lors de l'intégration de notre processus. Nous concluons que la qualité des vidéos reconstruites est proche de celle d'origine, et que le processus produit un faible pourcentage de bits ajoutés.

Nous pourrions étendre l'évaluation en évaluant le taux de bonne reconnaissance des attributs des personnes (par exemple, le sexe, l'âge, l'origine ethnique). Nous pourrions également évaluer subjectivement l'efficacité de la protection de la vie privée et de l'intelligibilité en effectuant un sondage sur des images protégées par notre méthode. Par exemple, en demandant aux gens, l'identité des personnes célèbres et les événements de la scène en proposant un «choix multiple» pour y répondre.

Bibliography

[1] Saleh Mosaddegh, Loic Simon, and Frédéric Jurie. Photorealistic face de-identification by aggregating donors' face components. In *Computer Vision–ACCV 2014*, pages 159–174. Springer, 2014.

[2] Serdar Çiftçi, Pavel Korshunov, Ahmet O Akyuz, and Touradj Ebrahimi. Using false colors to protect visual privacy of sensitive content. In *IS&T/SPIE Electronic Imaging*, pages 93941L–93941L. International Society for Optics and Photonics, 2015.

[3] Jithendra K Paruchuri, Sen-Ching S Cheung, and Michael W Hail. Video data hiding for managing privacy information in surveillance systems. *EURASIP Journal on Information Security*, 2009:7, 2009.

[4] Ralph Gross, Latanya Sweeney, Jeffrey Cohn, Fernando Torre, and Simon Baker. Face de-identification. *Protecting Privacy in Video Surveillance*, pages 129–146, 2009.

[5] Liang Du, Meng Yi, Erik Blasch, and Haibin Ling. Garp-face: Balancing privacy protection and utility preservation in face de-identification. In *Biometrics (IJCB), 2014 IEEE International Joint Conference on*, pages 1–8. IEEE, 2014.

[6] Lin Yuan, Pavel Korshunov, and Touradj Ebrahimi. Secure jpeg scrambling enabling privacy in photo sharing. In *Automatic Face and Gesture Recognition (FG), 2015 11th IEEE International Conference and Workshops on*, volume 4, pages 1–6. IEEE, 2015.

[7] Frédéric Dufaux and Touradj Ebrahimi. A framework for the validation of privacy protection solutions in video surveillance. In *ICME*, pages 66–71, 2010.

[8] Yongsheng Wang, Fatih Kurugollu, et al. Privacy region protection for h. 264/avc with enhanced scrambling effect and a low bitrate overhead. *Signal Processing: Image Communication*, 35: 71–84, 2015.

[9] N Khlif, T Damak, F Kammoun, and N Masmoudi. Motion vectors signs encryption for h. 264/avc. In *Advanced Technologies for Signal and Image Processing (ATSIP), 2014 1st International Conference on*, pages 1–6. IEEE, 2014.

[10] Iain E Richardson. *H. 264 and MPEG-4 video compression: video coding for next-generation multimedia*. John Wiley & Sons, 2004.

[11] Frederic Dufaux. Video scrambling for privacy protection in video surveillance: recent results and validation framework. In *SPIE Defense, Security, and Sensing*, pages 806302–806302. International Society for Optics and Photonics, 2011.

[12] Humphrey Taylor. Most people are "privacy pragmatists" who, while concerned about privacy, will sometimes trade it off for other benefits. *The Harris Poll*, 17(19):44, 2003.

[13] P Grother and M Ngan. Face recognition vendor test (frvt) performance of face identification algorithms. *NIST Interagency Report*, 8009:2, 2014.

[14] Jose M Such, Agustín Espinosa, and Ana García-Fornes. A survey of privacy in multi-agent systems. *The Knowledge Engineering Review*, 29(3):314–344, 2014.

[15] Antitza Dantcheva, Carmelo Velardo, Angela D'angelo, and Jean-Luc Dugelay. Bag of soft biometrics for person identification. *Multimedia Tools and Applications*, 51(2):739–777, 2011.

[16] Paul Viola and Michael J Jones. Robust real-time face detection. *International journal of computer vision*, 57(2):137–154, 2004.

[17] Navneet Dalal and Bill Triggs. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 886–893. IEEE, 2005.

[18] Andrea Frome, German Cheung, Ahmad Abdulkader, Marco Zennaro, Bo Wu, Alessandro Bissacco, Hartwig Adam, Hartmut Neven, and Luc Vincent. Large-scale privacy protection in google street view. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 2373–2380. IEEE, 2009.

[19] Timothy Gerstner, Doug DeCarlo, Marc Alexa, Adam Finkelstein, Yotam Gingold, and Andrew Nealen. Pixelated image abstraction. In *Proceedings of the Symposium on Non-Photorealistic Animation and Rendering*, pages 29–36. Eurographics Association, 2012.

[20] Tomofumi Koyama, Yuta Nakashima, and Noboru Babaguchi. Real-time privacy protection system for social videos using intentionally-captured persons detection. In *Multimedia and Expo (ICME), 2013 IEEE International Conference on*, pages 1–6. IEEE, 2013.

[21] Karen Lander, Vicki Bruce, and Harry Hill. Evaluating the effectiveness of pixelation and blurring on masking the identity of familiar faces. *Applied Cognitive Psychology*, 15(1):101–116, 2001.

[22] Ce Liu, Heung-Yeung Shum, and William T Freeman. Face hallucination: Theory and practice. *International Journal of Computer Vision*, 75(1):115, 2007.

[23] Ludovico Cavedon, Luca Foschini, and Giovanni Vigna. Getting the face behind the squares: Reconstructing pixelized video streams. In *WOOT*, pages 37–45, 2011.

[24] E. Newton, L. Sweeney, and B. Malin. Preserving privacy by de-identifying face images. *IEEE Trans. on Knowledge and Data Engineering*, 17(2):232–243, February 2005.

[25] Hajer Fradi, Yiqing Yan, and Jean-Luc Dugelay. Privacy protection filter using shape and color cues, 2014.

[26] Stuart Lloyd. Least squares quantization in pcm. *IEEE transactions on information theory*, 28(2): 129–137, 1982.

[27] Pavel Korshunov and Touradj Ebrahimi. Using Face Morphing to Protect Privacy. In *IEEE International Conference on Advanced Video and Signal-Based Surveillance (AVSS)*, pages 208 – 213, August 2013.

[28] Philip J Benson. Morph transformation of the facial image. *Image and Vision Computing*, 12(10): 691 – 696, 1994. ISSN 0262-8856. doi: http://dx.doi.org/10.1016/0262-8856(94)90044-2. URL http://www.sciencedirect.com/science/article/pii/0262885694900442.

[29] Pavel Korshunov and Touradj Ebrahimi. Using warping for privacy protection in video surveillance. In *Digital Signal Processing (DSP), 2013 18th International Conference on*, pages 1–6. IEEE, 2013.

[30] Wei Zhang, Sen-Ching S Cheung, and Minghua Chen. Hiding privacy information in video surveillance system. In *ICIP (3)*, pages 868–871, 2005.

[31] Matusek Florian. *Selective Privacy Protection for Video Surveillance*. PhD thesis, Thesis, 2014.

[32] Maneesh Upmanyu, Anoop M Namboodiri, Kannan Srinathan, and CV Jawahar. Efficient privacy preserving video surveillance. In *Computer Vision, 2009 IEEE 12th International Conference on*, pages 1639–1646. IEEE, 2009.

[33] Ralph Gross, Edoardo Airoldi, Bradley Malin, and Latanya Sweeney. Integrating utility into face de-identification. In *International Workshop on Privacy Enhancing Technologies*, pages 227–242. Springer, 2005.

[34] Latanya Sweeney. k-anonymity: A model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(05):557–570, 2002.

[35] Terence Sim and Li Zhang. Controllable face privacy. In *Automatic Face and Gesture Recognition (FG), 2015 11th IEEE International Conference and Workshops on*, volume 4, pages 1–8. IEEE, 2015.

[36] Himali R Nemade and GD Bonde. Reversible de-identification for lossless image compression using reversible watermarking. *International Journal of Engineering Science*, 2860, 2016.

[37] Benedikt Driessen and Markus Dürmuth. Achieving anonymity against major face recognition algorithms. In *IFIP International Conference on Communications and Multimedia Security*, pages 18–33. Springer, 2013.

[38] Chikahito Nakajima, Massimiliano Pontil, Bernd Heisele, and Tomaso Poggio. Full-body person recognition system. *Pattern recognition*, 36(9):1997–2006, 2003.

[39] Jeffrey E Boyd and James J Little. Biometric gait recognition. In *Advanced Studies in Biometrics*, pages 19–42. Springer, 2005.

[40] Charu C Aggarwal. On k-anonymity and the curse of dimensionality. In *Proceedings of the 31st international conference on Very large data bases*, pages 901–909. VLDB Endowment, 2005.

[41] Charu C Aggarwal and S Yu Philip. A general survey of privacy-preserving data mining models and algorithms. In *Privacy-preserving data mining*, pages 11–52. Springer, 2008.

[42] Ming Yang, Nikolaos Bourbakis, and Shujun Li. Data-image-video encryption. *IEEE potentials*, 23 (3):28–34, 2004.

[43] Amit Pande, Prasant Mohapatra, and Joseph Zambreno. Securing multimedia content using joint compression and encryption. *IEEE MultiMedia*, 20(4):50–61, 2013.

[44] Wenjun Zeng and Shawmin Lei. Efficient frequency domain selective scrambling of digital video. *IEEE Transactions on Multimedia*, 5(1):118–129, 2003.

[45] Shailender Gupta, Ankur Goyal, and Bharat Bhushan. Information hiding using least significant bit steganography and cryptography. *International Journal of Modern Education and Computer Science (IJMECS)*, 4(6):27, 2012.

[46] Ankur Chattopadhyay and Terrance E Boult. Privacycam: a privacy preserving camera using uclinux on the blackfin dsp. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–8. IEEE, 2007.

[47] Terrance Edward Boult. Pico: Privacy through invertible cryptographic obscuration. In *Computer Vision for Interactive and Intelligent Environment, 2005*, pages 27–38. IEEE, 2005.

[48] Hosik Sohn, Wesley De Neve, and Yong Man Ro. Privacy protection in video surveillance systems: analysis of subband-adaptive scrambling in jpeg xr. *Circuits and Systems for Video Technology, IEEE Transactions on*, 21(2):170–177, 2011.

[49] Sk Md Mizanur Rahman, M Anwar Hossain, Hussein Mouftah, Abdulmotaleb El Saddik, and Eiji Okamoto. Chaos-cryptography based privacy preservation technique for video surveillance. *Multimedia systems*, 18(2):145–155, 2012.

[50] Andrea Melle and Jean-Luc Dugelay. Scrambling faces for privacy protection using background self-similarities. In *Image Processing (ICIP), 2014 IEEE International Conference on*, pages 6046–6050. IEEE, 2014.

[51] Feng Dai, Lingling Tong, Yongdong Zhang, and Jintao Li. Restricted h. 264/avc video coding for privacy protected video scrambling. *Journal of Visual Communication and Image Representation*, 22(6):479–490, 2011.

[52] C Narsimha Raju, Ganugula Umadevi, Kannan Srinathan, and CV Jawahar. Fast and secure real-time video encryption. In *Computer Vision, Graphics & Image Processing, 2008. ICVGIP'08. Sixth Indian Conference on*, pages 257–264. IEEE, 2008.

[53] Andreas Unterweger, Kevin Van Ryckegem, Dominik Engel, and Andreas Uhl. Building a post-compression region-of-interest encryption framework for existing video surveillance systems. *Multimedia Systems*, 22(5):617–639, 2016.

[54] Moo-Ryong Ra, Ramesh Govindan, and Antonio Ortega. P3: Toward privacy-preserving photo sharing. In *NSDI*, pages 515–528, 2013.

[55] SimYing Ong, KokSheik Wong, and Kiyoshi Tanaka. Scrambling–embedding for jpeg compressed image. *Signal Processing*, 109:38–53, 2015.

[56] Lingling Tong, Feng Dai, Yongdong Zhang, Jintao Li, and Dongming Zhang. Compressive sensing based video scrambling for privacy protection. In *Visual Communications and Image Processing (VCIP), 2011 IEEE*, pages 1–4. IEEE, 2011.

[57] Frederic Dufaux and Touradj Ebrahimi. Scrambling for privacy protection in video surveillance systems. *Circuits and systems for video technology, IEEE Transactions on*, 18(8):1168–1174, 2008.

[58] Ci Wang, Hong-Bin Yu, and Meng Zheng. A dct-based mpeg-2 transparent scrambling algorithm. *Consumer Electronics, IEEE Transactions on*, 49(4):1208–1213, 2003.

[59] Hosik Sohn, Esla T AnzaKu, Wesley De Neve, Yong Man Ro, and Konstantinos N Plataniotis. Privacy protection in video surveillance systems using scalable video coding. In *Advanced Video and Signal Based Surveillance, 2009. AVSS'09. Sixth IEEE International Conference on*, pages 424–429. IEEE, 2009.

[60] Heinz Hofbauer, Andreas Unterweger, and Andreas Uhl. Encrypting only ac coefficient signs considered harmful. In *Image Processing (ICIP), 2015 IEEE International Conference on*, pages 3740–3744. IEEE, 2015.

[61] Po-Chyi Su, Wei-Yu Chen, Shao-Yu Shiau, Ching-Yu Wu, and Addison YS Su. A privacy protection scheme in h. 264/avc by data hiding. In *Signal and Information Processing Association Annual Summit and Conference (APSIPA), 2013 Asia-Pacific*, pages 1–7. IEEE, 2013.

[62] Fei Peng, Xiao-wen Zhu, and Min Long. An roi privacy protection scheme for h. 264 video based on fmo and chaos. *IEEE transactions on information forensics and security*, 8(10):1688–1699, 2013.

[63] Yi Chang, Rong Yan, Datong Chen, and Jie Yang. People identification with limited labels in privacy-protected video. In *Multimedia and Expo, 2006 IEEE International Conference on*, pages 1005–1008. IEEE, 2006.

[64] Datong Chen, Yi Chang, Rong Yan, and Jie Yang. Protecting personal identification in video. *Protecting Privacy in Video Surveillance*, pages 115–128, 2009.

[65] David H Hubel and Torsten N Wiesel. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of physiology*, 160(1):106, 1962.

[66] Régis Vaillant, Christophe Monrocq, and Yann Le Cun. Original approach for the localisation of objects in images. *IEE Proceedings-Vision, Image and Signal Processing*, 141(4):245–250, 1994.

[67] Steve Lawrence, C Lee Giles, Ah Chung Tsoi, and Andrew D Back. Face recognition: A convolutional neural-network approach. *Neural Networks, IEEE Transactions on*, 8(1):98–113, 1997.

[68] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012.

[69] P. Korshunov, H. Nemoto, A. Skodras, and T. Ebrahimi. Crowdsourcing-based evaluation of privacy in HDR images. In *SPIE Photonics Europe 2014, Optics, Photonics and Digital Technologies for Multimedia Applications*, Brussels, Belgium, April 2014.

[70] P. Korshunov, S. Cai, and T. Ebrahimi. Crowdsourcing approach for evaluation of privacy filters in video surveillance. In *Proceedings of the ACM Multimedia 2012 Workshop on Crowdsourcing for Multimedia*, CrowdMM'12, pages 35–40, Nara, Japan, October 2012. ISBN 978-1-4503-1589-0. doi: $10.1145/2390803.2390817$. URL http://doi.acm.org/10.1145/2390803.2390817.

[71] Christian Keimel, Julian Habigt, Clemens Horch, and Klaus Diepold. Qualitycrowd a framework for crowd-based quality evaluation. In *Picture Coding Symposium (PCS), 2012*, pages 245–248. IEEE, 2012.

[72] Ádám Erdélyi, Thomas Winkler, and Bernhard Rinner. Privacy protection vs. utility in visual data. *Multimedia Tools and Applications*, pages 1–28, 2017.

[73] P. Korshunov and W. T. Ooi. Video quality for face detection, recognition, and tracking. *ACM Trans. Multimedia Comput. Commun. Appl.*, 7(3):14:1–14:21, September 2011. ISSN 1551-6857. doi: $10.1145/2000486.2000488$. URL http://doi.acm.org/10.1145/2000486.2000488.

[74] F. Dufaux and T. Ebrahimi. A framework for the validation of privacy protection solutions in video surveillance. In *Proceedings of IEEE International Conference on Multimedia & Expo (ICME 2010)*, Singapore, July 2010.

[75] Timo Ahonen, Abdenour Hadid, and Matti Pietikainen. Face description with local binary patterns: Application to face recognition. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 28(12):2037–2041, 2006.

[76] Matthew A Turk and Alex P Pentland. Face recognition using eigenfaces. In *Computer Vision and Pattern Recognition, 1991. Proceedings CVPR'91., IEEE Computer Society Conference on*, pages 586–591. IEEE, 1991.

[77] Oscar Déniz, Gloria Bueno, Jesús Salido, and Fernando De la Torre. Face recognition using histograms of oriented gradients. *Pattern Recognition Letters*, 32(12):1598–1603, 2011.

[78] Pedro F Felzenszwalb, Ross B Girshick, David McAllester, and Deva Ramanan. Object detection with discriminatively trained part-based models. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 32(9):1627–1645, 2010.

[79] Pavel Korshunov, Andrea Melle, Jean-Luc Dugelay, and Touradj Ebrahimi. Framework for objective evaluation of privacy filters. In *SPIE Optical Engineering+ Applications*, pages 88560T–88560T. International Society for Optics and Photonics, 2013.

[80] Hua Huang and Huiting He. Super-resolution method for face recognition using nonlinear mappings on coherent features. *Neural Networks, IEEE Transactions on*, 22(1):121–130, 2011.

[81] P Jonathon Phillips, Hyeonjoon Moon, Syed A Rizvi, and Patrick J Rauss. The feret evaluation methodology for face-recognition algorithms. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(10):1090–1104, 2000.

[82] Ferdinando S Samaria and Andy C Harter. Parameterisation of a stochastic model for human face identification. In *Applications of Computer Vision, 1994., Proceedings of the Second IEEE Workshop on*, pages 138–142. IEEE, 1994.

[83] Mislav Grgic, Kresimir Delac, and Sonja Grgic. Scface–surveillance cameras face database. *Multimedia tools and applications*, 51(3):863–879, 2011.

[84] Brandon Amos, Bartosz Ludwiczuk, and Mahadev Satyanarayanan. Openface: A general-purpose face recognition library with mobile applications. Technical report, CMU-CS-16-118, CMU School of Computer Science, 2016.

[85] Hong-Wei Ng and Stefan Winkler. A data-driven approach to cleaning large face datasets. In *Image Processing (ICIP), 2014 IEEE International Conference on*, pages 343–347. IEEE, 2014.

[86] Dong Yi, Zhen Lei, Shengcai Liao, and Stan Z Li. Learning face representation from scratch. *arXiv preprint arXiv:1411.7923*, 2014.

[87] Gary B Huang and Erik Learned-Miller. Labeled faces in the wild: Updates and new reporting procedures. *Dept. Comput. Sci., Univ. Massachusetts Amherst, Amherst, MA, USA, Tech. Rep*, pages 14–003, 2014.

[88] Itay Maoz. *Face Recognition in Unconstrained Videos with Matched Background Similarity*. Tel Aviv University, 2012.

[89] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 815–823, 2015.

[90] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

[91] Sen Jia and Nello Cristianini. Learning to classify gender from four million images. *Pattern Recognition Letters*, 58:35–41, 2015.

[92] Grigory Antipov, Sid-Ahmed Berrani, Natacha Ruchaud, and Jean-Luc Dugelay. Learned vs. hand-crafted features for pedestrian gender recognition. In *Proceedings of the ACM International Conference on Multimedia*. ACM, 2015.

[93] Yubin Deng, Ping Luo, Chen Change Loy, and Xiaoou Tang. Pedestrian attribute recognition at far distance. In *Proceedings of the ACM International Conference on Multimedia*, pages 789–792. ACM, 2014.

[94] Jamie D Shutler, Michael G Grant, Mark S Nixon, and John N Carter. On a large sequence-based human gait database. In *Applications and Science in Soft Computing*, pages 339–346. Springer, 2004.

[95] Khurram Soomro and Amir R Zamir. Action recognition in realistic sports videos. In *Computer Vision in Sports*, pages 181–208. Springer, 2014.

[96] Yinian Mao and Min Wu. A joint signal processing and cryptographic approach to multimedia encryption. *IEEE Transactions on Image Processing*, 15(7):2061–2075, 2006.

[97] Zhou Wang, Alan C Bovik, Hamid R Sheikh, and Eero P Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

[98] Richard McPherson, Reza Shokri, and Vitaly Shmatikov. Defeating image obfuscation with deep learning. *arXiv preprint arXiv:1609.00408*, 2016.

[99] Tobias Hossfeld, Christian Keimel, Matthias Hirth, Bruno Gardlo, Julian Habigt, Klaus Diepold, and Phuoc Tran-Gia. Best practices for QoE crowdtesting: QoE assessment with crowdsourcing. *IEEE Transactions on Multimedia*, PP(99):1–1, 2013. ISSN 1520-9210. doi: $10.1109/\mathrm{TMM}.2013.2291663$.

[100] Recommendation ITU-R BT.500-13. *Methodology for the subjective assessment of the quality of television pictures*. International Telecommunication Union, Geneva, Switzerland, 2012.

[101] Mohamad Forouzanfar. This work has been published by scitech publishing in" principles of waveform diversity and design" book available at http://www. scitechpub. com/wdd/. please cite this book chapter as follows: M. forouzanfar and h. abrishami-moghaddam, ultrasound speckle reduction in the complex wavelet domain, in principles of waveform diversity and design, m. wicks, e. mokole, s. blunt, r. schneible, and v.

[102] Qi Shan, Jiaya Jia, and Aseem Agarwala. High-quality motion deblurring from a single image. In *ACM Transactions on Graphics (TOG)*, volume 27, page 73. ACM, 2008.

[103] Pankaj Hedaoo and Swati S Godbole. Wavelet thresholding approach for image denoising. *International Journal of Network Security & Its Applications*, 3(4):16–21, 2011.

[104] Jae S Lim. Two-dimensional signal and image processing. *Englewood Cliffs, NJ, Prentice Hall, 1990, 710 p.*, 1, 1990.

[105] Sachin D Ruikar and Dharmpal D Doye. Wavelet based image denoising technique. *IJACSA) International Journal of Advanced Computer Science and Applications*, 2(3), 2011.

[106] Robert Keys. Cubic convolution interpolation for digital image processing. *Acoustics, Speech and Signal Processing, IEEE Transactions on*, 29(6):1153–1160, 1981.

[107] Sung Cheol Park, Min Kyu Park, and Moon Gi Kang. Super-resolution image reconstruction: a technical overview. *Signal Processing Magazine, IEEE*, 20(3):21–36, 2003.

[108] Bernard Chalmond. Psf estimation for image deblurring. *CVGIP: Graphical Models and Image Processing*, 53(4):364–372, 1991.

[109] Weisheng Dong, Lei Zhang, Guangming Shi, and Xiaolin Wu. Image deblurring and super-resolution by adaptive sparse domain selection and adaptive regularization. *Image Processing, IEEE Transactions on*, 20(7):1838–1857, 2011.

[110] Jan Kotera, Filip Šroubek, and Peyman Milanfar. Blind deconvolution using alternating maximum a posteriori estimation with heavy-tailed priors. In *Computer Analysis of Images and Patterns*, pages 59–66. Springer, 2013.

[111] Markus A Mayer, Anja Borsdorf, Martin Wagner, Joachim Hornegger, Christian Y Mardin, and Ralf P Tornow. Wavelet denoising of multiframe optical coherence tomography data. *Biomedical optics express*, 3(3):572–589, 2012.

[112] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Image super-resolution using deep convolutional networks. *IEEE transactions on pattern analysis and machine intelligence*, 38 (2):295–307, 2016.

[113] Christian J Schuler, Michael Hirsch, Stefan Harmeling, and Bernhard Schölkopf. Learning to deblur. *IEEE transactions on pattern analysis and machine intelligence*, 38(7):1439–1451, 2016.

[114] Xiaojiao Mao, Chunhua Shen, and Yu-Bin Yang. Image restoration using very deep convolutional encoder-decoder networks with symmetric skip connections. In *Advances in Neural Information Processing Systems*, pages 2802–2810, 2016.

[115] 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. URL http://www.multimediaeval.org/mediaeval2015/droneprotect2015/.

[116] Atta Badii, Pavel Korshunov, Hamid Oudi, Touradj Ebrahimi, Tomas Piatrik, Volker Eiselein, Natacha Ruchaud, Christian Fedorczak, Jean-Luc Dugelay, and Diego Fernandez Vazquez. Overview of the mediaeval 2015 drone protect task. In *MediaEval*, 2015.

[117] Margherita Bonetto, Pavel Korshunov, Giovanni Ramponi, and Touradj Ebrahimi. Privacy in mini-drone based video surveillance. In *Workshop on De-identification for privacy protection in multimedia*, number EPFL-CONF-206109, 2015.

[118] Pavel Korshunov and Touradj Ebrahimi. Pevid: privacy evaluation video dataset. In *Proceedings of SPIE Volume 8856*, volume 8856. Spie-Int Soc Optical Engineering, 2013.

[119] Zoran Zivkovic. Improved adaptive gaussian mixture model for background subtraction. In *Pattern Recognition. ICPR 2004. Proceedings of the 17th International Conference on*, 2004.

[120] Carsten Rother, Vladimir Kolmogorov, and Andrew Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. In *ACM transactions on graphics (TOG)*, volume 23, pages 309–314. ACM, 2004.

[121] Yuri Y Boykov and Marie-Pierre Jolly. Interactive graph cuts for optimal boundary & region segmentation of objects in nd images. In *Computer Vision, 2001. ICCV 2001. Proceedings. Eighth IEEE International Conference on*, volume 1, pages 105–112. IEEE, 2001.

[122] Edward R Dougherty. An introduction to morphological image processing. *Tutorial texts in optical engineering*, 1992.

[123] Djalil Chafaï and Florent Malrieu. Permutations, partitions, et graphes. In *Recueil de Modèles Aléatoires*, pages 57–68. Springer, 2016.

[124] Erik Learned-Miller, Gary B Huang, Aruni RoyChowdhury, Haoxiang Li, and Gang Hua. Labeled faces in the wild: A survey. In *Advances in face detection and facial image analysis*, pages 189–248. Springer, 2016.

[125] Ralph Gross, Latanya Sweeney, Fernando De La Torre, and Simon Baker. Semi-supervised learning of multi-factor models for face de-identification. In *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, pages 1–8. IEEE, 2008.

[126] Lior Wolf, Tal Hassner, and Itay Maoz. Face recognition in unconstrained videos with matched background similarity. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 529–534. IEEE, 2011.

[127] SuGil Choi, Jong-Wook Han, and Hyunsook Cho. Privacy-preserving h. 264 video encryption scheme. *ETRI Journal*, 33(6):935–944, 2011.

[128] Kirsten Bock. Europrise trust certification. *Datenschutz und Datensicherheit-DuD*, 32(9):610–614, 2008.

[129] Nasir Ahmed, T_ Natarajan, and Kamisetty R Rao. Discrete cosine transform. *IEEE transactions on Computers*, 100(1):90–93, 1974.