Future Generation Computer Systems 96 (2019) 23–31

Contents lists available at ScienceDirect

Future Generation Computer Systems

journal homepage: www.elsevier.com/locate/fgcs

DC coefficient recovery for JPEG images in ubiquitous communication systems

Han Qiu [a], Gerard Memmi [a], Xuan Chen [b,*], Jian Xiong [c]

[a] Telecom-ParisTech, Paris, 75013, France
[b] Sorbonne University, Paris, 75005, France
[c] Shanghai Jiao Tong University, Shanghai, 200433, China

HIGHLIGHTS

- A DC coefficients recovery method is proposed for JPEG images.
- This paper improves previous work by giving a reliable math definition.
- The proposed method can improve error resistance for JPEG transmission.
- This method can be a potential compression candidate for lossy image transmissions.

ARTICLE INFO

Article history:
Received 23 November 2018
Received in revised form 7 January 2019
Accepted 15 January 2019
Available online 4 February 2019

Keywords:
Multimedia data transmission
Error recovery
DCT
JPEG
Fault tolerance

ABSTRACT

With the development of data based technology, data transmission methods for ubiquitous computing and communication systems are more and more needed. There are various kinds of data processing techniques such as compression or coding to improve the transmission efficiency for heterogeneous networks with different requirements. However, the methods for improving fault tolerance in ubiquitous communication systems are still lacked especially for the multimedia data driven applications at the execution ends. In this paper, we propose an image content recovery method for JPEG images that can recover the image content by estimating the DC coefficients without any pre-know knowledge. This method can also be used to transmit the rough image content by reducing the data amount needed to be transmitted. Thus, fault tolerance can be achieved at the receivers' ends for ubiquitous communication systems. The result analysis with different images compared with previous works proved the effectiveness of our method.

1. Introduction

During the last decades, the data generation and transmission have been evolving and increasing significantly. As one of the developments, hybrid and heterogeneous network architectures have enabled a variety of cloud-based infrastructure solutions. As pointed in [1], pervasive and ubiquitous systems are widely deployed and power consumption and latency are becoming issues for data sharing in some ubiquitous systems such as mobile cloud systems.

Nowadays, for the ubiquitous communication systems, images and videos are the most commonly transmitted digital data including social media, healthcare applications [2], cloud based data storage systems [3]. Except transmission efficiency, there are many other issues that need to be considered in such ubiquitous communication systems such as security [4] and fault tolerance. Although there are various kinds of multimedia processing techniques for improving the efficiency of data transmission at the senders' end, the techniques based on a viewpoint from a receiver's end is still lacked. For instance, compression technologies [5] before outsourcing are commonly used to reduce the data amount of transmission. In particular, efficient image/video transmission are realized by combining the transform, quantization, and coding technologies to achieve the compression purpose such as JPEG [6] and MPEG-4 [7] standards. However, there is also a need for ubiquitous communication systems to improve the data transmission success ratio from a receiver's end [8]. One of the problems is to improve the error tolerance by recovering the data at the receiver's end [9].

Although there are many different techniques to resist errors by recovery at source coding or channel coding ends, error resistance still needs to be considered at the data content level. One example would be the possible loss for the JPEG images transmitted over

* Corresponding author.
E-mail addresses: han.qiu@telecom-paristech.fr (H. Qiu), gerard.memmi@telecom-paristech.fr (G. Memmi), xuan.chen@etu.sorbonne-universite.fr (X. Chen), xjarrow@sjtu.edu.cn (J. Xiong).

https://doi.org/10.1016/j.future.2019.01.037
0167-739X/

the Wireless Sensor Network (WSN) [10]. The WSN is a kind of network with a low data transmission bandwidth but a high data error rate [11]. There are many tiny sensor nodes equipped with power limited batteries [9] which cannot meet the needs for error recovery. As the JPEG images are highly reduced and the data redundancy has been removed, the transmission error will lead to visual degrading for the recovered image contents or to totally unrecoverable results. Thus, transmission error will lead to the repeated image transmission which will further lead to energy wasting. The worst case is that the next incoming image is still suffering from the data error or loss. Thus, smart image processing techniques to recover the image from the loss on the receivers' end can effectively improve the error resistance capacity for the WSNs which will further optimize the energy consumption [1].

Most of today's multimedia data are compressed based on the transform theory. Discrete Cosine Transform (DCT) [12] is the most widely used transform in multimedia coding, which covers image and video coding standards such as JPEG, MPEG-1/2, MPEG-4, AVC/H.264 [13], and the more recent HEVC [14]. DCT is a Fourier-like transform, which was first proposed by [12]. The purpose of DCT is to perform de-correlation of the input signal and to present the output in the frequency domain just like other transformation algorithms. Compared with the Fourier Transform, which represents a signal as a combination of sines and cosines, DCT performs only the cosine-series expansion. DCT, as an orthogonal transform, can compact most energy of a highly-correlated discrete signal into a few coefficients. Then the basic methodology is to determine the importance levels of the DCT coefficients in the frequency domain and to use the quantization step to compress these coefficients.

Among all DCT coefficients, the first (i.e., the one with the lowest frequency) coefficient is called the DC coefficient while the rest are called as AC coefficients. Each coefficient carries distinct information of the transformed signal, although the DC coefficient is considered the most important one because it carries the average intensity of the transformed signal. Most multimedia coding standards working in DCT domain apply the DCT transform to smaller blocks sequentially to reduce the overall time complexity of the transformation. As a result, the assembling of DC coefficients from all transformed blocks resembles the original signal, but at a lower resolution.

The previous researches on recovering the DCT coefficients are mostly used to improve the image/video quality by recovering the high frequency coefficients (less important ones) at the execution ends. This could help to provide better image/video quality which can improve the efficiency and accuracy of multimedia based data mining [15]. However, on the other hand, recovering the low frequency coefficients especially DC coefficients from the high frequency coefficients are always ignored. For some special network environment such as networks with a lossy transmission, error on low frequency coefficients especially the DC coefficients will lead to totally unrecognizable image/video at the execution ends. Therefore, any multimedia based data mining will be affected as the received data suffer from data loss.

In our previous work [16], we have shown that recovering low frequency coefficients based on high frequency coefficients is possible in DCT based use cases such as bitmap images which are processed without quantization step. This is not practical to be used in a ubiquitous communication system as most images are compressed before transmission. Thus, in order to help the execution ends to improve fault tolerance, DCT coefficient recovery techniques for compressed images such as JPEG are necessary to be developed. In this paper, we pick JPEG images as an example to illustrate our recovery method. As DC coefficients are more important and recovery based on DC coefficients are already shown in previous works [17]. We show how to guess the DC coefficients from the rest AC coefficients. We believe our proposal can help to improve the fault tolerance at the receivers' ends in a ubiquitous communication system for JPEG image transmissions.

Our contribution in this paper includes: We develop an optimized method to recover the JPEG image by recovering the DC coefficients from the remaining AC coefficients. This method can be then further used for any DCT based multimedia data for resisting transmission errors at the execution ends which will further help the efficient multimedia data processing.

The roadmap for this paper is as follows: in Section 2, some related research background knowledge is presented and discussed while existing problems are introduced; in Section 3, our method is introduced with a brief example and some details; in Section 4, we analyze the recovery results with more tests and some corner cases; in Section 5, we discuss the problems that need to be further solved; in Section 6 we conclude our work.

2. Research backgrounds

2.1. JPEG compression

JPEG is the most commonly used method of lossy compression for digital images. JPEG image is normally generated following the method shown in Fig. 1. Firstly, a transform which is DCT is deployed on initial images. This mathematical transform converts each data source from the spatial (2D) domain into the frequency domain (a.k.a. transform domain).

DCT has different types shown in [18]. The most popular DCT algorithm is a two-dimensional symmetric variation of the transform that operates on 8×8 blocks (DCT 8×8) and its inverse (iDCT 8×8). This DCT 8×8 is utilized in JPEG compression routines [19] and has become an important standard in image and video transformation algorithms defined as follows:

$$C(u, v) = \alpha(u)\alpha(v) \sum_{x=0}^{7} \sum_{y=0}^{7} f(x, y) \cos\left[\frac{\pi(2x+1)u}{16}\right] \times \cos\left[\frac{\pi(2y+1)v}{16}\right] \tag{1}$$

The inverse of two-dimensional DCT 8×8 is defined as:

$$f(x, y) = \sum_{u=0}^{7} \sum_{v=0}^{7} \alpha(u)\alpha(v) C(u, v) \cos\left[\frac{\pi(2x+1)u}{16}\right] \times \cos\left[\frac{\pi(2y+1)v}{16}\right] \tag{2}$$

where

$$\alpha(u) = \begin{cases} \sqrt{\frac{1}{8}}, u = 0 \\ \frac{1}{2}, u \neq 0 \end{cases} \tag{3}$$

As can be seen from Eq. (1), especially, in the forward DCT 8×8, the substitution of $u, v = 0$ yields:

$$C(0, 0) = \alpha(0)\alpha(0) \sum_{0}^{7} \sum_{0}^{7} f(x, y) \tag{4}$$

which is eight times the mean of 8×8 sample. In fact, this value is called the DC coefficient of the transform results and the others are called the AC coefficients which are independent of the average value. Normally in the image compression case, the DC coefficient is relatively large in magnitude while the AC terms become lower in magnitude as they move farther from the DC coefficient. This means that by performing the DCT 8×8 on the input raw image, the representation of the image (the main elements carried by an image) is concentrated in the upper left coefficients of each of the output 8×8 matrix (i.e. low frequency area), while the lower

Fig. 1. A general process of how multimedia content is generated.

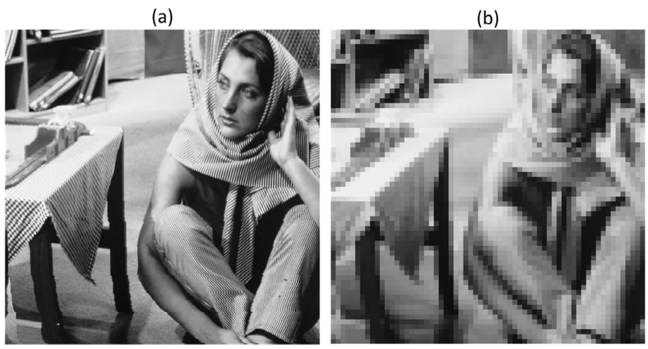

Fig. 2. (a) original bitmap image; (b) iDCT results while keeping only DC coefficients for each 8×8 block.

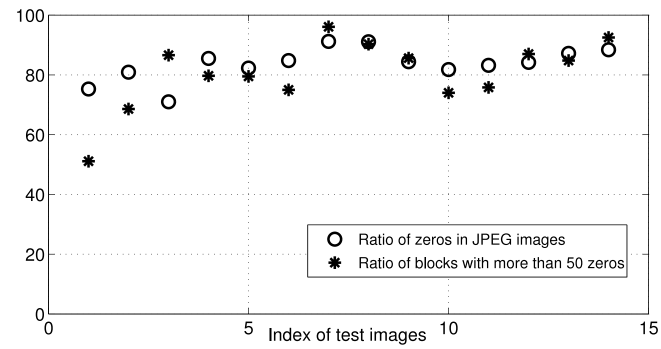

Fig. 3. Ratio of zero values in JPEG images and ratio of blocks that have more than 50 zero values.

right coefficients of the output matrix contains less important information like details (high frequency area).

The next step is called quantization which is to reduce the information in the frequency domain. The basic rule is that for more important coefficients, the lighter compression will be performed while heavier compression will be done for the less important coefficients. In simpler terms, quantization is a method for optimally reducing a large number scale (with different occurrences of each number) into a smaller one, and the transform domain is a convenient representation of the image because the high frequency coefficients, which contribute less to the overall picture than other coefficients, are characteristically small-values with high compressibility. The quantized coefficients are then sequenced and losslessly packed into the output bitstream which is called the coding step.

In fact, the JPEG compression will turn most of the DCT coefficients of the initial images become zeros which will lead to an unrecoverable loss. The quantization table Q_{50} [20] is used as the default to reduce the coefficient values according to the distribution of the frequency. Coefficients distributed in the higher frequency area will be divided with large values which lead to more zeros in high frequency areas than low frequency areas.

2.2. DC coefficient

The results of a DCT 8×8 transform are 1 DC coefficient and 63 AC coefficients. The DC coefficient represents the average color of the 8×8 region while the 63 AC coefficients represent color change across the block.

Low frequency coefficients (AC coefficients near to DC coefficient) represent the basic elements of one image. If most of the AC coefficients are zero values, in the time domain, the color changing trends in this specific block are not obvious. However, in the contrary, if there are many AC coefficients that are not zero values, the color of this block in spatial domain will be changing dramatically. In Fig. 2, we remove all AC coefficients while keeping only the DC coefficients for each of the 8×8 blocks in the frequency domain and show the visual results for the spatial domain.

Only DC coefficients reconstruct a decimated (and low-pass filtered) version of the original image. From a visual result shown in Fig. 2, as we can see, the DC-only spatial domain could reflect the basic element of one image and some details are also given. Thus many previous works are based on this theory to protect the image by protecting the low frequency coefficients [21]. Most previous works do not point out the exact image format of the research which is significant. For instance, the DC coefficient is less important for bitmap reconstruction than for JPEG reconstruction. As in [22], some of the AC coefficients in low frequency area are also protected. However, images contents can be reconstructed from the high frequency coefficients in a rough manner [16]. In this paper, for the JPEG case, most of the AC coefficients are quantized and such unrecoverable loss will make all the remaining coefficients very vital for image reconstruction.

2.3. Problem description

Our previous work [16] for recovering the missing low frequency coefficients will be difficult to implement for JPEG images. As pointed in [16,23], the recovery is based on most of the nonzero AC coefficients. In a bitmap case, even more than half of the low frequency coefficients are missing, a rough recovery for the image content is still possible. For some extreme corner cases when images have much energy in high frequency area, the recovery can even be done just based on a few highest frequency coefficients. However, this is not the case dealing with in this paper.

For the quantized DCT coefficients, most DCT coefficients are turned into zeros. In Fig. 3, we listed the ratio of zero DCT coefficients after JPEG compression in red points. Most of the DCT coefficients are compressed to zeros due to the quantization step. Also, as the JPEG standard is deployed by blocks of size 8×8, there are many blocks with most coefficients inside are zeros. As listed as the blue points in Fig. 3, the ratio of blocks with more than 50 zero coefficients are mostly around 80%.

However, if more than one AC coefficient is removed from the low frequency area, more than half of the 8×8 DCT blocks are all zeros which are hard to be recovered. Thus, the problem description is that the DC coefficients for each of the 8×8 blocks are the expected output values while the rest AC coefficients remain without a loss. Moreover, some further cases where some ACs are also missing are discussed in Section 4.1.

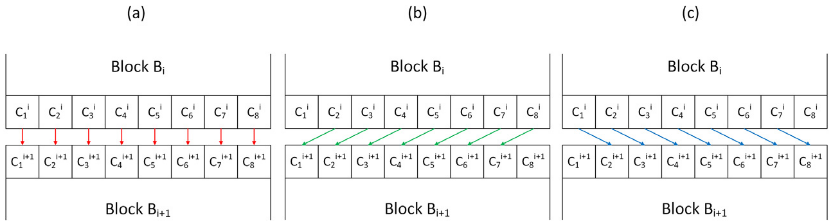

Fig. 4. Three directions to calculate the MSE between pixel pairs of neighbor blocks.

3. DCT coefficients recovery

In this section, an improved method for recovering image content by guessing DC values for each 8 × 8 block is given. We are aiming to illustrate that image content including basic elements and details can be effectively recovered by estimating the DC coefficients for each of the 8 × 8 block.

3.1. Basic property

As pointed in [24], there is an important property used in DCT coefficients recovery: the difference between two neighboring pixels is a Laplacian variate with zero mean and small variance. Basically, the authors in [24] considered there must be the smoothest passing between two neighbor blocks. The corresponding pixel pairs of the two neighbor blocks have three possible directions as shown in Fig. 4 and the mean-square error (MSE) [25] is used to test the variance between two neighbor blocks.

The MSE shown in Fig. 4 is calculated as in Eqs. (5)–(7):

$$MSE_a(B_i, B_{i+1}) = \frac{1}{8}\sum_{j=1}^{8}(c_j^i - c_j^{i+1})^2 \tag{5}$$

$$MSE_b(B_i, B_{i+1}) = \frac{1}{7}\sum_{j=1}^{7}(c_{j+1}^i - c_j^{i+1})^2 \tag{6}$$

$$MSE_c(B_i, B_{i+1}) = \frac{1}{7}\sum_{j=1}^{7}(c_j^i - c_{j+1}^{i+1})^2 \tag{7}$$

As mentioned before, the DC values for each 8 × 8 blocks are the average values of the pixel values. This leads to the fact that if the DC values of neighbor blocks are similar and they are removed, the mean value of the pixels in the spatial domain is removed. However, the grayscale gradient which is calculated with MSE stays similar.

In order to estimate the DC coefficient according to the known DC coefficient of the neighbor block, this property can be further described as a more specific question. In this paper, as long as we are working on the recovery of DC coefficients in a specific format of JPEG, the DC values in all 8 × 8 blocks are all integers within a range: $DC \in [-64, +64]$. Then the question of recovering unknown DC values can be expressed as follows.

Assume DC_i is the DC value for block B_i which is known and DC_{i+1} is the DC value for block B_{i+1} which is unknown. The other AC coefficients in block B_i and block B_{i+1} are all known. The question then can be expressed as:

$$\begin{aligned}
&For\ \forall x \in [-64, +64]\\
&Find\ x\ for\ the\ min\{MSE\}\\
&which\ MSE = min\{MSE_a,\ MSE_b,\ MSE_c\}\\
&let\ DC_{i+1} = x
\end{aligned} \tag{8}$$

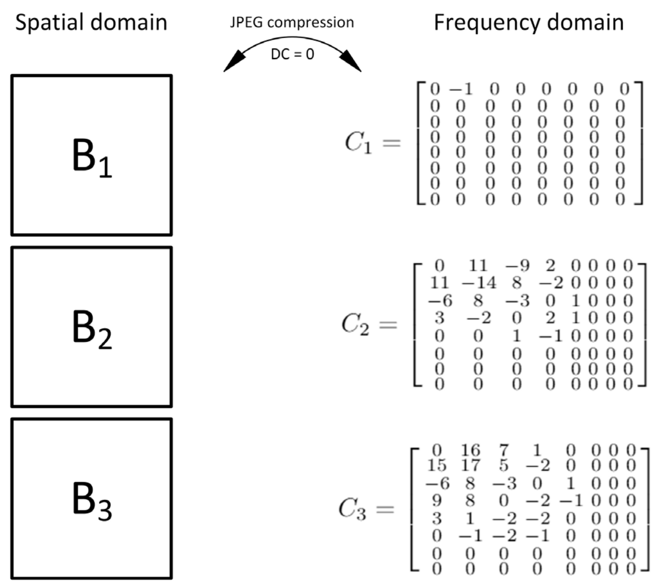

Fig. 5. Blocks in spatial domain and in frequency domain.

3.2. One example

In this subsection, one example will be given for illustrating the method for guessing the coefficients based on the grayscale changing trends. As shown in expression (8), the way to estimate the missing DC coefficients is to use brute-force way to calculate all possibilities of DC values in a known range. There are three 8 × 8 blocks picked from the image Fig. 8(a) as B_1 is on top of B_2 and B_2 is on top of B_3. The C_i are the DCT coefficients after JPEG compression (DCT, JPEG quantization).

Firstly, assume the DC coefficient D_1 for C_1 is known as 24, the D_2 for C_2 is the DC coefficient we want to estimate. Once an estimated value is calculated as D_2', we estimate D_3 based on D_2'. In order to estimate D_2, B_1 is firstly calculated with the reverse steps of JPEG compression: first put the known DC coefficient D_1 at $C_1(1, 1)$, then reverse the quantization step to get the DCT coefficients, then the reverse DCT is calculated to recover the B_1 in spatial domain. The B_2 is also needed for the calculation of the MSE in Eq. (8). As D_2 is known, all possible values belongs to $[-64, +64]$ are tried as D_2 to calculate the B_2 which will be further used to calculate the MSE according to Eqs. (5)–(7).

The distribution of MSE_a, MSE_b, and MSE_c for estimating D_2 is shown in Fig. 6(a). It can be observed that the MSE of the three directions shown in Fig. 4 are all have a min value nearly to zero when the estimated value of D_2 is 18. Thus, $D_2' = 18$ and this value is used to estimate the value D_3' with the same method. The distribution of MSE_a, MSE_b, and MSE_c for estimating D_3 is shown in Fig. 6(b). In this case, the MSE_b which is calculated based on the direction in Fig. 4(b) has the min value at −43. Thus, the estimated $D_3' = -43$.

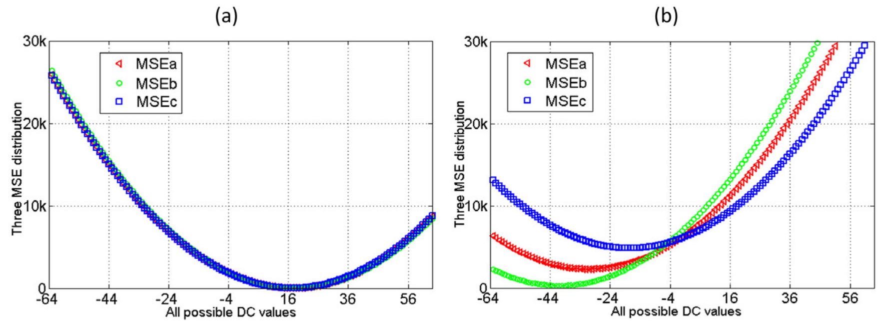

Fig. 6. When DC value in D_2 is known: (a) Distribution of the three MSEs for Block B2; (b) Distribution of the three MSEs for Block B3.

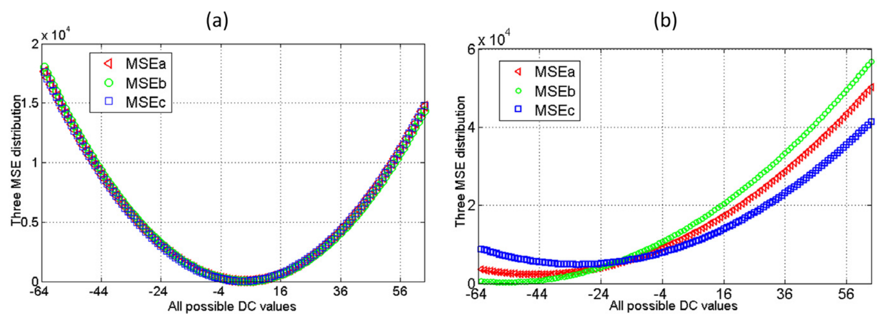

Fig. 7. When DC value in D_2 is assumed as 10: (a) Distribution of the three MSEs for Block B2; (b) Distribution of the three MSEs for Block B3.

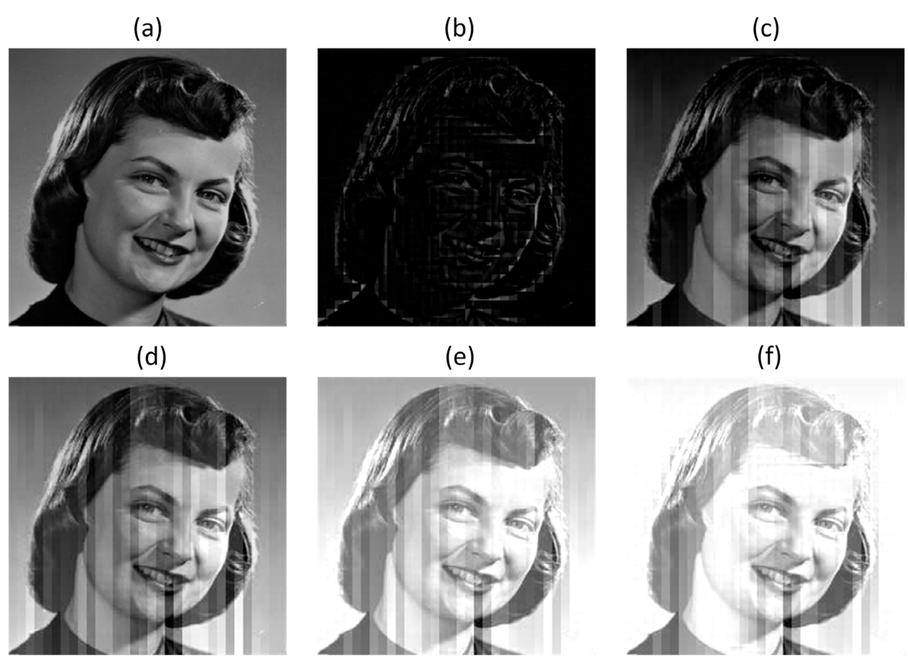

Fig. 8. (a) original JPEG image; (b) iDCT results without DC coefficients; (c) Recovered image scan from up to down with all DC coefficients in first row as −60; (d) Recovered image scan from up to down with all DC coefficients in first row as −20; (e) Recovered image scan from up to down with all DC coefficients in first row as 20; (f) Recovered image scan from up to down with all DC coefficients in first row as 60.

In fact, the original DC coefficients in B_1, B_2, and B_3 are $D_1 = 24$, $D_2 = 18$, and $D_3 = -47$ respectively. Thus the estimation of D_2 is correct without error while the estimation of D_3 is similar but with error. One further step is to test the estimation of DCs without the correct D_1. This means there is no pre-known DC coefficients for all blocks. In this case, we use the same blocks as in Fig. 5 but the assumed $D_1 = 10$. The same method is used and the distribution of the MSE_a, MSE_b, and MSE_c for estimating D_2 and D_3 is shown in Fig. 7(a) and (b) respectively. The estimated results are: $D_2' = 4$ and $D_3' = -56$. Thus the grayscale changing trends for the neighbor blocks stay the same. This is due to the reason that the DC coefficients are just the mean value of grayscale values, the rest AC values will also determine the grayscale variation between neighbor blocks.

Table 1
Major notations used in this algorithm and their definitions.

Notation	Definition
$B_m(i, j)$	8×8 Block $B_m(i, j)$ with location (i, j)
$C_m(i, j)$	64 DCT coefficients of block $B_m(i, j)$
$A_m(i, j)$	63 AC coefficients of block $B_m(i, j)$
$D_m(i, j)$	DC coefficient of block $B_m(i, j)$
$iDCT$	Reverse Discrete Cosine Transform
Q_{50}	Quantization table in JPEG
$NumL$	Number of 8×8 blocks in vertical direction
$NumW$	Number of 8×8 blocks in horizontal direction
MSE	MSE for two neighbor blocks: $\{MSE_a, MSE_b, MSE_c\}$

3.3. Recovery for images

In this subsection, the algorithm to estimate DC coefficients based on calculating the distribution of MSEs will be presented and tested for different images. For the image case, the neighbor 8×8 blocks are not just correlated in the vertical direction but also in the horizontal direction. The MSEs will be calculated also with the direction from left to right to estimate the DC values.The setting is shown in Table 1 for the parameters are used for the algorithms.

So the algorithm for one image without any knowledge of DC coefficients will start with a guess DC value. In normal JPEG image case, the four lines at the edge are normally in very similar grayscale. Thus, for the estimation without any pre-knowledge of DC values, the first step is to assume a value for the DC coefficient at the block of the four corners of this image. This assumed value is also within the range of a DC coefficient which is an integer between $[-64, +64]$. As shown in the example in Figs. 6 and 7, the value of known DC will not affect the difference between the two DCs in neighbor blocks. Thus, in this experimentation, we assume this four DC coefficients as total seven possibilities: $\{-60, -40, -20, 0, 20, 40, 60\}$. There will be seven times calculations for each of the values and the results will be listed to illustrate changing this starting point will not affect recovering the image content but only affect the brightness of the images.

ALGORITHM 1: Estimation of DC coefficients from up to down.

Input: AC coefficient blocks $A_m(i, j)$ with $i, j \in$ {NumL,NumW}
Output: Recovered DC coefficients $D_m(i, j)$

for $i \leftarrow 1$ to $NumW$ **do**
for $j \leftarrow 1$ to $NumL$ **do**
$C_m(i, j) = A_m(i, j) + D_m(i, j)$
$B_m(i, j) = iDCT(C_m(i, j). * Q_{50})$ /*Calculate the spatial domain $B_m(i, j)$.*/
for $x \leftarrow -64$ to $+64$ **do**
/*Calculate the MSE values for all possible DC values.*/
$C_m(i, j+1) = A_m(i, j+1) + x$
$B_m(i, j+1) = iDCT(C_m(i, j+1). * Q_{50})$
$f_{MSE}(x) = \{MSE_a(B_m(i, j), B_m(i, j+1)), MSE_b(B_m(i, j), B_m(i, j+1)), MSE_c(B_m(i, j), B_m(i, j+1))\}$
end for
$D_m(i, j+1) = \text{argmin}_x\{f_{MSE}(x)\}$/*The value that minimizes MSE is chosen as the estimated DC coefficient .*/
end for
end for

Assume there are $NumL \times NumW$ blocks in this image, $B_m(i, j)$ means the spatial domain of the 8×8 block located at (i, j). The $A_m(i, j)$ is the 63 AC coefficients of block $B_m(i, j)$ which is known. The $D_m(i, j)$ is the DC coefficient of block $B_m(i, j)$ which is unknown except for the first row in this step. Q_{50} is used to get the correct DCT coefficients for JPEG image with a decompression. MSE functions are calculated through three directions as shown in Fig. 4.

For the first estimation of DC coefficients, the DC coefficients of blocks in the first row of the image are seen as the same value with the DC coefficient of the block located at the up-left corner. Then for each column, DC coefficients in the upper block will be used to estimate the DC coefficient in the block below the upper one. After estimating all the DC coefficients in this row, this step will repeat to estimate the next row with the calculated coefficients in this step. This calculation will scan the blocks row by row and from up to down.

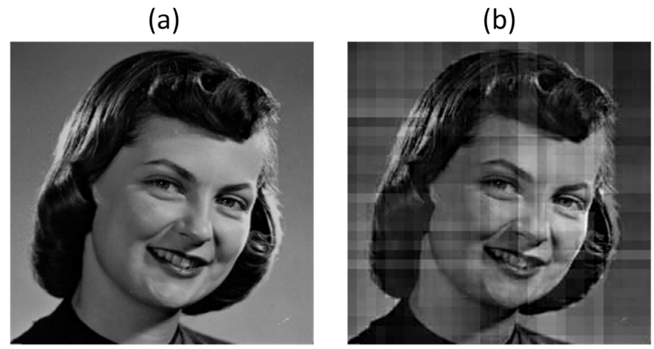

Fig. 9. (a) original JPEG image; (b) Recovered image with DC coefficients calculated from average of four different scanning directions.

The result of scanning from up to down is shown in Fig. 8. As shown in Fig. 8(b), basically the initial point of this recovery is just very blurred edges. The recovered images with different pre-set DC coefficients are shown in Fig. 8(c)–(f) respectively. The different pre-set DC coefficients just changed the brightness of this image while not affecting recovering the image contents. This is due to the difference of the pre-set DC values will only change the baseline of the grayscale for the image. While our method can still recover the grayscale's gradient distribution.

As we can observe in Fig. 8(c) which is the most similar brightness as the original image, some columns have more difference at the bottom part of the image. This is due to the error propagation due to the recursive estimation of the DC coefficients.

In order to reduce the error propagation, we calculate the other three estimations from other directions to scan the image: from down to up, from left to right, and from right to left. Then the average DC coefficients are calculated from this four times of scans to get the final results as shown in Fig. 9. In this result, the error for the backgrounds with pure colors is increased while the error for the details of the image is reduced. Especially the most important edges of the original image are recovered. In summary, our proposed method can effectively recover the image content for the JPEG case that DC coefficients are missing.

4. Analysis and discussion

In this section, we first give some experimental results with some unusual images. Then the PSNR [26] is measured between the original JPEG image and the recovered image. The comparison with previous works is also given to show the improvement based on the results. We also discuss the error propagation problems in the end and give a further research direction that may further improve the recovered image quality by reducing the error propagation.

4.1. Results analysis

More recovered images are shown in Fig. 10. In order to further test our proposed method, the experiments used three different kinds of images. For image of Fig. 10(a), the distribution of the grayscale is common to see: there are smooth color changes and also some sharp edges. For image of Fig. 10(d), the four corners are very smooth colors but in the middle of the image, there are many sharp edges and some points with huge difference on grayscale

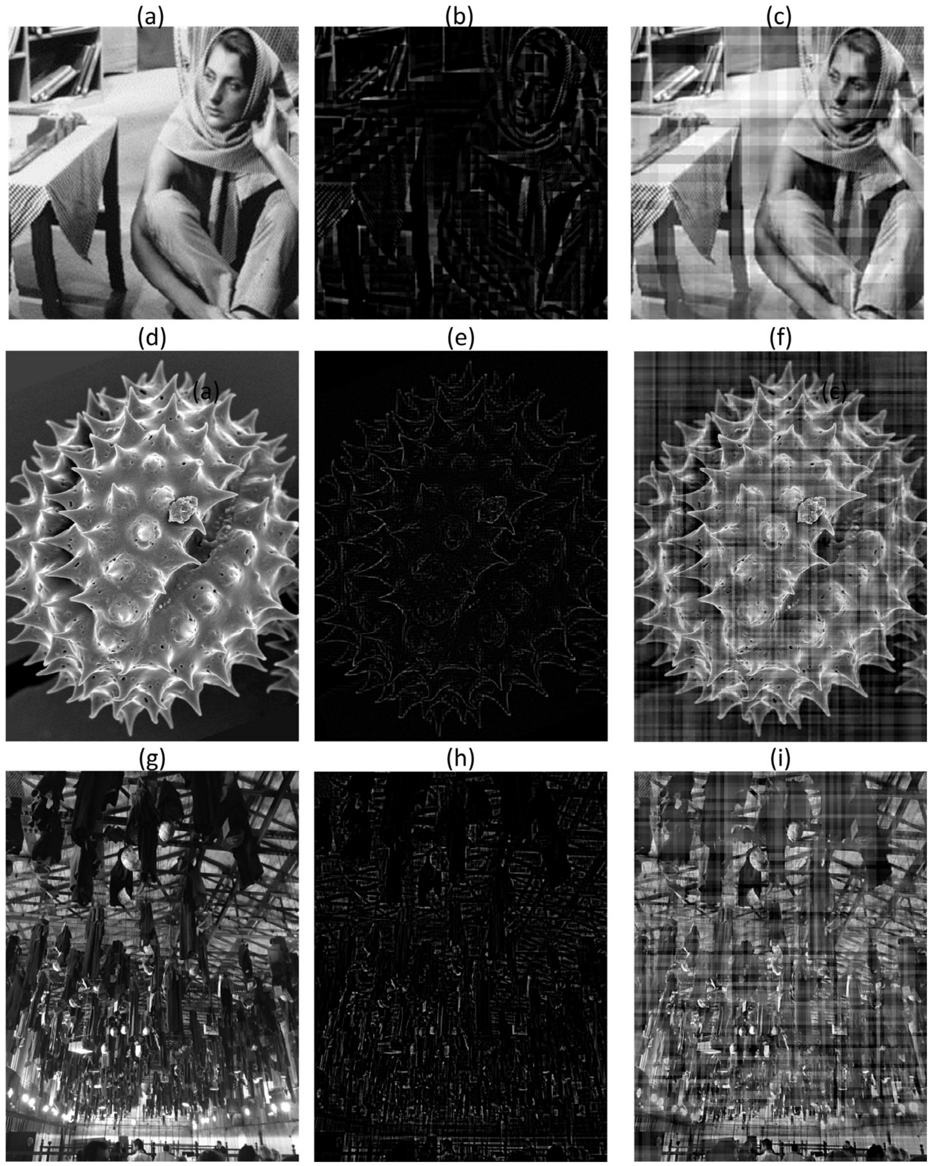

Fig. 10. (a), (d), and (g): original image; (b), (e), and (h): Image missing DC coefficients for each blocks; (c), (f), and (i): Recovered image with DC coefficients based on the proposed method.

between neighbor blocks. The blocks on the four corners of the image are hard to recover as the DC values are the only non-zero values of the blocks. For image of Fig. 10(g), it is full of sharp edges without any smooth grayscale distribution and there are many elements inside. There are many details in this image combined with the pure color blocks. The recovery results in Fig. 10(i) are still visible compared with the image without DC coefficient values Fig. 10(h).

Based on observation for the visual results, the proposed method can recover better the edges and details of the image. For the smooth color area, the recovery will introduce noises which cannot guarantee the smooth grayscale distribution in the results. This is due to the specialty of the JPEG image. As shown in Section 2, the JPEG image has intensive compression based on the quantization step. In fact, for the smooth color areas, the compression results are always DCT coefficient blocks with one large DC coefficient and seldom non-zero AC coefficients. Once the DC coefficients are missing, remaining AC coefficients in many neighbor blocks are all zero which does not give any hint about the original grayscale value. A small error will lead to the non-smooth in the results because there is no way to recover from all-zero blocks.

Table 2
PSNR between original JPEG images and recovered images in this paper.

Images	Fig. 8(a)	Fig. 10(a)	Fig. 10 (d)	Fig. 10(g)
PSNR	24.86 dB	25.64 dB	24.22 dB	23.13 dB

PSNR is used to test the noise ratio for images. In Table 2, we list the PSNR values for all images tested in this paper. All the PSNR values for each image are calculated between the JPEG images with the default quality (referred as JPEG 50% [27]) and the recovered images. It can be observed that the recovery quality is better than the results in [24] but is still far from good quality according to the standard defined in [26].

4.2. Discussion

In the last subsection, we compare our method with the previous works for recovering only the DC coefficients more limited pre-known information. One further experimentation would be to test the recovery results when more DCT coefficients are missing

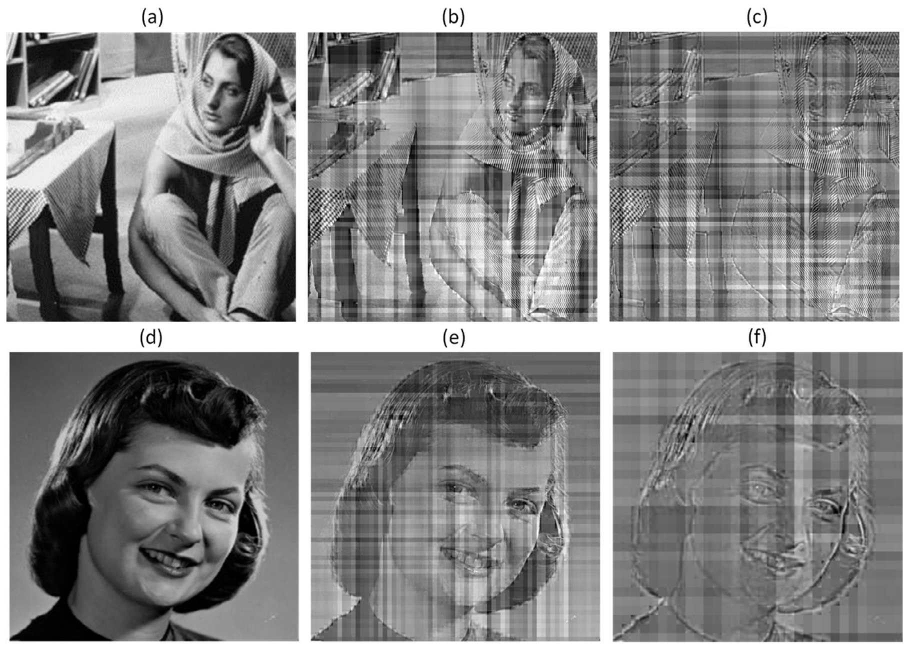

Fig. 11. (a), (d): original image; (b), (e): Recovered image with DC and first AC are missing; (c), (f): Recovered image with DC and first two ACs are missing.

which means the input is not all the non-zero AC coefficients. As shown in [16], the method proposed in [24] cannot deal with the bitmap situation when more DCT coefficients are missing. In this paper, we also test the recovery results with some ACs missing as shown in Fig. 11.

However, as most of the DCT coefficient blocks have only very few non-zero AC coefficients, it is not possible to recover the image when many ACs are missing. For instance, for the image in Fig. 8(a), once the first AC coefficient for each 8 block is missing, there will be 26.3% of the blocks that all DCT coefficients inside are zeros. Moreover, there are 75.9% of the DCT coefficient blocks have only one non-zero AC coefficients and this only one AC coefficient equals to one. Basically, this means there is a quarter of the image is recovered based on nothing and three quarters of the image are recovered from only one frequency coefficient with value 1. However, our method can still recover many of the image elements especially the human face as shown in Fig. 11(b) and (e). Furthermore, if there are two first AC coefficients missing, the ratio of the blocks that all DCT coefficients inside are zeros increases to be 40.7% which means almost half of the image is recovered from basically nothing. As shown in Fig. 11(c) and (f), the recovered results based on our method can show only some hints about the image content.

Apparently, if there are 3 first AC coefficients are missing, the ratio of DCT blocks full with only zeros are more than 50% and DCT blocks with only one non-zero coefficient with value 1 are almost 90%. Thus the recovery turns to be unpractical as no recovery can be done from all zeros and definitely it is not the question we need to solve for JPEG use case.

5. Discussion and future work

In this paper, error propagation is always the problem for such recovery methods. As the recovery is recursively recovering the DC coefficients according to the nearest neighbor block. Two problems need to be solved in future. The first problem is that there are many DCT coefficient blocks full with zeros. Such a situation means this block in the spatial domain is a block with pure color. Even the original DC coefficient may have slight differences, there is basically no way to detect the very small color change. Thus the error will propagate and it is difficult to detect and recover.

The second problem is that the recovery method is actually measuring the trends of grayscale changing. Once the grayscale changing trends between two neighbor blocks is sharp, the estimation will always introduce small errors. Moreover, our proposed method is always picking the estimated DC coefficient that can let this grayscale changing smoothest (minimum of the MSE between pixel pairs of neighbor blocks) which is not always the actual situation. In some cases, the grayscale changing trends of the original image is sharp which is hard to estimate with accuracy. One possible solution may be to use the energy distribution to detect the possible sharp edge in the grayscale changing trends. For instance, once DC coefficients are removed, the sum of the square value of the ACs are large means in this block, grayscale will change rapidly which always means there is a sharp edge around this block. For future work, this feature will be investigated for reducing error.

Also, as pointed out in Section 1, there are many other needs in the ubiquitous communication systems such as data security. Nowadays, the smart data analytics are developing to meet the needs for not only secure data transmission but also secure data computing. One example is given in [28] which can improve the existing Fully Homomorphic Encryption (FHE) methods practically. Thus, the idea to use our proposal for data protection methods can also be feasible for some specific use cases. For instance, this method can test the recovery level when more and more DCT coefficients are missing which can determine how many coefficients are protected when image content is secure enough for JPEG images.

6. Conclusion

In this paper, we presented a DC recovery method for JPEG images which can be used to resist the transmission error at the execution ends for ubiquitous communication systems. Firstly we assumed that all AC coefficients are correct and there is no pre-known knowledge about the DC coefficients. We presented our method by detecting the grayscale changing trends and using brute force to estimate the DC coefficients from all possible values. Then we analyzed the results and tested this method with the situation that more DCT frequency coefficients are missing. The recovery results are presented to prove the effectiveness of our proposal and compared with previous work. This method can be used to improve the fault tolerance for JPEG transmission in ubiquitous communication systems by recovering image contents at the receivers' ends when DC coefficients are missing. In summary, we proposed a JPEG image recovery method that improve the fault tolerance for JPEG image transmissions in the ubiquitous communication systems.

References

[1] M. Qiu, Z. Chen, Z. Ming, X. Qin, J. Niu, Energy-aware data allocation with hybrid memory for mobile cloud systems, IEEE Syst. J. 11 (2) (2017) 813–822.
[2] Y. Zhang, M. Qiu, C.-W. Tsai, M.M. Hassan, A. Alamri, Health-CPS: Healthcare cyber-physical system assisted by cloud and big data, IEEE Syst. J. 11 (1) (2017) 88–95.
[3] K. Gai, M. Qiu, H. Zhao, Cost-aware multimedia data allocation for heterogeneous memory using genetic algorithm in cloud computing, IEEE Trans. Cloud Comput. (2016).
[4] M. Qiu, K. Gai, B. Thuraisingham, L. Tao, H. Zhao, Proactive user-centric secure data scheme using attribute-based semantic access controls for mobile clouds in financial industry, Future Gener. Comput. Syst. 80 (2018) 421–429.
[5] A.M. Tekalp, Digital Video Processing, Prentice Hall Press, 2015.
[6] W.B. Pennebaker, J.L. Mitchell, JPEG: Still Image Data Compression Standard, Springer Science & Business Media, 1992.
[7] I.E. Richardson, H. 264 and MPEG-4 Video Compression: Video Coding for Next-Generation Multimedia, John Wiley & Sons, 2004.
[8] F. Hu, M. Qiu, J. Li, T. Grant, D. Taylor, S. McCaleb, L. Butler, R. Hamner, A review on cloud computing: Design challenges in architecture and security, J. Comput. Inform. Technol. 19 (1) (2011) 25–55.
[9] M. Qiu, E.H.-M. Sha, Cost minimization while satisfying hard/soft timing constraints for heterogeneous embedded systems, ACM Trans. Des. Autom. Electron. Syst. 14 (2) (2009) 25.
[10] M. Wu, L. Tan, N. Xiong, Data prediction, compression, and recovery in clustered wireless sensor networks for environmental monitoring applications, Inform. Sci. 329 (2016) 800–818.
[11] S. Kurt, H.U. Yildiz, M. Yigit, B. Tavli, V.C. Gungor, Packet size optimization in wireless sensor networks for smart grid applications, IEEE Trans. Ind. Electron. 64 (3) (2017) 2392–2401.
[12] N. Ahmed, T. Natarajan, K.R. Rao, Discrete cosine transform, IEEE Trans. Comput. 100 (1) (1974) 90–93.
[13] H. Schwarz, D. Marpe, T. Wiegand, Overview of the scalable video coding extension of the H. 264/AVC standard, IEEE Trans. Circuits Syst. Video Technol. 17 (9) (2007) 1103–1120.
[14] J. Zeng, O.C. Au, W. Dai, Y. Kong, L. Jia, W. Zhu, A tutorial on image/video coding standards, in: Signal and Information Processing Association Annual Summit and Conference (APSIPA), 2013 Asia-Pacific, IEEE, 2013, pp. 1–7.
[15] B. Thuraisingham, Data Mining: Technologies, Techniques, Tools, and Trends, CRC Press, 2014.
[16] H. Qiu, E. Nathalie, G. Memmi, A case study for practical issues of DCT based bitmap selective encryption methods, in: IEEE International Conference on Security of Smart Cities, Industrial Control System and Communications, IEEE, 2018.
[17] S. Ong, S. Li, K. Wong, K. Tan, Fast recovery of unknown coefficients in DCT-transformed images, Signal Process., Image Commun. 58 (2017) 1–13.
[18] R. Kresch, N. Merhav, Fast DCT domain filtering using the DCT and the DST, IEEE Trans. Image Process. 8 (6) (1999) 821–833.
[19] G.K. Wallace, The JPEG still picture compression standard, IEEE Trans. Consum. Electron. 38 (1) (1992) xviii–xxxiv.
[20] M. Tuba, N. Bacanin, JPEG Quantization tables selection by the firefly algorithm, in: Multimedia Computing and Systems (ICMCS), 2014 International Conference on, IEEE, 2014, pp. 153–158.
[21] L. Krikor, S. Baba, T. Arif, Z. Shaaban, Image encryption using DCT and stream cipher, Eur. J. Sci. Res. 32 (1) (2009) 47–57.
[22] H. Qiu, G. Memmi, Fast selective encryption methods for bitmap images, Int. J. Multimedia Data Eng. Manag. 6 (3) (2015) 51–69.
[23] S. Li, A. Karrenbauer, D. Saupe, C.C.J. Kuo, Recovering missing coefficients in DCT-transformed images, in: Image Processing (ICIP), 2011 18th IEEE International Conference on, IEEE, 2011, pp. 1537–1540.
[24] T. Uehara, R. Safavi-Naini, P. Ogunbona, Recovering DC coefficients in block-based DCT, IEEE Trans. Image Process. 15 (11) (2006) 3592–3596.
[25] T. Chai, R.R. Draxler, Root mean square error (RMSE) or mean absolute error (MAE)?–Arguments against avoiding RMSE in the literature, Geosci. Model Dev. 7 (3) (2014) 1247–1250.
[26] Q. Huynh-Thu, M. Ghanbari, Scope of validity of PSNR in image/video quality assessment, Electronics letters 44 (13) (2008) 800–801.
[27] T. Pevny, J. Fridrich, Merging Markov and DCT features for multi-class JPEG steganalysis, in: Security, Steganography, and Watermarking of Multimedia Contents IX, 6505, International Society for Optics and Photonics, 2007, p. 650503.
[28] K. Gai, M. Qiu, Blend arithmetic operations on tensor-based fully homomorphic encryption over real numbers, IEEE Trans. Ind. Inf. 14 (8) (2018) 3590–3598.

Han Qiu has earned a Ph.D. degree in Computer Science from the Department of Network and Computer Science at Telecom-ParisTech, Paris, France. He also holds degrees from Beijing University of Posts and Telecommunications (BE), Institute Eurecom (MS). Currently, he is a PostDoc researcher at the Department of Network and Computer Science at Telecom-ParisTech, Paris, France. His research interests include heterogeneous computing, cyber security, applied cryptography, multimedia security.

Gerard Memmi currently is professor and Head of the Networks & Computer Science Department at Telecom ParisTech since 2009. He is member of the executive board of the IRT SystemX since 2012. Before joining Telecom ParisTech, Gerard Memmi held various executive positions in American start-ups. He succeeded in delivering the industry's best-in-class equivalency checker used to verify electronic design; and focused on improving its architecture and performances. While founding and developing the Applied Research Laboratory for Groupe Bull in the US, he was honored as Principal Investigator for a DARPA grant on Collaborative Software. He has over 80 publications including patents, co-authored a book, gave key notes presentations in International Conferences. He is holding a thèse d'Etat in Computer Science from Université Pierre et Marie Curie, Paris. Gerard Memmi is constantly involved in the development of key scientific and industrial partnership. Today, his main scientific topics of interest are about data protection and privacy, energy profiling of software programs, and verification of distributed systems.

Xuan Chen is currently a Ph.D. student majored in Philosophy in Sorbonne University, Paris, France. She also holds two Master degrees from Sorbonne University majored in Sociology and Philosophy, respectively. Her research interests include social data science, trust issues in social networks and social computing.

Jian Xiong received the B.Sc. and M.Sc. degrees from the University of Electronic Science and Technology of China, Chengdu, China, and the Ph.D degree from Shanghai Jiao Tong University (SJTU), Shanghai, China, in 1999, 2002, and 2006, respectively. He was a visiting scholar of Columbia University during 2015. He is currently an Associate Professor of the Image Communication and Networking Engineering Institute, SJTU. His current research interests include wireless wideband transmission technologies, networking, and caching technologies of converged wideband and broadcast systems. He has published over 40+ journal or conference papers and holds 40+ patents, including 39 awarded patents. He has won one Best Journal Paper Award (IEEE TB'14) and 2 Conference Best Paper Awards (IEEE/BMSB'16, IEEE/SSC'16). He is the TPC Co-Chair of the IEEE International Symposium on Broadband Multimedia Systems and Broadcasting from 2010 to 2018.