# Localization of Forgeries in MPEG-2 Video through GOP Size and DQ Analysis

D. Labartino [#1], T. Bianchi [*2], A. De Rosa [#3], M. Fontani [†4], D. Vázquez-Padín [‡5], A. Piva [#6], M. Barni [†7]

[#] *Dept. of Information Engineering, Università di Firenze (Italy)*
[1] `daniele.labartino@gmail.com` [3] `alessia.derosa@unifi.it` [6] `alessandro.piva@unifi.it`

[*] *Dept. of Electronic and Telecommunications, Politecnico di Torino (Italy)*
[2] `tiziano.bianchi@polito.it`

[†] *Dept. of Information Engineering and Mathematical Sciences, Università di Siena (Italy)*
[4] `marco.fontani@unisi.it` [7] `barni@dii.unisi.it`

[‡] *Signal Theory and Communications Department, University of Vigo (Spain)*
[5] `dvazquez@gts.uvigo.es`

*Abstract*—**This work addresses forgery localization in MPEG-2 compressed videos. The proposed method is based on the analysis of Double Quantization (DQ) traces in frames that were encoded twice as intra (i.e., I-frames). Employing a state-of-the-art method, such frames are located in the video under analysis by estimating the size of the Group Of Pictures (GOP) that was used in the first compression; then, the DQ analysis is devised for the MPEG-2 encoding scheme and applied to frames that were intra-coded in both the first and second compression. In such a way, regions that were manipulated between the two encodings are detected. Compared to existing methods based on double quantization analysis, the proposed scheme makes forgery localization possible on a wider range of settings.**

## I. Introduction

Images and videos are probably the most important media in everyday's communications. They also became important in fields like medicine, military and security, so the possibility of knowing their provenience and certifying their credibility is of great interest. A number of techniques to investigate the processing history of digital contents are emerging, under the name of Multimedia Forensics; these techniques exploit the fact that some imperceptible traces are left in the media during its acquisition, modification and compression.

The research community started working on images first, and a huge number of methods exist today for analyzing the authenticity of images (a recent survey is given in [1]). On the contrary, video forensics is still an emerging field, for several reasons: creating fake images is much easier than creating fake videos; images are usually available either in uncompressed or JPEG format, while videos can be encoded with many different schemes and, finally, videos usually undergo a stronger compression compared to images, making their forensic analysis more difficult. This contrasts with the fact that nowadays digital videos are probably used more than images for security tasks (just think about video-surveillance

systems), so their trustability must be strengthened.

As it will be shown in Section III, a great deal of the research activity in video forensics focused on detection of double compression. Although double compression can be considered a necessary step for creating a tampered video, saying that a sequence has been encoded twice is not a sufficient proof for claiming its non-authenticity: it may be the case, for example, that the video is automatically re-encoded when it is downloaded from the acquisition device. For this reason, investigating the authenticity of a digital video requires to take one more step in the analysis. Furthermore, a distinction must be made between intra-frame and inter-frame video forgeries [2]: in the former, the attacker changes the content of some frames (e.g., by adding or removing an object), while in the latter one or more frames are entirely added/removed from the video. It is easy to guess that different techniques are needed to investigate these two tampering scenarios.

This paper presents a method for forgery localization in the frames of a MPEG-2 compressed video, that is, a method allowing to determine which parts of a frame have been altered. The method basically works by searching for traces of double quantization at a spatial level, allowing to build a fine-grained probability map of tampering for each analyzed frame. This is done by adapting and extending the method proposed in [3] to the MPEG-2 encoding scheme. Due to how the MPEG-2 encoding is performed, this kind of analysis is only possible on frames that have been intra-coded[1] twice. Therefore, we first adopt a state-of-the-art method for localizing the position of such frames in the first encoding, and then perform the proposed analysis on the suitable frames.

Compared to state-of-the-art approaches targeting the same task, the proposed method achieves forgery localization under more realistic working scenarios (e.g., video encoded using motion prediction, that are the vast majority), and exploits some advantages of the MPEG-2 encoding standard to improve the robustness of the analysis.

---

[1] Intra-coding is described in Section II.

The paper is structured as follows: Section II makes a very brief tour on MPEG-2 compression, then related works are considered in Section III; the proposed method is explained in Section IV and experimentally validated in Section V.

## II. MPEG-2 Video Compression

MPEG-2 video standard (ISO/IEC 13818-2/ITU-T recommendation H.262) is a widely employed method for video compression, that basically works by reducing both spatial and temporal redundancy in a captured video sequence. The standard follows a block-based hybrid video coding approach and defines different types of pictures: intra-coded pictures, referred to as I-frames (only progressive videos are considered on this paper), and predictive-coded pictures, commonly named P-frames and B-frames. Given the block-based structure, each frame of a video sequence is divided into macroblocks (MBs), i.e., blocks of $16 \times 16$ samples, which are encoded following several coding modes that are available according to the selected type of frame.

In a similar way as it happens with JPEG images, the MBs in I-frames are encoded without making reference to other frames: each MB of the luminance component (we do not consider the chrominance for brevity) is divided in blocks of $8 \times 8$ pixels that are transformed according to the DCT and whose coefficients are later quantized (details about this step will be given in Section IV). Quantization in the DCT domain allows to remove spatial redundancy in a perceptually convenient way.

By compressing the whole video using only intra-coded pictures, this would lead to the so-called Motion-JPEG encoding, where temporal redundancy is not exploited. Notice that, although being very similar to the JPEG compression scheme, the mentioned procedure uses a slightly different quantization function. In MPEG-2, the coarseness of the quantization is selected by the encoder through the quantizer scale factor, denoted as Q, that ranges from 0 to 31 and maps the values of the multiplier $k$ that is applied to the quantization matrix. Two different mappings are available in the standard, but in this paper we will assume the one that corresponds to $k = 2Q$ (except for Q = 0, where no value is assigned to $k$). Therefore, by fixing the multiplier to a certain value, the factor Q enables to control the trade-off between the quality and bitrate of a compressed video. If the value of Q is constant, then a fixed quantizer will be used and a Variable BitRate (VBR) will be provided; while if it is adapted on a frame to frame (or even on a MB to MB) basis, then a Constant BitRate (CBR) can be achieved.

In a general scenario, a strong correlation between adjacent frames will be present since the scene is captured at several frames per second, and this temporal redundancy should be exploited to increase the level of compression. This is obtained through motion compensation. For instance, when encoding a picture as a P-frame, each MB is compared with the respective area in neighborhood positions within the previous encoded and reconstructed frame (i.e., a reference frame), in order to find the region that better resembles the MB to encode.

If a good match is found, then the MB is predictive-coded: the displacement vector (i.e., a motion vector) is stored and the residual difference with the reference MB is $8 \times 8$-DCT transformed and further quantized. However, if a good match is not available, then the MB is intra-coded like in an I-frame and we will refer to this type of macroblock as I-MB. Finally, if after performing the predictive-coding there is no need to transmit the motion vector (because it is null) and the residual difference after quantization is also negligible, then the standard defines a specific type of macroblock, called skipped MB and we will address to it as S-MB.

The only difference between P- and B-frames is that the MBs on B-frames can be bidirectionally predictive-coded, in such a way that the motion compensation can be carried out from a past and/or a future reference frame.

## III. Previous Works

Although not being as much advanced as for images, the forensic analysis of digital videos already counts several achievements. Only previous works that are related to the proposed method are mentioned in this section, while a thorough overview and taxonomy of video forensic techniques can be found in [2].

One of the most studied tasks in video forensics is the detection of double encoding and/or transcoding, probably because a tampered video is usually encoded (at least) twice, namely during the acquisition and after the manipulation. Several works are based on the effects introduced by double quantization in the DCT coefficients. In [4], the I-frames of the video are considered and the histogram of two quantized DCT coefficients is studied in order to search a convex pattern, that characterizes double-encoded videos. By adopting a simple yet effective approach, the method is extended also to the challenging CBR case: macroblocks are separated in different sets, according to their quantization parameter, and the analysis is carried separately for each set. An approach based on Benford's law is presented in [5], where the first-digit distribution of DCT coefficients of I-frames is considered and a 12-dimensional feature is extracted to be classified using Support Vector Machines. Besides detecting double encoding, the method also classifies the second encoding as being at a higher or lower bitrate with respect to the first one. On the other hand, this method may not work when the two encodings are performed using different implementation of the MPEG-2 standard.

Generalizing also to other video coding standards, a very recent work has been published about double encoding detection for MPEG-4 videos [6], based on Markov statistics extracted from DCT coefficients. Recently, Benford's law together with SVMs has been employed by authors of [7] for detecting multiple (i.e., even more than two) encodings of the same video. Finally, a completely different method to detect double video encoding has been proposed in [8], where the authors show that when a video is encoded twice, a characteristic behavior appears in those P-frames that are the re-encoded version of the I-frames of the original sequence.

The method works both when VBR and CBR coding is used and, noticeably, also works when different codecs are used in performing the two compressions.

So far we have only mentioned works targeting the detection of double compression. Switching to tampering detection, an effective method for detecting removal of frames was proposed in [9], where the de-synchronization (induced by the tampering) between the GOP used for the first and for the second encoding is detected, by searching for a periodic behavior in the magnitude of motion vectors. Another method is presented in [10], where the different characteristics of quantization matrices employed for intra- and predictive- coded frames is considered: if an I-frame is re-encoded as P or B, a different quantization matrix will be used that preserves more energy in the high-frequency DCT coefficients. This fact is studied to detect periodic anomalies in the energy of some DCT coefficients, thus exposing that a de-synchronization in the GOP structure occurred. Finally, authors of [4] show that their method for double encoding detection can also be used to detect frame removal.

Intra-frame forgery localization is probably the less studied field in video forensics today, and most of the existing approaches work only under strict assumptions [2]. The most recent method is the one proposed in [11], where a DQ analysis is applied separately for each macroblock. The underlying idea is that when some of the MBs of a frame show the effects of double quantization and some others do not, the last ones have been probably pasted from another sequence. This idea is borrowed from JPEG image forensics and, as such, the analysis makes sense only on frames that have been encoded twice as intra. The authors work around this problem by assuming that Motion-JPEG encoding has been performed (i.e., only intra-coded pictures are used), thus heavily restricting the applicability of the method. Furthermore, in [11] the double quantization analysis is performed separately on each MB, leading to a computationally intensive analysis.

## IV. PROPOSED METHOD

In this section, we present a new method for localizing forgeries in MPEG-2 videos. We focus on the intra-frame forgery scenario, and we assume that, starting from a MPEG-2 video sequence, the attacker decodes the video, alters the content of a group of frames, and finally encodes the resulting sequence again with MPEG-2, using a different GOP size. In the following, we assume a fixed quantizer and that the default quantization matrix is employed, leading to a VBR coding. The approach makes use of the method presented in [8] to retrieve the GOP size of the first compression. By knowing this, the location of the I-frames in the first encoding is inferred and the DQ effect is studied in those frames that have been encoded as intra both in the first and second encoding. In this work, we do not consider the removal/addition of whole frames: this would cause a misalignment in the GOP structures, complicating the localization (based on the method in [8]) of frames that have been encoded twice as intra.

### A. Detection of Frames Encoded Twice as Intra

Recently, the Variation of Prediction Footprint (VPF) has been proposed as a method to detect double video encoding [8]. This footprint captures a characteristic phenomenon that occurs when an I-frame is re-encoded as a P-frame: in such a frame, the number of S-MBs noticeably decreases, while the number of I-MBs strongly increases. By measuring the presence and the periodicity of this phenomenon, an algorithm has been proposed to detect double compression and to estimate the size of the GOP used for the first encoding.

Let us assume that a video, composed by $N$ frames, has been encoded twice using $G_1$ and $G_2$ as the GOP size for the first and second encoding respectively, where $G_1 \neq m \cdot G_2, \forall m \in \mathbb{N}$. Assuming a fixed GOP structure, the set of indices of the frames that have been intra-coded twice is

$$\mathcal{C}_{G_1,G_2} = \{n \in \mathbb{N} : n = m \cdot \mathrm{lcm}(G_1, G_2) \wedge n \leq N, \forall m \in \mathbb{N}\},$$

where $\mathrm{lcm}(G_1, G_2)$ represents the least common multiple between $G_1$ and $G_2$. The cardinality $|\mathcal{C}_{G_1,G_2}|$ of the set is simply given by:

$$|\mathcal{C}_{G_1,G_2}| = 1 + \left\lfloor \frac{N}{\mathrm{lcm}(G_1, G_2)} \right\rfloor,$$

where $\lfloor \cdot \rfloor$ stands for the floor function. In other words, forgery localization can be performed every $\mathrm{lcm}(G_1, G_2)$ frames. Therefore, for relatively prime values of $G_1$ and $G_2$ the analysis can be carried out only once every $G_1 \cdot G_2$ frames, and this value could be not appealing. On the other hand, the GOP size is usually chosen from a set of possibilities, like 12 for PAL videos, 15 for NTSC videos, while recording devices often choose a GOP size around 30. At a frame rate of 25 fps, combinations of the mentioned values for $G_1$ and $G_2$ result in a satisfactory time resolution for the analysis.

There is one important fact that must be considered to justify the adoption of the method proposed in [8], since their experiments are conducted on double encoded videos, without modifications between the two encodings. On the contrary, we are assuming that the video is manipulated (by altering the content of a group of frames) before the second compression takes place. The robustness of the VPF in this scenario must be evaluated, and this task will be addressed in Section V.

### B. Forgery Localization Based on DQ Analysis

According to the assumed forgery scenario, tampered frames that have been encoded twice as intra will consist of two groups of pixels: the one that has not been modified, thus undergoing a double quantization, and the one that has been introduced between the encodings. Even when these latter pixels come from a compressed sequence, they will unlikely be pasted respecting the $8 \times 8$ quantization grid of the host frame and, therefore, will not show traces of double quantization after the second encoding, thus making localization possible. A thorough explanation of this model is given in [12]. That said, if we consider the histogram of a specific DCT coefficient (e.g., the one in position (0,1) in all $8 \times 8$ blocks), we should see a mixture of two components: a comb-shaped component
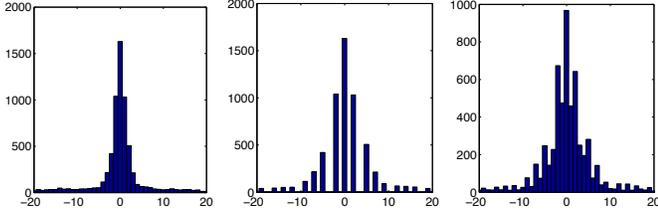
Fig. 1. Histogram of a DCT coefficient from a single compressed frame (left), double compressed frame (center) and tampered frame (right). Notice that the tampered histogram can be seen as a mixture of the previous two.

due to unaltered, double compressed regions, and a "standard" component due to regions that have been introduced (see Figure 1). In [3] a Bayesian inference method is proposed that first assigns to each DCT coefficient of a block its probability of belonging to each one of these components, and then accumulates these probabilities for all the coefficients of a block, producing an aggregated probability for the whole block of being/not being doubly compressed. The output is a map associating to each $8 \times 8$ block of pixels its probability of being tampered (i.e., not showing the DQ effect) or untouched (i.e., showing the effect). In order to compute such a map, the mentioned algorithm basically performs the following steps (see [3] for a formal presentation) for each group of DCT coefficients sharing the same position:

1) from the observed DCT coefficients, estimate the histogram $\tilde{h}$ that would result after a single encoding with the quantization step used in the second compression;
2) estimate the quantization step that was used during the first compression;
3) knowing both quantization steps, compute a function $n(x)$ that gives the number of bins of the original histogram that are mapped in the bin corresponding to the value $x$ in the double quantized histogram.

Then, denoting with $\mathcal{H}_0$ and $\mathcal{H}_1$ the hypothesis of being tampered and original, respectively, for each coefficient $x$ authors in [3] obtain:

$$p(x|\mathcal{H}_0) = \tilde{h}(x) \qquad (1)$$

and

$$p(x|\mathcal{H}_1) = n(x) \cdot \tilde{h}(x), \quad x \neq 0. \qquad (2)$$

These steps are carried out separately for each DCT coefficient (usually only the first dozen of AC coefficients are used for the analysis). Then, for each $8 \times 8$ block, the probability of being tampered is "accumulated" as:

$$p = 1 / \left( \prod_{i|x_i \neq 0} n_i(x_i) + 1 \right), \qquad (3)$$

where $n_i(x)$ is the $n(x)$ function for the $i$-th coefficient.

Since DCT coefficients quantization is a key step both in JPEG and MPEG-2 coding, the above method can been borrowed from image to video forensics, as suggested in [11]. However, some significant differences must be considered to devise a correct model for MPEG-2:

1) the dequantization formula in JPEG differs from that of MPEG-2 [13];
2) in JPEG, the $8 \times 8$ quantization matrix is declared in the header and usually it is not governed by the quality factor; in MPEG-2, instead, the adopted matrix (the default or a custom one) is parameterized by the multiplier $k$ (see Section II), to adjust the quantization strength;
3) in JPEG, the quantization matrix is the same for all the image; this holds also for MPEG-2 when a fixed quantizer is used, while the quantization matrix may change from frame to frame or MB to MB, for instance, for CBR coding.

Each of these facts has a direct implication on the model described in [3]. Since a different quantization formula is used, the function $n(x)$ will likely change; the fact that all quantization coefficients are obtained multiplying by $k$ makes it not necessary to estimate a different quantization step for each coefficient (we can directly estimate $k$); finally, in the case of CBR coding, that is left for future work, MBs that are not quantized using the same $k$ must be analyzed separately.

Inspired by [3], the approach we follow is to model the histogram of DCT coefficients in tampered frames as a mixture between a double quantized component and a single quantized component. To get a reliable estimate of coefficients quantized only once (by the quantization factor employed in the last encoding), we make use of the calibration technique [14]: the frame is cropped by one row and one column, and the result is quantized with the second quantization matrix. Consistently with [3], this component will be indicated by $\tilde{h}(x)$. Therefore, given a coefficient $x$, its probability of belonging to a tampered region is estimated as in (1).

To get an estimate of the other component, we need to derive the appropriate function $n(x)$ for the MPEG-2 quantization scheme. In the following, we denote the never-compressed DCT coefficient on the $i$-th row and $j$-th column of an $8 \times 8$ block with $x(i,j)$, where $i, j \in \{0, \ldots, 7\}$. Similarly, we denote with $u_1(i,j)$ the quantized version of the coefficient, with $x_1(i,j)$ its de-quantized version, and with $u_2(i,j)$ its re-quantized version. We also denote each element in the $8 \times 8$ quantization matrix with $W(i,j)$, and we call $k_1$ and $k_2$ the multipliers that parameterized the quantization matrix in the first compression and in the second compression, respectively. According to the MPEG-2 standard [13], and following the proposed notation, the de-quantized version of the DCT coefficients coming from a single compressed intra-coded frame is:

$$x_1(i,j) = \text{sign}\left(u_1(i,j)\right) \left\lfloor \frac{W(i,j) \cdot |u_1(i,j)| \cdot k_1}{16} \right\rfloor \qquad (4)$$

for all coefficients apart from the DC, where $|\cdot|$ is the absolute value operator. Starting from (4), the most intuitive way to define the quantization is:

$$u_1(i,j) = \left\lceil \frac{16 \cdot x(i,j)}{k_1 \cdot W(i,j)} \right\rceil, \qquad (5)$$
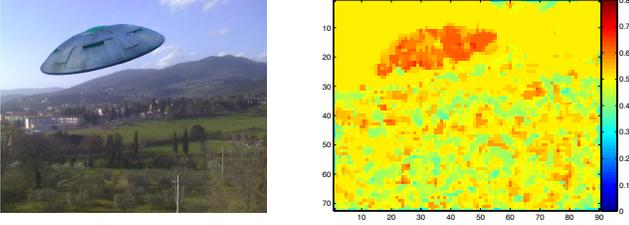
Fig. 2. An intra-frame tampering (left figure) and the produced probability map (right figure). For showing purposes, a $3 \times 3$ median filter has been applied to the map.

where $[\cdot]$ represents the rounding to nearest integer operation. According to (4) and (5), the re-quantized version of the DCT coefficients, (i.e., the double quantized coefficients), can be written (omitting the position indices) as:

$$u_2 = \left[ \frac{16}{k_2 \cdot W} \left( \text{sign} \left( \left[ \frac{16 \cdot x}{k_1 \cdot W} \right] \right) \left[ \frac{W \cdot \left| \left[ \frac{16 \cdot x}{k_1 \cdot W} \right] \right| \cdot k_1}{16} \right] \right) \right]$$

From this formula, the function $n(x)$ can be proved to be:

$$n(x) = \frac{k_1 \cdot W}{16} \left( \left\lceil \frac{16}{k_1 \cdot W} \left\lceil \frac{k_2 \cdot W}{16} \left( u_2 + \frac{1}{2} \right) \right\rceil \right\rceil - \left\lceil \frac{16}{k_1 \cdot W} \left\lceil \frac{k_2 \cdot W}{16} \left( u_2 - \frac{1}{2} \right) \right\rceil \right\rceil \right), \quad (6)$$

where $\lceil \cdot \rceil$ denotes the ceiling function. In the above equation, $k_1$ is the only parameter that must be estimated, given that $k_2$ and the values of $W$ are available from the bitstream. The multiplier $k_1$ is defined by its relation with the quantizer scale factor used in the first encoding $Q_1$ (see Section II), yielding to a possible value within the set $\mathcal{K}_1 = \{2Q_1 : 1 \leq Q_1 \leq 31\}$. If we assume to have the correct $k_1$, the histogram of doubly quantized coefficients can be obtained from $\tilde{h}(x)$ as $n(x; k_1) \cdot \tilde{h}(x)$, and we could write the probability distribution of the observed coefficients as the following mixture:

$$p(x; k_1, \alpha) = \alpha \cdot n(x; k_1) \cdot \tilde{h}(x) + (1 - \alpha) \cdot \tilde{h}(x), \quad (7)$$

where $\alpha \in [0, 1]$. As suggested in [3], an effective way to get an estimate of $k_1$ is to iteratively search the value $\hat{k}_1$ that minimizes the difference between the observed histogram $h(x)$ and $p(x; k_1, \alpha)$, choosing the optimal $\alpha$ in the least square sense (formula is given in [3]). With respect to the JPEG case, the minimization is simplified by the fact that the quantization matrix is known, and all the coefficients share the same $k_1$. Thus, we define the following vector

$$\mathbf{h} = [h_1(-\tfrac{B}{2}) \dots h_1(-1)\, h_1(1) \dots h_1(\tfrac{B}{2})\, h_2(-\tfrac{B}{2}) \dots h_C(\tfrac{B}{2})]^T$$

where $B+1$ is the number of bins of $h(x)$ and $C$ is the number of considered coefficients. We similarly define $\tilde{\mathbf{h}}$ and $\mathbf{n}$. Then, we can write:

$$\mathbf{p}(k_1, \alpha) = \alpha \cdot \mathbf{n}(k_1) \cdot \tilde{\mathbf{h}} + (1 - \alpha) \cdot \tilde{\mathbf{h}},$$

where the product between vectors is made element-by-element. Finally, $\hat{k}_1$ is obtained as

$$\hat{k}_1 = \arg \min_{k_1 \in \mathcal{K}_1} ||\mathbf{h} - \mathbf{p}(k_1, \alpha)||^2.$$

By using all the coefficients to estimate $k_1$, a more robust estimation is obtained; this is a crucial benefit, especially if we consider that: i) values in $W$ are quite high even for small $i$ and $j$, ii) the spatial resolution of videos is usually lower than that of images; and both these facts reduce the number of DCT coefficients that can be fruitfully exploited for the estimation. Using $\hat{k}_1$ and the $n(x)$ function defined in (6) for the MPEG-2 case, we can compute the probability in (2). Finally, the probability for each $8 \times 8$ block of being tampered is obtained through equation (3). Figure 2 shows a forged frame along with the probability map generated by the proposed method.

## V. EXPERIMENTAL RESULTS

Experiments have been carried out on a set of well known videos, selected so to have heterogeneous scenes[2], cropped to a resolution of $720 \times 576$ pixels. MPEG-2 VBR coding with a fixed quantizer is performed using the FFmpeg coding software[3].

The experimental validation follows this path: the video is compressed with a quantizer scale factor $Q_1$; then it is decoded and a square block of $200 \times 200$ pixels is replaced with the same content coming from the uncompressed version of the video; finally, the resulting video is re-encoded with a factor $Q_2$. Using the uncompressed version of the same video as a source for tampered pixels, it is possible to create a forgery that is practically imperceptible to the eye, thus mimicking the work of an editing expert.

Given that performance of the VPF do not strongly depend on the size of GOPs [8], we employed fixed GOP sizes $G_1 = 12$ and $G_2 = 15$ for the first and second compression respectively (this choice is motivated in Section IV-A). Furthermore, we limit ourselves to use P-frames, since GOP estimation in presence of B-frames is not possible with the mentioned method. As shown in [3], forgery localization generally works when the second compression is not as strong as the first one. For this reason, we choose $Q_1 \in \{6, 8, 10, 12\}$, $Q_2 \in \{2, 3, 4, 5\}$, and all the possible combinations between these two sets are used for generating tampered videos. Finally, since the model proposed in Section IV-B has been derived assuming that the fixed quantizer is uniform, the dead-zone of the quantizer implemented in FFmpeg is fixed to the interval $[-\Delta/2, \Delta/2]$ (where $\Delta$ denotes the quantization step), by setting the parameter ibias equal to 128. Note that the model can be easily adjusted to work with different dead-zones.

First of all, we have investigated the reliability of the VPF-based GOP estimation in the considered scenario since, in the case of a wrong estimation, the proposed method would fail. The GOP size was retrieved from the available set of tampered videos and the number of exact estimations of $G_1$ was calculated. The estimation never failed under the considered settings, thus confirming that VPF can be safely used in the proposed chain of analysis.

[2] Selected videos are: *ducks_take_off*, *in_to_tree*, *old_town_cross*, *park_joy*, *shields*, *sunflower* and *touchdown_pass*, freely available at http://media.xiph.org/video/derf/.
[3] http://www.ffmpeg.org/

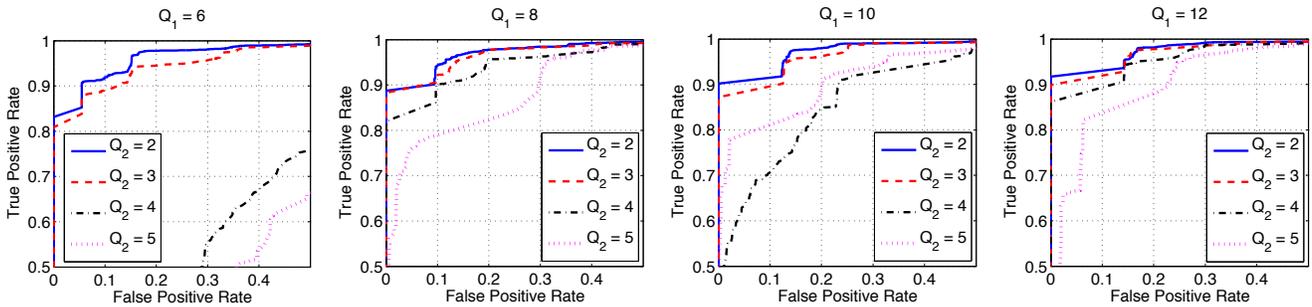Fig. 3. ROC curves obtained for the examined combination of $Q_1$ and $Q_2$. From left to right, increasing values of $Q_1$ are considered, and performance for varying values of $Q_2$ are plotted (notice that curves have been magnified to improve readability). As expected, lower values for the second compression facilitate the localization.

After retrieving the GOP size of the first compression (denoted with $\hat{G}_1$) for each tampered video, the DQ analysis is carried out, specifically in frames indexed by elements in the set $\mathcal{C}_{\hat{G}_1,15}$, defined in Section IV-A. Only the first 5 AC coefficients in the zig-zag ordering are used for the analysis. The probability map produced from each frame is then thresholded and compared to the ground truth mask, allowing us to calculate the true positive and false positive rate; these values are averaged over all videos sharing the same combination of $Q_1$ and $Q_2$. By varying values of the threshold, for all the explored combinations of $Q_1$ and $Q_2$ Receiver Operating Characteristic (ROC) curves are obtained (Figure 3) and their Area Under Curve (AUC) is calculated (Table I) . We see clearly that, for a given $Q_1$, lower values of $Q_2$ facilitate the localization; on the other hand, higher values of $Q_1$ favour the performance of the method.

TABLE I
AUC OBTAINED WITH THE PROPOSED METHOD.

| $Q_1$ \ $Q_2$ | 2 | 3 | 4 | 5 |
|---|---|---|---|---|
| 6 | 0.98 | 0.97 | 0.68 | 0.63 |
| 8 | 0.98 | 0.98 | 0.96 | 0.93 |
| 10 | 0.98 | 0.98 | 0.91 | 0.94 |
| 12 | 0.98 | 0.98 | 0.97 | 0.94 |

## VI. DISCUSSION

In this paper we have proposed a method for localizing forgeries in the frames of a MPEG-2 compressed video. The method works by first locating frames that have been intra-coded twice, and then applying a double quantization analysis to them, based on a proposed model specific for MPEG-2.

As a key contribution, our method exploits the characteristics of MPEG-2 coding, and it is the first allowing to apply DQ analysis to videos that have been encoded using P-frames. Experimental results are promising and lay the basis for future work, that will focus on: i) facing the case where one or both compressions are at CBR; ii) taking into account the presence of B-frames, that are currently not supported by the feature used for GOP estimation; iii) extend the experimental validation to realistic, hand-made, video forgeries.

## REFERENCES

[1] A. Piva, "An overview on image forensics," *ISRN Signal Processing*, vol. 2013, 2013.

[2] S. Milani, M. Fontani, P. Bestagini, M. Barni, A. Piva, M. Tagliasacchi, and S. Tubaro, "An overview on video forensics," *APSIPA Transactions on Signal and Information Processing*, vol. 1, 11 2012.

[3] T. Bianchi, A. De Rosa, and A. Piva, "Improved DCT coefficient analysis for forgery localization in JPEG images," in *Acoustics, Speech and Signal Processing (ICASSP), IEEE International Conference on*, 2011.

[4] J. Xu, Y. Su, and Q. Liu, "Detection of double MPEG-2 compression based on distributions of DCT coefficients," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 27, no. 01, p. 1354001, 2013.

[5] T. Sun, W. Wang, and X. Jiang, "Exposing video forgeries by detecting MPEG double compression," in *Acoustics, Speech and Signal Processing (ICASSP), IEEE International Conference on*, 2012, pp. 1389–1392.

[6] X. Jiang, W. Wang, T. Sun, Y. Shi, and S. Wang, "Detection of double compression in MPEG-4 videos based on Markov statistics," *Signal Processing Letters, IEEE*, vol. 20, no. 5, pp. 447–450, 2013.

[7] S. Milani, P. Bestagini, M. Tagliasacchi, and S. Tubaro, "Multiple compression detection for video sequences," in *Multimedia Signal Processing (MMSP), IEEE 14th International Workshop on*, 2012, pp. 112–117.

[8] D. Vazquez-Padin, M. Fontani, T. Bianchi, P. Comesana, A. Piva, and M. Barni, "Detection of video double encoding with GOP size estimation," in *Information Forensics and Security (WIFS), IEEE International Workshop on*, 2012, pp. 151–156.

[9] W. Wang and H. Farid, "Exposing digital forgeries in video by detecting double MPEG compression," in *MM&Sec*, 2006, pp. 37–47.

[10] Y. Su, W. Nie, and C. Zhang, "A frame tampering detection algorithm for MPEG videos," in *Information Technology and Artificial Intelligence Conference, 6th IEEE Joint International*, vol. 2, 2011, pp. 461–464.

[11] W. Wang and H. Farid, "Exposing digital forgeries in video by detecting double quantization," in *Proceedings of the 11th ACM workshop on Multimedia and security*, ser. MM&Sec '09. New York, NY, USA: ACM, 2009, pp. 39–48.

[12] Z. C. Lin, J. F. He, X. Tang, and C. K. Tang, "Fast, automatic and fine-grained tampered JPEG image detection via DCT coefficient analysis," *Pattern Recognition*, vol. 42, no. 11, pp. 2492–2501, Nov. 2009.

[13] ISO, "Information technology - generic coding of moving pictures and associated audio information - part 2: Video," International Organization for Standardization, Geneva, Switzerland, ISO/IEC IS 13818-2, 2008.

[14] J. Lukáš and J. Fridrich, "Estimation of primary quantization matrix in double compressed JPEG images," in *Digital Forensic Research Workshop*, 2003.