VIDEO CODEC IDENTIFICATION

P. Bestagini, A. Allam, S. Milani, M. Tagliasacchi, S. Tubaro

Dipartimento di Elettronica e Informazione - Politecnico di Milano
P.zza Leonardo da Vinci, 32 - Milano, Italy

ABSTRACT

Video content is routinely acquired and distributed in digital format. Therefore, it is customary to have the content encoded multiple times. In this paper we consider a processing chain of two coding steps and we propose a method that aims at identifying the type of codec used in the first step, by analyzing its coding-based footprints. The method relies on the fact that lossy coding is an almost idempotent operation, i.e., re-encoding the reconstructed sequence with the same codec and coding parameters produces a sequence that is highly correlated with the input one. As a consequence, it is possible to analyze this sort of correlation to identify the first codec provided that the second codec does not introduce severe quality degradation. The proposed solution finds several applications in the field of multimedia forensics, e.g. to identify the device that generated the original video stream or detect collages of different sequences.

Index Terms— Video forensics, coding-based footprints

1. INTRODUCTION

Video content is typically available in a lossy compression format. Over the last decades, several video codec architectures have been standardized with the goal of improving coding efficiency, leading to the definition of a rich set of coding tools. Some of them are included in more than one standard, whereas others are distinctive of a specific coding scheme.

Due to the lossy nature of video compression, each codec performs some non-invertible operations on the video sequence leaving peculiar coding-based footprints that can be revealed by properly analyzing the decoded video sequence. These traces might be due to either: i) normative coding tools, i.e. explicitly defined by the standard (e.g. block size, type of transform, etc.), or; ii) non-normative tools, i.e. optionally selected at the encoder/decoder, in a way that is dependent on the specific implementation (e.g. motion estimation algorithm, rate control, error concealment, etc.).

Coding-based footprints have been largely studied for digital images [1][2][3][4] whereas little has been investigated for the case of video. In [5], a deblocking strategy is used to compute the quantization parameter of MPEG-2 I-frames from quantized coefficients. The work in [6] describes the detection of double MPEG-2 compression, for the case of I-frames only. However, conventional video coding standards leverage motion-compensated prediction in order to estimate temporal redundancy. Each group of pictures (GOP) contains frames of different kind (e.g. I-, P- and B-frames), depending on the reference frames used for prediction. The GOP structure is detected in [7] based on the strength of spatial blocking artifacts. More recently, we showed how how to estimate quantization parameters and motion vectors in H.264/AVC video from decoded pixels [8].

In this paper we turn our attention to the problem of identifying the type of video codec when the input sequence is coded twice. This is a rather common scenario that arises, for example, when a video sequence is uploaded to video sharing web sites, or when it is the result of video editing. The codec type of the second coding step is readily available, as it is determined by the syntax of the bitstream. Hence, the proposed approach aims at characterizing the codec adopted in the first coding step, by determining the corresponding coding standard. The proposed method is based on recompressing the available video sequence with different codecs and coding parameters, looking for similarities between the input and output sequences of this additional coding step. Despite the simplicity of the approach, we show experimentally that identification can be performed correctly on different video sequences, especially when the second coding step does not adopt coarse quantization. A forensic analyst might exploit this piece of information to identify the device that generated the original video stream or detect video sequences that are the result of collages of different sequences.

The rest of this paper is organized as follows. Section 2 provides an overview of the main building blocks in a conventional video coding architecture, while Section 3 illustrates the proposed identification algorithm. Section 4 reports the results of experimental tests and Section 5 draws the final conclusions.

2. BACKGROUND

In a conventional video coder, each block of pixels $\mathbf{x}$ in a frame is processed according to a set of operations that can be summarized by the first line of blocks in the diagram of Fig. 1 (entropy coding is omitted since it is a reversible operation and does not leave traces). An (optional) predictor is generated by $\mathcal{P}_1$ (exploiting either spatio and/or temporal correlation). Then, prediction residuals are transformed by means of an orthonormal transform $\mathbf{T}_1$, e.g. the DCT, and scalar quantization $\mathcal{Q}_1$ is applied to each transform coefficient to obtain $\hat{\mathbf{y}}_1$. Finally, the block is reconstructed in the pixel domain by inverting the transform and adding back the predictor. In order to represent the pixel values in finite integer arithmetics, rounding might be applied. Moreover, an optional in-loop filter might be applied to remove blocking artifacts. Rounding and in-loop filtering are summarized in a single block $\mathcal{R}_1$.

Coding footprints are introduced by the non-invertible operations in Figure 1, i.e. quantization and rounding. Since the impact of the former is more pronounced, we focus on quantization-based footprints to identify the underlying codec. Each coding standard usually defines a finite set of possible quantization steps that are indirectly selected adjusting an integer-valued Quantization Parameter

The project REWIND acknowledges the financial support of the Future and Emerging Technologies (FET) programme within the Seventh Framework Programme for Research of the European Commission, under FET-Open grant number:268478.

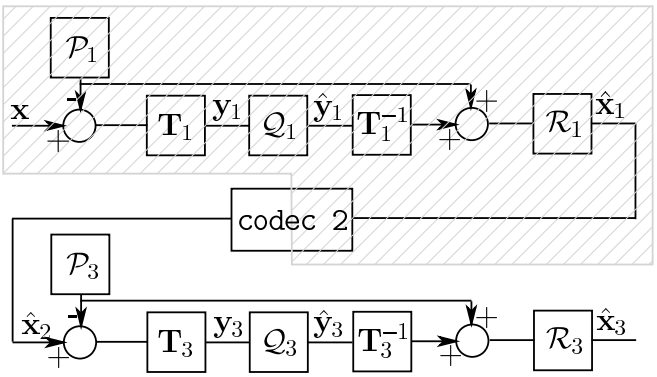

Fig. 1. A series of three coding steps.

(QP) [1]. Coding standards typically differ in the way quantization and dequantization are performed, even when they share the same underlying coding architecture. Hence, the footprint inserted by the quantization process represents a distinctive element.

3. CODEC IDENTIFICATION

We consider the setting depicted in Figure 1. A video sequence is encoded by the first codec, denoted $\mathtt{c}_1$, and later encoded by $\mathtt{c}_2$. The two codecs might be implementations of different coding architectures. For example, in a real world scenario the first coding step might be performed by the acquisition device, while the second coding step might take place in the case of transcoding, e.g., when the sequence, or part of it, is uploaded to a video sharing system or when the sequence is re-encoded after having been edited. In the following, we will refer to $\mathbf{X}$ as the original sequence, and $\hat{\mathbf{X}}_j$ as the sequence reconstructed after the j-th coding pass.

In this paper we aim at identifying the codec type being used by $\mathtt{c}_1$. Specifically, we wish to identify the adopted coding standard, assuming that the forensic analyst has access only to the reconstructed sequence $\hat{\mathbf{X}}_2$ provided by $\mathtt{c}_2$ and to the corresponding bitstream. In our experimental setting, the codec adopted by $\mathtt{c}_1$ is compliant with either MPEG-2, MPEG-4 or AVC standard. Here, $\mathtt{c}_2$ acts as a source of noise, since it might mask the coding traces of $\mathtt{c}_1$. Therefore, we also aim at studying the amount of quantization noise that can be tolerated by the proposed method. Indeed, we argue that in the case of aggressive lossy coding by $\mathtt{c}_2$ the identification of $\mathtt{c}_1$ might become very difficult, or even unfeasible.

In order to perform identification, the forensic analyst re-encodes $\hat{\mathbf{X}}_2$ with $\mathtt{c}_3$, iterating over each of the candidate codecs possibly used in $\mathtt{c}_1$ and coding parameters, and obtains $\hat{\mathbf{X}}_3$. The key observation is that lossy coding is an (almost) idempotent operation. That is, when a video sequence is re-encoded using the same coding architecture and coding parameters, the input and output sequences are alike. In our setting, when $\mathtt{c}_2$ is lossless, i.e. noise is neglected, we expect $\hat{\mathbf{X}}_3$ to be equal to $\hat{\mathbf{X}}_2$.

In order to provide an intuitive justification supporting this statement, consider a block $\hat{\mathbf{x}}_2$ taken from $\hat{\mathbf{X}}_2$. Let us assume an ideal case, in which $\mathtt{c}_2$ is lossless, i.e. $\hat{\mathbf{x}}_2 = \hat{\mathbf{x}}_1$, and an oracle informs $\mathtt{c}_3$ about the exact coding parameters used by $\mathtt{c}_1$ (e.g. coding mode, motion vectors, etc.). This way, $\mathtt{c}_3$ can form the same predictor as $\mathtt{c}_1$ and compute the same prediction residuals $\mathbf{y}_3 = \hat{\mathbf{y}}_1 = \mathcal{Q}_1(\mathbf{y}_1)$ in the transform domain. Then, $\mathtt{c}_3$ performs quantization to compute $\hat{\mathbf{y}}_3 = \mathcal{Q}_3(\mathbf{y}_3) = \mathcal{Q}_3(\hat{\mathbf{y}}_1) = \mathcal{Q}_3(\mathcal{Q}_1(\mathbf{y}_1))$. If $\mathcal{Q}_3 \equiv \mathcal{Q}_1$, due to the idempotent property of scalar quantization, $\hat{\mathbf{y}}_3 = \mathcal{Q}_1(\mathbf{y}_1) = \hat{\mathbf{y}}_1$ and, consequently, $\hat{\mathbf{x}}_3 = \hat{\mathbf{x}}_1$.

When $\mathtt{c}_3$ does not match $\mathtt{c}_1$, the use of different predictors, transform and quantizers leads to $\hat{\mathbf{x}}_3 \neq \hat{\mathbf{x}}_1$. Conversely, when a match occurs, $\hat{\mathbf{x}}_3$ and $\hat{\mathbf{x}}_1$ are not identical, although, in practice, $\hat{\mathbf{x}}_3 \simeq \hat{\mathbf{x}}_1$. This is due to the adoption of different coding options related to non-normative aspects of the standard (e.g. motion estimation, rate-distortion optimization, spatial prediction, etc.), the rounding operations, and the in-loop filtering applied on the pixels of each frame to reduce blocking artifacts.

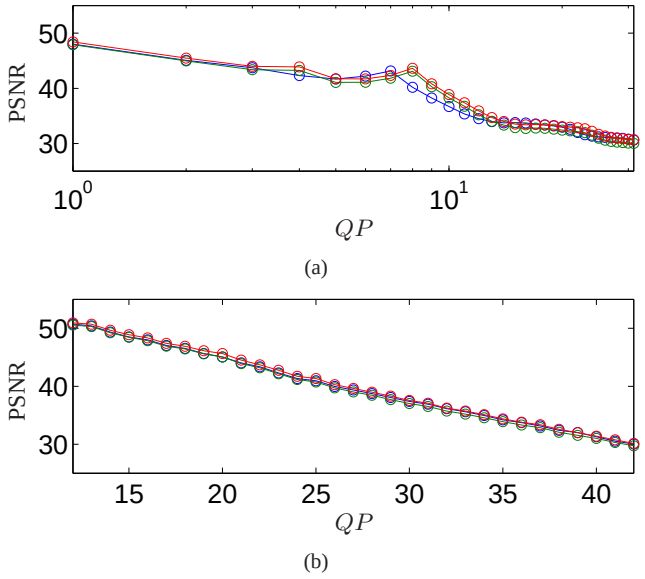

Fig. 2. An example of $\mathbf{P}^{\mathtt{c}_3}$ for three frames of the *Foreman* sequence, when $\mathtt{c}_1$ = MPEG-2, $\mathtt{c}_2$ = AVC (at $QP = 20$) and a) $\mathtt{c}_3$ = MPEG-2, b) $\mathtt{c}_3$ = AVC.

In order to find a match between $\mathtt{c}_3$ and $\mathtt{c}_1$, we re-encode $\hat{\mathbf{X}}_2$ with $\mathtt{c}_3$ at different target values of the quantization parameter QP and compute the PSNR between $\hat{\mathbf{X}}_2$ and $\hat{\mathbf{X}}_3$ for each frame. By analyzing how the PSNR varies as a function of QP, we observed the following:

- If $\mathtt{c}_3$ matches $\mathtt{c}_1$, the PSNR vs. QP function typically presents a local maximum corresponding to the QP value originally used by $\mathtt{c}_1$ to encode the frame. As an example, see Figure 2(a), where $\mathtt{c}_1 = \mathtt{c}_3 =$ MPEG-2.
- Otherwise, the PSNR vs. QP function is smooth and monotonically decreasing. When $\mathtt{c}_3$ adopts AVC, this function is approximately linear. In the case of MPEG-2 or MPEG-4, the function is linear when warped to a log-scale. See, for example, Figure 2(b), where $\mathtt{c}_1$ = MPEG-2 and $\mathtt{c}_3$ = AVC.

Starting from the considerations above, we propose the following identification algorithm.

- Let $\mathtt{c}_3 \in \{\text{MPEG-2}, \text{MPEG-4}, \text{AVC}\}$. For each target codec type $\mathtt{c}_3$, encode the sequence $\hat{\mathbf{X}}_2$ using only the intra coding mode for different values of QP (for MPEG-2 and MPEG-4 $QP \in [1, 31]$, while for AVC $QP \in [12, 43]$). Let $\hat{\mathbf{X}}_3^{\mathtt{c}_3,q}$ de-

[1] The relation between QP and quantization step q varies according to the specific standard.

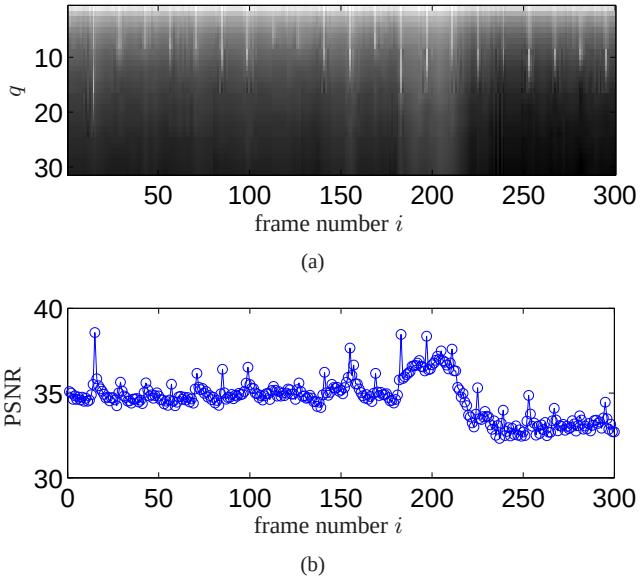

Fig. 3. An example of a) $\mathbf{P}^{\mathtt{c}_3}$ for 300 frames of the *Foreman* sequence, when $\mathtt{c}_1$ = MPEG-2, $\mathtt{c}_2$ = AVC (at $QP = 20$) and $\mathtt{c}_3$ = MPEG-2, where lighter grey indicates higher values of $PSNR(\hat{\mathbf{X}}_2, \hat{\mathbf{X}}_3^{\mathtt{c}_3,q})$; b) mean value along each column of $\mathbf{P}^{\mathtt{c}_3}$.

note the reconstructed sequence, where $q = QP$ for MPEG-2 and MPEG-4 and $q = QP - 11$ otherwise.

- Construct a $31 \times N$ matrix $\mathbf{P}^{\mathtt{c}_3}$, whose entries are computed as follow. $\mathbf{P}^{\mathtt{c}_3}(q,i) = PSNR(\hat{\mathbf{X}}_2(i), \hat{\mathbf{X}}_3^{\mathtt{c}_3,q}(i))$, i.e. the PSNR value computed comparing the i-th frame of $\hat{\mathbf{X}}_2$ and $\hat{\mathbf{X}}_3^{\mathtt{c}_3,q}$. An example of $\mathbf{P}^{\mathtt{c}_3}$ is illustrated in Figure 3(a).
- Detect the GOP size and the indexes of the frames that were originally intra-coded by $\mathtt{c}_1$. To this end, detect the peaks of the N-element sequence obtained by averaging the entries of $\mathbf{P}^{\mathtt{c}_3}$ along the columns. See Figure 3(b) for an example. Let $\mathcal{I}$ denote such a set of indexes.
- For each frame $i \in \mathcal{I}$, we consider the following model

$$\hat{\mathbf{P}}^{\mathtt{c}_3}(q,i) = \begin{cases} \alpha_i q + \beta_i & \text{if } \mathtt{c}_3 = \text{AVC;} \\ \alpha_i \log(q) + \beta_i & \text{otherwise.} \end{cases} \quad (1)$$

and we compute the normalized mean square error of the residuals. That is,

$$\mathcal{E}^{\mathtt{c}_3}(i) = \frac{\sqrt{(1/31)\sum_{q=1}^{31} |\hat{\mathbf{P}}^{\mathtt{c}_3}(q,i) - \mathbf{P}^{\mathtt{c}_3}(q,i)|^2}}{(1/31)\sum_{q=1}^{31} \mathbf{P}^{\mathtt{c}_3}(q,i)} \quad (2)$$

- The identified codec is computed by selecting the model that leads to the largest average residuals, as it indicates a deviation from the linear trend observed when $\mathtt{c}_3$ and $\mathtt{c}_1$ do not match.

$$\mathtt{c}_3^* = \underset{\mathtt{c}_3 \in \{\text{MPEG-2, MPEG-4, AVC}\}}{\text{argmax}} \sum_{i \in \mathcal{I}} \mathcal{E}^{\mathtt{c}_3}(i) \quad (3)$$

4. EXPERIMENTAL RESULTS

We tested the performance of the proposed method on a dataset of six video sequences: four at CIF spatial resolution (352×288), namely *Foreman*, *Mobile*, *Paris*, *News*; two at 4CIF spatial resolution (704×576), namely *Ice* and *Harbour*. Each original sequence was encoded with either MPEG-2, MPEG-4 or AVC. For each codec, we selected three different target bitrates by enabling rate control in order to obtain three sequences at, respectively, *low*, *medium* and *high* quality.

As for the second coding pass, we re-encoded all sequences with either MPEG-2, MPEG-4 or AVC. In order to unify the notation, the set of possible QP values for $\mathtt{c}_2$ can be identified with $\{a, b, c, d, e, f\}$, which corresponds to the set $\{1, 2, 4, 5, 7, 10\}$ for MPEG-x codecs and to the set $QP \in \{10, 20, 23, 26, 29, 32\}$ for AVC (equalizing the value of quantization steps among the codecs).

Figure 4 shows the probability of correct codec identification as a function of the codec adopted by $\mathtt{c}_2$, denoted as masking codec. The identification method is operated at the sequence level by aggregating the observations extracted from all detected intra-coded frames. These results are averaged across all tested sequences and values of QP for the second coding step.

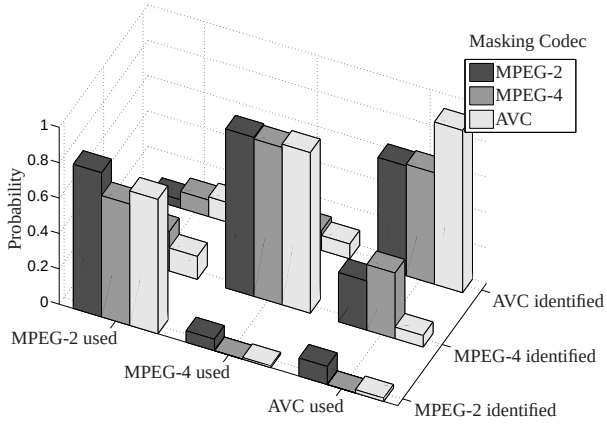

Fig. 4. Probability of codec identification averaged on all the sequences.

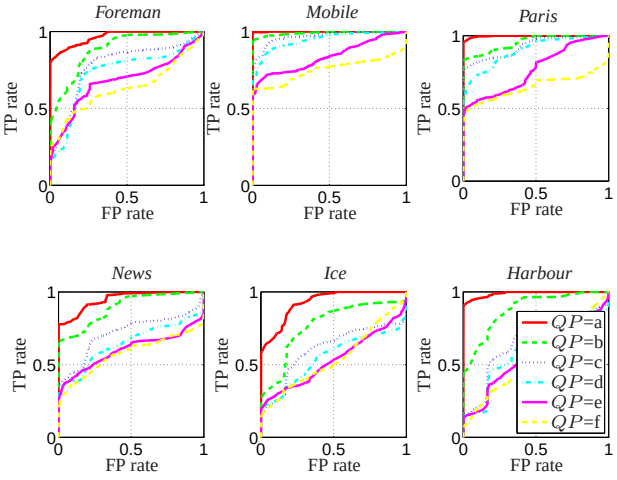

Fig. 5. Detection ROC for each sequence. Results are averaged on $\mathtt{c}_1$ and $\mathtt{c}_2$. As shown in Table 1, codec identification is sequence-dependent, although good results can be achieved for low QP values of $\mathtt{c}_2$ since the masking effect is less influential. due to the second encoder is not so evident.

Table 1. Identification accuracy.

(a)

QP ($\mathtt{c}_2$ = MPEG-2)	1	2	4	5	7	10
foreman	1.00	1.00	0.89	0.67	0.67	0.56
mobile	1.00	1.00	1.00	1.00	1.00	1.00
paris	1.00	1.00	1.00	1.00	0.56	0.56
news	1.00	1.00	0.78	0.56	0.56	0.44
ice	1.00	0.89	0.56	0.44	0.56	0.33
harbour	1.00	0.89	0.67	0.56	0.56	0.33

(b)

QP ($\mathtt{c}_2$ = MPEG-4)	1	2	4	5	7	10
foreman	1.00	0.89	0.67	0.67	0.56	0.56
mobile	1.00	1.00	1.00	1.00	1.00	0.89
paris	1.00	1.00	0.89	0.78	0.56	0.44
news	1.00	1.00	0.67	0.56	0.56	0.44
ice	1.00	0.78	0.44	0.44	0.33	0.33
harbour	1.00	0.89	0.56	0.56	0.56	0.44

(c)

QP ($\mathtt{c}_2$ = AVC)	10	20	23	26	29	32
foreman	1.00	0.89	0.89	0.78	0.78	0.56
mobile	1.00	1.00	1.00	1.00	0.67	0.67
paris	1.00	1.00	1.00	1.00	0.89	0.78
news	1.00	0.89	0.78	0.78	0.67	0.56
ice	1.00	1.00	1.00	0.78	0.78	0.56
harbour	1.00	1.00	0.89	0.89	0.89	0.56

In order to analyze the masking effect further, Table 1 shows the accuracy obtained at different values of QP of the second coding step, i.e. the fraction of correct identifications of the first coding step. In nearly lossless conditions (low QP) the proposed method successfully identifies the first codec in all cases. Notice that the influence of lossy compression on the effectiveness of the proposed identification algorithm is content-dependent. Indeed, for *Mobile* and *Paris*, accuracy remains at 1.0 also at higher values of QP, whereas for *Foreman* the method might fail when QP is moderately increased. This is due to the fact that, in the latter case, most of the coefficients are quantized to zero, due to the presence of relatively smooth textures.

Finally, we tested the performance of the identification algorithm when it is applied on a frame-by-frame basis on detected intra-coded frames. To this respect, we show the receiver-operating-characteristic (ROC) curves obtained at different values of QP for the second coding step. Let τ denote a threshold value. The proposed method flags a frame i as encoded with $\mathtt{c}_3$ whenever $\mathcal{E}^{\mathtt{c}_3}(i) > \tau$. The true positive rate is the fraction of frames originally encoded with $\mathtt{c}_1$ for which $\mathcal{E}^{\mathtt{c}_3}(i) > \tau$. Conversely, the false positive rate is the fraction of frames not encoded with $\mathtt{c}_1$ for which $\mathcal{E}^{\mathtt{c}_3}(i) > \tau$. ROC curves are traced by varying the value of τ. Figure 6 shows the ROC curves for each masking codec $\mathtt{c}_2$, averaging results across all sequences and codecs $\mathtt{c}_1$. This allows us to study the impact of the masking codec in terms of identification accuracy. We notice that, at approximately the same quality level, AVC is a stronger masker than MPEG-2 and MPEG-4. This is due to the presence of deblocking filter, which conceals parts of the traces left by quantization, especially for high values of QP. In order to study the dependency on the video content, Figure 5 shows individual ROC curves for each sequence, this time averaging results across both $\mathtt{c}_1$ and $\mathtt{c}_2$. These charts confirm that codec identification is content-dependent, as already observed analyzing the results in Table 1.

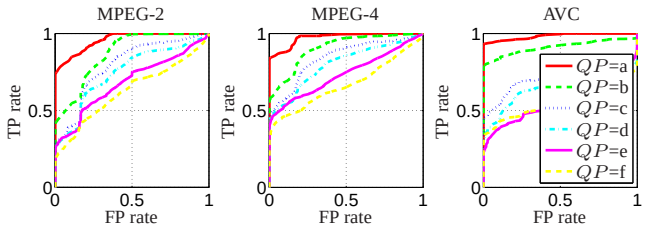

Fig. 6. Detection ROC for each codec $\mathtt{c}_2$. Results are averaged on sequences and $\mathtt{c}_1$. Note the dependency with respect to $\mathtt{c}_2$. When AVC is used, low QPs give better performances than MPEG-2 or MPEG-4, while increasing the QP the results are the opposite.

5. DISCUSSION

In this paper we propose an algorithm that is able to identify the coding standard used to lossy compress a video sequence. Although the preliminary results are promising, there are several issues that need to be faced and stimulate future research work. First, we considered a closed-group setting where the different codec implementations are known and can be enumerated. As a matter of fact, the proposed strategy needs to be extended to an open-group scenario. Second, the current version of the method did not exploit the available knowledge on the second coding step, which acts as masker, nor the properties of the video content. Third, the experimental validation needs to be extended including additional codec types for the second coding step (e.g. codecs employed by video sharing sites like YouTube).

6. REFERENCES

[1] Zhigang Fan and Ricardo L. de Queiroz, "Identification of bitmap compression history: JPEG detection and quantizer estimation," *IEEE Transactions on Image Processing*, vol. 12, no. 2, pp. 230–235, 2003.

[2] Jan Lukas and Jessica Fridrich, "Estimation of primary quantization matrix in double compressed JPEG images," in *Digital Forensic Research Workshop*, Cleveland, USA, 2003.

[3] Steven K. Tjoa, Wan-Yi Sabrina Lin, H. Vicky Zhao, and K. J. Ray Liu, "Block size forensic analysis in digital images," in *ICASSP (1)*, 2007, pp. 633–636.

[4] Tiziano Bianchi, Alessia De Rosa, and Alessandro Piva, "Improved DCT coefficient analysis for forgery localization in JPEG images," in *ICASSP (1)*, Prague, Czech Republic, 2011.

[5] Huiying Li and Søren Forchhammer, "MPEG2 video parameter and no reference PSNR estimation," in *Proceedings of the 27th conference on Picture Coding Symposium*, Piscataway, NJ, USA, 2009, PCS'09, pp. 149–152, IEEE Press.

[6] Weihong Wang and Hany Farid, "Exposing digital forgeries in video by detecting double quantization," in *Media Forensics and Security*, 2009, pp. 39–48.

[7] Weiqi Luo, Min Wu, and Jiwu Huang, "MPEG recompression detection based on block artifacts," in *SPIE Conference on Security, Forensics, Steganography, and Watermarking of Multimedia Contents*, San Jose, USA, 2008.

[8] Giuseppe Valenzise, Marco Tagliasacchi, and Stefano Tubaro, "Estimating QP and motion vectors in H.264/AVC video from decoded pixels," in *ACM Workshop on Multimedia in Forensics, Security and Intelligence*, Florence, Italy, 2010.