# Transmitting What Matters: Task-oriented video composition and compression

Fernanda A. Andaló, Otávio A. B. Penatti, Vanessa Testoni
Samsung Research Institute
Campinas, Brazil
E-mail: feandalo@ic.unicamp.br, {o.penatti,vanessa.t}@samsung.com

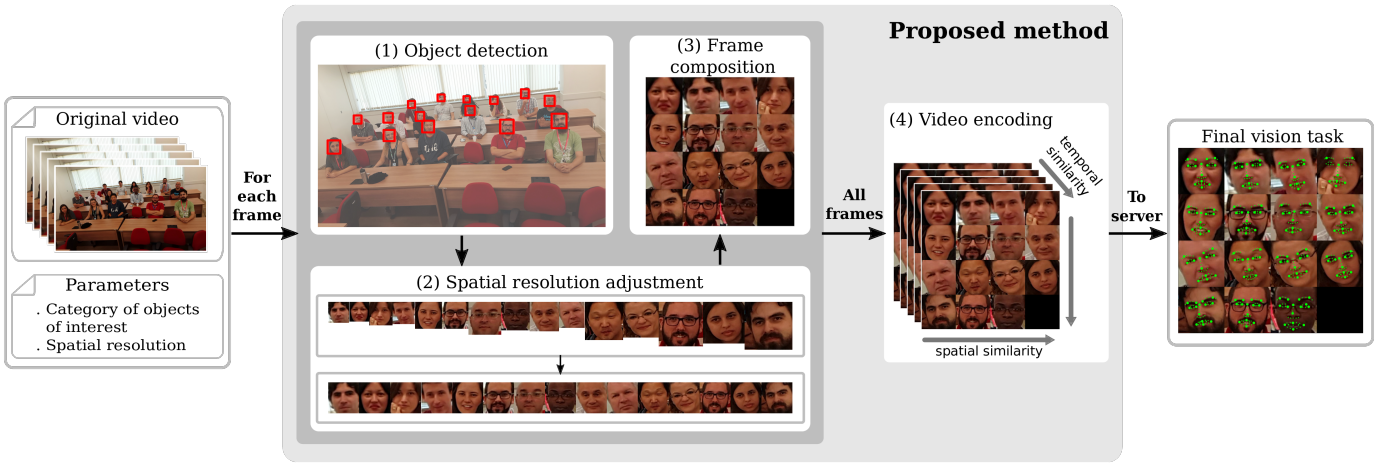


Fig. 1. Proposed framework: creating compressed videos taking into account a target computer vision task. For each frame of the input video, steps (1) Object detection, (2) Spatial resolution adjustment, and (3) Frame composition are performed. All created frames are encoded in a compressed video at step (4) Video encoding. Finally, the compressed video can be transferred to a server where the final computer vision task takes place.

***Abstract*—We present a simple yet effective framework – *Transmitting What Matters* (TWM) – to generate compressed videos containing only relevant objects targeted to specific computer vision tasks, such as faces for the task of face expression recognition, license plates for the task of optical character recognition, among others. TWM takes advantage of the final desired computer vision task to compose video frames only with the necessary data. The video frames are compressed and can be stored or transmitted to powerful servers where extensive and time-consuming tasks can be performed. We experimentally present the trade-offs between distortion and bitrate for a wide range of compression levels, and the impact generated by compression artifacts on the accuracy of the desired vision task. We show that, for one selected computer vision task, it is possible to dramatically reduce the amount of required data to be stored or transmitted, without compromising accuracy.**

***Keywords*-frame composition; object detection and tracking; video compression; computer vision applications;**

## I. Introduction

The volume of Internet traffic is constantly growing. By 2018, it will reach 1 Pbps in the busiest hour, 79% of which will be video data [1]. In such conditions, it is important to develop solutions that can reduce the amount of video to be transferred, with the additional benefit of requiring less storage.

When concerning high-resolution images and videos for computer vision tasks, simply reducing their spatial or temporal resolution would be a straightforward solution, but it is not always an option since low resolution data make most computer vision techniques much less precise.

Consider a classroom scenario in which students' faces are recorded and transmitted to an external server which performs face recognition. Faces recorded at 5 to 10 meters away from the camera with high resolution of 1080p contain around 65 to 30 pixels horizontally, i.e., critically close to the lowest resolution required by current face identification applications [2]. Therefore, at this minimum required resolution, an already compressed video of a class would require gigabytes of storage space. Taking into account that multiple classes may be recorded daily and simultaneously, the school would need to store and transmit an incredible amount of information. Naturally, this huge amount of generated video information is not only a problem in the school scenario, but also in other systems that also require high-resolution videos, like surveillance systems, sports, etc.

In this paper, we propose an interesting alternative framework, named *Transmiting What Matters* (TWM), that saves storage and still keeps enough resolution, by composing videos with only relevant data for the considered computer vision task. The solution creates an interesting opportunity for com-

pression while keeping enough resolution for the final tasks. Besides, there is a double gain, one related to the content generation and another to the optimized compression.

In addition to the proposed framework, another contribution is the evaluation of the impact in terms of: compression rate, by computing how much we can reduce the videos to be transmitted and stored; accuracy of the final computer vision task, by analyzing the compression trade-offs.

For evaluation purposes, we consider the school scenario, in which videos of a classroom are recorded and the facial features of students need to be detected. We show that we can obtain up to four times of reduction in the amount of data to be transferred and stored. At the same time, we show that despite the high compression in the data, the facial features can still be effectively detected in the decoded video.

The remainder of the paper is the following. Section II presents related work. In Section III, we detail the proposed framework. Section IV presents the experiments and obtained results. Finally, Section V concludes the paper and shows opportunities for future work.

## II. Related work

Current solutions to compress videos and reduce bandwidth usage do not address the entire process of optimized creation and compression depending on the desired final task.

In the literature, there are basically two related solutions: *Tiled streaming* [3], [4], [5] and *Region-of-Interest (RoI) video encoding* [6], [7], [8], [9]. Other methods focus on video codecs [10] or protocols [11] to perform adaptive transmission.

*Tiled streaming* methods encode a video sequence by dividing frames into a grid of independent tiles which are scalably encoded and stored. This content can then be streamed with a spatial or quality resolution compatible with the available bandwidth. A lower resolution version of the sequence can be initially transmitted until a user selects a region of interest, by zooming-in. After that, only the additional bits for representing the tiles covering the selected RoI in higher resolution are transferred. Therefore, these methods potentially reduce bandwidth consumption but do not reduce storage requirements due to the higher size of the scalably coded file.

In *RoI video encoding* methods, foreground-background identification is conducted so that background regions are more compressed at the encoding step. Even though the background is highly compressed, these methods still need to encode it, since the whole frames have to be reconstructed in the typical application scenarios. *RoI video encoding* is also included in perceptual video coding [12], which is a rising research area that includes perceptual properties of the human visual system in the coding models and implementations.

We propose a framework to transmit only what matters, saving storage, bandwidth and still keeping enough resolution for performing complex computer vision tasks at the receiver side.

TWM is not a new codec or protocol, but rather is a scheme that enables current codecs to produce more effective results considering a desired task. In this sense, to the best of our knowledge, our framework is not comparable with any other in the literature.

## III. Transmitting what matters

In this section, we detail the proposed framework TWM which generates compressed videos containing only the objects of interest[1]. Our explanation and examples are based on the school scenario, in which the final task is to detect facial features of students recorded in a classroom. However, the framework is general enough to work in many other scenarios, like face recognition in surveillance systems, visual analysis of athletes in sports systems, crop identification or plague analysis in agricultural systems, license plate recognition in traffic surveillance systems, and others.

The pipeline illustrated in Figure 1 summarizes the framework. The system receives as input a digital video as well as parameters that inform the category of objects of interest and the desired spatial resolution for these objects. Based on the provided data, **for each frame** of the input video, TWM:

(1) detects and extracts the objects of interest, considering the informed category (faces, for example);
(2) adjusts the spatial resolution of the extracted objects according to the resolution parameter;
(3) composes a final frame with the extracted and adjusted objects grouped spatially in a grid;

Finally, the new frames are joined and encoded with a state-of-the-art video codec which benefits from the visual similarities and local correlations, both spatially in each frame and temporally across several frames. These visual similarities considerably improve the effectiveness of the video codec, consequently increasing the compression capacity.

After the final video encoding step, the generated compressed video can be transmitted to a server where the computer vision task (in our example, facial feature detection) will take place. If there is more than one category of objects (for instance, faces and hands), the whole process is repeated for each category, therefore generating several compressed videos.

In the following subsections, we detail each step of the proposed framework.

### A. Object detection

The object detection step receives as input the original video and parameters specifying the category of the objects of interest (face, for instance). Object detection is then performed considering the category informed as parameter. One can implement this step using specific pre-trained object detectors, like face detectors [13] or other schemes for detecting different types of objects [14], [15], [16]. Other possibilities include the use of distinctive local features [17], [18], [19] or even more general descriptors [20], [21].

In our example, where the objects of interest are faces, we used the OpenCV [22] implementation of the Viola and Jones face detector [13] improved by [23]. It is a classic method

[1]TWM is patent pending under the application number US 14/663,637 filed on March 20, 2015.

which employs a cascade of boosted trained classifiers to search for the object of interest, at different sizes, in an image.

At this step, objects can also be tracked across frames to help the frame composition step. Depending on the category, a myriad of tracking algorithms can be used [24], [25]. We employed a simple tracking procedure, by matching detected objects using the absolute difference norm between objects in the previous frame and the actual frame.

### *B. Spatial resolution adjustment*

This step receives as input the objects of interest represented as image tiles already cropped from the original frame and the spatial resolution informed as parameter.

All the image tiles are adjusted in order to be represented according to the spatial resolution parameter. The target spatial resolution needs to be selected according to the final desired vision task, because different tasks often require a specific minimum image resolution for accurateness.

Naturally, if the object's current resolution is lower than the desired resolution, an up-sampling process is performed. Otherwise, a down-sampling process is performed.

To adjust the spatial resolution of the detected objects, we used cubic interpolation for the up-sampling process, and area interpolation for the down-sampling process.

### *C. Frame composition*

For each input video frame, the tiles with the detected objects (already spatially adjusted by the previous step) are organized in a grid. The grid can have different forms, like a single row or a single column, for instance. However, for better exploiting the video codec capabilities of taking advantage of both horizontally and vertically local correlations, the grid should be configured as a rectangle or square.

One possibility to determine the grid configuration (width and height) is to consider the information of the maximum number of objects of interest that could be possibly detected in the video. For example, in a classroom, one may know beforehand the maximum number of students. Therefore, the grid could be configured as a square with size equal to the square root of the maximum number of objects of interest.

By using the tracking information generated by the object detection step, TWM places the same object at the same grid position at all frames, in order to obtain even higher compression rates in the final video. Without the tracking information, each object can be freely placed in any position in the grid at the cost of less compression in the final video.

### *D. Video encoding*

Finally, in the video encoding step, the generated frames are joined and encoded with a video codec which benefits from the visual similarities and local correlations, both spatially in each frame and temporally across several frames.

Current codecs, such as the popular H.264/MPEG-4 Advanced Video Coding (AVC) [26] or the current state-of-the-art High Efficiency Video Coding (HEVC) [27], employ very efficient intrapicture and interpicture prediction methods as well as advanced motion estimation and compensation techniques [28], [29]. Therefore, they are able to exploit all the high spatial and temporal correlations introduced in the generated videos where the frames contain similar objects placed in similar positions.

On top of that, codecs enable several configuration parameters that can be optimized according to the final main pursued compression result. If it is more important to ensure that a fixed number of frames, named a group of pictures (GOP), will be encoded with approximately the same bitrate (despite of the actual content of each GOP), the codec parameters can be chosen to target a final specific bitrate through its inherent rate control mode. On the other hand, if the final application requires an approximately constant quality for all the decoded frames, then the codec parameters can be set to reach a fixed quality at the cost of a final variable bitrate. The main codec parameter that varies and is responsible for each compression result is the quantization parameter (QP).

It is key to ensure that all the decoded frames will achieve a minimum quality required to the satisfactory performance of the selected computer vision task. Therefore, each object (or pixel area, since the codec does not perform object recognition) will be encoded with a QP optimally chosen by the codec. For instance, a high-resolution object extracted from the original video has a lot of information that will be already discarded by the down-sampling operation performed in the previous spatial resolution adjustment step. However, even after the down-sampling operation, this object will probably present more details than another low-resolution object extracted from the original video that was up-sampled by the spatial resolution adjustment step. The up-sampled object will probably be blurred and cannot afford to be highly quantized in the encoding step, missing even more information.

Both previously mentioned codecs (H.264/MPEG-4 Advanced Video Coding and the state-of-the-art High Efficiency Video Coding) have internal mechanisms that work to effectively encode videos that contain regions with different qualities, such as the videos produced by the spatial resolution adjustment step of TWM. Therefore, on the average, a higher compression level, or a higher QP, will be applied to previously down-sampled objects while and a lower compression level, or a lower QP, will be applied to previously up-sampled objects.

It is important to note that a system that does not employ our framework would still have to encode the original high-resolution video. The amount of raw 4K UHD video generated in the considered scenario, and in other similar scenarios, easily reaches the terabyte range for only one 30-minute recorded video. Encoding such 4K UHD videos with current state-of-the-art video codecs requires a considerable amount of time [30], even when optimized implementations of the codecs are used.

Therefore, by reducing the total amount of data to be encoded and by creating a much more compressible final video content, we not only save bandwidth and storage, but also dramatically reduce the video encoding processing time.

## IV. Experiments and Results

The experiments, which were conducted on a captured video dataset (Section IV-A), verified how much video compression can be obtained with TWM (Section IV-B), and how the introduced compression artifacts affect the final computer vision task (Section IV-C).

### A. Video sequences

Our illustrative scenario is the classroom and the selected computer vision task is the detection of students facial features. Since there are no standard 1080p HD or 4K UHD video datasets with the required characteristics, we captured a dataset of 15 video sequences in 4K UHD resolution ($3840 \times 2160$ pixels) at 30 fps. All sequences are progressively scanned and use the YUV 4:2:0 color format with 8 bit per color sample. Each video sequence has 420 frames and was acquired in an environment simulating a classroom, with the camera in front of the room recording the scenario.

Each one of the 15 original UHD video sequences is identified by $full_i$, where $i$ is the video sequence number. The notation $full_i^{BR}$ identifies each video sequence compressed with fixed bitrate (BR). Similarly, $full_i^{QP}$ identifies each video sequence compressed with fixed QP.

By applying the initial three steps of our framework and prior to the video encoding step, the raw videos containing only the faces of the students are identified by $face_i$. After the video encoding step, the corresponding compressed videos are identified by $face_i^{BR}$ when compressed using fixed bitrate, or by $face_i^{QP}$ when using fixed QP.

When composing the $face_i$ videos, TWM can benefit from the tracking algorithm. These sequences are identified by the "+trk" string after the video sequence number.

For reporting results in the following subsections, we selected two typical video sequences from our dataset numbered as 14 and 15. These two sequences correspond to different moments in the class after students changed places.

### B. Video encoding results

We selected the HEVC as the most suitable video codec because it is shown to be especially effective for low bitrates and high-resolution video content [31]. The official HEVC HM 16.4 test model software [32] was used. For the HM encoder parameter definition, we employed the common test conditions and software reference configurations officially recommended by the Joint Collaborative Team on Video Coding [33]. Therefore, we selected the main profile (MP) and the random access (RA) mode, with random access points about every 1 second. Since our sequences were recorded at 30 fps, the intra refresh period was defined as 32. And since the random access mode was selected, we used the dyadic hierarchical prediction structure with groups of 8 frames, where all frames are coded as B frames except at the random access refresh points (where I frames are used). Because we are encoding high-resolution video sequences, we chose the maximum coding unit size of 64 samples.

For the average fixed bitrate mode, HEVC internally employs its Rate-Distortion Optimized Quantization (RDOQ) method [34]. For fixed QP, the quantization parameter was set only for the I frame and internally increased according to the hierarchy level of each subsequent B frame. The QP for I frames varied from 22 to 38. Since we have groups of 8 frames, there are 4 hierarchy levels for the B frames. The quantization step size is increased by about 12% from one hierarchy level to the next, and the quantization step size for the B frames of the lowest hierarchy level is increased by 12% relative to that of the I frames.

Each one of the above configurations was optimally defined according to the purpose of our framework, which is achieving the highest compression in UHD videos while keeping enough resolution for performing complex vision tasks.

Video encoding results are shown through distortion versus bitrate curves and visual quality comparisons. We show the rate-distortion curves of the combined luminance and chrominance components. The combined Peak Signal-to-Noise Ratio (PSNR) is computed as

$$\text{YUV-PSNR} = (6.\text{PSNR}_Y + \text{PSNR}_U + \text{PSNR}_V)/8 \qquad (1)$$

where $\text{PSNR}_Y$, $\text{PSNR}_U$, and $\text{PSNR}_V$ are each computed as

$$\text{PSNR} = 10 \log_{10} \frac{(2^B - 1)^2}{\text{MSE}} \qquad (2)$$

where B = 8 is the number of bits per sample of the video signal to be encoded and the Mean Squared Error (MSE) is the Sum of Squared Differences (SSD) divided by the number of samples in the signal. The PSNR measurements per video sequence are computed by averaging the per-frame measurements.

Figure 2 presents YUV-PSNR $\times$ bitrate curves. For fixed QP mode, the QP for the I frames was varied in the range from 22 to 38. For fixed bitrate mode, each sequence was encoded at 13 different bitrates.

The first interesting result shown in Figure 2 regards the encoding performance for the original 4K $full_{14}$ and $full_{15}$ sequences. As reported in [35], HEVC reaches around 3 Mbps when encoding 4K sequences with good quality. Since the content of our dataset is highly compressible, once it consists of the static classroom background and seated students watching the class with a limited range of motion, HEVC impressively reaches less than 1 Mbps while keeping more than 40 dB. Besides, also due to the nature of our video sequences, the encoding performance with fixed QP (sequences $full_{14}^{QP}$ and $full_{15}^{QP}$) is always better than the encoding performance with fixed bitrate (sequences $full_{14}^{BR}$ and $full_{15}^{BR}$). The gains obtained with fixed QP are more perceptible at the challenging low bitrate scenario (below 450 kbps), where the quality difference between the two modes achieves around 2 dB for similar bitrates.

The second worth mentioning point regards the significant gains obtained with the tracking algorithm in the encoding

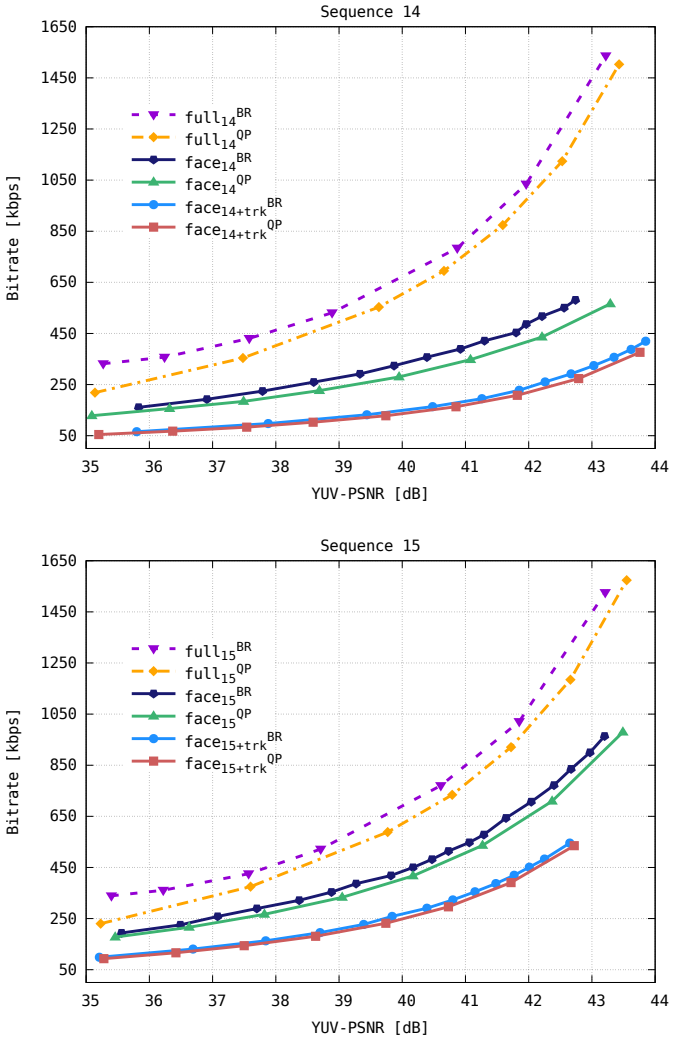


Fig. 2. YUV-PSNR [dB] $\times$ bitrate [kbps] curves for two video sequences. The lower the curve, the better. The best result is reached by TWM when fixed QP and tracking are both employed. This result is shown in red as $face_{i+trk}^{QP}$. In order to better correlate these results with the ones presented in Figure 4, the axes are inverted in relation to the most common usage in the literature.

of the face videos. These gains were expected because, even tough faces in general are already similar, placing the same faces in similar positions through the video frames adds even more similarities and temporal correlations which are properly exploited with the optimized configuration of the HEVC. It can be noted in Figure 2 that, for similar bitrates, the quality gains obtained with the tracking for $face_{15+trk}$ sequence are up to 3.5 dB, and for $face_{14+trk}$ sequence the same gains are up to 5.5 dB.

The differences between the two sequences are due to the fact that $face_{15}$ sequence has, on the average, more information than $face_{14}$ sequence, since more faces are detected in the former by the object detection step. This is also the reason why, for similar YUV-PSNR qualities, the bitrates achieved for $face_{15}$ sequence are always higher than the bitrates achieved for $face_{14}$ sequence. Even though the faces videos are not 4K and are comprised by only faces, they still contain a considerable amount of data to be compressed since each face is represented as a $128 \times 128$ square and the whole frames, with at most 15 faces, have a resolution of $512 \times 512$ pixels.

In Section IV-C it will be shown that YUV-PSNR qualities above 41 dB are sufficient to effectively perform the selected vision task. As can be seen in Figure 2, the gains obtained with our framework are increasingly higher in the quality range above 40 dB. For sequence 14 at 42 dB, for instance, the bitrate can be reduced 4 times, from around 1 Mbps for $full_{14}^{QP}$ to 250 kbps for $face_{14+trk}^{QP}$, when compared to the baseline where the full video is encoded. At the same quality of 42 dB for sequence 15, the bitrate can be reduced 2.2 times, from around 1 Mbps for $full_{15}^{QP}$ to 450 kbps for $face_{15+trk}^{QP}$.

A visual quality comparison is presented in Figure 3. Frames 107 and 33 extracted from sequences $face_{14+trk}$ and $face_{15+trk}$ are shown, respectively, in Figure 3(a) and in Figure 3(d).

It is not expected that all the faces will show the same quality in the original video due to the effect of the spatial resolution adjustment step. The students in the back of the classroom were captured in small resolutions and their faces had to be up-sampled, which generated the blurred effect. The faces of the students in front of the classroom (closer to the camera) were captured in high resolutions and, even after being down-sampled, were still preserving enough details.

Frame 107 of sequence $face_{14+trk}$ is shown encoded at 41.8 dB in Figure 3(b) and at 35.2 dB in Figure 3(c). As expected from the PSNR $\times$ bitrate curves of the sequence $face_{14+trk}^{QP}$, the faces in Figure 3(b) show a very good visual quality comparable to the original faces in Figure 3(a).

One can see that the quality degradation is already noticeable in most of the faces in Figure 3(c) and also that this degradation is not uniform. Some faces that were full of details in Figure 3(b) are even more blurred in Figure 3(c) than other faces that were already blurred in Figure 3(b) because the last ones were previously up-sampled and could not afford missing information. This result shows that the bitrate is being appropriately distributed by the codec through the frame and is a key result of our proposed framework, since it is important to maintain a minimum quality for all the faces for the final computer vision task.

The same observations can be made for sequence $face_{15+trk}$, whose frame 33 is shown encoded in similar qualities, at 41.7 dB in Figure 3(e) and at 35.3 dB in Figure 3(f). As previously explained, for similar YUV-PSNR qualities, the bitrates achieved for $face_{15}$ sequence are always higher because more faces were detected, on the average, for this sequence.

We opted for not reporting the processing times because we are using the official HEVC HM 16.4 reference software, which is not hardware optimized. However, the encoding of a 4K video sequence, even with a highly compressible content, is a complex task. Since with TWM, instead of encoding a 4K video sequence, it is necessary to only encode

(a) Frame 107 from original $face_{14+trk}$ (b) $face^{QP}_{14+trk}$ with 41.8 dB at 208 kbps (c) $face^{QP}_{14+trk}$ with 35.2 dB at 54 kbps

(d) Frame 33 from original $face_{15+trk}$ (e) $face^{QP}_{15+trk}$ with 41.7 dB at 391 kbps (c) $face^{QP}_{15+trk}$ with 35.3 dB at 93 kbps

Fig. 3. Visual quality comparison results for sequences $face_{14+trk}$ and $face_{15+trk}$, encoded with different YUV-PSNR qualities.

a $512 \times 512$ resolution video, there is a significant gain in processing time. The processing times required for performing the other steps (object detection, spatial resolution adjustment and frame composition) are negligible when compared to the video encoding step times.

### C. Facial feature detection results

To analyze the impact of the proposed framework on tasks performed after the decoding of the generated videos, we chose the particular and interesting task of facial feature detection, which refers to the detection of keypoints in faces. These keypoints can be applied, for instance, to aid the analysis of facial expressions [36].

In order to detect the facial features, we employed the Intraface method [37], which uses Supervised Descent Method for aligning a face model consisting of 49 face landmarks.

We compared the displacement of facial features detected by the Intraface method in several versions of the videos, considering $i = 1, \ldots, 15$:

- before the application of TWM ($full_i$);
- after the three initial steps of TWM, but prior to the encoding step ($face_{i+trk}$);
- after the encoding step with fixed QP ($face^{QP}_{i+trk}$) and different compression levels.

Note that we are not considering the video sequences compressed with fixed bitrate nor the ones without the tracking procedure, because of the better results achieved with fixed QP and tracking, as shown in Section IV-B.

The mean displacement error is computed as the sum of per-feature displacements ($L_1$-norm) for the same face between correspondent frames of two different sequences, divided by the total number of 49 facial features. Then, it is normalized by the number of faces in the frames and by the number of total frames in the considered sequence.

The computed mean displacement error between features detected on the original high-resolution $full_i$ videos and the $face_i$ sequences before encoding are negligible, being less than 1 pixel and standard deviation of 0.1 pixel. This small difference is due mainly to round-off errors caused by the up and down-sampling processes, which consider real numbers as scale factors.

Figure 4 presents YUV-PSNR $\times$ displacement $\times$ bitrate curves for sequences 14 and 15. They show mean displacement errors, and standard deviations, between facial features detected in the raw sequences $face_{i+trk}$ and in the respective compressed sequences $face^{QP}_{i+trk}$, considering different compression levels.

For YUV-PSNR values greater than 41 dB, the displacement error for the considered task is lower than 1 pixel, and the detection displacement for both sequences yields almost the same behavior. However, for lower reconstruction qualities, errors for sequence 14 are higher than for sequence 15, including higher standard deviations, because of the different content of the sequences. This is also corroborated by the previous results presented in Figure 2 and by the achieved bitrates for the same YUV-PSNR quality in Figure 4, which are lower for sequence 14 than for sequence 15.

Figure 5 shows examples of facial features in two frames of sequences $face_{14+trk}$, $face^{QP}_{14+trk}$, $face_{15+trk}$, and $face^{QP}_{15+trk}$. One can observe larger displacements in features detected in frames with lower YUV-PSNR, and an almost perfect detection at qualities higher than 41 dB.

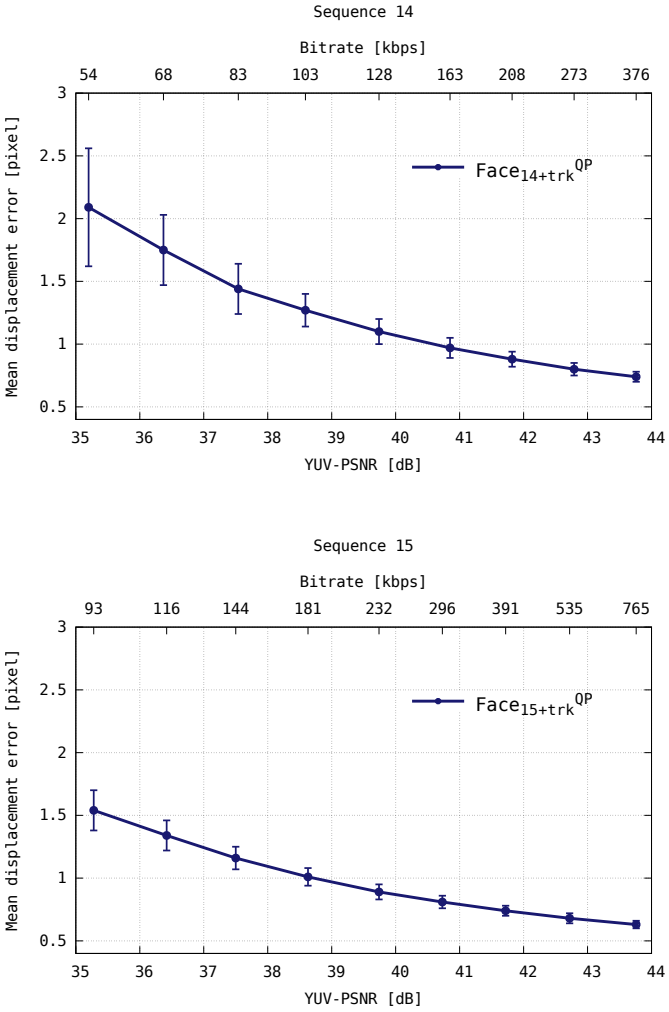


Fig. 4. Facial features displacement results (YUV-PSNR [dB] $\times$ mean displacement error [pixel] $\times$ bitrate [kbps]). Errors were computed between raw sequence $face_{i+trk}$ and compressed sequence $face^{QP}_{i+trk}$, considering several compression levels. Error bars indicate the standard deviation of the computed errors. Note that above quality 41dB, errors are below 1 pixel for both video sequences.

## V. Conclusions

We presented a framework, named *Transmitting What Matters* (TWM), for generating compressed videos aiming at a computer vision task. TWM creates videos containing only the information of interest for the desired task.

The solution is specially relevant in the yet challenging, but increasingly common, scenarios that require the transmission and storage of UHD videos and where the simple reduction of spatial or temporal resolutions is not acceptable due to computer vision requirements.

Experimental results using 4K UHD videos and HEVC showed that TWM is very effective for video compression without harming accuracy. The bitrate was reduced up to four times while the detection of facial features was affected by only $\sim 1$ pixel.

It is also safe to affirm that with TWM there is a significant gain in processing time, since it is only necessary to encode a much smaller resolution video, instead of a 4K video sequence.

We envision opportunities for future work by considering perceptual encoding and by evaluating TWM with other computer vision tasks (license plate recognition, for instance) and new datasets related to new scenarios.

## References


[1] "Cisco Visual Networking Index: Forecast and Methodology, 2013–2018," http://www.cisco.com/c/en/us/solutions/collateral/service-provider/ip-ngn-ip-next-generation-network/white_paper_c11-481360.html, CISCO, Tech. Rep., 2014.

[2] Axis Communications, "Pixel density," (as of April 2, 2015). [Online]. Available: http://www.axis.com/academy/pixel_count/pixel_density.htm

[3] R. Guntur, A. Shafiei, W. T. Ooi, and Q. M. K. Ngo, "System and method for enabling user control of live video stream(s)," 2014, wO Patent App. PCT/SG2013/000,341. [Online]. Available: https://www.google.com.br/patents/WO2014025319A1?cl=en

[4] N. Q. M. Khiem, G. Ravindra, A. Carlier, and W. T. Ooi, "Supporting zoomable video streams with dynamic region-of-interest cropping," in *ACM Conference on Multimedia Systems*, 2010, pp. 259–270.

[5] N. Q. M. Khiem, G. Ravindra, and W. T. Ooi, "Adaptive encoding of zoomable video streams based on user access pattern," *Signal Processing: Image Communication*, vol. 27, no. 4, pp. 360–377, 2012.

[6] C. Bulla, C. Feldmann, and M. Schink, "Region of interest encoding in video conference systems," in *International Conferences on Advances in Multimedia*, 2013, pp. 119–124.

[7] C. Rhodes, "Systems and methods for adaptive transmission of data," 2012, uS Patent 8,184,069. [Online]. Available: https://www.google.com.br/patents/US8184069

[8] M. Xu, X. Deng, S. Li, and Z. Wang, "Region-of-interest based conversational HEVC coding with hierarchical perception model of face," *IEEE Journal of Selected Topics in Signal Processing*, vol. 8, no. 3, pp. 475–489, June 2014.

[9] D. Grois and O. Hadar, "Region-of-interest: Processing and coding techniques," *Intelligent Multimedia Technologies for Networking Applications: Techniques and Tools*, pp. 126–155, 2013.

[10] D. DeForest, N. Lee, R. Pizzorni, and C. P. Pace, "Context based video encoding and decoding," 2013, uS Patent App. 13/725,940. [Online]. Available: https://www.google.com.br/patents/US20130107948

[11] A. Balk, M. Gerla, D. Maggiorini, and M. Sanadidi, "Adaptive video

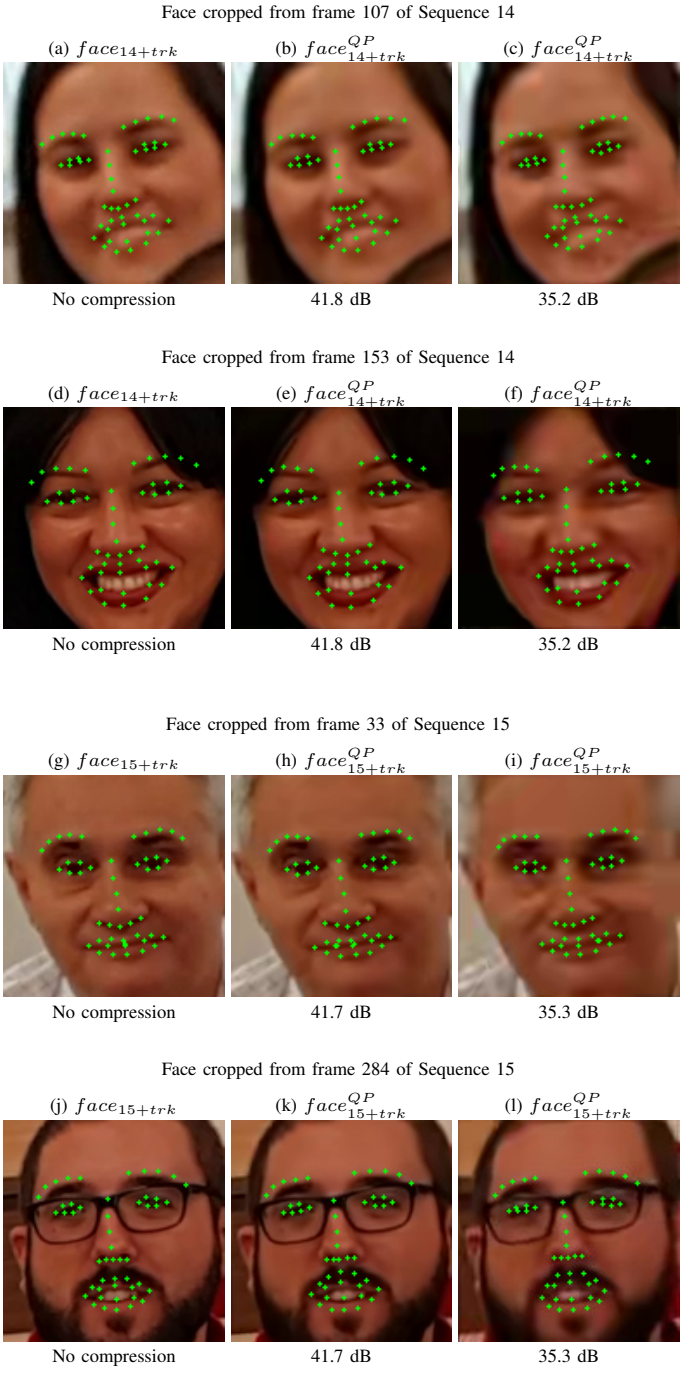


Fig. 5. Faces cropped from two frames of sequences 14 and 15, with two different compression levels (the more to the right, the more compression). For faces (b), (f), (h), and (l), compression artifacts affect the detection more than the mean displacement error for their respective sequence. Face (b) was affected by 0.93 px; face (f) by 2.57 px; face (h) by 0.78 px; and face (l) by 1.78 px. Errors are more visible in the eyes and nose areas.

streaming: Pre-encoded MPEG-4 with bandwidth scaling," *Computer Networks*, vol. 44, no. 4, pp. 415–439, 2004.

[12] J.-S. Lee and T. Ebrahimi, "Perceptual video compression: A survey," *IEEE Journal of Selected Topics in Signal Processing*, vol. 6, no. 6, pp. 684–697, Oct 2012.

[13] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Conference on Computer Vision and Pattern Recognition*, vol. 1, 2001, pp. 511–518.

[14] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, "Overfeat: Integrated recognition, localization and detection using convolutional networks," *arXiv preprint arXiv:1312.6229v4*, 2014.

[15] P. F. Felzenszwalb, R. B. Girshick, and D. McAllester, "Cascade object detection with deformable part models," in *Conference on Computer Vision and Pattern Recognition*, 2010, pp. 2241–2248.

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

[17] K. Mikolajczyk and C. Schmid, "A performance evaluation of local descriptors," *Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1615–1630, 2005.

[18] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[19] H. Bay, T. Tuytelaars, and L. Gool, "Surf: Speeded up robust features," in *European Conference on Computer Vision*, vol. 3951, 2006, pp. 404–417.

[20] J. Sivic and A. Zisserman, "Video google: a text retrieval approach to object matching in videos," in *International Conference on Computer Vision*, vol. 2, 2003, pp. 1470–1477.

[21] K. E. A. van de Sande, T. Gevers, and C. G. M. Snoek, "Evaluating color descriptors for object and scene recognition," *Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1582–1596, 2010.

[22] G. Bradski, "The OpenCV Library." *Dr. Dobb's Journal of Software Tools*, 2000.

[23] R. Lienhart and J. Maydt, "An extended set of haar-like features for rapid object detection," in *International Conference on Image Processing*, vol. 1, 2002, pp. 900–903.

[24] G. R. Bradski, "Computer vision face tracking for use in a perceptual user interface," *Intel Technology Journal*, vol. Q2, 1998.

[25] S. Avidan, "Ensemble tracking," *Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 2, pp. 261–271, 2007.

[26] ITU-T/ISO/IEC JTC 1, "Advanced Video Coding for Generic Audio-Visual Service," *ITU-T Rec. H.264 and ISO/IEC 14496-10 (AVC)*, 2003 (and subsequent editions).

[27] B. Bross, W. Han, G. J. Sullivan, J. Ohm, and T. Wiegand, "High Efficiency Video Coding (HEVC) Text Specification Draft 9," *Document JCTVC-K1003 - ITU-T/ISO/IEC Joint Collaborative Team on Video Coding (JCT-VC)*, 2012.

[28] G. J. Sullivan, J. Ohm, W.-J. Han, and T. Wiegand, "Overview of the High Efficiency Video Coding (HEVC) Standard," *Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1649–1668, Dec 2012.

[29] T. Wiegand, G. Sullivan, G. Bjontegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 560–576, July 2003.

[30] F. Bossen, B. Bross, K. Suhring, and D. Flynn, "HEVC complexity and implementation analysis," *Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1685–1696, 2012.

[31] J.-R. Ohm, G. J. Sullivan, H. Schwarz, T. K. Tan, and T. Wiegand, "Comparison of the coding efficiency of video coding standardsincluding high efficiency video coding (HEVC)," *Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1669–1684, 2012.

[32] JCT-VC, "Hevc model (hm) repository," (as of April 2, 2015). [Online]. Available: https://hevc.hhi.fraunhofer.de/svn/svn_HEVCSoftware/

[33] F. Bossen, "Common test conditions and software reference configurations," *Document JCTVC-F900 - Joint Collaborative Team on Video Coding (JCT-VC)*, 2011.

[34] M. Karczewicz, I. Ye, and I. Chong, "Rate distortion optimized quantization," *Document VCEG AH21 - ITU T SG16/Q.6*, 2008.

[35] G. Correa, P. Assuncao, L. Agostini, and L. A. da Silva Cruz, "Performance and computational complexity assessment of high-efficiency video encoders," *Transactions on Circuits and Systems for Video Technology*, vol. 22, no. 12, pp. 1899–1909, 2012.

[36] W.-S. Chu, F. De la Torre, and J. F. Cohn, "Selective transfer machine for personalized facial action unit detection," in *Conference on Computer Vision and Pattern Recognition*, 2013, pp. 3515–3522.

[37] X. Xiong and F. De la Torre, "Supervised descent method and its applications to face alignment," in *Conference on Computer Vision and Pattern Recognition*, 2013, pp. 532–539.