Optimal Grouping-of-Pictures in IoT Video Streams

K. Panagidi[a], C. Anagnostopoulos[b], S. Hadjiefthymiades[a]

[a]*Department of Informatics and Telecommunications,National and Kapodistrian University of Athens,Greece*
[b]*School of Computing Science, University of Glasgow, United Kingdom*

Abstract

We study a dynamic video encoder that detects scene changes and tunes the synthesis of Groups-of-Pictures accordingly. Such dynamic encoding can be applied to infrastructures with restricted resources, like IoT facilities where multimedia streams are of use. In such facilities the scarcity of resources (energy, bandwidth, etc.) is a dominant solution design factor. In the domain of video capturing/transmission content-driven approaches should be adopted to improve efficiency while maintaining quality at acceptable levels. We propose a time-optimized decision making model that yields different sizes of groups-of-pictures (frames) to meet the previously discussed objectives i.e., transmit video sequences in acceptable quality with rational use of the wireless resources. Our quantitative findings show that the propose scheme performs quite efficiently while dispatching video sequences with different characteristics.

Keywords: Scene detection, Content-driven, Group-of-Pictures, Optimal Stopping Theory

1. Introduction

1.1. Motivation

The use of multimedia applications has risen nowadays in diverse areas like video-on-demand, distance learning, spatial monitoring etc. A special category of wireless sensors networks, in which multimedia data such as voice,

Email addresses: kakiap@di.uoa.gr (K. Panagidi), Christos.Anagnostopoulos@glasgow.ac.uk (C. Anagnostopoulos), shadj@di.uoa.gr (S. Hadjiefthymiades)

Preprint submitted to Special Issue on Mobile Video in the 5G/IoT EraNovember 27, 2017

image and video are disseminated, is called Wireless Sensor Multimedia Networks (WSMNs) [1]. Currently, WSMNs are attracting significant attention due to the variety of applications in which they can be applied such as traffic congestion, environmental, habitat and patient monitoring and recording unusual events. One of the challenges of WSMNs is the lifetime of the network, since the nodes are mostly battery-operated. Although providing better quality for images and videos is necessary, it shortens the network lifetime as the energy sources are rapidly drained. One of the features, which is energy consuming in WSMNs, is multimedia streaming. Multimedia streaming is the process of sending and delivering multimedia content to end users or to the fixed infrastructure, where it will pass through further processing. Multimedia streaming requires efficient compressing methods which minimize the consuming power without harming the content of the distributed data.

The most popular standard for motion compensated video compression is MPEG. Even though it was originally designed for digital storage media, its capabilities have been increased to support a high spectrum of bit rates in order to be used in streaming multimedia applications over the Internet or over lossy wireless networks. Although in this paper we assess the performance of our scheme using the MPEG-2 standard our technique is also applicable to the MPEG-1 and 4 standards as well as the H.26* family of standards. This wide applicability is based on the intra-frame calculations that we undertake in order to throttle our decision making process. In this paragraph we briefly present the broader MPEG video compression technique. A key feature of MPEG is the ability to compress a video signal to a fraction of the original size by coding only the differences between two sequential frames instead of an entire frame. This compression method is called differential encoding. MPEG uses three types of frames, i.e. I, P and B frames to implement different compression methods and exploit inter-frame dependencies within the video stream. Typically, the repeated sequence of I, P, and B frames is known as Group of Pictures (GOP). Each GOP is characterized by a specific number of I, P and B frames. I frame means an intra-coded frame and can be treated as a standalone image. I frames are often used as a reference point to a new scene or a big change to the already transmitted sequence of frames. A P frame contains only predictive information. P frame is generated by looking at the deltas between the present and the previous frame. B frames are created by examining the differences between the previous and the next reference frame, i.e. either I or P, in a sequence of frames. P and B frames do not contain sufficient information to view the related video frame but they

have the advantage of requiring significantly less resources when stored or transmitted. P and B frames can be decoded in the context of GOP. Ideally a GOP should represent a similar continuous related scene. The encoders mostly use fixed GOP size to encode video sequences. A fixed encoder can operate with different size of GOPs but once a target size for the GOPs is selected, the same size is applied to the whole coded sequence. Fixed encoders are easy to implement but they prevent the encoding process to be adaptive to changes in video sequences due to i) scene cuts (abrupt/gradually), ii) changes of video capturing settings e.g. camera focus and iii) degradation of frame quality based on transmission noise.

Challenge 1: Bandwidth is limited: Imagine a video from a surveillance camera of a parking lot. Except from the movement of a car or a passenger all the remaining scene remains static over times. It is expected that the surveillance video demonstrates "limited" activity thus frequent transmission of I frames is not needed, which in turn require network resources and energy. In contrast a football match contains many scene changes because the camera or the objects in the scene are constantly in movement, which logically corresponds to frequent I frames. If scenes with small video content variance, e.g. parking lot, are coded with the same GOP structure frequency with high rate changing frames, e.g. football match, this would lead to a considerable waste of network resources. Constant rate of I frame generation from fixed encoders requires significantly more bandwidth than the actually needed to support the considered multimedia applications.

Challenge 2: Video Streaming in 'accepted' quality: Scene changes can be divided into two categories: abrupt and gradual. The difference between abrupt and gradual scene changes lies in the number of frames needed to conclude the change. If the change is contained only in one frame it is defined as a abrupt scene change. Gradual scene change involves several frames to complete the transition from one scene to another. Encoding process is influenced by GOP structure because it is based on predictive coding techniques along the temporal axis such as motion prediction and compensation. If I-frames are created independently of scene changes then encoding efficiency will suffer from severe error drifting on video transmission. Again high-rate changing frames should be shorter than slow motion videos in order to achieve better coding efficiency.

This paper is organized as follows. In Section 2 we present the preliminaries needed for both GOP structure and OST formulation problem. In section 3 we present the scene Detection problem combined with the description of

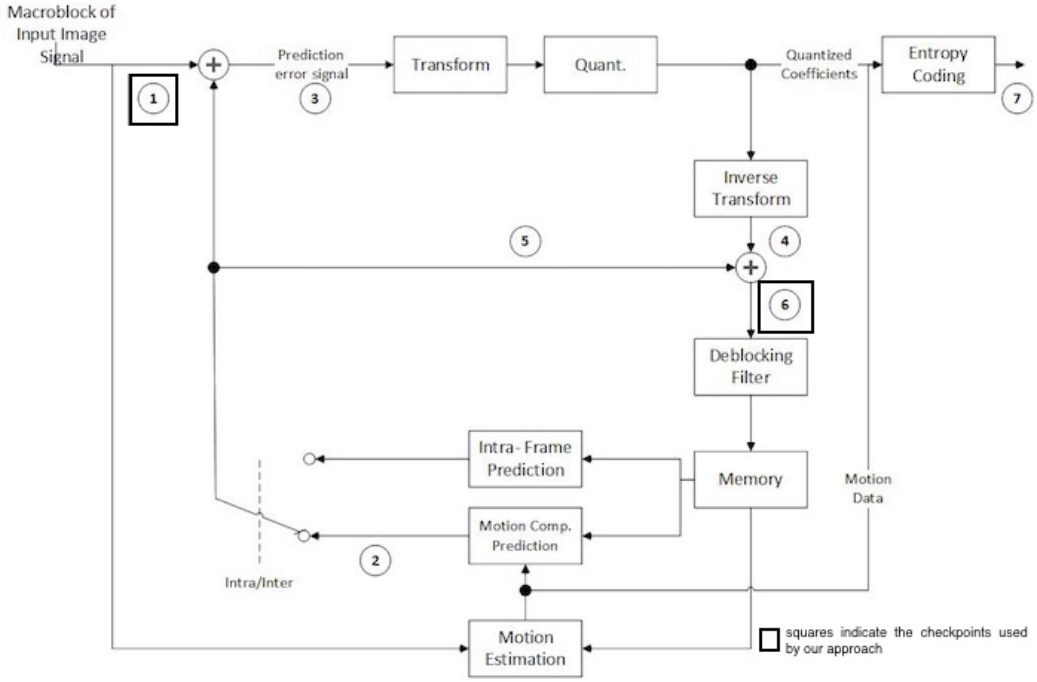


Figure 1: MPEG Encoder

our solution. Section 4 presents the experiments performed and the corresponding discussion followed by the conclusions in section 5. Experiments were conducted by using several types of slow and fast motion video samples from a media library [2].

1.2. Preliminaries

1.2.1. Group-Of-Pictures structure

The main goal of the MPEG standard is to compress a video sequence to a fraction of the original prior to transmission or storage. This is achieved by transmitting the changes between frames, which are sampled at specific time intervals, and not the whole sequence of frames. The basic processing blocks in the encoder, shown in Figure 1, are Discrete Cosine Transform (DCT) coefficient quantizer, run-length amplitude / variable length coder, and block-based motion compensated prediction, using motion estimation.

Starting with the first frame of a Group-Of-Pictures (GOP), an I (intra-coded) frame is created. The encoder can predict a target frame. This is commonly referred to as a P (Predicted) frame, and it may also be predicted

from other P frames, although only in a forward-time manner. Each P frame in a sequence is predicted from the frame immediately preceding it, whether it is an I frame or a P frame. Note that, I frames are autonomously compressed spatially with no reference to any other frame in the sequence. The temporal prediction technique used in MPEG video is based on motion estimation. The basic assumption of motion estimation is that, in most cases, consecutive video frames will be similar except for changes induced by objects moving within the frames. In the trivial case of zero motion between frames (and no other differences caused by noise), the encoder predicts the current frame as a duplicate of the prediction frame. When this is done, the only information necessary to transmit to the decoder becomes the syntactic overhead, which is necessary to reconstruct the picture from the original reference frame.

1.2.2. Optimal Stopping Theory

In our context we establish a content-driven and structure different GOP size adaptive to changes. GOP size is provided by an optimal stopping rule based on the principles of Optimal Stopping Theory (OST), which provides the best time instance to maximize an expected pay off.

Specifically, let $\mathbb{F}_n$ is defined as the σ-algebra generated by the random $Y_1, Y_2, \cdots, Y_n$ variables in a probability space $(\Omega, \mathbb{F}, P)$. A *stopping rule* is a random variable τ with realization values in a set of natural numbers such that $\{\tau = n \in \mathbb{F}_n\}$ for $n = 1, 2, \ldots$ and $P(\tau < \infty) = 1$. We denote with $\mathbb{M}(n, N)$ the class of all stopping rules τ in which $P(n \leq \tau \leq N) = 1$ for any $n = 1, 2, \ldots$. The real-valued pay off function in OST is defined as the mapping $W : \mathbb{R} \to \mathbb{R}$ being a Borel measurable function which values $W(y)$ interpret the pay off of a decision maker (encoder in our context) when it stops the Markov chain $(Y_n, \mathbb{F}_n)$ at the state $y \in \mathbb{R}$.

Assume now that for a given state y and for a given stopping rule τ the expectation $\mathbb{E}[W(Y_\tau)|Y_1 = y]$ exists. Then the expected pay off $\mathbb{E}[W(Y_\tau)|Y_1 = y]$ corresponding to a chosen stopping rule τ exists for all states $y \in \mathbb{R}$, which refers to the value of the stopping problem. Based on the *principles of optimality* the *value* $V_N(y)$of the optimal stopping problem is the supremum of the expected pay off of all the stopping rules belonging to $\mathbb{M}(1, N)$, i.e.,

$$V_N(y) \quad = \quad \sup_{\tau \in \mathbb{M}(1,N)} \mathbb{E}[W(Y_\tau)|Y_1 = y], \tag{1}$$

where the supremum is taken for all stopping rules $\tau \in \mathbb{M}(1, N)$ for which

the expectation $\mathbb{E}[W(Y_\tau)|Y_1 = y]$ exists for all $y \in \mathbb{R}$. Based on the optimal value $V_N(y)$, where the supremum in (1) is attained, the *optimal stopping rule* $t^* \in \mathbb{M}(1, N)$ satisfies the condition:

$$V_N(y) \;=\; \mathbb{E}[W(Y_t^*)|Y_1 = y], \forall y \in \mathbb{R}. \tag{2}$$

It is clear that the optimal value $V_N(y)$ is the maximum possible excepted pay off to be obtained observing the random variables $Y_1, \ldots, Y_N$ up to N-th observation. Consider now that the expectations $\mathbb{E}[W(Y_\tau)|Y_1 = y$ exist for all $y \in \mathbb{R}$ and, based on the principles of optimality, introduce the operator $\mathcal{Q}$ over the pay off function $W \in \mathbb{R}$ such that:

$$\mathcal{Q}W(y) \;=\; \max\{W(y), \mathbb{E}[W(Y_t^*)|Y_1 = y]\}. \tag{3}$$

Then, the optimal stopping rule t^* which attains the optimal value in (2) is estimated by the Theorem 1:

Theorem 1 ([3]) Assume that $W \in \mathbb{R}$. Then:

- $V_n(y) = \mathcal{Q}^n W(y)$, $n = 1.2, \ldots$;
- $V_n(y) = \max\{W(y), \mathbb{E}[V_{n-1}(Y_1)]\}$, where $V_0(y) = W(y)$
- The stopping rule t_n^* evaluated as

$$t_n^* \;=\; \min\{0 \leq k \leq n : V_{n-k}(y) = W(y)\}, \tag{4}$$

refers to an optimal stopping rule in $\mathbb{M}(1, n)$. If $\mathbb{E}[|W(Y_k)|] < \infty$, for $k = 1, \ldots, n$, then the stopping rule t_n^* in (4) is *optimal* in the class $\mathbb{M}(1, n)$.

2. Related Work & Contribution

2.1. Related Work

Scene change detection is the main criterion which defines GOP length in many research approaches. Therefore we present below related works on scene detection and adaptive GOP structuring. All the following approaches are based on the following steps: authors extract some statistics from consecutive frames like color histograms or block differences and then compare this information with a specific threshold. Especially in compressed videos we can use several well studied indicators like discrete cosine transform (DCT)

coefficients [4],[5], and block modes/types [6] and motion vectors [7],[8] and [9].

Authors in [10] study the scene detection problem. The use of texture variation indicators, like interframe variations combined with a parallel processing method is proposed for video encoding with adaptive GOP structure. They detect both types of scene changes, i.e. gradual and abrupt scene changes, at less computation effort and the creation of new GOP is based on this detection. They also propose also balanced frame-level parallel scheduling algorithms that first determine frame priority, followed by the thread priority assignment. However this approach is mostly focused in the parallelization of video processing and not on the implementation of more sophisticated algorithms for scene detection. Scene detection can be based on other approaches like the pixel-based method in [11].The differences between the pixel values of two sequential frames is measured and if this value is higher than a specific threshold a change is detected. The disadvantage for the pixel-based method is that it is sensitive to object motion in the scene. Histogram comparison is proposed in [12] where the difference between histograms of two sequential frames is computed in order to determine the scene change. It should be mentioned that histograms are not sufficient information for scene change detection as long as different scenes can have similar histogram values. Scene change detection by using Markov Chain Monte Carlo (MCMC) algorithm [13] and k-means clustering-based [14] approaches also provide feasible solutions. The posterior probability calculation of the MCMC algorithm is computed based on the data likelihood of the video and it requires important computation effort. However statistical techniques, e.g. pixel-based and block-based luminance difference approaches, involve lower complexity than clustering-based approaches as shown in [15].

Adaptive GOP size is also a well-known problem in the related literature but most approaches follow intuitive processes. Dumitras and Haskell [16] developed a frame type decision algorithm, which employs the motion similarity information. Authors show that the optimal number of B frames between reference frames must be between 0 and 2. In [17] the author proposes to place I frames to the positions of detected cuts during the process of video encoding. Our model follows a different approach compared to these research efforts. Mainly these methods compare two or more consecutive frames and not by taking the advance of interframe result. This approach requires a significant computational effort. In addition histograms and other traditional methods based on statistics cannot be applied on real-time fast

flow video in which changes can occur stochastically. Our model lets the encoder decide the GOP size by autonomously determining the appropriate time to conclude the GOP.

Methods derived from the optimal stopping theory have been applied to information dissemination in ad-hoc networks. The data delivery mechanisms in [18] and [19]deal with the delivery of quality information to context-aware applications in static and mobile ad-hoc networks respectively assuming epidemic-based information dissemination schemes. The mechanism in [18] is based on the probabilistic nature of the "secretary problem" [3] and the optimal online problem. In [20] authors make optimal stopping decisions on the collection of contextual data from WSNs. Authors try to determine the best time to switch from decision to learning phase of Principal Component-based Context Compression (PC3) model while data inaccuracy is taken into account. If data inaccuracy remains at low levels, then any deterministic switching from compression to learning phase leads to unnecessary energy consumption. OST rules are applied between compression and learning phases of the observations.

2.2. Contribution

We propose a model of dynamic encoder adaptive to changes in video sequences by dynamically adjusting GOP size based on an Optimal Stopping Theory (OST) rule in order to transmit video sequences in an acceptable quality with the simultaneous rational use of WSMN resources. More specifically we report:

1. what it is defined as a scene change problem and quantify this event
2. the optimal stopping rule for the discussed problem and how it is applied
3. the performance evaluation of the proposed scheme.

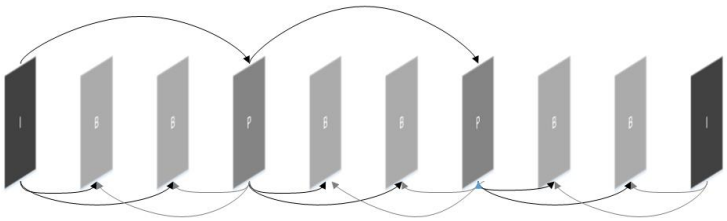


Figure 2: GOP in the H.263 video flow

3. Time-optimized Grouping-of-Pictures

3.1. Rationale & Problem Formulation

A prediction scheme inside the encoder is used in order to foresee any scene changes. In our case, it is assumed that each GOP structure can be large and finite. Each next frame is encoded as a P frame at discrete time step $t \in \{1, \ldots, n\}$. At time instance $t = 1$ an I frame F_I is constructed and in $t = N$ the last frame F_{P_N} is created. The main goal is to continue to add P frames into the same GOP sequence, if and only if a scene change does not occur. At this point we must quantify a scene change. Based on the definition provided in [10] let us consider a video frame F_{C_t} coming inside the encoder encoder at the checkpoint (1) of figure 1. A P frame F_{P_t} is encoded using the motion vectors between the previous reference frame and the current frame F_{C_t} inserted in the MPEG encoder and the output is an encoded frame F_{P_t-enc} which mainly contains the differences betweenF_{C_t} and the previously I or P frame as shown in Figure 2. This bit-stream is sent to the decoder. The decoder based on these differences creates the new frame F_{DC_t} exiting from checkpoint (6) in figure 1. The metric indicating a possible scene change is defined as the sum of absolute differences [10] between the two frames (SATD) F_{C_t} and F_{DC_t} where $F_{i,j}^{F_{C_t}}$ is the pixel value at location (i, j) of frame F_{C_t} and I_w and H_h are the width and height of a frame, respectively:

$$SATD(F_{C_t}, F_{DC_t}) = \sum_{i=0}^{I_w-1} \sum_{j=0}^{H_h-1} |F_{i,j}^{F_{C_t}} - F_{i,j}^{F_{DC_t}}| \tag{5}$$

As a decision maker, we desire to get as close as possible to a given limit in which a scene change occurs but the limit should not be exceeded. Given the incoming values of SATD between the incoming and the outgoing frame, i,e, checkpoints 1 and 6 from the encoder encoder in figure 1, we would like to find the closest distribution which fits the data. We used a distribution comparison function which returns the fit of all valid parametric probability distributions to the input data and plot the Probability Density Functions (PDFs) to compare them graphically. In our case we can see the results of the function in figure 3. In this work we will deal with the two most prevalent distributions, i.e. gamma and normal distributions.

3.1.1. Gamma Distribution

Specifically, let us consider that $S_1, S_2, \cdots, S_n$ be a sequence of sequentially observed random variables having a gamma distribution $\Gamma(\alpha, \beta)$ where

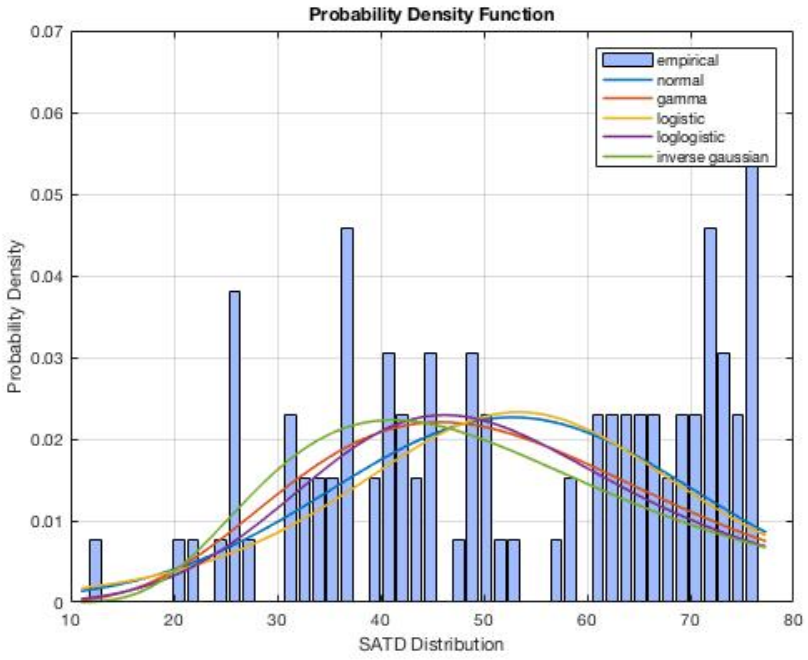


Figure 3: Fitting probability distribution functions based on the actual SATD values.

$\alpha, \beta > 0$ and each corresponds to $S_n = SATD(F_{C_n}, F_{DC_n})$ at time instance (step) $t = n$:

$$f(s \mid \alpha, \beta) = \frac{\beta^{\alpha} s^{\alpha-1} e^{-s\beta}}{\Gamma(\alpha)} \tag{6}$$

The encoder observes the random sequence $\{S_1, \ldots, S_n\}$ and decides whether to 'stop' or to 'continue'. The encoder wants to pull as many frames as possible. If the encoder decides to stop at the moment n, then it will gain a real-valued pay off $(y + \sum_{i=1}^{n} S_i)$, if the sum $\sum_{i=1}^{n} S_i$ is not greater than a specified threshold T. The threshold T corresponds to cumulative error when a scene changes occurs. If the encoder passes the limit T, then the gain is zero. A given nonnegative real number y appearing in the above gain definition is another characteristic of the problem and may be interpreted as an initial state of the process of observations. Formally, we consider a Markov

chain $(Y_n, \mathbb{F}_n)$ for $n = 1, \ldots, N$ with

$$Y_n = y + \sum_{i=1}^{n} S_i, \tag{7}$$

with $\mathbb{F}_n$ being generated by the observations $S_1, S_2, \ldots, S_n$ and $y > 0$. We define as pay off the real-valued function $W(y) \in \mathbb{R}$ such that:

$$W(Y_n; y, T) \;=\; \begin{cases} (y + \sum_{i=1}^{n} S_i) & \text{, if } y + \sum_{i=1}^{n} S_i \leq T \\ 0 & \text{, otherwise,} \end{cases} \tag{8}$$

with error tolerance threshold $T > 0$. The threshold T indicates the tolerance of the encoder to *delay* the cumulative sum of the frame variations in light of pulling as many fames as possible. However, the sum of those variations is stochastic, thus, the encoder has to find an optimal rule for stopping the surge of the random sum just before reaching its maximum tolerance value T. Based on the pay off $W(Y_n)$ with initial state $y > 0$ and tolerance threshold T, we define our optimal stopping time problem for the encoder:

Problem 1. Given observations of SATD values $\{S_1, \ldots, S_n\}$ and tolerance cumulative sums $Y_1 = y+S_1, Y_2 = y+S_1+S_2, \ldots, Y_n = y+\sum_{i=1}^{n} S_n$, find the optimal stopping time t^* to maximize the expected pay off $\mathbb{E}[W(Y_{t^*})|Y_1]$ where the pay off is defined in (8).

3.2. Solution Fundamentals

Before proceeding with a solution of Problem 1, we refer to the Proposition 1 to analyze the expectation of the optimal value $V_n(y)$.

Proposition 1. If there exists a real number t^*, $0 \leq t^* \leq T$ such that the conditions of Theorem 1 hold true, the optimal value $V_n(y)$ is calculated for $y < t^*$ as follows, where $n = 2, \ldots, N$:

$$V_n(y) = \int_0^{t^*-y} V_{n-1}(y+s) f(s) ds + \int_{t^*-y}^{\infty} W(y+s) f(s) ds \tag{9}$$

with the initial condition $V_1(y) = \int_0^{\infty} W(y+s) f(s) ds$.

Proof: This derives immediately from the principle of optimality in Theorem 1 by taking the expectation of the optimal value of $V_n(y)$.

Let us now provide a solution to Problem 1. We need first to find the form of $V_n(y) = \mathcal{Q}^n(y)$, $n = 1, \ldots, N$. By definition of the operator $\mathcal{Q}$, we have for every $y \in (0, T]$:

$$\begin{aligned}
\mathcal{Q}W(y) &= \max\{W(y), \mathbb{E}[W(Y_1)]\} = \max\{W(y), \mathbb{E}[W(y + S_1)]\} \\
&= \max\{W(y), \int_0^\infty W(y+s) f(s \mid \alpha, \beta) ds\} \\
&= \max\{W(y), \mathbb{I}_1(y)\}
\end{aligned}$$

For $y < T$ and given Proposition 1, the integral function $\mathbb{I}_1(y) = \int_0^\infty W(y+s) f(s \mid \alpha, \beta) ds$ is expressed as follows:

$$\begin{aligned}
\mathbb{I}_1(y) &= \int_0^\infty W(y+s) f(s) ds = \\
&= \int_0^{T-y} (y+s) f(s) ds + \int_{T-y}^\infty 0 f(s) ds \\
&= \int_0^{T-y} (y+s) \frac{\beta^\alpha s^{\alpha-1} e^{-s\beta}}{\Gamma(\alpha)} ds \\
&= \int_0^{T-y} y \frac{\beta^\alpha s^{\alpha-1} e^{-s\beta}}{\Gamma(\alpha)} ds + \int_0^{T-y} s \frac{\beta^\alpha s^{\alpha-1} e^{-s\beta}}{\Gamma(\alpha)} ds \\
&= \frac{1}{\Gamma(a)} (y \beta^\alpha (T-y)^\alpha (\beta(T-y))^{-\alpha} (\Gamma(\alpha) - \Gamma(\alpha, \beta(T-y))) \\
&+ \beta^{\alpha-1} (T-y)^\alpha (\beta(T-y))^{-\alpha} (\Gamma(\alpha+1) - \Gamma(\alpha+1, \beta(T-y)))) \quad (10)
\end{aligned}$$

Figure 4 shows an exemplary graph of the pay off function $W(y)$ and the integral function $\mathbb{I}_1(y)$ for $y \leq T$.

It is easy to verify that for any given tolerance threshold T the functions W and I_1 have equal values at $t_1 \in (0, T]$ at which the function $\mathbb{I}_1$ takes its only maximum on the interval $(0, T]$ because $\mathbb{I}_1(y) > W(y)$ for $y \in (0, t_1)$ and $\mathbb{I}_1(y) < W(y)$ for $y \in (t_1, T]$. Then the value of t_1 is estimated by solving the following equation:

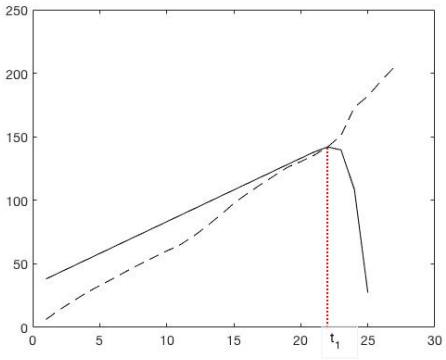


Figure 4: Graphs of functions $\mathbb{I}_1$ (continuous line) and W (dotted line), T=30

$$\mathbb{I}(t_1) = \int_0^{T-t_1} W(t_1+s)f(s)ds = W(t_1), \tag{11}$$

and depends on the probability density function $f(s)$ and tolerance threshold T. Given the pay off function in (8), we obtain t_1 by solving the equation:

$$\begin{aligned} \int_0^{T-t_1} (t_1+s)f(s)ds &= t_1 \Leftrightarrow \\ t_1 \int_0^{T-t_1} f(s)ds + \int_0^{T-t_1} sf(s)ds &= t_1 \Leftrightarrow \\ t_1 \frac{(1-F_S(T-t_1))}{F_S(T-t_1)} &= \mathbb{E}[S|S \leq T-t_1], \end{aligned} \tag{12}$$

where $F_S(x) = P(S \leq x) = \int_0^x f(s)ds$ is the cumulative probability function of S and $\mathbb{E}[S|S \leq T-t_1]$ is the conditional expectation of S given that $S \leq T-t_1$. The optimal value function $V_1 = \mathcal{Q}W$ is the maximum of the two ones presented in Figure4. Based on the optimality in Theorem 1, one step before the end of the observations the decision maker should continue the observations if it is at any state y which is less than t_1 and should stop

otherwise. Obviously the functions $\mathbb{I}_1$, W and V_1 are equal to 0 for arguments greater than T.

Given that $\mathbb{I}_1(y)$ denotes the expectation $\mathbb{E}[V_{n-1}(y+S_1)], n = 1 \cdots N$, we provide Proposition 2 that holds true for any integral function $\mathbb{I}_n(y)$, $n = 1 \cdots N$ by induction.

Proposition 2. For any natural number n, for t_1 derived from (11), and for every $T > 0, \alpha > 0, \beta > 0$ the integral function $\mathbb{I}_n(y)$ satisfies the following conditions:

1. $\mathbb{I}_n(y) > W(y)$ for $y \in (0, t_1)$
2. $\mathbb{I}_n(y) < W(y)$ for $y \in (t_1, T]$
3. $\mathbb{I}_n(y) = 0$ for $y > T$

Proof. The conditions (1),(2) and (3) for $n = 1$ derive from Proposition 1. Now, let us assume that the conditions (1)–(3) hold for $\mathbb{I}_{n-1}(y)$. Then, by definition of $V_{n-1}(y)$ and by induction assumption for $y \in (0, t_1)$ we obtain:

$$\begin{aligned}
\mathbb{I}_n(y) &= \int_0^{\infty} V_{n-1}(y+s)f(s)ds \\
&= \int_0^{t_1-y} I_{n-1}(y+s)f(s)ds + \int_{t_1-y}^{T-y} W(y+s)f(s)ds + \int_{T-y}^{\infty} 0 \cdot f(s)ds \\
&\geq \int_0^{t_1-y} W(y+s)f(s)ds + \int_{t_1-y}^{T-y} W(y+s)f(s)ds = \mathbb{I}_1(y) > W(y)
\end{aligned}$$

Hence the condition (1) is satisfied. In addition, condition (2) is satisfied when $y \in [t_1, T)$ since we obtain that:

$$\mathbb{I}_n(y) = \int_0^{\infty} V_{n-1}(y+s)f(s)ds = \int_0^{T-y} W(y+s)f(s)ds = \mathbb{I}_1(y) < W(y) \quad (13)$$

The condition (3) is obvious, thus, the proof of Proposition 2 is completed.

It follows from Proposition 2 immediately that for $n = 1, \ldots, N$, the optimal values $V_n(y)$ have the form:

$$V_n(y) = \mathbb{I}_n(y)\mathbf{1}_{(0,t_1]}(y) + W(y)\mathbf{1}_{(t_1,T]}(y), \quad (14)$$

where the value of t_1 is provided in (11). Based on this, we provide the optimal stopping rule for the Problem 1:

Proposition 3. Given a sequence of SATD realizations $S_1, \ldots, S_N$ with probability density function $f(s)$ and pay off function $W(Y_n; y, T)$ defined in (8) with cumulative sum $Y_n = y + \sum_{i=1}^{n} S_i$, the optimal stopping rule t^* for the Problem 1 with initial state y is given by:

$$t^* = \min\{0 \leq k \leq N : Y_k = y + \sum_{i=1}^{k} S_i \geq t_1\}, \tag{15}$$

where t_1 is estimated in (11).

Proof: The result follows directly from Theorem 1 and Proposition 2.

From Proposition 3, the optimal stopping rule model is interpreted as follows: the encoder continues to observe, i.e. add P frames in the GOP sequence, as long as the sum of the initial state y and the sum of already observed values s_i do not exceed the value t_1. Hence, we have to compute the threshold value t_1 which requires the estimation of the probability density function $f(s)$ given a tolerance threshold T. In case that SATD follow the gamma distribution $\Gamma(\alpha, \beta)$, then the integral function $\mathbb{I}_1$ is directly provided in (10) and t_1 is obtained by solving the equation in (11).

Remark 1. It is worth mentioning that the optimal value $V_N(y)$ of Problem 1 is inductively calculated for $y < t_1$ from the recursive equation:

$$V_n(y) = \int_0^{t_1 - y} V_{n-1}(y + s) f(s) ds + \int_{t_1 - y}^{T - y} (y + s) f(s) ds,$$

where the initial condition is given by $V_1(y) = \mathbb{I}_1(y)$ for any $N > 0$ and $n = 2, 3, \ldots, N$.

3.2.1. Normal Distribution

The normal distribution is investigated as the probability distribution that fits the actual SATD values. However, error figures are limited to values greater than zero. Let $S \sim N(\mu, \sigma^2)$ follow a normal distribution and lie within the interval $S \in [0, +\infty)$. Then, the random variable S conditioned on the interval $[0, \infty)$ is described by the *truncated* probability function:

$$f(s \mid \mu, \sigma, 0, \infty) = \frac{\frac{1}{\sqrt{2\pi}} e^{\frac{-1}{2}(\frac{s-\mu}{\sigma})^2}}{\sigma(\Phi(\infty) - \Phi(\frac{0-\mu}{\sigma}))}, \text{with } \Phi(x) = \frac{1}{2}(1 + erf(x/\sqrt{2})). \tag{16}$$

By definition, $\Phi(\infty) = 1$ and, thus, the probability function is re-written as:

$$f(s \mid \mu, \sigma, 0, \infty) = \frac{1}{\sqrt{2\pi}\sigma(1 - \Phi(\frac{-\mu}{\sigma}))} e^{\frac{-1}{2}(\frac{s-\mu}{\sigma})^2} \tag{17}$$

According to the truncated normal distribution and Preposition 1 for $y < T$, the integral function $\mathbb{I}_1(y) = \int_0^\infty W(y+s) f(s \mid \mu, \sigma) ds$ is expressed as follows:

$$\begin{aligned}
\mathbb{I}_1(y) &= \int_0^\infty W(y+s) f(s) ds = \\
&= \int_0^{T-y} (y+s) f(s) ds + \int_{T-y}^\infty 0 f(s) ds \\
&= \int_0^{T-y} (y+s) \frac{1}{\sqrt{2\pi}\sigma(1 - \Phi(\frac{-\mu}{\sigma}))} e^{\frac{-1}{2}(\frac{s-\mu}{\sigma})^2} ds \\
&= (\frac{1}{\sqrt{2\pi}\sigma(1 - \Phi(\frac{-\mu}{\sigma}))})(\int_0^{T-y} y e^{\frac{-1}{2}(\frac{s-\mu}{\sigma})^2} ds + \int_0^{T-y} s e^{\frac{-1}{2}(\frac{s-\mu}{\sigma})^2} ds) \\
&= y\sqrt{\frac{\pi}{2}}\sigma(erf(\frac{\mu}{\sqrt{2}\sigma}) - erf(\frac{\mu - T + y}{\sqrt{2}\sigma})) + \sigma\frac{1}{2}(-\sqrt{2\pi}\mu erf(\frac{\mu - T + y}{\sqrt{2}\sigma}) + \\
&+ \sqrt{2\pi}\mu erf(\frac{\mu}{\sqrt{2}\sigma}) + 2\sigma(e^{\frac{-\mu^2}{2\sigma^2}} - e^{\frac{-(\mu - T + y)^2}{2\sigma^2}}))
\end{aligned} \tag{18}$$

It is easy to verify that, for any given tolerance threshold T, the functions W and I_1 have equal values at $t_1 \in (0, T]$ at which the function $\mathbb{I}_1$ takes its only maximum on the interval $(0, T]$ because $\mathbb{I}_1(y) > W(y)$ for $y \in (0, t_1)$ and $\mathbb{I}_1(y) < W(y)$ for $y \in (t_1, T]$. Then, the optimal stopping rule t^* is derived from Proposition 3.

3.3. Complexity & Model Design Parameters

The complexity of the encoder for triggering the optimal stopping rule as derived from Proposition 3 is based on the calculation of the current SATD value. Specifically, the SATD value calculation requires $O(I_w H_h)$ time since I_w and H_h are the width and height of a frame. The encoder then increases

Type	α	β	μ	σ	T
Slow motion Video	16.50761	0.07891	0.9766	0.6694	45
Medium motion Video	4.516779	2.99732	7.5131	2.2424	25
Fast motion	7.5712	6.96713	22.6879	4.7797	12

Table 1: α and β values for different types of video

the current summation Y_n at step n by the new S_n SATD value, which is achieved in $O(1)$. If this sum exceeds the optional threshold t_1 provided by Proposition 3, then the encoder is triggered. Hence, the overall complexity for the decision making requires $O(I_w H_h)$.

The design parameters of the problem are the following: the limit (threshold tolerance) T of the cumulative sum of inter-frame deviations, and α, and β fitted parameters of the Gamma distribution. Let us assume that the initial state y of the process equals to 0, i.e., after each triggering of the encoder, and let us confine ourselves to this situation where the value of the problem $V_N(0)$ is positive, i.e., the decision maker (encoder) should make at least one observation (receives at least one frame). Using sample videos from the Test Media Library [2], we have tried to evaluate the design parameters of our approach in different MPEG streams with different needs. For example, a video containing only one shot of a waterfall from a stable camera reception can be characterized as a *slow* motion video. In contrast, a sequence from a football match can be considered as a *fast* motion video. A *medium* motion vector can be defined as a man who is talking to the camera by moving his head. For these three different examples of motion videos α, and β values of the Gamma distribution were computed and presented in Table 1.

4. Performance Evaluation

4.1. Simulation setup

The simulation setup has as follows: we have used an MPEG-2 simulator. This simulator is based on the work presented in [21]. MPEG-2 simulators is enhanced with additional functions in order to support the creation of GOPs with dynamic size based on an OST rule.

The performance metrics of the proposed encoder with dynamic grouping of pictures of GOPs adapted to stream behavior are i) the produced error of the encoding process and ii) the size of generated video stream in bits.

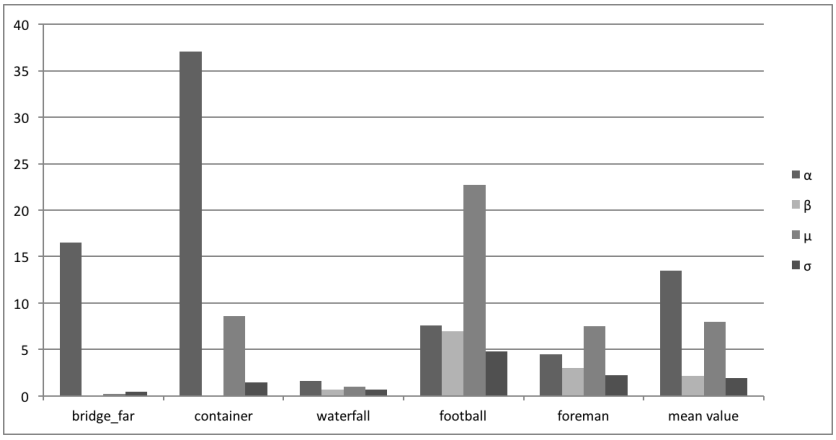


Figure 5: α, β values of Γ distribution for the set of videos used in our experiments

In this way we are trying to map the two challenges referred to Section 1 related to limited bandwidth and the 'quality' of the derived video stream with the experimental results. The metric related to the quality is SATD shown in equation 5 and was measured between the control points (1) and (6) as depicted in MPEG encoder at figure 1. The dynamic grouping of pictures method is compared with a classic fixed-length version of an MPEG-2 encoder which creates a GOP with one I frame and then adds a constant number of P frames e.g. IPPPPPPPPPP. In our case the length of P frames is equal to 10. The pool of videos is downloaded from [2]. Every video was examined in a sequential stream of 100 frames. A short description of the videos follows to illustrate the dynamic character of streams:

1. bridge-far: a slow motion video showing a bridge from remote;
2. waterfall: a slow motion video with the constant recording of a waterfall;
3. hall-objects: a fast motion video from a camera in an office corridor. At some point two people walk in;
4. highway: a shooting of a vacant highway recorded by a camera in a car - medium motion video;
5. foreman: a person talking to camera - medium motion video;
6. football: a fast motion video from a football match;
7. container: a fixed camera showing the course of a tanker - fast motion video;

For each of these videos α, β, μ and σ values are presented in figure 5. The presented values are generated by a single GOP with one I frame and an "infinite" number of P frames. A single GOP of a video stream can provide us with a holistic overview of the SATD of frames. The approach that we follow in order to configure α, β, μ and σ values is the following. A pool of twenty different kind of videos was analyzed and the mean values of the aforementioned parameters were extracted. These mean values are the initial α_{mean}, β_{mean}, μ_{mean} and $\sigma mean$ values when the encoder starts to operate, i.e. $t = 0$. The OST rule for the first incoming frames is based on these initial values. User can select gamma or normal functionality for implementing the dynamic encoding module. We use the abbreviation DGPE describing the dynamic grouping of pictures encoder for the gamma distribution and NDGPE describing the normal distribution. The time when the first GOP concludes, α and β or μ and σ values are re-calculated fitting in the cumulative SATD of the already processed video stream, i.e. GOP=1.

4.2. Discussion of Simulation results

The results of the simulations are described below. The classic encoder (CE) created 10 fixed length GOPs. The number of GOPS created by DGPE and NDGPE are depicted in table 2. In the same table we compare the total size transmitted for each video (*inbits*) from the CE and DGPE encoders. We can notice that in slow motion videos the GOP size is extended in order to avoid unnecessary transmissions of I frames. For example in the waterfall video the number of GOPs is reduced to 2 and 3 per 60 frames in DGPE and DGPE respectively. In contrast in fast motion video the GOPs created are increased while the size of the generated bitstream stays belows the generated bitstream of CE in average. It can be noticed that the volume transmitted in most of the cases from dynamic encoder is smaller than classic encoder. This is expected as fixed encoders are not content-driven and lead to waste of bits and resources. By comparing the dynamic encoders, we may notice that DGPE is more "sensitive" in fast-motion videos by capturing more scene changes than DGPE while DGPE shows tolerance to the medium motion videos.

In addition, through figures 6, 7, 8 and 9 we provide a comparison overview of SATD measured between CE and DGPE. In figure 6 it is observed that the error values coming from CE are higher than the dynamic encoders DGPE and NDGPE. The median value of SATD corresponds to 107.4 for CE. The median value of DGPE is 27.56 and 23.52 of NDGPE.

Video	$DGPE_{GoPs}$	$NDGPE_{GoPs}$	$DGPE_{Size}$	CE_{Size}	$NDGPE_{Size}$
bridge	8	7	832477	876182	841583
waterfall	2	3	1499705	1642998	1500144
hall	9	15	1019986	1032109	1098023
container	15	11	2158422	2017824	2084643
foreman	6	9	2705621	2819314	2847462
football	27	13	6608428	6510336	6288473

Table 2: Size of bitstreams transmitted in network

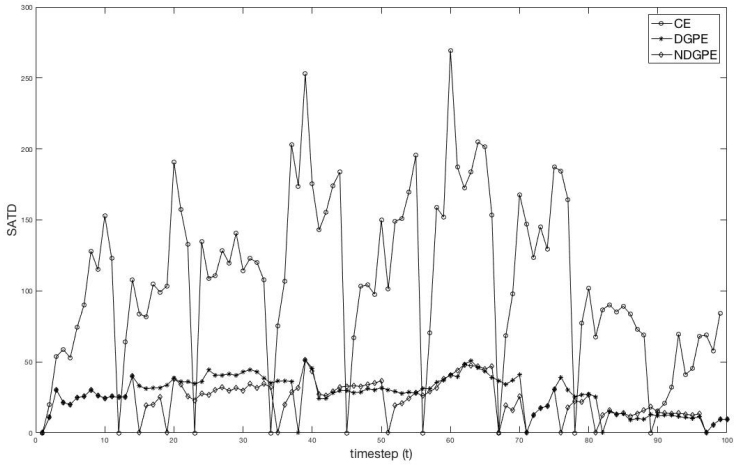
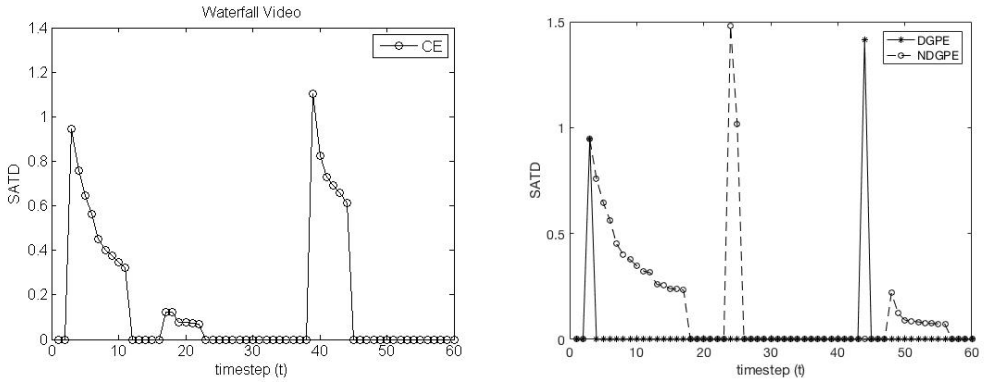


Figure 6: SATD between classic approach and OST- football video

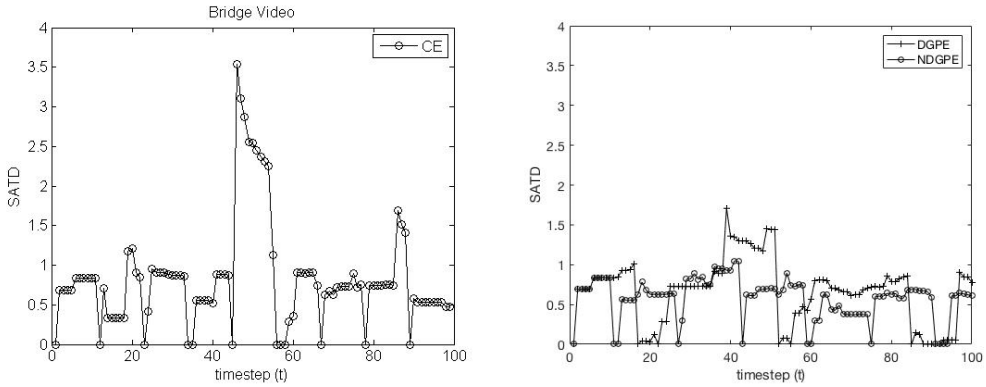
The fewer GOPs created by truncated normal encoder also corresponds to a reduction of 4% in the total transmitted volume of bits as shown in table 2. In figure 7, DGPE has the best video stream performance. The error remains close to the zero values. The first GOP is based on initial mean values of α and β and the next GOPs are based on the refitting of the design values to the incoming data distribution. NGOE needs time to fit μ and σ values to the slow motion video distribution. The mean and std values of the output error are the following: DGPE$\{0.0394, 0.2177\}$ and NDGPE$\{0.1625, 0.2934\}$.

From the results in figures 8 and 9 the encoder which uses normal distribution to compute t^* performs better than the other assessed encoders. We



(a) SATD of CE (b) SATD of DGPE and NDGPE

Figure 7: SATD between classic approach and OST - waterfall video

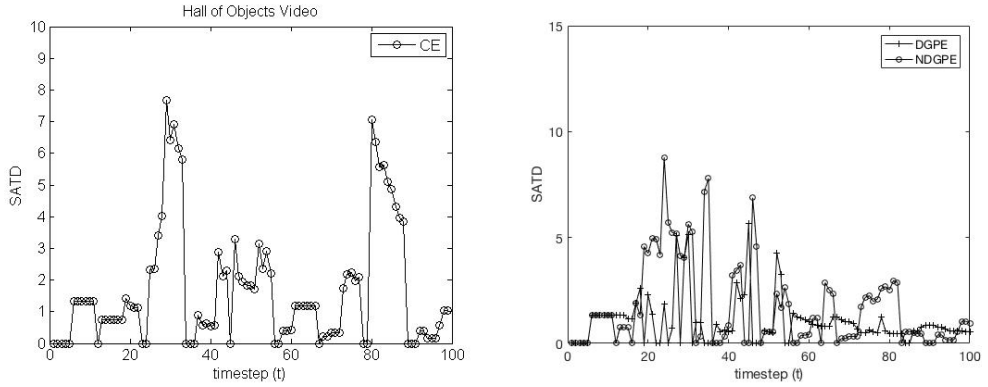


(a) SATD of CE (b) SATD of DGPE and NDGPE

Figure 8: SATD between classic approach and OST - bridge video

can notice that NDGPE needs more time to be adaptive to the changes of the incoming distribution but then SATD error values generated between the frames in the GOP created correspond to small values. For example at hall video the error values after the first 30 frames are quite low when compared with DGPE and CE methods

From the description above, it shown that the dynamic encoders perform better than the fixed length encoder. The notion of adoption to video content is important as I frames are depended on scene changes and thus the encoding efficiency suffers from the error drifting on video transmission. The NDGPE



(a) SATD of CE (b) SATD of DGPE and NDGPE

Figure 9: SATD between classic approach and OST - hall video

shows better performance but a number of training incoming frames are required to fit the data distribution. If video streaming is quite short then the gamma-based encoder DGPE is the better candidate.

5. Conclusions and Future Work

In this paper, we focus on content-based MPEG encoder and propose an OST decision rule for the conclusion of GOP and the transmission of intra-coded frames. Dynamic encoding applied to infrastructures with restricted resources, like IoT camera networks, is needed in order to support media-rich applications in such infrastructures. Limited bandwidth and battery lifetime require nowadays content-driven transmission rates and processing of the video sequences. One major contribution of this paper is the adaptation to video changes; I frames are created when scene changes are detected which leads to significant resource savings while retaining equal quality levels. Our encoder can be applied to facilities with restricted resources like WSMNs in order to transmit video sequences in an acceptable quality. The aim is twofold: to create different size of GOPs adaptive to the transmitted video streams and to try to save resources with a small SATD error. Experiments show that the GOP size was extended in order to avoid unnecessary transmissions. We observe that the stream volume transmitted in most of the cases is smaller than the CE created bitstream which justifies that fixed encoders which are not content-driven lead to waste of network resources.

The encoder focuses on the transmitted video content and, thus, the values of SATD stay lower than the classic approach. Our future agenda includes the expansion of our study toward the inclusion of bidirectional (B) frames in the OST controlled video stream. B frames are created by examining the difference between the previous and the next reference frame and this surely imposes changes in the OST strategy applied for the GOP inclusion. However B frames require less resources when stored or transmitted and this can further lead to savings on the resources employed for video transmission. Additionally, the combined assessment of spatiotemporal differences within and among frames of the video sequence is a significant challenge that we intend to address in our future work.

6. References

[1] J. X. Azim M., Wireless Sensor Multimedia Netwoks: Architectures, Protocols and Applications, CRC Press, October 27,2015.

[2] S. Hoelzer, Xiph org, `https://media.xiph.org/video/derf/`, 2010 (Accessed: 2017-02-24).

[3] T. S. Ferguson, Optimal Stopping and Applications, Mathematics Department, UCLA, Accessed May, 2015.

[4] B. Yeo, B. Liu, Rapid scene analysis on compressed videodynamic vision, IEEE Trans. Circuits and Systems for Video Technology 5 (Dec. 1995) 533–544.

[5] H. Liu, G. Zick, Automatic determination of scene changes in mpeg compressed video, pp. 764–767.

[6] B. Yeo, B. Liu, Novel error concealment method with adaptive prediction to the abrupt and gradual scene changes, IEEE Trans. Multimedia 6 (2004) 158–173.

[7] I. Koprinska, S. Carrato, Detecting and classifying video shot boundaries in mpeg compressed sequences, in: IX European Signal Processing Conf. (EUSIPCO), pp. 1729–1732.

[8] A. N. Anthony Y. Lan, J.-N. Hwang, Scene context-dependent reference-frame placement for mpeg video coding, IEEE Transactions on Circuits and Systems for Video Technology 9 (1999) 478–489.

[9] X. Gu, H. Zhang, Implementing dynamic gop in video encoding.

[10] H.-F. Hsiao, C.-T. Wu, Balanced parallel scheduling for video encoding with adaptive gop structure, IEEE Trans. Parallel Distrib. Syst 24 (2013) 2355–2364.

[11] A. K. H.J. Zhang, S. Smoliar, Automatic partitioning of full-motion video, Multimedia Systems 1 (1993) 10–28.

[12] J. N. Y. X. Z. F. L. Wu, X. Huang, Y. Zhou, Fdu at trec2002: Filtering, qa, web and video tasks, in: 11th Text Retrieval Conference.

[13] Y. Zhai, M. Shah, Video scene segmentation using markov chain monte carlo, IEEE Trans. Multimedia 8 (2006) 686–697.

[14] A. F. B. Gunsel, A. Tekalp, Temporal video segmentation using unsupervised clustering and semantic object tracking, Electronic Imaging 7 (1998) 592–604.

[15] J. H. S. Lefevre, N. Vincent, A review of real-time segmentation of uncompressed video sequences for content- based search and retrieval, Real-Time Imaging 9 (2003) 73–98.

[16] A.Dumitras, B. Haskell, I/p/b frame type decision by collinearity of displacements, in: IEEE Int. Conf. Image Process, volume 4, pp. 2769–2772.

[17] K. L., A novel method of adaptive gop structure based on the positions of video cuts, in: ELMAR, pp. 67–70.

[18] C. Anagnostopoulos, S. Hadjefthymiades, Delay-tolerant delivery quality information in ad hoc networks, Journal of Parallel and Distributed Computing 71(7) (2011) 974–987.

[19] C. Anagnostopoulos, S. Hadjefthymiades, Optimal quality-aware scheduling of data consumption in mobile ad hoc networks, Journal of Parallel and Distributed Computing 72(10) (Oct. 2012) 1269–1279.

[20] C. Anagnostopoulos, S. Hadjiefthymiades, Advanced principal component-based compression schemes for wireless sensor networks, ACM Trans. Sen. Netw. 11 (2014) 7:1–7:34.

[21] S. Hoelzer, Mpeg-2 overview and matlab codec project, http://users.cs.cf.ac.uk/Dave.Marshall/Multimedia/Lecture_Examples/Compression/mpegproj/, 2005.