

A Support Vector Machine Approach for Detection and Localization of Transmission Errors Within Standard H.263++ Decoders

Reuben A. Farrugia, *Student Member, IEEE*, and Carl James Debono, *Senior Member, IEEE*

Abstract—Wireless multimedia services are increasingly becoming popular boosting the need for better quality-of-experience (QoE) with minimal costs. The standard codecs employed by these systems remove spatio-temporal redundancies to minimize the bandwidth required. However, this increases the exposure of the system to transmission errors, thus presenting a significant degradation in perceptual quality of the reconstructed video sequences. A number of mechanisms were investigated in the past to make these codecs more robust against transmission errors. Nevertheless, these techniques achieved little success, forcing the transmission to be held at lower bit-error rates (BERs) to guarantee acceptable quality.

This paper presents a novel solution to this problem based on the error detection capabilities of the transport protocols to identify potentially corrupted group-of-blocks (GOBs). The algorithm uses a support vector machine (SVM) at its core to localize visually impaired macroblocks (MBs) that require concealment within these GOBs. Hence, this method drastically reduces the region to be concealed compared to state-of-the-art error resilient strategies which assume a packet loss scenario. Testing on a standard H.263++ codec confirms that a significant gain in quality is achieved with error detection rates of 97.8% and peak signal-to-noise ratio (PSNR) gains of up to 5.33 dB. Moreover, most of the undetected errors provide minimal visual artifacts and are thus of little influence to the perceived quality of the reconstructed sequences.

Index Terms—Error detection and concealment, error resilient coding, learning systems, multimedia communications, video coding, wireless network.

I. INTRODUCTION

THE demand for advanced information services is growing, and the trend is pushing global information networks to offer flexible multimedia services, such as digital TV broadcasting, mobile TV, and videoconferencing applications, to users having an always-on communication link [1]–[3]. These services employ video compression standards to reduce transmission costs [4]. However, the compressed video bitstream is very sensitive to transmission errors, where a single corrupted bit will cause loss of synchronization between the encoder and decoder until the next synchronization marker. This results in a number of visually impaired macroblocks (MBs) causing

significant degradation in the quality of the reconstructed video sequence.

To alleviate this problem, a number of solutions were presented in the literature which can be grouped into three categories: 1) encoder-decoder interactive error control approach [5]–[7], 2) error resilient encoding [8]–[14], and 3) error concealment [15]–[21]. The encoder-decoder interactive error control approach introduces additional delays, making it unusable in real-time wireless applications. The error resilient mechanisms presented in [8], [10], [11], [13], and [14] require additional complexities in both the encoder and the decoder while the solutions in [9] and [12] do not manage to detect a number of corrupted MBs. Concealment methods are generally adopted to alleviate the effect of transmission errors by means of interpolation. However, the performance of these methods depends on the error detection capabilities of the decoder, since only the MBs which are detected as being corrupted will be concealed.

The standard H.263++ [22] has limited error detection capabilities with only 40%–60% of the corrupted MBs being detected [8]. For this reason, it generally adopts the error detection capabilities of the transport protocol to detect corrupted segments, which are dropped and concealed. This gives rise to large amounts of pixels being concealed (several of which are undistorted pixels), thereby reducing the quality of the video sequence.

Transmission errors can be detected without any increase in transmission bit rate by exploiting the inherent redundancies within and outside a macroblock [23]–[30] at image level. However, these features achieve low error detection rates when applied individually and some uncorrupted MBs may be detected as erroneous, further reducing the quality of the reconstructed video sequence. The authors in [31] have obtained a 93% error detection rate by combining a number of features and applying heuristic thresholds to detect corrupted MBs at image level. They have also applied an iterative error correction algorithm which has achieved a significant gain in peak signal-to-noise ratio (PSNR) at the expense of a significant increase in complexity. This makes its application unpractical in real-time systems.

Classification methods have already been adopted in the areas of multimedia retrieval [32], [33], error concealment [18], [20], optimal cross-layer strategies for multimedia transmission [34], and several other areas. However, to the knowledge of the authors, classification methods for H.263++ systems were only used to detect visually distorted MBs in [35] where a probabilistic neural network (PNN) was used.

This paper presents a novel error detection and localization algorithm which exploits the error detection capabilities of the

Manuscript received June 30, 2008; revised May 25, 2009. First published August 18, 2009; current version published October 16, 2009. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Qian Zhang.

The authors are with the Department of Communications and Computer Engineering, University of Malta, Msida MSD 2080, Malta (e-mail: rrfarr@eng.um.edu.mt; cjdebo@eng.um.edu.mt).

Digital Object Identifier 10.1109/TMM.2009.2030651

transport protocols to flag potentially corrupted group-of-blocks (GOBs). The algorithm, which utilizes a support vector machine (SVM) classifier at its core, is only applied on the MBs contained within this set of GOBs. Results show that this solution achieves an error detection rate of 97.8%. This means that the area over which the error concealment algorithm is applied is drastically reduced, producing a significant gain in quality. Moreover, PSNR gains of up to 5.33 dB have been experienced. Another important feature is that the SVM classifier is only applied to potentially corrupted MBs, minimizing false detections and ensuring that no loss in quality is experienced when the video is transmitted over an error-free channel.

This paper is organized as follows. An overview of the error detection and concealment algorithms employed in standard codecs and their limitations are presented in Section II. The details concerning the proposed error detection and localization algorithm are presented in Section III, followed by the delivery of simulation results in Section IV. The final comments and conclusion are presented in Section V.

II. PROBLEM DEFINITION

The video coding standards generally adopted in wireless communication systems employ variable length codes (VLCs) to achieve high compression ratios. The compressed video bitstreams are extremely vulnerable to transmission errors, the frequency of which is much more pronounced in wireless environments. The detection of any corrupted MB is dependent on the syntax and semantic violation tests adopted by the codec. However, the corrupted codewords can still form valid entries in the VLC table, and therefore the decoder may not detect the occurrence of an error [26]. This will result in a degradation of the end-to-end quality of the video system, as shown in Fig. 1(a).

To alleviate this problem, video systems generally use the bit-level error detection capabilities of the transport protocols, such as UDP [36]. When a UDP segment, which encapsulates a number of GOBs, is detected as corrupted, it is dropped and all the MBs contained within it concealed. This approach is pessimistic because a number of uncorrupted MBs are also concealed, as shown in Fig. 1(b) (areas between the dashed lines), resulting in a significant reduction in the quality of the reconstructed video sequence. Standard codecs may apply advanced error resilient methods such as slices or interleaving; however, a number of uncorrupted MBs are still concealed.

The solution proposed in this paper utilizes the inherent redundancies present between each MB and its neighboring MBs in both the spatial and the temporal domains to detect and localize the first residual corrupted MB which provides major visual distortions. Hence, this algorithm conceals only the first distorted MB and the following MBs (which are generally distorted due to the loss in synchronization) contained within a GOB. As shown in Fig. 1(c), this method significantly reduces the number of pixels to be concealed, thus improving the quality of the reconstructed sequences.

A corrupted MB may have different levels of distortion, some of which provide major artifacts while others imperceptible artifacts. The visual distortion level of the corrupted MBs used was inspired by the Single Stimulus Test [37]. These levels were scaled according to a survey where the audience was given a set of corrupted and uncorrupted frames to scale in reference to

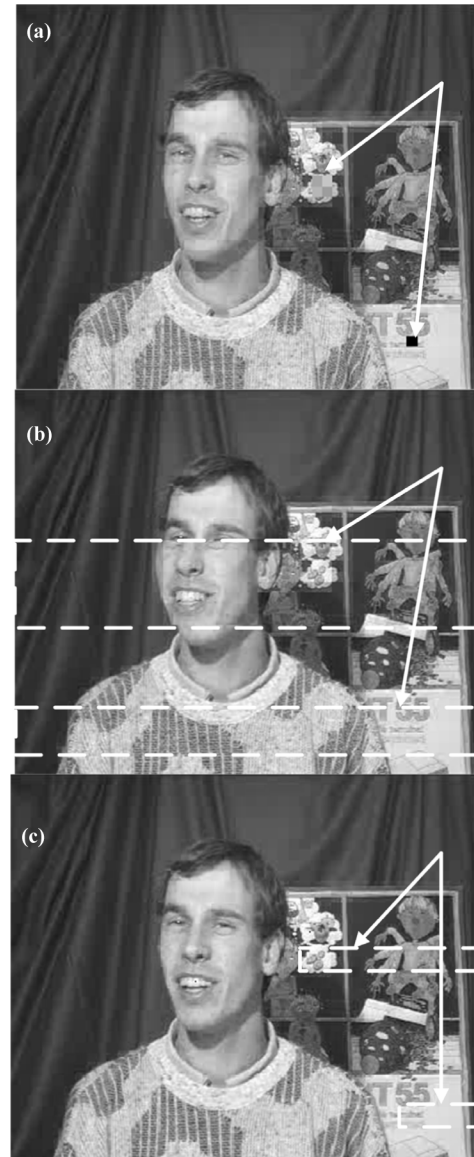


Fig. 1. Reconstructed video using (a) syntax and semantic violations test of the standard codec, (b) UDP checksum test, and (c) proposed solution. The arrows in (a) indicate the visually impaired regions, while in (b) and (c), they indicate the concealed regions.

TABLE I
OBJECTIVE SCALE LEVEL OF VISUAL DISTORTIONS OF THE CONSIDERED MB

Description	Distortion Level
Uncorrupted / Imperceptible	0
Perceptible but not annoying	1
Slightly annoying	2
Annoying	3
Very Annoying	4

Table I. The Mean Opinion Score (MOS) was used to provide a subjective rating of the distortion level of the corrupted MB. The aim of the proposed solution is to maximize the detection of severely damaged MBs while being more lenient with others that provide insignificant loss in the perceptual quality.

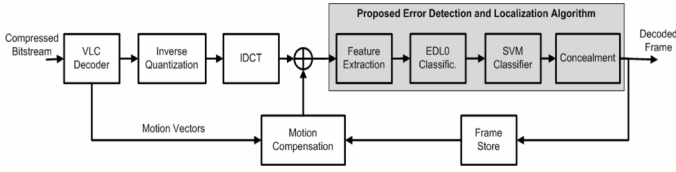
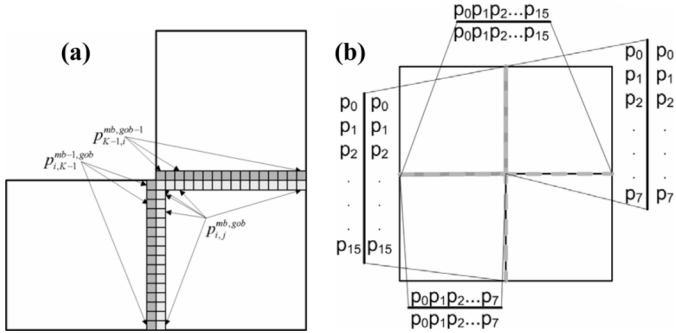


Fig. 2. Block diagram of the proposed error detection algorithm.

Fig. 3. Block analysis for error detection (a) AIDB, (b) IAIDB_{block}, and IAIDB.

III. ERROR DETECTION AND LOCALIZATION ALGORITHM

The proposed algorithm is divided into three layers and follows the standard decoder, as shown in Fig. 2. The feature extraction layer extracts the features which describe the fitness of the MB in the spatio-temporal domain. Some of these features are compared to heuristic thresholds within the Early Distortion Level 0 Classification (EDL0C) module. This module significantly reduces the computational time required by the algorithm as most distortion level 0 (DL0) MBs are detected and immediately classified as uncorrupted, thus bypassing the SVM classification module. This can be done because the features of most of the uncorrupted MBs will lie below the thresholds.

The feature vector of the remaining MBs is passed through an SVM classifier which discriminates between the corrupted and the uncorrupted MBs. This module can be developed using any supervised-learning machine algorithm. However, as it will be illustrated in the following sections, the SVM classifier achieves the highest recognition rates and consequently has been adopted in this work.

A. Feature Extraction

The visual impairments caused by transmission errors generally provide MBs whose color and texture do fit in neither the spatial nor the temporal domains. The set of ten features adopted in our solution exploit these properties and can be divided into two groups: the pixel domain features, which were computed in the perceptually uniform CIE LUV color space model, and the DCT domain features, which were extracted in the DCT domain. These features exploit the inherent redundancy provided by neighboring pixels in order to detect corrupted MBs and are described in some detail in the following subsections.

1) *Average Intersample Difference Across Boundaries (AIDB)*: In natural video sequences, the pixels across a boundary generally vary smoothly. The AIDB [23] feature exploits this characteristic and is illustrated in Fig. 3(a). It computes the sum of Euclidean distances across MB boundaries in

the CIE LUV color space model. To avoid additional delays, only the upper $X_{mb,gob-1}$ and the left $X_{mb-1,gob}$ MBs are considered. The AIDB for a $K \times K$ dimensional MB $X_{mb,gob}$ is given by

$$f_{AIDB} = \frac{1}{2K} \sum_{i=0}^{K-1} \left\{ \left\| p_{i,0}^{mb,gob} - p_{i,K-1}^{mb-1,gob} \right\| + \left\| p_{0,i}^{mb,gob} - p_{K-1,i}^{mb-1,gob} \right\| \right\} \quad (1)$$

where $K = 16$, $p_{i,0}^{mb,gob}$ and $p_{0,i}^{mb,gob}$ represent the leftmost and upper boundary pixels of the current MB, while $p_{i,K-1}^{mb-1,gob}$ and $p_{K-1,i}^{mb-1,gob}$ represent the rightmost boundary pixels of the neighboring MB on the left and the lower boundary pixels of the neighboring MB on the top, respectively.

2) *Internal Average Intersample Difference Across Boundaries Block (IAIDB_{block})*: The IAIDB_{block} [31] feature is similar to the AIDB but exploits the dissimilarity across 8×8 pixels block boundaries instead. As illustrated in Fig. 3(b), the IAIDB_{block} computes the sum of Euclidean distances across boundary block pixels in the CIE LUV color space. The IAIDB_{block} metric for each block b is computed using

$$f_{IAIDB_{block}}^b = \frac{1}{K} \sum_{i=0}^{K/2-1} \left\{ \left\| p_{i,b}^{above} - p_{i,b}^{below} \right\| + \left\| p_{i,b}^{left} - p_{i,b}^{right} \right\| \right\} \quad (2)$$

where in this case, $K = 8$, $p_{i,b}^Z$ is the pixel around the current block b and $Z \in \{ \text{above, below, left, right} \}$. The resultant IAIDB_{block} feature is then derived using

$$f_{IAIDB_{block}} = \max(f_{IAIDB}^b) - \max_2(f_{IAIDB}^b) \quad (3)$$

where $\max(\bullet)$ and $\max_2(\bullet)$ provide the maximum and second maximum from the array, respectively. This feature was designed to detect a single 8×8 corrupted block within an MB.

3) *Internal Average Intersample Difference Across Boundaries (IAIDB)*: The IAIDB feature [31] is required when two corrupted 8×8 blocks are adjacent to each other, as shown in Fig. 3(b). In order to detect these artifacts, the accumulated Euclidean distance between internal horizontal and vertical boundaries which cross the current MB are considered. The IAIDB is given by

$$f_{IAIDB} = \frac{1}{2K} \sum_{i=0}^{K-1} \left\{ \left\| p_{i,K/2-1} - p_{i,K/2} \right\| + \left\| p_{K/2-1,i} - p_{K/2,i} \right\| \right\} \quad (4)$$

where in this case, $K = 16$, and the terms enclosed between the parenthesis, are the Euclidean distance across the vertical and horizontal boundary, respectively. Here, $p_{i,K/2-1}$ and $p_{K/2-1,i}$ represent the leftmost and upper boundary pixels of the current block, while $p_{i,K/2}$ and $p_{K/2,i}$ represent the rightmost boundary pixels of the neighboring block on the left and the lower boundary pixels of the neighboring block on top, respectively.

4) *Sum of Euclidean Pixel Distance (SED)*: The SED [29] feature is based on the fact that the pixel transition of an MB X_t and the corresponding MB in the previous frame X_{t-1} varies smoothly when the MB is not corrupted. This feature tries to

detect entirely corrupted MBs and is given by

$$f_{\text{IAIDB}} = \frac{1}{K^2} \sum_{i=0}^{K-1} \sum_{j=0}^{K-1} \|x_t(i, j) - x_{t-1}(i, j)\| \quad (5)$$

where $K = 16$, and x represent the pixel values with indexes i and j .

5) *Internal Sum of Euclidean Pixel Distance (ISED)*: To enhance the detection of corrupted 8×8 pixel blocks, the *ISED* [29] feature was adopted. This is similar to SED but operates on 8×8 blocks within the MB instead of the MB. The *ISED* metric for each block b is evaluated using

$$f_{\text{ISED}}^b = \frac{4}{K^2} \sum_{i=0}^{K/2-1} \sum_{j=0}^{K/2-1} \|x_{t,b}(i, j) - x_{t-1,b}(i, j)\| \quad (6)$$

where $K = 8$, and x represents the pixel values with indexes i and j . The resulting *ISED* feature is then calculated using

$$f_{\text{ISED}} = \max(f_{\text{ISED}}^b) - \max_2(f_{\text{ISED}}^b) \quad (7)$$

where $\max(\bullet)$ and $\max_2(\bullet)$ provide the maximum and second maximum from the array, respectively.

6) *DCT Domain Features*: The corrupted MBs can be detected using the DCT coefficients, since these contain both color and textural information [28]. Fig. 4(a) illustrates the features that can be used to enhance the error detection capabilities of DCT based codecs, such as H.263++. Each feature contains information about the color, together with 1) vertical, 2) diagonal, and 3) horizontal edge information. These metrics are computed by summing the coefficients, p , along the respective direction and are given in the following:

$$F_1 = \sum_{i=0}^{N-1} p(i, 0) \quad (8)$$

$$F_2 = \sum_{i=0}^{N-1} p(i, i) \quad (9)$$

$$F_3 = \sum_{i=0}^{N-1} p(0, i). \quad (10)$$

The spatial DCT features are based on the fact that there is a correlation between the DCT coefficients of the current block being decoded and the coefficients of the neighboring blocks in space. Therefore, the DCT coefficients of the current MB can be approximated from the surrounding blocks using [30]. Fig. 4(b) illustrates the DCT block prediction algorithm which adopts both upper and left MBs. The DC coefficient is derived using the average DC value from the above and left block. The horizontal AC coefficients are derived from the first row of the block above the one under test while the vertical AC coefficients are derived from the first column of the block to its left.

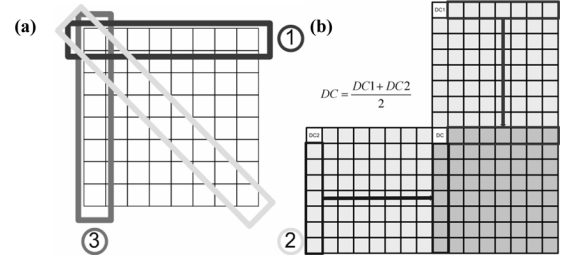


Fig. 4. DCT metrics containing both color and textural information.

The DCT block prediction algorithm derives only the results of the first row and first column of each block. Therefore, only the F_1 and F_3 metrics can be extracted from the predicted DCT block. For each color component, the metrics F_{1p} and F_{3p} are extracted from the predicted block, while F_{1t} and F_{3t} are extracted from the current block under test. The features f_{S1} and f_{S3} can then be computed using (11) at the bottom of the page, where Y and C represent the luminance and chrominance components, respectively.

The temporal DCT features are based on the fact that the difference between the color and textural information of the current block and the corresponding block in the previous frame is usually small when the block is not corrupted. For each color component, the metrics F_{1t-1} , F_{2t-1} and F_{3t-1} are extracted from the previous block using (8)–(10). Similarly, metrics F_{1t} , F_{2t} , and F_{3t} are extracted from the current block. The features f_{T1} , f_{T2} and f_{T3} are then computed using (11).

B. Early DL0 Classification

The EDL0C module combines the features extracted in order to derive four new features. These features were found to offer the best performance following extensive testing during the algorithm development stage. These new features are computed as follows:

$$f_{\text{pel-t}} = f_{\text{SED}} + f_{\text{ISED}} \quad (12)$$

$$f_{\text{pel-s}} = f_{\text{AIDB}} + f_{\text{IAIDB}_{\text{block}}} \quad (13)$$

$$f_{\text{dct-t}} = \frac{1}{3}(f_{T1} + f_{T2} + f_{T3}) \quad (14)$$

$$f_{\text{dct-s}} = \frac{1}{2}(f_{S1} + f_{S3}). \quad (15)$$

All the four features were found to follow distributions similar to the one illustrated in Fig. 5, where most of the uncorrupted MBs have small feature values while corrupted MBs have large feature values. It was further noticed that corrupted MBs whose features are below a certain value generally contain minimal visual distortion.

The EDL0C module thus adopts a global threshold for each feature described in (12)–(15). When a metric is below the threshold, the MB is classified as uncorrupted; otherwise, it

$$f_n = \sqrt{\left\{ \frac{1}{4} \sum_{i=0}^3 |F_{npY(i)} - F_{ntY(i)}| \right\}^2 + |F_{npCb} - F_{ntCb}|^2 + |F_{npCr} - F_{ntCr}|^2} \quad (11)$$

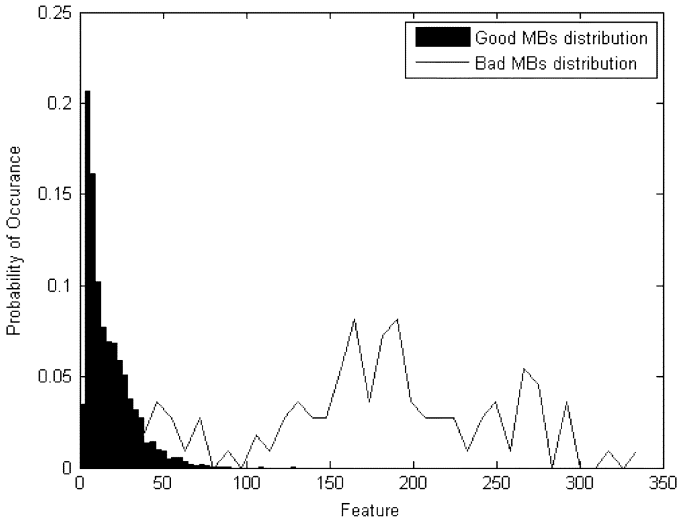


Fig. 5. Typical distribution of the features (12)–(15) for corrupted and uncorrupted macroblocks.

TABLE II
GLOBAL THRESHOLDS CHOSEN FOR THE EDLOC MODULE

Feature	Global Threshold
f_{pel-t}	13.9307
f_{pel-s}	10.8984
f_{dct-t}	62.0000
f_{dct-s}	49.5558

is passed to the following module for further processing. The thresholds used by the EDLOC module are listed in Table II.

C. Support Vector Machine Classifier

The classification method used in the proposed system is based on a support vector machine [38]. The classifier is trained offline using a modified version of the sequential minimal optimization (SMO) [39]. During the training stage, the SVM uses a training set S denoted by

$$S = ((\vec{x}_1, y_1), \dots, (\vec{x}_l, y_l)) \vec{x} \in \mathfrak{R}^N, y \in \{-1, 1\} \quad (16)$$

where \vec{x} are the feature vectors, y_i are their corresponding labels, l is the number of feature vectors, and N is the number of dimensions of the feature vector. In our case, $N = 10$ as discussed in Section III-A and $l = 1000$.

In order to discover nonlinear relations within a linear machine, the SVM employs an implicit nonlinear mapping of the data onto a high-dimensional feature space via a positive semi-definite kernel method K . The Gaussian kernel was used in this work and is given by

$$K(x, y) = \exp\left(\frac{-\|x - y\|^2}{2\sigma^2}\right) \quad (17)$$

where σ is the smoothing parameter, which, after extensive testing, was assigned a value of 0.1969.

The SVM classifier has several advantages in this classification problem. First of all, the probability of misclassifying a previously unseen feature vector drawn at random from a fixed but

unknown probability function is minimized. Secondly, SVMs outperform most classifiers when the training data set is limited [34], and hence, the classifier can be trained with a small amount of training data and still achieve reasonable performance. Finally, an SVM can deal with nonlinearly separable data.

The training of the SVM classifier is highly computational intensive. However, since it is performed offline, it does not affect the real-time performance of the proposed system. In fact, when an unknown vector \vec{x}_i is inputted to the SVM, the MB is assigned to a class based on the following decision rule:

$$f(x) = \sum_{i \in sv} \alpha_i y_i K(\vec{x}_i, \vec{x}) + b \quad (18)$$

where α_i and b are the Lagrange multipliers and the bias derived during training, respectively, and sv are the support vectors. The MB is detected to be corrupted when $f(x) < 0$.

Four video sequences, *Coastguard*, *Erik*, *Paris*, and *Stefan*, were encoded and corrupted at different noise levels. From the resulting sequences, 1000 MBs were chosen at random. The training set was made up of 500 feature vectors representing uncorrupted MBs (Class 0) and another 500 feature vectors representing corrupted MBs (Class 1). The same procedure was used to select the testing set; however, this time, the *Silent*, *News*, *Football*, and *Foreman* sequences were adopted to ensure that the classifier is not biased.

D. Error Detection and Localization Algorithm

The SVM classifier will derive an optimal hyperplane in order to detect and localize corrupted MBs at image level. Although this method achieves high error detection rates (EDR), there will be a number of uncorrupted MBs which will be detected as being corrupted, thus forcing superfluous concealment of uncorrupted MBs even in an error-free channel. The bit-level error detection module applies the error detection capabilities of the transport layer protocols to determine whether or not a packet is corrupted. Whenever a packet is detected as corrupted, the GOBs encapsulated within it are flagged as being potentially corrupted and the features describing each MB are extracted. The EDLOC module examines the flagged MBs and classifies those MBs whose features are below a certain threshold as being uncorrupted. The EDLOC module manages to detect 85.68% of the uncorrupted MBs which will bypass the SVM classifier, thus reducing the complexity of the system. The remaining MBs are then passed through the SVM classifier which detects the corrupted MBs. The schematic diagram of the proposed error detection and localization algorithm is illustrated in Fig. 6.

This procedure significantly reduces the false detection rate (FDR) making the SVM solution feasible. Therefore, in error-free channel conditions, no potentially corrupted GOB will be detected and normal decoding of the compressed bit-stream is applied. This also reduces the computational time required, making it more applicable to real-time applications.

IV. SIMULATION RESULTS

The performance of the algorithm depends on the classification method being used. Six different classification methods have been considered in this work: AIDB-based classification [23], spatial feature-based classification [29], Fisher discriminant analysis (FDA) [40], probabilistic neural networks [41], backpropagation neural networks (BPNN) [42], and support

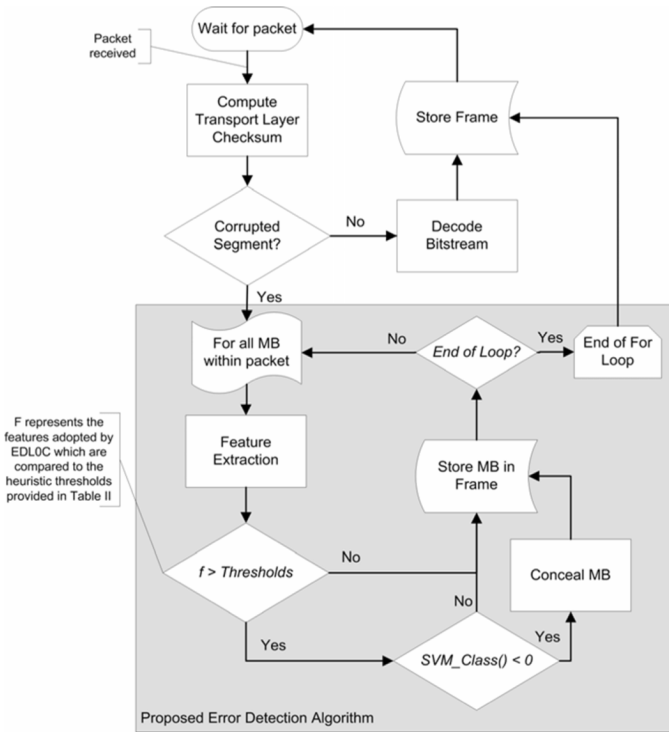


Fig. 6. Proposed error detection and localization algorithm.

vector machines [38]. The last four classification methods were trained with 1000 training vectors of which 500 represented corrupted MBs while the others represented uncorrupted MBs as described in Section III-C. The remaining two classification methods adopt optimal thresholds in order to detect the presence of corrupted MBs.

Table III summarizes the performance of these classification methods in terms of error detection rate (EDR) and false detection rate (FDR). It can immediately be noticed that the SVM achieves by far the best classification and manages to detect on average 97.8% of the corrupted MBs. This result outperforms the best published results in [31] and [35], where a 93% and a 95.74% detection rate were claimed, respectively. On the other hand, the AIDB, FDA, and spatial feature have lower FDR values compared to the SVM classifier at the expense of a low EDR. This means that with these solutions, numerous corrupted MBs are undetected and consequently not concealed. The FDR problem is overcome with the help of the bit-level error detection algorithm which limits the propagation of false detection within potentially corrupted GOBs, and thus, the SVM classifier can never perform worse than the standard codecs relying on the syntax and semantic violations.

In the second experiment, four video sequences were considered, namely, 1) *Silent* sequence which is a typical videoconferencing sequence containing moderate motion, 2) *Football* sequence containing fast moving objects and abrupt camera movements, 3) *News* sequence which is a typical videoconferencing sequence with some movement in the background and shots, and 4) *Foreman* sequence consisting of moderate motion with camera movements. These video sequences were compressed using a baseline H.263++ encoder at CIF resolution and were simulated to be transmitted over a Digital Video Broadcasting

TABLE III
PERFORMANCE OF THE CONSIDERED CLASSIFYING METHODS

Classification Method	EDR	FDR
AIDB	0.4413	0.0160
Spatial Feature	0.6100	0.0140
Fisher Discriminant Analysis	0.7480	0.0020
Probabilistic Neural Network	0.9216	0.0800
Backpropagation Neural Network	0.9080	0.0860
Support Vector Machine	0.9780	0.0440

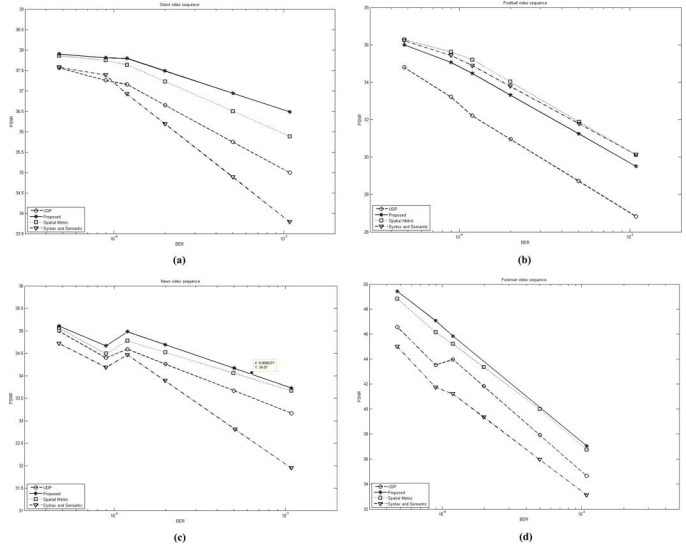


Fig. 7. Performance of the considered error detection mechanisms for (a) Silent, (b) Football, (c) News, and (d) Foreman sequences.

(DVB-RCS) [43] wireless channel, where the compressed bit-stream is encapsulated within UDP segments over IP, a scenario which corresponds to a typical videoconferencing system [36].

Four different error detection mechanisms were considered during simulation: 1) the error detection capabilities of the standard H.263++ decoder which relies on the syntax and semantic violations, 2) the bit-level error detection using UDP checksums and concealment of all the potentially corrupted GOBs as adopted in standard video transmission applications, 3) the proposed bit-level error detection algorithm using UDP checksums with an SVM as classifier module, and 4) the algorithm in 3) with the spatial feature-based classifier replacing the SVM. The MBs which were detected to be corrupted were concealed using the standard concealment method, which replaces the corrupted MBs with the motion compensated MBs from the previous frame. The motion vector was computed from the median of the motion vectors of the neighboring MBs. Fig. 7 illustrates the performance of these algorithms at different BERs.

From these figures, it is evident that the algorithms based on classification methods outperform the other two methods in terms of PSNR. These approaches achieve a significant gain in the quality of the reconstructed video sequences, with the SVM-based method achieving PSNR gains of up to 5.33 dB and 3.54 dB compared to the H.263++ standard decoder and the UDP-based checksum tests.

The proposed algorithm performs best in terms of PSNR for the *Silent*, *News*, and *Foreman* sequences. However, both the

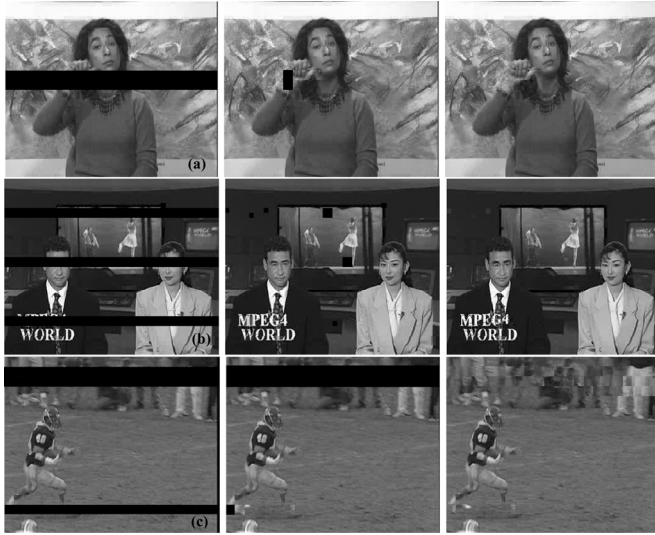


Fig. 8. Subjective evaluation of the video quality for (a) Silent, (b) News, and (c) Football sequences, showing (left) true error region without concealment, (center) the spatial feature with concealment and undetected errors, and (right) SVM classifier with concealment.

spatial feature-based method and the standard H.263++ decoder performed slightly better than the proposed algorithm for the *Football* sequence. This occurs because in the presence of fast moving objects and camera movements, the SVM classifier has a higher FDR and therefore conceals more uncorrupted MBs, resulting in a loss in PSNR. Moreover, the concealment method used does not perform well in the presence of fast moving objects and camera movements, and thus even if a corrupted MB is detected, the quality of the concealed frame will be relatively poor.

The proposed method with the SVM at its core still manages to outperform the other methods in the subjective quality experienced by the end-user. This can be observed in Fig. 8, where the figures on the left indicate the region affected by transmission errors and that will be discarded and concealed by standard codecs. However, most of the MBs enclosed within a GOB are not corrupted and thus completed discarding the GOB generally reduces the quality of the reconstructed video sequence. To enhance the experience of the user, both the spatial and the SVM approaches try to conceal only those MBs which contain significant visual distortions. However, as shown in the central images, the spatial metric does not manage to detect a number of artifacts. This deficiency is corrected by the SVM solution, as illustrated in the right images and summarized in Table IV, where the major visually impaired regions are generally detected. This significant gain in error detection rate over the spatial metric approach corresponds to a gain in QoE.

These results confirm that although applying the SVM may result in more false detections compared to the AIDB, spatial feature, and FDA, which in turn may result in a loss in PSNR, most of the time, better subjective quality is experienced when concealing a few uncorrupted MBs rather than leaving corrupted MBs which provide major visual distortions pass undetected. Furthermore, the FDR of the SVM is relatively small compared to the other machine learning algorithms considered in this work and is bounded within potentially corrupted GOBs.

TABLE IV
ERROR DETECTION RATE OF CORRUPTED MBs AT DIFFERENT OBJECTIVE DISTORTION LEVELS WHEN ADOPTING THE SVM AND SPATIAL FEATURE

Distortion Level	SVM	Spatial Feature
1	0.9655	0.2069
2	0.9175	0.1237
3	0.9772	0.5000
4	1.0000	0.8776

V. COMMENTS AND CONCLUSION

This paper has presented a novel solution which can be used to enhance the QoE through improved error detection capabilities within standard H.263++ decoders. This solution manages to achieve an overall error detection rate of 97.8% by adopting an SVM classifier at its core. In addition, since all classification methods provide false positives, a bit-level error detection mechanism which flags the GOBs which are potentially corrupted is implemented before the SVM. This ensures that the SVM classifier is only applied to the potentially corrupted GOBs, limiting the propagation of false detections. Therefore, the quality experienced when transmitting over an error-free channel is not degraded.

The performance of the proposed algorithm was found to outperform other error detection mechanisms adopted by the standard decoders. The algorithm ensures that only visually distorted MBs are concealed compared to standard techniques that conceal all corrupted packets. This solution has produced PSNR gains of up to 5.33 dB compared to the standard. The result is even more impressive when the video quality is evaluated subjectively, since almost all the MBs which provide major visual distortions are detected and concealed. The results presented in this paper are based on the H.263++ codec; however, similar performance is expected when the algorithm is applied to other video codecs in other video compression standards.

REFERENCES

- [1] F. Hartung, N. Niebert, A. Schieder, R. Rembarz, S. Schmid, and L. Eggert, "Advances in network-supported media delivery in next generation mobile systems," *IEEE Commun. Mag.*, vol. 44, no. 8, pp. 82–89, Aug. 2006.
- [2] J. Farserotu and R. Prasad, "A survey of future broadband multimedia satellite systems, issues and trends," *IEEE Commun. Mag.*, vol. 38, no. 6, pp. 128–133, Jun. 2000.
- [3] A. Jahn, M. Holzbock, J. Müller, R. Keibel, M. de Sanctis, A. Rogoyski, E. Trachtman, O. Franzrahe, M. Werner, and F. Hu, "Evolution of aeronautical communications for personal and multimedia services," *IEEE Commun. Mag.*, pp. 36–43, Jul. 2003.
- [4] M. Ghanbari, *Standard Codecs: Image Compression to Advanced Video Coding*. London, U.K.: IEE Press, 2003.
- [5] J. T. Wang and P. C. Chang, "Error propagation prevention technique for real-time video transmission over ATM networks," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, no. 3, pp. 513–523, Apr. 1999.
- [6] B. Girod and N. Färber, "Feedback-based error control for mobile video transmission," *Proc. IEEE*, vol. 97, no. 10, pp. 1707–1723, Oct. 1999.
- [7] M. Budagavi and J. D. Gibson, "Multiframe video coding for improved performance over wireless channels," *IEEE Trans. Image Process.*, vol. 10, no. 2, pp. 252–265, Feb. 2001.
- [8] W. Park and B. Jeon, "Error detection and recovery by hiding information into video bitstream using Fragile watermarking," in *Proc. SPIE Visual Communications and Image Processing 2002*, Jan. 2002, vol. 4671, pp. 1–10.

- [9] J. Wen and J. D. Villasenor, "Reversible variable length codes for robust image and video transmission," in *Proc. 31st Asilomar Conf. Signals, Systems, and Computers*, Pacific Grove, CA, Nov. 1997, pp. 973–979.
- [10] M. Chen, Y. He, and R. L. Lagendijk, "A Fragile watermarking error detection scheme for wireless video communications," *IEEE Trans. Multimedia*, vol. 7, no. 2, pp. 201–211, Apr. 2005.
- [11] A. Yilmaz and A. A. Alatan, "Error concealment of video sequences by data hiding," in *IEEE Proc. 10th Int. Conf. Image Processing*, Barcelona, Spain, Sep. 2003, pp. 679–682.
- [12] Y. Zhang, W. Gao, Q. Huang, and D. Zhao, "Joint source-channel rate-distortion optimization for H.264 video coding over error-prone networks," *IEEE Trans. Multimedia*, vol. 9, no. 3, pp. 445–454, Apr. 2007.
- [13] H. Wang, S. A. Tsafaris, and A. K. Jatsaggelos, "Joint source-channel coding for wireless object-based video communications utilizing data hiding," *IEEE Trans. Image Process.*, vol. 15, no. 8, pp. 2158–2169, Aug. 2006.
- [14] J. Song and K. J. R. Liu, "A data embedding video coding scheme for error prone channels," *IEEE Trans. Multimedia*, vol. 3, no. 4, pp. 415–423, Dec. 2001.
- [15] Y. Wang and Q. Zhu, "Error control and concealment for video communication: A review," *Proc. IEEE*, vol. 86, no. 5, pp. 974–997, May 1998.
- [16] J. Suh and Y. Ho, "Error concealment techniques for digital TV," *IEEE Trans. Broadcast.*, vol. 48, no. 4, pp. 299–306, Dec. 2002.
- [17] S. Tsekeridou and I. Pitas, "MPEG-2 error concealment based on block-matching principles," *IEEE Trans. Circuit Syst. Video Technol.*, vol. 10, no. 4, pp. 646–658, Jun. 2000.
- [18] J. Kim, "Geometric-based error concealment for concealing transmission errors and improving visual quality," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 8, pp. 974–981, Aug. 2006.
- [19] S. C. Hsia, S. C. Cheng, and S. W. Chou, "Efficient adaptive error concealment technique for video decoding system," *IEEE Trans. Multimedia*, vol. 7, no. 5, pp. 860–868, Oct. 2005.
- [20] M. Chen, Y. Zheng, and M. Wu, "Classification-based spatial error concealment for visual communications," *EURASIP J. Appl. Signal Process.*, vol. 2006, pp. 57–74, Article ID 13438, 2006.
- [21] Y. Wang, S. Wenger, J. Wen, and A. K. Katsaggelos, "Error resilient video coding techniques," *IEEE Signal Process. Mag.*, vol. 17, no. 4, pp. 61–82, Jul. 2000.
- [22] Video Coding for Low Bit-Rate Communications, 2005, ITU-T Rec. H.263.
- [23] W. J. Chu and J. J. Leou, "Detection and concealment of transmission errors in H.261 images," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 8, no. 1, pp. 74–84, Feb. 1998.
- [24] H. C. Shyu and J. J. Leou, "Detection and concealment of transmission errors in MPEG-2 images—A genetic algorithm approach," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, no. 6, pp. 937–948, Sep. 1999.
- [25] S. Ye, X. Lin, and Q. Sun, "Content based error detection and concealment for image transmission over wireless channel," in *Proc. IEEE Int. Symp. Circuits and Systems.*, Bangkok, Thailand, 2003, pp. 368–371.
- [26] M. R. Pickering, M. R. Frater, and J. F. Arnold, "A statistical error detection technique for low bit-rate video," in *Proc. IEEE TENCON Conf.*, Brisbane, Australia, Dec. 1997, pp. 773–776.
- [27] O. Lehtoranta, T. D. Härmäläinen, and V. Lappalainen, "Detecting corrupted intra macroblocks in H.263 video," in *Proc. IEEE Workshop Multimedia Signal Processing*, St. Thomas, Virgin Islands, Dec. 2002, pp. 33–36.
- [28] K. Bhattacharyya and H. S. Jamadagni, "DCT coefficient-based error detection technique for compressed video stream," in *Proc. IEEE Int. Conf. Multimedia and Expo.*, Lausanne, Switzerland, Aug. 2000, pp. 1483–1486.
- [29] R. A. Farrugia and C. J. Debono, "Enhancing the error detection capabilities of the standard video decoder using pixel domain dissimilarity metrics," in *Proc. IEEE EUROCON 2007 Conf.*, Warsaw, Poland, Sep. 2007.
- [30] R. A. Farrugia and C. J. Debono, "Enhancing the error detection capabilities of the DCT based codecs using compressed domain dissimilarity metrics," in *Proc. IEEE Proc. EUROCON 2007 Conf.*, Warsaw, Poland, Sep. 2007.
- [31] E. Khan, S. Lehmann, H. Gunji, and M. Ghanbari, "Iterative error detection and correction of H.263 coded video for wireless networks," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 14, no. 12, pp. 1294–1307, Dec. 2004.
- [32] M. Dimitrova, H. J. Zhang, B. Shahraray, I. Sezan, T. Huang, and A. Zakhor, "Applications of video content analysis and retrieval," *IEEE Multimedia*, vol. 9, no. 3, pp. 42–55, Jul.–Sep. 2002.
- [33] J. Fan, H. Lua, Y. Gao, and R. Jain, "Incorporating concept ontology for hierarchical video classification, annotation, and visualization," *IEEE Trans. Multimedia*, vol. 9, no. 5, pp. 939–957, Aug. 2007.
- [34] M. van der Schaar, D. S. Turaga, and R. Wong, "Classification-based system for cross-layer optimized wireless video transmission," *IEEE Trans. Multimedia*, vol. 8, no. 5, pp. 1082–1095, Oct. 2006.
- [35] R. A. Farrugia and C. J. Debono, "Enhancing error resilience in wireless transmitted compressed video sequences through a probabilistic neural network core," in *Proc. IEEE Int. Picture Coding Symp.*, Lisbon, Portugal, 2007.
- [36] R. A. Farrugia and C. J. Debono, "Modeling the end-to-end quality of a practical videoconferencing system," in *Proc. OPNETWORK 2006 Conf.*, Washington, DC, Aug. 2006.
- [37] Subjective Video Quality Assessment Methods for Multimedia Applications, 1999, ITU-T Rec. P.910.
- [38] N. Cristianini and J. D. Taylor, *Support Vector Machines and Other Kernel-Based Learning Methods*. Cambridge, U.K.: Cambridge Univ. Press, 2000.
- [39] S. S. Keerthi, S. K. Shevade, C. Bhattacharyya, and K. R. K. Murthy, "Improvements to Platt's SMO algorithm for SVM classifier design," *Neural Comp.*, vol. 13, pp. 637–649, Mar. 2001.
- [40] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd ed. New York: Wiley, 2000, pp. 161–350.
- [41] D. F. Specht, "Probabilistic neural networks for classification, mapping or associative memory," in *Proc. IEEE Int. Conf. Neural Networks*, San Diego, CA, Jul. 1988, pp. 525–532.
- [42] M. M. Gupta, L. Jin, and N. Homma, *Static and Dynamic Neural Networks*. New York: Wiley/IEEE Press, 2003, pp. 105–171.
- [43] R. A. Farrugia and C. J. Debono, "A statistical bit error generator for emulation of complex forward error correction schemes," in *Proc. IEEE Int. Conf. Communications*, Glasgow, U.K., Jun. 2007.



Reuben A. Farrugia (S'04) received the first degree in electronic engineering from the University of Malta, Msida, Malta. He is currently pursuing the Ph.D. degree in communications engineering with the Department of Communications and Computer Engineering at the University of Malta.

Between 2004 and 2007, he was employed as a Research Engineer with the Department of Communications and Computer Engineering of the University of Malta, and in January 2008, he was appointed Assistant Lecturer with the same department. His research

interests are in robust multimedia transmission, multimedia processing, forward error correction codes, and machine learning.



Carl James Debono (S'97–M'01–SM'07) received the B.Eng. (Hons.) degree in electrical engineering from the University of Malta, Msida, Malta, in 1997 and the Ph.D. degree in electronics and computer engineering from the University of Pavia, Pavia, Italy, in 2000.

Between 1997 and 2001, he was employed as a Research Engineer in the area of Integrated Circuit Design with the Department of Microelectronics at the University of Malta. In 2000, he was also engaged as a Research Associate with Texas A&M University,

College Station. In 2001, he was appointed as a Lecturer with the Department of Communications and Computer Engineering at the University of Malta and is now a Senior Lecturer. His research interests are in RF and microwave systems development and applications, resilient multimedia transmission, and modeling of communication systems.

Dr. Debono is currently chair of the IEEE Malta Section.