Quality Optimization of H.264/AVC Video Transmission over Noisy Environments Using a Sparse Regression Framework

K. Pandremmenou[a], N. Tziortziotis[a], S. Paluri[b], W. Zhang[c],
K. Blekas[a], L. P. Kondi[a], S. Kumar[c]

[a]Department of Computer Science and Engineering, University of Ioannina,
Ioannina, GR 45110, Greece;
[b]Computational Science Research Center, San Diego State University,
San Diego, CA 92182, USA;
[c]Department of Electrical and Computer Engineering, San Diego State University,
San Diego, CA 92182, USA

ABSTRACT

We propose the use of the Least Absolute Shrinkage and Selection Operator (LASSO) regression method in order to predict the Cumulative Mean Squared Error (CMSE), incurred by the loss of individual slices in video transmission. We extract a number of quality-relevant features from the H.264/AVC video sequences, which are given as input to the LASSO. This method has the benefit of not only keeping a subset of the features that have the strongest effects towards video quality, but also produces accurate CMSE predictions. Particularly, we study the LASSO regression through two different architectures; the Global LASSO (G.LASSO) and Local LASSO (L.LASSO). In G.LASSO, a single regression model is trained for all slice types together, while in L.LASSO, motivated by the fact that the values for some features are closely dependent on the considered slice type, each slice type has its own regression model, in an effort to improve LASSO's prediction capability. Based on the predicted CMSE values, we group the video slices into four priority classes. Additionally, we consider a video transmission scenario over a noisy channel, where Unequal Error Protection (UEP) is applied to all prioritized slices. The provided results demonstrate the efficiency of LASSO in estimating CMSE with high accuracy, using only a few features.

Keywords: Cumulative mean squared error, LASSO, slice loss, slice prioritization, unequal error protection.

1. INTRODUCTION

Nowadays, more and more users access the Internet services through their mobile phones, tablets or laptops. However, practical limitations of the various network infrastructures, such as bandwidth constraints, create the need for video compression in order to save data storage space or enhance the channel's transmission capacity. However, it is widely accepted that the process of compression leads to video quality deterioration, where depending upon its intensity, it can introduce different visual artifacts that would decrease the perceptual video quality as compared to the original version of the video. Additionally, quality degradation can also occur due to packet losses in video transmissions over lossy networks. Losses of video data in a network can happen for a number of reasons such as network fluctuations, multi-path fading, channel congestion, and any operational management procedure. Therefore, suitable error-resilient mechanisms should be developed in order to cope with such issues and guarantee high end-to-end Quality of Service (QoS).

In most video transmission scenarios, the contents of some parts of a video are considered to be more important than others, and thus, better protection should be provided to ensure higher reliability of the crucial pieces of information. For this purpose, Unequal Error Protection (UEP) is applied to accommodate for receivers with

E-mail: apandrem@cs.uoi.gr (K. Pandremmenou), ntziorzi@cs.uoi.gr (N. Tziortziotis), spaluri@mail.sdsu.edu (S. Paluri), zhangw@rohan.sdsu.edu (W. Zhang), kblekas@cs.uoi.gr (K. Blekas), lkon@cs.uoi.gr (L. P. Kondi), skumar@mail.sdsu.edu (S. Kumar).

Visual Information Processing and Communication VI, edited by Amir Said, Onur G. Guleryuz,
Robert L. Stevenson, Proc. of SPIE-IS&T Electronic Imaging, SPIE Vol. 9410, 94100D
 · CCC code: 0277-786X/15/$18 · doi: 10.1117/12.2077243

poor link quality and assure a more graceful video degradation.[1,2] In the literature, several studies attempt to evaluate the packet loss effect on video quality. For the estimation of the distortion caused by packet losses and the resulting error propagation, the authors in Ref. 3 proposed a model that evaluates video quality degradation in terms of Peak Signal to Noise Ratio (PSNR). In the same work, a distortion-based video packet prioritization mechanism for streaming over networks, which can be used for UEP, was also introduced. P. Pérez et al.[4] proposed a model for measuring the packet loss effect, where it was shown that packet priority and packet loss prediction models significantly improve the network Quality of Experience (QoE).

In Ref. 5 the effect of packet loss in terms of Mean Squared Error (MSE) on video quality directly from the video bitstream was studied through the use of three different methods; a model that exploits only network-level measurements, another that extracts the spatiotemporal extent of the impact of the loss and yet another model that extracts sequence-specific information including spatiotemporal activity and effects of error propagation. The goal of Ref. 6 was to develop a packet loss visibility model, applicable to different Group Of Picture (GOP) structures. The effectiveness of the proposed model was validated on a packet prioritization scenario, where the network gets congested at an intermediate router and the router has to decide about which packets to drop such that the visual quality of the video is minimally impacted. Furthermore, in the same work, a Generalized Linear Model (GLM) was employed in order to predict the probability that a packet loss will be visible to a viewer. In Ref. 7 a GLM-based model was presented, targeting at slice prioritization of video flows for real-time H.264/AVC streaming, based on the estimated values of Cumulative MSE (CMSE) incurred by individual slice losses. In addition, the GLM-based CMSE prediction model developed in Ref. 8 was evaluated in a large variety of GOP structures and lengths, as well as in different encoding bitrates. Also, the same work examined both the cases of GOP-level slice prioritization and frame-level slice prioritization.

The present article is mainly focused on improving the accuracy of CMSE estimations, through the use of the Least Absolute Shrinkage and Selection Operator Regression (LASSO) tool.[9–11] More precisely, the framework of this paper is summarized as follows. Initially, we calculate the true CMSE values as they result from the loss of each individual slice. In the following, a number of quality-relevant features are extracted from each slice of a video sequence during the encoding process, in order to be used for the estimation of the CMSE values. LASSO regression is performed so as to indicate the most important features of the dataset, providing accurate CMSE estimations at the same time. Based on the CMSE measured and estimated values, we group the slices into four priority classes, respectively, and a scenario where video sequences are transmitted over an Additive White Gaussian Noise (AWGN) channel, by applying UEP is also considered.

To the best of our knowledge, LASSO regression is applied for the first time to address slice prioritization issues, principally aiming at precise CMSE estimations, which will next guide the quartile-based slice classification. The specific linear regression method is able to select the features that have the strongest effects towards video quality, rejecting those that do not essentially capture the effect incurred by each individual slice loss. In this way, the problem's complexity is significantly reduced, since now a smaller set of features is necessary for CMSE estimation, something especially important in time-critical applications. Moreover, LASSO approach is simpler compared to regression techniques that require additional methods for feature selection prior to the estimation of a response variable (for example, combining Principal Component Analysis (PCA) for feature selection with Ordinary Least Squares (OLS) for the estimation of the response variable).

Particularly, we study the LASSO regression through two different architectures; Global LASSO (G.LASSO) and Local LASSO (L.LASSO). In G.LASSO, a single regression model is trained for all slice types together. Moreover, motivated by the fact that the values for some features are closely dependent on the considered slice type, in L.LASSO we examine the case where each slice type has its own sparse regression model, in an effort to capture more precisely the effect of a slice loss. In addition, in L.LASSO the estimation results for the separate models are combined so as to compute the performance statistics for all slice types together. The experimental results prove that the CMSE estimations using both LASSO approaches closely follow the measured values, and this fact is explained by the low percentages of slice misclassification errors, the provided measures of model's performance as well as the results that arise from the video transmission scenario. Nevertheless, L.LASSO is proved to be more robust and adaptive compared to G.LASSO, both using only a few features for making estimations.

The rest of the article is organized as follows: Section 2 presents the features we extract from each video slice, and Section 3 includes the regression method used. Section 4 describes the procedure followed in order to perform the slice prioritization, while Section 5 elaborates on a scenario, where the CMSE-based prioritized bitstreams are transmitted over an AWGN channel. Lastly, in Section 6 conclusions are drawn.

2. FEATURES CAPTURING SLICE DISTORTION

It is well-known that the loss of a slice can introduce distortion not only in the frame where the slice loss occurs, but also in the subsequent frames belonging in the same GOP, due to error propagation. In this context, we summarize here the features we extract from each slice of the video sequences, which affect perceptual video quality. Specifically:

- **Motx, moty** represent the mean motion vectors for the x and y directions, respectively, averaged over all the MacroBlocks (MBs) in a slice. These features are calculated so as to represent the magnitude of the slice distortion in both x and y directions.
- **Avginterparts** refers to the number of MB sub-partitions averaged over all the MBs in a slice. The higher the motion of a video scene, the higher the "avginterparts" value and vice versa.
- **Maxresengy** is equal to the maximum residual energy of a MB, over all the MBs included in a slice. The residual energy for a MB is computed by taking the sum of squares of all its integer transform coefficients, after motion compensation. It is to be noted that if a video scene includes high motion, the "maxresengy" value would be high.
- **Sigmean, Sigvar** correspond to the mean and variance, respectively, of the Y-component of the signal.
- **Tmdr** captures the error propagation length due to a slice loss, which is heavily dependent on the considered slice type.
- **Imse** captures the exact error measurement in terms of the MSE between the corresponding slices of the reconstructed frames without a slice loss and the reconstructed frames with possible slice losses, after applying error concealment at the decoder.
- **Issim** captures the exact error measurement in terms of the initial Structural SIMilarity (SSIM) index,[12] between the corresponding slices of the reconstructed frames without a slice loss and the reconstructed frames with possible slice losses, after applying error concealment at the decoder.

In addition, it is worth mentioning that we label each slice as of Instantaneous Decoding Refresh (IDR), Predictive (P) and Bidirectionally predictive (B) type. Also, in this paper, the **CMSE** index is employed in order to account for the impact of individual slice losses on video quality, by accurately describing the error propagation within a GOP. It is computed by systematically discarding one video slice at each time and summing the MSE of the current and subsequent frames in the same GOP. The specific index correlates reasonably well with subjective assessment and it does not involve the conduction of subjective tests. In fact, subjective assessment is a time-consuming and costly process that should be carefully designed and performed under specific conditions, and the human viewers are not always willing or available to perform the specific task. Therefore, CMSE provides the "ground truth" of video distortion in this work, and based on its measurements, we are able to prioritize the slices.

3. SPARSE REGRESSION MODELING

In order to produce the CMSE estimations, we need to apply a regression model able to perform the specific task. Through the literature there are various types of regression models that have been used in several applications, including hidden Markov models, polynomial and spline regression models, AutoRegressive Moving Average (ARMA) models or even Gaussian processes. However, these methods are suffering from the drawback of not automatically addressing the problem of model order selection, which is a very important issue in regression. If the order of the regression model is too large, it overfits the observations and does not generalize well. On the other hand if it is too small, it might miss trends in the data.

Sparse Bayesian regression offers a convenient solution to the problem of CMSE estimation, by introducing an l_1 penalty term on the model parameters. Enforcing sparsity is a fundamental machine learning regularization principle and has been used to tackle several problems, such as feature selection. LASSO[9–11] is such a penalized regression method for simultaneous feature selection and regression coefficient estimation that has received a great deal of attention in recent years, due to its generalization capabilities. The key idea is that during the training process, the least important features are assigned regression coefficients that are equal to zero, and only a few of them are retained as significant. Also, two additional LASSO features are that it is able to improve the estimation accuracy of ill-posed problems, and produces interpretable models like subset selection, by exhibiting the stability of ridge regression[13] at the same time.[9]

Let us first describe a linear regression model. A linear regression model is a model of the form:

$$\hat{y}_i = w_0 + \sum_{j=1}^{m-1} w_j \phi_j(x_i) = w^\top \phi(x_i), \quad \text{for } i = 1, \ldots, n \tag{1}$$

where n is the total number of observations, namely the total number of examined slices of all frames of the examined video sequences; $\hat{y}_i$ is the estimated value of CMSE at observation x_i; the basis function $\phi(x_i)$ is a vector of m-by-1 values at observation x_i, which includes the values for all examined features for a particular slice, and w is an m-by-1 vector of regression coefficients including the intercept factor w_0. Such a model is *linear* in the coefficients w.

Feature selection in regression is crucial when a variety of input features are available and we wish to select only the most important of them for the efficient estimation of a response variable. LASSO is able to simultaneously select features and produce estimations, while it also features the benefit of not only shrinking some regression coefficients close to zero, but also setting some others exactly to zero, producing interpretable models. The specific method minimizes the residual sum of squares subject to the sum of the absolute value of the regression coefficients being less than a constant. In other words, for a given nonnegative λ value, LASSO solves the following minimization problem:

$$\min_w \left(\frac{1}{2} \sum_{i=1}^{n} \left(y_i - w^\top \phi(x_i) \right)^2 + \frac{\lambda}{2} \sum_{i=1}^{n} |w_i| \right). \tag{2}$$

The regression coefficients w for the LASSO methodology have no closed form and the solution involves quadratic programming techniques from convex optimization. The tuning parameter λ controls the amount of regularization, meaning that the larger the λ values are, the more regression coefficients are driven to zero, leading to a sparse model representation. Alternatively, for $\lambda = 0$, no shrinkage is performed. It is worth mentioning that in this study, a set of regularization coefficients λ within a predefined range were examined in a preliminary dataset, where the $\lambda > 0$ value that corresponds to the lowest MSE for each of the models was selected for the rest of our experiments.

For our experiments we used the video database of Refs. 7,8. Table 1 shows the regression coefficient estimates obtained by using both LASSO approaches (G.LASSO and L.LASSO), and the selected λ values per case are also included in the same table. From Table 1, LASSO's sparsity is obvious, since it keeps only a small subset of the features required by the GLM model presented in Refs. 7, 8 in order to produce CMSE estimations. We observe that only six features out of the 13 in total are employed by G.LASSO and similarly, six features on average for all slice types are employed by L.LASSO as well.

The Pearson Correlation Coefficient (PCC), the Spearman Rank Order Correlation Coefficient (SROCC) and the Root MSE (RMSE) measures of performance were used to evaluate the effectiveness of the regression models, as recommended by Video Quality Experts Group (VQEG).[14] Table 2 summarizes these results for the "Foreman" and "Akiyo" video sequences, each of 300 frames, encoded at 1 Mbps, at Common Intermediate Format (CIF) resolution. The slice size is 300 bytes for the "Foreman" and 600 bytes for the "Akiyo", while the GOP structure is IDR B P B and IDR P P P, respectively.

Features	G.LASSO	L.LASSO		
		IDR	P	B
0.intercept	15.4	1.2	-2.8	22.1
1.motx	0.053	**0**	0.152	**0**
2.moty	**0**	**0**	-0.057	-0.024
3.avginterparts	-6.30	**0**	-3.29	-0.02
4.maxresengy	**0**	**0**	-3.64×10^{-9}	**0**
5.sigmean	**0**	-0.0222	**0**	**0**
6.sigvar	2.1×10^{-3}	**0**	9×10^{-4}	**0**
7.slice type f2	**0**	**0**	**0**	**0**
8.slice type f3	**0**	**0**	**0**	**0**
9.tmdr	-0.466	**0**	-0.008	**0**
10.imse	0.624	12.637	4.101	0.530
11.issim	**0**	**0**	**0**	-21.9
12.imse$\times$tmdr	0.560	0.239	0.384	0.443
13.imse$\times$maxresengy	**0**	**0**	-2.40×10^{-11}	**0**
λ	2.2742	2.0932	2.3002	0.8802

Table 1: Regression coefficients and employed λ values.

A close inspection of the results of both tested video sequences reveals that both LASSO architectures are able to provide accurate CMSE estimations, as it is clear from the provided PCC values. Also, the SROCC results show that the CMSE estimations are monotonically related with the measured values, following an increasing monotonic trend, and the small RMSE values signify that the CMSE estimations closely follow the measured ones. A comparison between the G.LASSO and L.LASSO models justifies our choice of building a regression model, separately for each considered slice type. As it is evident from the same table (Table 2), improved performance statistics are obtained in such a case, as compared to the coarser approach of building a single regression model for all slice types together. In addition, it is to be noted that our proposed models behave better when the "Akiyo" video sequence is assessed compared to the "Foreman" video sequence, in terms of the PCC and RMSE results. However, "Foreman" gathers better SROCC results, meaning that there is a higher monotonic relationship between the measured and estimated CMSE results as compared to "Akiyo".

	"Foreman"		"Akiyo"	
	G.LASSO	L.LASSO	G.LASSO	L.LASSO
PCC	0.8108	**0.8282**	0.9789	**0.9908**
SROCC	0.8745	**0.8930**	0.7199	**0.8671**
RMSE	29.5563	**28.3504**	21.7261	**12.9547**

Table 2: Performance statistics.

In the following, Figs. 1 and 2 graphically illustrate the measured CMSE values as well as the estimated CMSE values achieved by both LASSO approaches, for the "Foreman" and "Akiyo" video sequences, respectively, when each of the slices consisting a video sequence is assumed to get lost. Both figures verify our conclusions about the efficiency of the proposed schemes drawn earlier by examining the performance statistics for both tested video sequences.

4. SLICE PRIORITIZATION

Based on the measured and estimated CMSE values using both G.LASSO and L.LASSO, we classify the slices into four priority classes applying a Quartile-Based Prioritization (QBP) scheme. Our ultimate goal is to assign different channel coding rates to each differently prioritized class in order to enhance the video quality that reaches the end-user.

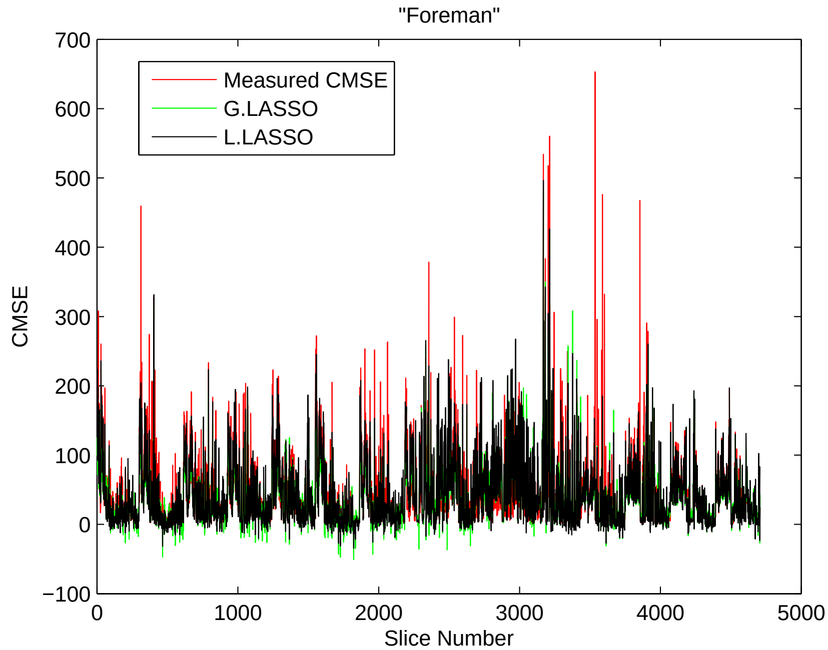

Figure 1: Measured versus estimated CMSE for “Foreman” video sequence.

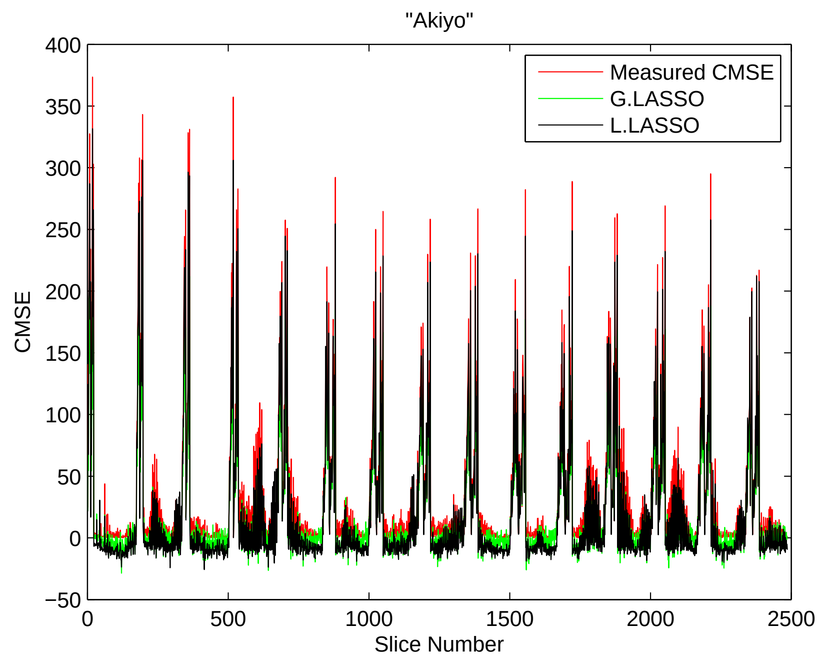

Figure 2: Measured versus estimated CMSE for “Akiyo” video sequence.

With regard to the QBP procedure, it can be described as follows. The CMSE values (measured and estimated) are sorted in ascending order and we calculate the median value, which is the middle value of the dataset. The same procedure is also followed with the lower and upper half of the dataset. Therefore, the 75^{th}, 50^{th} and 25^{th} quartiles that result, split the CMSE values into four priority classes. The class including the highest CMSE values corresponds to the 1^{st} priority class, the class including the CMSE values between the 75^{th} and 50^{th} quartiles corresponds to the 2^{nd} priority class, the 3^{rd} priority class includes the CMSE values that fall within the 50^{th} and 25^{th} quartiles, while the class with the lowest CMSE values represents the 4^{th} priority class.

Since QBP is conducted on both the measured and estimated CMSE values, the performance comparison between measured CMSE values and estimated CMSE values using both LASSO approaches is straightforward. Let us assume that a slice is assigned a priority p, for $p = 1, 2, 3, 4$, based on its measured CMSE value. We consider a first degree misclassification error if the priority assigned to the corresponding slice based on its estimated CMSE value is $p \pm 1$ and second and third degree misclassification errors, when the assigned slice priorities based on the estimated CMSE values are $p \pm 2$ and $p \pm 3$, respectively.

Two interesting remarks are that i) a first degree slice misclassification error is less important than a second or third degree misclassification error, since it represents a moderate CMSE estimation error compared to the other cases where the estimated values are considerably different from the measured ones, ii) the minimization of the CMSE estimation error of higher priority slices is more essential, since a possible loss of the specific slices incurs a stronger impact on perceptual quality.

"Foreman"						
	Priority 1			Priority 2		
	$1^{\text{st}} \to 2^{\text{nd}}$	$1^{\text{st}} \to 3^{\text{rd}}$	$1^{\text{st}} \to 4^{\text{th}}$	$2^{\text{nd}} \to 1^{\text{st}}$	$2^{\text{nd}} \to 3^{\text{rd}}$	$2^{\text{nd}} \to 4^{\text{th}}$
G.LASSO	4.59%	0.55%	0.00%	4.50%	5.35%	0.19%
L.LASSO	4.08%	0.96%	0.00%	4.63%	3.99%	0.64%
	Priority 3			Priority 4		
	$3^{\text{rd}} \to 1^{\text{st}}$	$3^{\text{rd}} \to 2^{\text{nd}}$	$3^{\text{rd}} \to 4^{\text{th}}$	$4^{\text{th}} \to 1^{\text{st}}$	$4^{\text{th}} \to 2^{\text{nd}}$	$4^{\text{th}} \to 3^{\text{rd}}$
G.LASSO	0.55%	4.65%	7.24%	0.08%	0.81%	6.54%
L.LASSO	0.38%	4.74%	4.74%	0.02%	0.45%	4.91%

Table 3: Percentages of slice misclassifications for the "Foreman" video sequence.

"Akiyo"						
	Priority 1			Priority 2		
	$1^{\text{st}} \to 2^{\text{nd}}$	$1^{\text{st}} \to 3^{\text{rd}}$	$1^{\text{st}} \to 4^{\text{th}}$	$2^{\text{nd}} \to 1^{\text{st}}$	$2^{\text{nd}} \to 3^{\text{rd}}$	$2^{\text{nd}} \to 4^{\text{th}}$
G.LASSO	1.73%	0.00%	0.00%	1.69%	4.26%	5.43%
L.LASSO	1.33%	0.00%	0.00%	1.33%	3.90%	1.01%
	Priority 3			Priority 4		
	$3^{\text{rd}} \to 1^{\text{st}}$	$3^{\text{rd}} \to 2^{\text{nd}}$	$3^{\text{rd}} \to 4^{\text{th}}$	$4^{\text{th}} \to 1^{\text{st}}$	$4^{\text{th}} \to 2^{\text{nd}}$	$4^{\text{th}} \to 3^{\text{rd}}$
G.LASSO	0.04%	5.71%	10.70%	0.00%	3.94%	12.19%
L.LASSO	0.00%	4.02%	9.25%	0.00%	0.88%	9.37%

Table 4: Percentages of slice misclassifications for the "Akiyo" video sequence.

Tables 3 and 4 depict the percentages of slice misclassifications, for the "Foreman" and "Akiyo" video sequences, respectively. From both tables, we confirm that the misclassification errors are low and the most common case of misclassification is observed between "neighboring" priority classes, meaning that our CMSE estimations do not differ significantly from the corresponding measured CMSE values. Additionally, the misclassification errors of the 1^{st} priority class are considerably lower compared to those of the other priority classes, irrespectively of the followed LASSO approach. This fact is especially important since by providing a stronger channel coding rate during wireless transmissions to the slices belonging to the specific priority class, we will assure an improved end-to-end video quality. Furthermore, as it was also expected by the results of Table 2, L.LASSO achieves lower slice misclassification percentages compared to the G.LASSO approach.

5. VIDEO TRANSMISSION SCENARIO

In order to assess the performance of the proposed CMSE estimation models, we consider a scenario, where the measured and estimated CMSE-based prioritized bitstreams are transmitted over an AWGN channel. The goal is the optimal determination of the channel coding rate, R_p, for each of the four priorities, $p = 1, 2, 3, 4$, in each GOP of each considered bitstream that could lead to average PSNR enhancement, by applying UEP.[7,8]

Let R_{CH} be the transmission bit rate of the channel in bits per second. The video is encoded at a frame rate of f_s frames per second, and the total outgoing bit budget B for a GOP of length L_G, is given by:

$$B = \frac{R_{CH}\, L_G}{f_s}.$$

The Rate Compatible Punctured Convolutional (RCPC) code rates[15] are chosen from a candidate set $\mathbf{R}$ of punctured code rates $\{R_1, R_2, \ldots, R_K\}$, while the expected video distortion within a GOP is the sum of the slice loss distortion over the AWGN channel. The expected distortion of the j^{th} slice depends on the CMSE due to its loss, $D(j)$, and the slice loss probability for a given channel Signal to Noise Ratio (SNR). The slice loss probability depends on i) the slice size $S(j)$ and ii) the bit error probability, p_b, after channel decoding. Likewise, p_b depends on the channel SNR and the channel coding rate, $R_p \in \mathbf{R}$, for a given priority p.

Accordingly, the optimization problem is formulated as follows:[7,8]

$$\{R_1^*, R_2^*, R_3^*, R_4^*\} = \arg \min_{R_1,R_2,R_3,R_4} \left\{ \sum_{p=1}^{4} \sum_{j=1}^{n_p} \left[1 - (1 - p_b(SNR, R_p))^{\frac{S(j)}{R_p}} \right] D(j) \} \right\} \tag{3}$$

subject to the following constraints:

1. $\sum_{p=1}^{4} \sum_{j=1}^{n_p} \frac{S(j)}{R_p} \leqslant B$
2. $R_1^* \leq R_2^* \leq R_3^* \leq R_4^*$.

The first constraint is the channel bitrate constraint and the second constraint guarantees that a higher priority CMSE class is assigned at least an equal or a stronger channel coding rate compared to a lower CMSE class. The resulting optimization problems are tackled using the *genetic algorithms toolbox* available in Matlab,[16] while the optimization is performed separately for each GOP of the video sequence, in order to avoid overflow at each second (one GOP lasts 2/3 second).

In this work, we consider that the channel bit rate is $R_{CH} = 2$ Mbps, and the GOP consists of 20 frames. The RCPC code rates that each priority class can be assigned are given by the set:

$$\mathbf{R} = \{8/9,\ 8/10,\ 8/12,\ 8/14,\ 8/16,\ 8/18,\ 8/20,\ 8/22,\ 8/24,\ 8/26,\ 8/28,\ 8/30,\ 8/32\}.$$

The mother code rate is 1/4 with memory M = 4 and puncturing period P = 8.

Figure 3 presents the average PSNR values for the “Foreman” video sequence computed over 100 realizations, for each given SNR value ($[0 \sim 5]$ dB). Apparently, as we observe from the simulation results, the PSNR is reducing for a lower SNR channel consideration, due to a higher slice loss probability caused by more channel errors. In addition, the higher the channel SNR the closer the estimated CMSE values are to the measured ones. Moreover, we can see that when UEP is applied to the CMSE estimations achieved by L.LASSO, the average PSNR values are more close to the measured ones, when compared to G.LASSO case, while this is more obvious for low channel SNR (0 dB and 1 dB). It is to be noted that for channel SNR 4 dB and 5 dB, both G.LASSO and L.LASSO achieve virtually identical performance with the measured CMSE case, as Fig. 3 depicts.

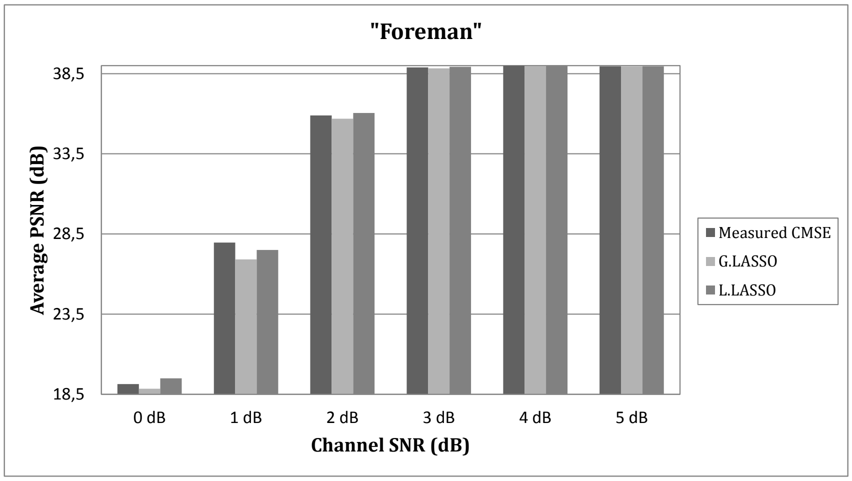

Figure 3: Average PSNR performance for "Foreman" video sequence transmitted over an AWGN channel.

6. CONCLUSIONS

In this work, we focus on the problem of accurately estimating the CMSE within each GOP of a video sequence, incurred due to possible slice losses during wireless video transmissions over noisy channels. Initially, we calculate the actual CMSE values, assuming that each individual slice is lost, and use this index as the "ground truth" of perceptual distortion. Continuing, we extract a number of features from a large collection of H.264/AVC video sequences and use all feature observations as input to the LASSO model. The specific regression technique is able to provide accurate CMSE estimations, by selecting only a subset of the features; the ones that are the most influential towards CMSE estimations. In more detail, we consider two different LASSO architectures; in G.LASSO a single regression model is built for all slice types, while in L.LASSO a separate regression model is considered for each different slice type. Based on the measured and estimated CMSE values, we group the slices into four priority classes for each case, by using a QBP scheme. Last, in order to evaluate the performance of our proposed scheme, we assume a video transmission scenario over an AWGN channel, where UEP is applied to each priority class, within each GOP of a video sequence. The provided experimental results verify the suitability of the employed video features as well as the efficiency of the proposed regression models in producing precise CMSE estimations. In addition, as it was expected, L.LASSO is proved to be a better choice than G.LASSO for enhancing further the accuracy of CMSE estimations. The slice misclassification percentages are low, the performance statistics high, and the average PSNR values of the video transmission scenario achieved by the proposed models are virtually identical to the PSNR values resulting from the measured CMSE case, especially for a channel SNR of 3 dB or higher.

REFERENCES

[1] Vosoughi, A., Cosman, P., and Milstein, L., "Joint source-channel coding and unequal error protection for video plus depth," *IEEE Signal Processing Letters* **22**, 31–34 (Jan. 2015).

[2] Zhou, C., Lin, C.-W., Zhang, X., and Guo, Z., "A novel JSCC scheme for UEP-based scalable video transmission over MIMO systems," *IEEE Transactions on Circuits and Systems for Video Technology* (2015). To appear.

[3] Bouazizi, I., "Estimation of packet loss effects on video quality," in [*First International Symposium on Control, Communications and Signal Processing*], 91–94 (2004).

[4] Pérez, P., Macias, J., Ruiz, J. J., and Garcia, N., "Effect of packet loss in video quality of experience," *Bell Labs Technical Journal* **16**, 91–104 (Jun. 2011).

[5] Reibman, A., Vaishampayan, V., and Sermadevi, Y., "Quality monitoring of video over a packet network," *IEEE Transactions on Multimedia* **6**, 327–334 (Apr. 2004).

[6] Lin, T.-L., Kanumuri, S., Zhi, Y., Poole, D., Cosman, P., and Reibman, A., "A versatile model for packet loss visibility and its application to packet prioritization," *IEEE Transactions on Image Processing* **19**, 722–735 (Mar. 2010).

[7] Paluri, S., Kambhatla, K., Kumar, S., Bailey, B., Cosman, P., and Matyjas, J., "Predicting slice loss distortion in H.264/AVC video for low complexity data prioritization," in [*19th IEEE International Conference on Image Processing (ICIP)*], 689–692 (Sept. 2012).

[8] Paluri, S., Kambhatla, K., Bailey, B., Cosman, P., Matyjas, and Kumar, S., "A low complexity model for predicting slice loss distortion for prioritizing H.264/AVC video," *Multimedia Tools and Applications* (Nov. 2014).

[9] Tibshirani, R., "Regression shrinkage and selection via the Lasso," *Journal of the Royal Statistical Society, Series B* **58**, 267–288 (1994).

[10] Tibshirani, R., "The Lasso method for variable selection in the cox model," *Statist. Med.* **16**(4), 385–395 (1997).

[11] Osborne, M., Presnell, B., and Turlach, B., "A new approach to variable selection in least squares problems," *IMA journal of numerical analysis* **20**(3), 389 (2000).

[12] Wang, Z., Bovik, A., Sheikh, H., and Simoncelli, E., "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing* **13**, 600–612 (Apr. 2004).

[13] Marquardt, D. W. and Snee, R. D., "Ridge regression in practice," *The American Statistician* **29**(1), 3–20 (1975).

[14] "Full reference television phase II subjective test plans," (2002). Objective Quality Model Evaluation Criteria.

[15] Hagenauer, J., "Rate-compatible punctured convolutional codes (RCPC codes) and their applications," *IEEE Transactions on Communications* **36**, 389–400 (Apr. 1988).

[16] MathWorks, "Global Optimization Toolbox: Users' Guide (R2011b)," (Nov. 2011).