

## **H.264/AVC Motion Vector Concealment Solutions Using Online and Offline Polynomial Regression**

Tamer Shanableh

Department of Computer Science and Engineering

American University of Sharjah, UAE

Fax: +971 6 515-2979

[tshanableh@aus.edu](mailto:tshanableh@aus.edu)

Khaled Assaleh

Electrical Engineering Department

American University of Sharjah, UAE

Fax: +971 6 515-2979

[kassaleh@aus.edu](mailto:kassaleh@aus.edu)

### **Abstract**

This paper introduces two polynomial regression solutions for error concealment by predicting the values of motion vectors of lost macroblocks. The two solutions are online and offline polynomial regression modeling. In the former solution, the regression model is built during the decoding process whilst in the latter solution; the model is built during the encoding or the transcoding process and then used at the decoder for concealment. Both solutions make use of the spatially and temporally neighboring motion vectors for building the regression models. The advantages and disadvantages of the proposed solutions are elaborated upon. In comparison to existing work, the experimental results show that the proposed solutions have clear advantages of computational time requirements and motion vector prediction accuracy.

**Keywords:** Video compression, error concealment, machine learning, regression.

## 1. Introduction

The H.264/AVC video codec [1] is designed with error resiliency capabilities. The error resiliency tools available can be divided into two categories. In one category, video is streamed from a pre-encoded content and there is no feedback channel available. In the other category, the video is streamed from a live source and a feedback channel is available. In the former category the following tools are available. Data partitioning which enables unequal error protection. Flexible MacroBlock (MB) ordering enables scattering errors to the whole frame. Redundant slices enable the decoder to replace a corrupted or a lost slice. And lastly, adaptive intra refresh reduces error propagation. On the other hand, in the latter category where a feedback channel is available, the above-mentioned tools are used in addition to two tools which are SP-frames and reference frame selection which are used to eliminate temporal error propagation without the use of I-frames. An introduction to the above tools can be found in [2].

However, in the H.264/AVC codec, error concealment is a normative part of the ISO/IEC standard, therefore different vendors can implement their own error concealment strategies. In general, to recover from errors, a compliant decoder starts by detecting the existence of an error and resynchronizes the decoding process. This is then followed by error concealment which can make use of the available error resiliency tools and apply various spatio-temporal error concealment.

The focus of this paper is on temporal error concealment through the use of regression models to predict the values of Motion Vectors (MVs) of a lost MB. Temporal error concealment is not specific to H.264/AVC, it has been previously proposed for MPEG-1 and MPEG-2 video as proposed in [3]. It was proposed to recover MV values of P-frame through the vectorial sum of forward and backward MVs of B-frames. It was also proposed to make use of neighboring MVs to compile a list of candidate MVs for concealment. Many of the existing concealment solutions were then customized and enhanced for the H.264/AVC codec with the major difference of the introduction of variable size motion compensation blocks and reference frame selection.

In the literature, error concealment solutions for recovering from the loss of full frames have been reported. Examples of such solutions are reported in [4] and [5]. It is more common though to assume slice loss as

opposed to frame loss since H.264/AVC coding is based on slice structures. Examples of partial frame concealment was proposed in [6] to recover sub-block MVs using weighting boundary match which is suitable for concealing areas with high motion activity. A hybrid concealment solution is proposed in [7] where it was proposed to use the temporal activity of the prediction source and the coding types of neighboring MBs to decide on the concealment mode to use (i.e. spatial or temporal). The outer boundaries of a lost MB can also be used to search for a good match in the previous frame as reported in [8]. Regression models are also used at the decoder to predict the values of lost MVs. For instance [9] and [10] proposed to construct regression models based on the spatial relationship between sub-block locations and their motion vectors.

The aforementioned existing work on error concealment using regression falls under online regression modeling where the regression model is built at the decoder when needed. In this work, we extend the existing work in terms of proposing an offline regression solution where the regression models are built prior to decoding. We also propose an online regression solution that takes both the spatial and the temporal dimensions into account. The latter solution is shown to offer higher concealment quality. On the other hand the former solution has an attractive feature of reduced processing time where the model building is done prior to decoding.

The paper is organized as follows. Section 2 introduces both the online and the offline regression concealment solutions. Section 3 presents the experimental results. Lastly Section 4 concludes the paper.

## **2. Proposed polynomial regression solutions**

In this work, a lost MB is concealed by predicting its MVs from the neighboring vertical, horizontal and temporal MVs. Four MVs from each direction are used for predicting the MVs of a lost MB. We assume that MBs are partitioned into 4x4 sub-blocks. If the MB partitioning is larger than 4x4, then the MV value is replicated accordingly. Likewise, the concealment is applied to 4x4 sub-blocks in a lost MB.

We propose two solutions for the MV concealment. The first solution is based on online regression where the regression model can be built and used at the decoder side. The second solution is based on offline regression where the regression model is built at the encoder or transcoder and used at the decoder. Each

solution has its advantages and disadvantages. Namely, the online regression solution does not require passing model parameters to the decoder. On the other hand, the offline regression model shifts the complexity of computing the model parameters away from the decoder. In the experimental results section, we further elaborate on the difference between the two solutions in terms of concealment accuracy and computational time requirements.

## 2.1 Online regression modeling

The input to the polynomial regression model in this solution are the predictor variables represented by the vector  $\mathbf{p}_d$  and the response variable represented by vector  $\mathbf{r}_d$ . The subscript refers to the direction of neighboring MVs used for building the model. When the MVs on the left or right of the lost MB are used for building the model then the direction is horizontal ‘ $h$ ’. Likewise the direction is vertical ‘ $v$ ’ if the MVs above or below the lost MB are used and lastly, the direction is temporal ‘ $t$ ’ if the co-located MVs from the previous frames are used. In all cases, a 4x4 partition size is assumed and 4 MVs are used from the neighboring MBs, i.e. left, right, top, bottom. In the temporal direction on the other hand, 4 MVs from the previous 4 co-located MBs are used. The predictor variables in this solution are simply the spatial indices of the neighboring MVs which will be represented as negative integers. On the other hand, the response variables are the actual values of the corresponding MVs. Clearly, the prediction models are repeated for both the  $x$  and  $y$  MV components. It is worth mentioning that H.264/AVC allows the use of dispersed MB ordering in which every other MB is assigned to a different slice. Hence when a slice loss occurs, the top, bottom, left and right MBs are available for concealment [12].

The proposed online regression solution is further illustrated in Figure 1. If the decoder detects a lost slice then each lost MB is concealed individually. Three regression models are built for each MB at the decoder, the model is then used to predict the values of lost MVs which will be used for error concealment.

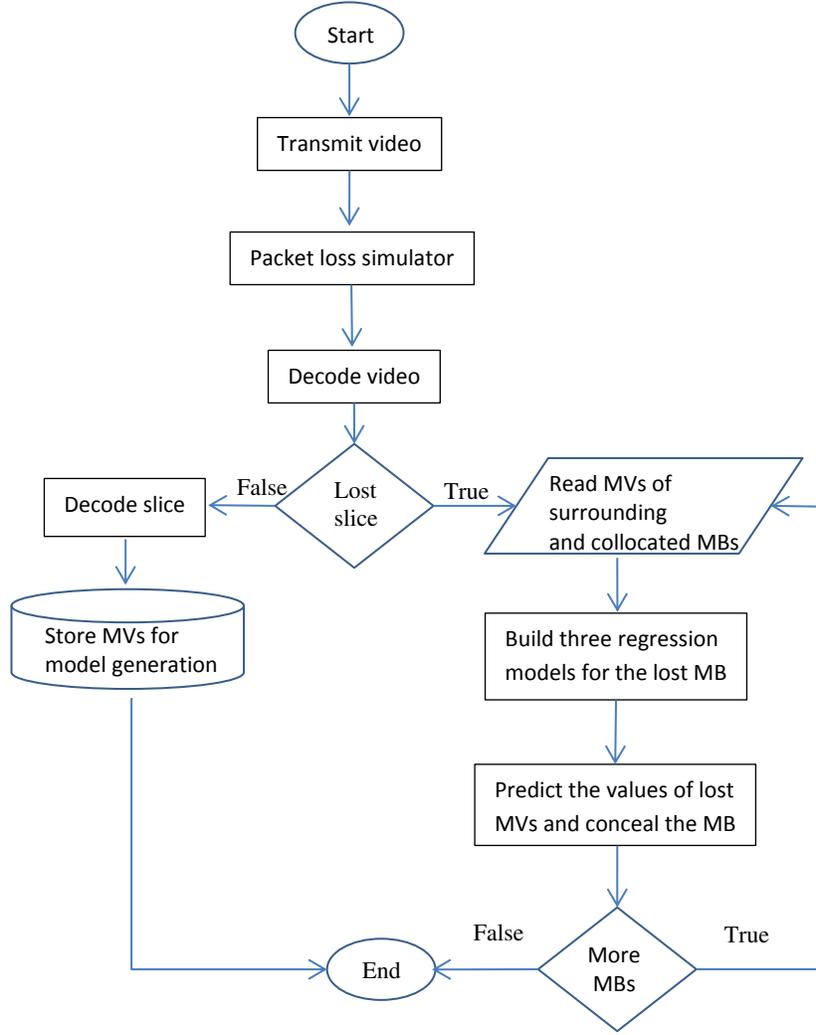


Figure 1. Flowchart illustrating the proposed online regression solution.

A MV is represented as  $V_{i,j,x,y}^t$  where the superscript indicates a frame at time  $t$ . The subscripts indicate the  $x,y$  coordinates of a 4x4 sub-block in a MB at the  $i,j$  coordinates. For a particular example of building regression models for the top left 4x4 sub-block in a lost MB, the predictor variables and response variables are represented by:

$$\mathbf{p}_h = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix} \quad (1), \quad \mathbf{r}_h = \begin{bmatrix} V_{i-1,j,3,0}^t \\ V_{i-1,j,2,0}^t \\ V_{i-1,j,1,0}^t \\ V_{i-1,j,0,0}^t \end{bmatrix} \quad (2), \quad \mathbf{p}_v = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix} \quad (3), \quad \mathbf{r}_v = \begin{bmatrix} V_{i,j-1,0,3}^t \\ V_{i,j-1,0,2}^t \\ V_{i,j-1,0,1}^t \\ V_{i,j-1,0,0}^t \end{bmatrix} \quad (4), \quad \mathbf{p}_t = \begin{bmatrix} 1 \\ 2 \\ 3 \\ 4 \end{bmatrix} \quad (5),$$

$$\mathbf{r}_t = \begin{bmatrix} V_{i,j,0,0}^{t-1} \\ V_{i,j,0,0}^{t-2} \\ V_{i,j,0,0}^{t-3} \\ V_{i,j,0,0}^{t-4} \end{bmatrix} \quad (6)$$

Where  $\mathbf{p}_h, \mathbf{p}_v$  and  $\mathbf{p}_t$  refer to the predictor vectors pertaining to the neighboring MVs in the horizontal, vertical and temporal directions. Likewise  $\mathbf{r}_h, \mathbf{r}_v$  and  $\mathbf{r}_t$  refer to the response vectors containing the values of the neighboring MVs in the horizontal, vertical and temporal directions.

Although the predictor vectors  $\mathbf{p}_h, \mathbf{p}_v$  and  $\mathbf{p}_t$  contain the same values, nonetheless, the spatio-temporal distances between the motion vectors in the spatial and the temporal directions are different. Additionally, in some cases, the MVs of the temporally co-located MBs resemble those of the current MB more than the spatially neighboring MVs and vice versa. Hence, it is important to take the three directions into account using different weights as proposed in Equation (9).

Notice that both the predictors and the response variables pertain to the neighboring MVs only, they do not contain information about the lost MVs. In this solution the regression models are built at the decoder hence the name online regression. For each lost MB the decoder computes three regression models for each of the 4x4 partitions. The models are then used to predict the values of lost MVs. The predictor variables in this case are the spatial indices of lost MVs which are represented as negative numbers.

In [10] it was reported that the best mapping between the predictors and the response variables can be achieved provided that the predictors are expanded into a second order regression model. Such a model contains constant, linear, cross product and squared terms [11]. The mapping of predictors into the response variables is achieved by using mean-squared error as the objective criterion such that:

$$\mathbf{w}_d^{opt} = \arg_{\mathbf{w}} \min \|\mathbf{x}_d \mathbf{w} - \mathbf{r}_d\|_2 \quad (7)$$

Where  $\|\cdot\|_2$  denotes the  $l_2$  norm,  $\mathbf{w}$  is a vector of model weights to be estimated and  $\mathbf{x}_d$  is the expanded predictor vector using polynomial expansion as introduced in [11]. Minimizing the objective function results in:

$$\mathbf{w}_d^{opt} = (\mathbf{x}_d^T \mathbf{x}_d)^{-1} \mathbf{x}_d^T \mathbf{r}_d \quad (8)$$

Where  $\mathbf{w}_d^{opt}$  are the model weights for one of the three prediction directions. The subscript  $d$  refers to ‘ $b$ ’, ‘ $v$ ’ or ‘ $t$ ’. Once the model weights are computed, a lost MV can be predicted by expanding its spatial index into a second order and then multiplying it with the model weights. This results in 3 predicted MVs per 4x4 partition. The predicted MVs can be merged using weighted averaging where the weights are inversely proportional to the variance of the neighboring MVs in that particular direction. In video compression, the homogeneity of neighboring MVs in a given direction indicates a similar motion pattern. Likewise the heterogeneity of MVs in given direction indicates that different parts on the same region are moving in different directions. As such, this homogeneity of motion can be assessed in terms of computing the standard deviation of the motion vectors.

For instance, if the variance of the MVs from the temporal direction is low then the corresponding predicted MV will have a higher weight and so forth. Consider a particular example where  $d$  refers to the temporal direction  $t$ . The corresponding weight used in computing the final MV is computed as:

$$w_{r_t} = 1 - \left( \frac{\sigma_{r_t}}{\sigma_{r_v} + \sigma_{r_h} + \sigma_{r_t}} \right) \quad (9)$$

Where  $\sigma_{r_t}$ ,  $\sigma_{r_v}$  and  $\sigma_{r_h}$  refer to the standard deviation of the neighboring MVs in the temporal, vertical and horizontal directions respectively.

Note that this solution is different than the one proposed in [10] in three ways. The first difference is due to the use of the temporal dimension to construct a third regression model. The second difference is due to the use of weighted averaging as explained in Equation 9. Lastly, a 4x4 partitioning structure is not forced at the encoder; rather, when building the regression models, the MVs are replicated to emulate a 4x4 partitioning structure if needed.

## 2.2 Offline regression modeling

In the second solution, an offline regression approach is proposed in which the model parameters are computed at the encoder or the transcoder and used at the decoder for concealment purposes. Again, 3 models are generated according to the direction of the neighboring MBs used for concealment.

Since the regression models are built at the encoder or transcoder then we can make use of the values of the MVs to be predicted in building the model. More specifically, the predictor variables in this case are generated from the neighboring MVs and the response variables are the actual values of the MVs to be predicted at the decoder. The proposed online regression solution is further illustrated in Figure 2. Part (a) of the figure illustrates the concept of building three regression models that will be used for predicting the values MVs for all lost MBs. This is in contrast to the proposed online regression solution of Section 2.1, where three regression models are built for each lost MB at the decoder side. Part (b) of the figure illustrates the MB concealment at the decoder side. If a slice is lost, then the decoder will predict the lost MVs of each MB using the three model weights that are received by the decoder. Therefore, the concealment process is fast and can be carried out in real-time as shall be illustrated in the experimental results section.

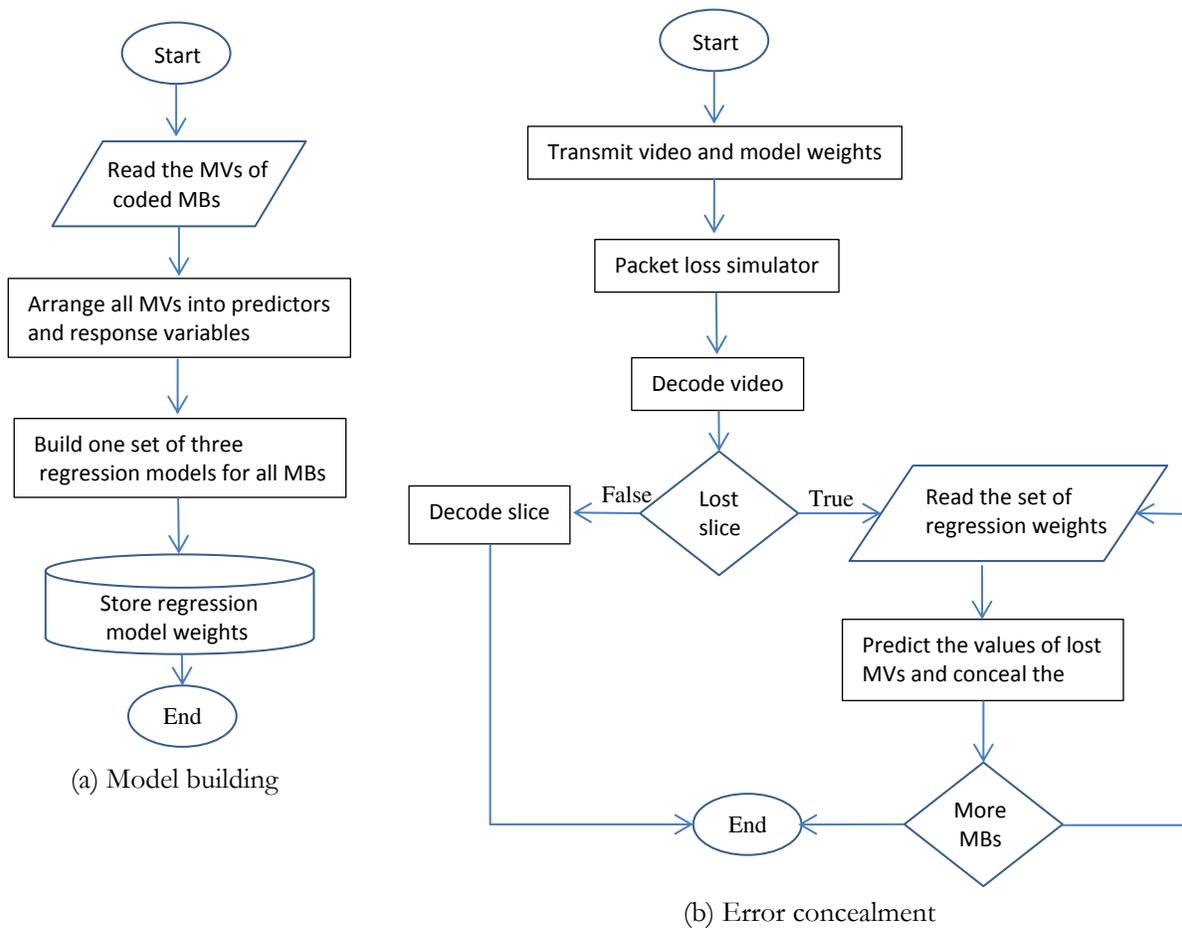


Figure 2. Flowchart illustrating the proposed offline regression solution. (a) Model building (b) Error concealment

The predictor variables are organized in a matrix format and the response variables are represented as a vector. For a particular example, consider the top left 4x4 partition in the lost MB. The predictor variables generated from the horizontal direction for a total of  $F$  frames each with  $I \times J$  MBs are represented by the following matrix:

$$P_h = \begin{bmatrix} 1 & V_{0,0,0,3}^1 & V_{0,0,0,2}^1 & V_{0,0,0,1}^1 & V_{0,0,0,0}^1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & V_{I,J-1,0,3}^1 & V_{I,J-1,0,2}^1 & V_{I,J-1,0,1}^1 & V_{I,J-1,0,0}^1 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & V_{0,0,0,3}^F & V_{0,0,0,2}^F & V_{0,0,0,1}^F & V_{0,0,0,0}^F \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & V_{I,J-1,0,3}^F & V_{I,J-1,0,2}^F & V_{I,J-1,0,1}^F & V_{I,J-1,0,0}^F \end{bmatrix} \quad (10)$$

The first row in the matrix starts with the constant ‘1’, this is required in multiple regression for the computation of the intercept term. The second item of the row;  $V_{0,0,0,3}^1$ , refers to the MV of the 4x4 block at coordinates 0,3 of the MB at index  $i=0, j=0$ . In the second row,  $V_{I,J-1,0,3}^1$  refers to the MV of the 4x4 block at coordinates 0,3 of the MB at index  $i=I, j=J-1$  and so forth.

Denote by  $V_{i,j,0,0}^f$  the value of the MV to be predicted at frame number  $f$ . The corresponding response variables for this example are represented by the following vector:

$$\mathbf{r}_h = [V_{0,1,0,0}^1 \dots V_{I,J,0,0}^1 \dots V_{0,1,0,0}^F \dots V_{I,J,0,0}^F]^t \quad (11)$$

Again, as the case with the online regression solution, the predictor matrix is expanded into a second order regression model. Thereafter, the nonlinear mapping between the expanded predictors and the response variables is performed using a similar arrangement to Equations 7 and 8 above. Unlike the online regression solution, in this case only three model weights are generated for each of the 4x4 sub-blocks, therefore a total of 16 sets of model weights are generated. Clearly one can propose a more robust solution in which the video sequence is split into a number of sub sequences and generate more sets of model weights.

The proposed solutions are compared against each other in terms of prediction and concealment accuracy. They are also compared in terms of computational time requirements.

### 3. Experimental results

The experiments are implemented using the H.264/AVC Joint Model reference software. Video sequences are encoded using the baseline profile. A slice structure is used in which macroblocks are assigned to slices

using dispersed macroblock ordering. As mentioned previously, when a slice is lost, every other macroblock, both horizontally and vertically is available in the bit stream. The motion vectors of such macroblocks are used in the concealment of the lost slice. Following reviewed work of [10], five video sequences are used in the experimental results, namely; Tempete, Foreman, Mobile, Coastguard and Carphone. The sequences are coded at a QCIF spatial resolution with a variable bitrate using a QP value of 16. Each sequence contains 200 frames, every 100<sup>th</sup> frame is intra-coded. This section presents the experimental results of the proposed online and offline regression solutions and compares the results to the reviewed work reported in [10].

One approach to assessing the accuracy of the motion vector prediction is to examine the Sum of Absolute Differences (SAD) between the predicted and the true motion vectors. The SAD results of the proposed solutions and the reviewed work are reported in Table 1. In this experiment the slice loss rate is set to 10%. The results in the table indicate the average SAD of the proposed solutions is less than that of the reviewed work. The results also indicate that the offline regression solution yields the lowest SAD values. The offline regression model globally minimizes the second norm between the predicted and the true motion vectors as indicated in Equation 7. However, it is shown in the experimental results to follow, that such minimization does not necessarily guarantee a higher concealment quality. This is so because a concealment motion vector that minimizes the second norm does not necessarily point to an area in the reference frame that is similar to the true best match area. Nonetheless, the results in Table 1 are indicative of the accuracy of the prediction model.

Sequence	Reviewed	Online regression	Offline regression
Tempete	5.08	3.37	1.84
Foreman	7.22	5.72	3.45
Mobile	2.5	1.76	1.11
Coastguard	3.63	2.73	1.91
Carphone	4.75	4.07	2.95
<b>Average</b>	<b>4.6</b>	<b>3.5</b>	<b>2.3</b>

Table 1. Average Sum of Absolute MV Differences per MB.

The PSNR results of the concealed videos are reported in Table 2. Because of the probabilistic nature of the packet loss simulator, the results are repeated 10 times and the averages are reported in the table. For a fair

comparison, the result of the packet loss simulator for a given repetition is saved and used for assessing the concealment quality of the reviewed and proposed solutions.

Part (a) of the figure reports the concealment results with 10% slice loss rate and part (b) reports the results for with 5% slice loss rate. The results indicate that the average PSNR of the online regression is superior to both the reviewed work and the offline regression. This result is intuitive because the regression model is computed on macroblock basis and it makes use of both spatial and temporal dimensions for concealment as explained previously. The results also indicate that the concealment PSNRs of the reviewed work and the offline regression are very similar. This result is interesting taking into account simplicity of the offline regression solution.

Sequence	Reviewed	Online regression	Offline regression
Tempete	23.87	25.92	25.7
Foreman	27.8	28.9	26
Mobile	23.67	25.85	23.6
Coastguard	26.12	27.9	24.7
Carphone	27.36	28.9	29.1
<b>Average</b>	<b>25.8</b>	<b>27.5</b>	<b>25.8</b>

(a) Slice loss rate of 10%

Sequence	Reviewed	Online regression	Offline regression
Tempete	25.74	28.4	28.39
Foreman	29.5	30.7	27.7
Mobile	25.9	27.5	26.35
Coastguard	27.3	29.1	26.6
Carphone	28	29.3	29.9
<b>Average</b>	<b>27.3</b>	<b>29.0</b>	<b>27.8</b>

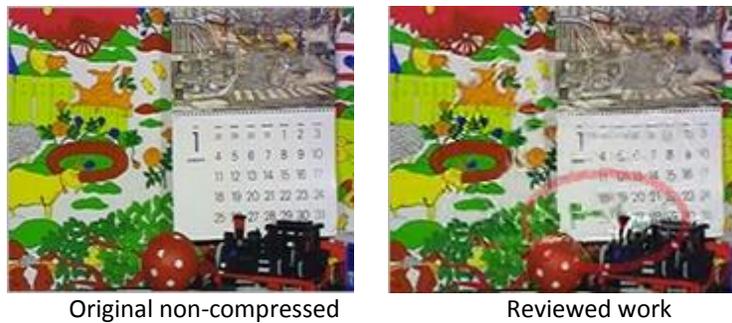
(b) Slice loss rate of 5%

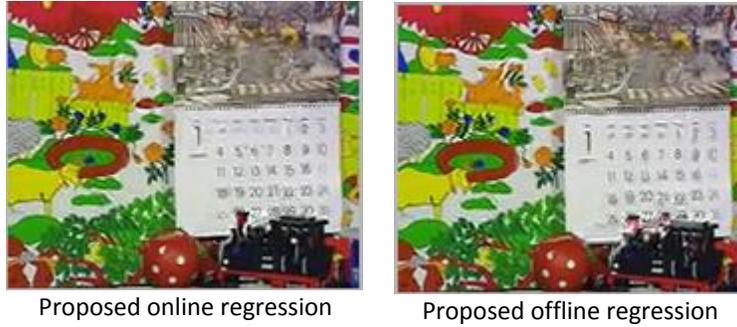
Table 2. Average PSNR with MV concealment.

Repeating the results of Table 2 for the proposed online regression solution without the use of the temporal direction for concealment, i.e. Equations (5) and (6), results in an average concealment quality of 26.7 dB and 28.7 dB for the 10% and 5% slices loss rates accordingly. This indicates the importance of using the temporal direction in error concealment as expected. Lastly, and for completeness, since the packet loss simulator was

ran many times as mentioned above, the average standard deviations of the concealment results are 0.35dB, 0.3dB and 0.29dB for the reviewed, online regression and offline regression accordingly.

In addition to the enhancements in the objective concealment results, subjective enhancements have also been observed. Figure 3 shows a number of examples where the erroneous regions are highlighted. The figure shows the original non-compressed images as a reference as well. The examples are repeated for the Tempete and the Mobiles sequences. It is shown that the proposed solutions result in less visual distortions as opposed to the reviewed work.





(b) Mobile test sequence frame number 60.

Figure 3. Example concealed images using the three concealment solutions of Table 2(a).

To further compare the proposed offline regression solution to the reviewed work, Table 3 examines the computational complexities of both approaches. The average time needed to compute the motion vectors of a concealed slice for all of the above-mentioned video sequences are reported in the table. It is shown that the proposed solution consumes a fraction of the time required to conceal the motion vectors using the reviewed work. This is so because in the offline regression solution, the model weights are computed offline prior to decoding. On the other hand, in the reviewed work, the model weights are computed using the horizontal and vertical information for each of the 4x4 sub-blocks in a MB. That is, the model weights are computed many times per MB during the decoding process. Hence the higher computational complexity as expected. Lastly, in the online regression solution, the computational complexity is expected to be higher than that of the reviewed work. This is so because an additional regression model is built for predicting the MVs based on the temporal direction. Therefore, the superior concealment results come at the expense of higher computational complexity.

	Reviewed	Offline regression	Online regression
Slice concealment time (sec)	0.138	0.03	0.22

Table 3. Average time needed to compute the MVs of a lost slice.

In summary, both of the proposed solutions use regression modeling for the prediction of lost MVs. However the two solutions are different in terms of model building, concealment quality and computational requirements. The online regression solution has the advantage of localizing the regression model to the neighboring region of a lost MB. Hence more accurate concealment results are achieved as reported in Table 2. The offline regression solution on the other hand, minimizes the squared differences between the actual

and predicted MVs globally using information from the whole sequence as reported in Table 1. Moreover, in the latter solution the model weights are transmitted to the decoder since no modeling takes place at the decoder side. Hence low computational requirement for error concealment at the decoder side as reported in Table 3.

#### **4. Conclusion**

Online and offline polynomial regression modeling solutions are proposed for predicting the values of lost MVs in H.264/AVC video. The former solution generated higher concealment quality mainly because it is performed on sub-block basis. The latter solution, on the other hand, reduced the computational complexity at the decoder side because the regression models are built beforehand. In comparison to existing work, the online regression solution scored an average increase of around 2 dB in concealment quality. Additionally, in terms of processing time, the offline regression solution offered a speed up of around 4 times in comparison to existing work. Future work include experimenting with various machine learning approaches for the offline regression solution and adapting the whole concealment solution to the new video coding standard which is known as High Efficiency Video Coding (HEVC).

#### **References**

- [1] Recommendation ITU-T H.264 | ISO/IEC 14496-10:2009, "Advanced Video Coding for generic audio-visual services," March 2009.
- [2] S. Kumar, L. Xu, M. Mandal, and S. Panchanathan, "Error Resiliency Schemes in H.264/AVC Standard," Elsevier J. of Visual Communication & Image Representation, 17(2), April 2006, pp. 425-450.
- [3] T. Shanableh and M. Ghanbari, "Loss Concealment using B-pictures Motion Information," IEEE Transactions on Multimedia. 5(2), June 2003, pp. 257-266.
- [4] B. Yan and H. Gharavi, "A Hybrid Frame Concealment Algorithm for H.264/AVC," IEEE Transactions on Image Processing, 19(1), January 2010, pp. 98-107.

- [5] M.-C. Hwang, J.-H. Kim, H.-Y. Yang, S.-J. Ko and A. Morales, "Frame Error Concealment Technique Using Adaptive Inter-Mode Estimation for H.264/AVC," *IEEE Transactions on Consumer Electronics*, 54(1), February 2008, pp. 1-2.
- [6] X. Yanling, and Z. Yuanhua, "H.264 video communication based refined error concealment schemes," *IEEE Transactions on Consumer Electronics*, 50(4), November 2004, pp. 1135-1141.
- [7] D. Agrafiotis, D. R. Bull, and C. N. Canagarajah, "Enhanced Error Concealment With Mode Selection," *IEEE Transactions on Circuits and Systems for Video Technology*, 16(8), August 2006, pp. 960-973.
- [8] C. Xiaoming, C. Yuk Ying, and B. Changseok, "Dynamic Multi-mode Switching Error Concealment Algorithm for H.264/AVC Video Applications," *IEEE Transactions on Consumer Electronics*, 54(1), January 2008, pp. 154-162.
- [9] X. Chen, Y.-Y. Chung, C. Bae, X. He, and W.-C. Yeh, "An Efficient Error Concealment Algorithm for H.264/AVC using Regression Modeling-Based Prediction," *IEEE Transactions on Consumer Electronics*, 56(4), November 2010, pp. 2694-2701.
- [10] J. Zheng and L.-P. Chau, "Efficient Motion Vector Recovery Algorithm for H.264 Based on a Polynomial Model," *IEEE Transactions on Multimedia*, 7(3), JUNE 2005, pp. 507-513.
- [11] D. G. Kleinbaum, and L. L. Kupper, *Applied Regression Analysis and other Multivariable Methods*, John Wiley & Sons: New Jersey, 2003
- [12] S. Wenger, "H.264/AVC over IP," *IEEE Transactions of Circuits Systems for Video Technology*, 13(7), July 2003, pp. 645-656.