View metadata, citation and similar papers at core.ac.uk

brought to you by CORE

provided by OAR@UM

# IMPROVING MOTION VECTOR PREDICTION USING LINEAR REGRESSION

*Reuben A. Farrugia*

Department of Communications and Computer Engineering
University of Malta, Msida
MSD 2080, Malta
Email: reuben.farrugia@um.edu.mt

## ABSTRACT

The motion vectors take a large portion of the H.264/AVC encoded bitstream. This video coding standard employs predictive coding to minimize the amount of motion vector information to be transmitted. However, the motion vectors still accounts for around 40% of the transmitted bitstream, which suggests further research in this area.

This paper presents an algorithm which employs a feature selection process to select the neighboring motion vectors which are most suitable to predict the motion vectors $\boldsymbol{mv}$ being encoded. The selected motion vectors are then used to approximate $\boldsymbol{mv}$ using Linear Regression. Simulation results have indicated a reduction in Mean Squared Error (MSE) of around 22% which results in reducing the residual error of the predictive coded motion vectors. This suggests that higher compression efficiencies can be achieved using the proposed Linear Regression based motion vector predictor.

***Index Terms***— H.264/AVC, linear regression, machine learning, motion vector prediction, video compression,

## 1. INTRODUCTION

H.264/AVC [1] is the state of the art video coding standard that is being extensively used for the compression and distribution of high definition video content. This is mainly attributed to the fact that H.264/AVC achieves twice the compression achieved by traditional MPEG-2 standard [2]. Moreover, it is more resilient to packet loss, which makes it more attractive for Internet Protocol Television (IPTV) and Video on Demand (VoD) services [3].

The motion vectors, which are transmitted within the H.264/AVC bitstream, can reach up to 40% of the total transmission bitrate [4]. This suggests that several research efforts must be done in order to improve the motion vector predictors that are currently being used by the standard. Several heuristic approaches can be found in literature which tries to combine a set of spatio-temporal neighboring motion vectors to improve the prediction of the current motion vector to be encoded [4, 5, 6, 7]. However, these heuristic approaches lack the scientific rigor that is required to ensure that the optimal set of parameters is being chosen. More recently, the author in [8] has employed linear regression to predict the motion vectors for sequences involving zoom motion. However, this method only considers spatial neighboring motion vectors and is only designed for sequences containing zooming.

This paper presents a novel motion vector prediction which exploits the correlation between spatio-temporal neighboring motion vectors to improve the performance of the motion vector predictor employed by the standard. A Genetic Search was applied on the set of spatio-temporal neighboring motion vectors to identify to motion vectors which are more suitable for prediction. The selected motion vectors are then combined using a linear regression method to derive the motion vector predictor. Simulation results have demonstrated that the proposed method achieves a reduction in Mean Square Error (MSE) of about 22%. This suggests that higher compression efficiencies can be achieved relative to the standard.

This paper is organized as follows: Section 2 presents the neighboring motion vectors that can be used for prediction. The following section describes the feature selection process while Section 4 introduces the Linear Regression method. Section 5 presents the proposed system followed by the simulation results in Section 6. The concluding remarks are drawn in Section 7.

## 2. MOTION VECTOR PREDICTION

The video compression standards available today need to transmit the motion information to be able to decode the video content. However, the overheads required to transmit this information are substantial. For this reason, recent video coding standards encode the motion vectors using lossless predictive coding to reduce the energy of the motion vector residual $\boldsymbol{\xi_{mv}}$ which is given by

$$\boldsymbol{\xi_{mv}} = \boldsymbol{mv} - \boldsymbol{mv_p} \tag{1}$$

where $\boldsymbol{mv}$ is the original motion vector and $\boldsymbol{mv_p}$ is the predicted motion vector. The H.264/AVC standard exploits the correlation between $\boldsymbol{mv}$ and the spatially neighboring motion

vectors (Fig. 1) to derive $\boldsymbol{mv_p}$. This standard generally employs the median between $\boldsymbol{mv_A}$, $\boldsymbol{mv_B}$ and $\boldsymbol{mv_C}$ when all the motion vectors are available. However, in the absence of $\boldsymbol{mv_C}$, the median between $\boldsymbol{mv_A}$, $\boldsymbol{mv_B}$ and $\boldsymbol{mv_D}$ is computed. To simplify the notation, the motion vector predictor provided by the standard is defined as $\boldsymbol{mv_M}$.

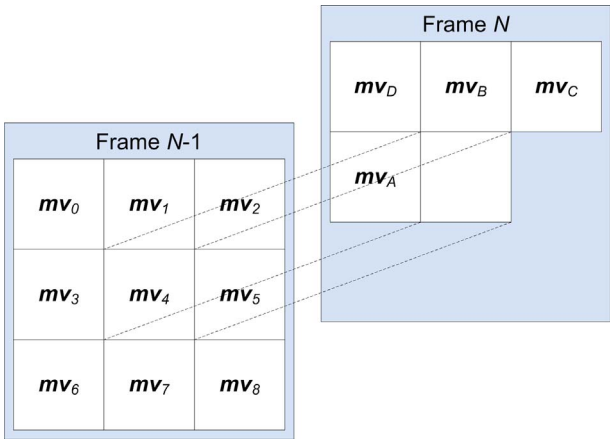


**Fig. 1**. Location of spatial and temporal motion vectors that can be used for prediction.

The predictor adopted by the standard manages to provide a quite accurate motion vector prediction. However, the authors in [4] have demonstrated that the motion vectors are also temporally correlated, and that they can be used to improve the prediction. Previous methods have considered the implementation of heuristic methods in order to exploit the spatiotemporal correlation of the motion vectors to improve motion vector prediction and thus increasing the compression efficiency.

This paper presents the motion vector prediction as a machine learning problem which can be solved using regression. The set of features $\boldsymbol{S}$ available for prediction is made up of the motion vectors $mv_{i,j}$ where $i \in \{A, B, C, D, M, 0, 1, \ldots, 8\}$ and $j \in \{x, y\}$. The aim of the presented method is to derive two functions, one to predict $mv_x$ and another to predict $mv_y$ respectively.

## 3. FEATURE SELECTION

The features residing in space $\Sigma$ do not provide the same amount of information. Some of the features are not correlated to the class and are therefore irrelevant for the regression function [9]. In addition, some features are highly correlated with others and are therefore redundant [10].

Therefore, the feature selection process must choose a subset $\sigma \in \Sigma$ of features which maximize the correlation to the class while minimizing the redundancy across the features. The merit $M_\sigma$, which provides a measure of the correlation of the subset, is calculated using

$$M_\sigma = \frac{k\bar{r}_{cf}}{\sqrt{k + k\,(k-1)\,\bar{r}_{ff}}} \tag{2}$$

where subset $\sigma$ contains $k$ features, $\bar{r}_{cf}$ is the average of the correlation between the components and the class variable and $\bar{r}_{ff}$ is the mean inter-correlation between features. The merit provides a ranking of the feature subset in the search space of all possible feature subsets, where subsets having larger merit values are preferred.

Exhaustive enumeration of all possible feature subsets is prohibitively expensive in terms of computation. The Genetic Algorithm (GA) [11], which is a search algorithm that models natural selection and natural genetics, can be used to derive the subset which maximizes the merit $M_\sigma$. The GA search starts with a population of subsets where the features contained within each subset are selected at random from the set $\Sigma$. Reproduction and crossover is executed according to their merit which assigns higher probability of deriving one or more healthy offspring in the following generations. In order to avoid premature loss of important features, a mutation process is used to occasionally generate subsets containing randomly selected features.

## 4. LINEAR REGRESSION

Linear regression is an approach which models the relationship between the category class $y$ and a set of features $\boldsymbol{x}$ of $D$ dimensions. This model involves a linear combination of the input variables $\boldsymbol{x}$ and the corresponding weights $\boldsymbol{w}$

$$y\,(\boldsymbol{x}, \boldsymbol{w}) = w_0 + w_1 x_1 + \cdots + w_D x_D \tag{3}$$

where $\boldsymbol{x} = (x_1, \ldots, x_D)^T$. The weights were estimated using the least squares method which finds the set of parameters that minimize the sum of the squared difference between the observed responses and the model [12] and is formally given by

$$E = \sum_{i=1}^{N} \left[t_i - y\,(\boldsymbol{x}_i, \boldsymbol{w})\right]^2 \tag{4}$$

where $N$ is the number of training samples, $t_i$ are the corresponding target values and $\boldsymbol{x}_i$ is the $i^{\text{th}}$ training sample.

## 5. PROPOSED SYSTEM

The proposed system, illustrated in Fig. 2, considers all the motion vectors available in the neighborhood $mv_{i,j} \in S_j$ to attain the optimal predictors. However, in order to improve accuracy and simultaneously reduce the computational complexity of the system, the neighboring motion vectors were processed using the Feature Selection process discussed in section 3. The GA Search was executed using a crossover probability of 0.6, mutation probability of 0.033, a population

size of 20 and a maximum number of generations of 20. This parameters were heuristically derived after several simulation tests.

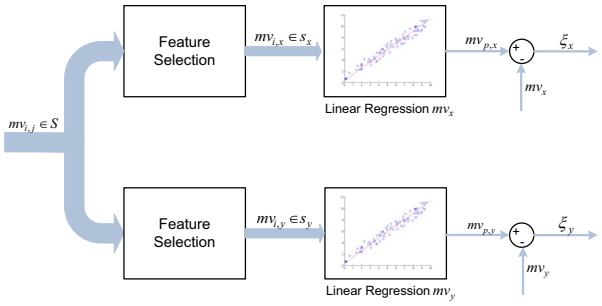


**Fig. 2**. Proposed motion vector prediction using Linear.

The selected set of motion vectors, $mv_{i,x} \in s_x$ and $mv_{i,y} \in s_y$, represent the set of motion vectors which provide the largest merit to predict the original motion vectors $mv_x$ and $mv_y$ respectively. The set

$s_x = \{mv_{A,x}, mv_{B,x}, mv_{1,x}, mv_{7,x}, mv_{M,x}\}$

was selected to predict $mv_x$ while the set

$s_y = \{mv_{A,y}, mv_{B,y}, mv_{4,y}, mv_{M,x}, mv_{M,y}\}$

was selected to predict $mv_y$.

The motion vector predictors, $mv_{p,x}$ and $mv_{p,y}$, were modeled using two different linear regression algorithms using the subsets $s_x$ and $s_y$ respectively. For both cases, a training set consisting of 10,000 samples was used. The linear regression function was trained using 10-fold cross-validation to derive the weights. Using this procedure, it was found that the motion vector predictors can be approximated using

$$mv_{p,x} = \lfloor 0.0818mv_{A,x} - 0.0414mv_{B,x} - 0.0406mv_{1,x} + 0.0527mv_{7,x} + 0.9314mv_{M,x} + 0.0016 \rfloor \quad (5)$$

$$mv_{p,y} = \lfloor 0.1035mv_{A,y} + 0.0359mv_{B,y} + 0.0463mv_{4,y} + 0.0265mv_{M,x} + 0.7861mv_{M,y} + 0.0715 \rfloor \quad (6)$$

## 6. SIMULATION RESULTS

The training and testing samples used in this work were derived using eight different CIF resolution video sequences{ *Coastguard*, *Container*, *Football*, *Foreman*, *Garden*, *Hall*, *Mobile* and *Tennis*} at different data rates ranging from 128kbps up to 1Mbps. All these video sequences were encoded according to H.264/AVC using the JM software [13]. The JM software was configured to enable only P-slice Inter block search $16 \times 16$ while disabled Bi-prediction coding.

**Table 1**. Performance of feature selection to predict $mv_x$

| Parameter | $mv_{i,x} \in S$ | $mv_{i,x} \in s_x$ | Gain |
|---|---|---|---|
| MAE | 14.5287 | 7.8484 | 45.9800% |
| RMSE | 32.0885 | 20.1213 | 37.2944% |
| Time (s) | 87 | 3 | 96.5517% |

**Table 2**. Performance of feature selection to predict $mv_y$

| Parameter | $mv_{i,y} \in S$ | $mv_{i,y} \in s_y$ | Gain |
|---|---|---|---|
| MAE | 17.2711 | 6.9523 | 59.7460% |
| RMSE | 33.2350 | 18.4123 | 44.5997% |
| Time (s) | 64 | 3 | 95.3125% |

A random sample of 20,000 samples was extracted from this database, where half was used for training and the remaining samples were used for testing. The training set was used to derive the features which are more descriptive and to derive the weights of the classification methods considered in this work. Tables 1 and 2 show the performance of the feature extraction method. It can be seen that for both predictions, the application of the genetic search has achieved a significant gain in both reducing the residual error and also reducing the complexity of the classifier. The application of the feature selection process has managed to provide a speedup of around 95% while reducing the Mean Absolute Error (MAE) up to 59%.

The performance of the proposed system was first compared to the predictor adopted by the H.264/AVC. The results are presented in Tables 3 and 4. It can be seen that the proposed method outperforms the Median predictor in both cases achieving an overall gain in terms of Mean Square Error (MSE) of 22%.

It was also considered the case that instead of using linear regression, two other robust regression functions are used. The regression systems considered were the Multilayer Perceptron [14] and the SMO Regression [15]. These classification methods were implemented using the Weka library [16]. It can be seen from Table 3 and 4 that the Linear regression method outperforms the Support Vector Machine approach in both cases. However, the neural approach outperforms the Linear regression to predict $mv_x$ while the Linear regression outperforms the Multilayer Preceptron method to predict $mv_y$. However, when considering the overall performance of the system and the complexity of both methods, it is advisable to adopt the Linear regression in both cases.

## 7. COMMENTS AND CONCLUSION

This paper has presented a novel Linear Regression based motion vector predictor. The proposed method first applies a genetic search to identify the neighboring motion vectors

**Table 3**. Performance of the proposed method to predict $mv_x$ relative to other predictors

| Method | MSE | $C_{\text{Pearson}}$ | $C_{\text{Spearman}}$ |
|---|---|---|---|
| H.264/AVC Median | 577.731 | 0.5807 | 0.6667 |
| Linear Regression | 561.0433 | 0.5866 | 0.6648 |
| **Multilayer Perceptron** | **549.8567** | **0.5913** | 0.6687 |
| SMO Regression | 573.8248 | 0.582 | 0.6694 |

**Table 4**. Performance of the proposed method to predict $mv_y$ relative to other predictors

| Method | MSE | $C_{\text{Pearson}}$ | $C_{\text{Spearman}}$ |
|---|---|---|---|
| H.264/AVC Median | 433.7536 | 0.3642 | 0.5597 |
| **Linear Regression** | **336.6011** | **0.6879** | **0.6189** |
| Multilayer Perceptron | 355.2771 | 0.7015 | 0.6136 |
| SMO Regression | 448.6092 | 0.368 | 0.5634 |

which are most appropriate to aid the prediction of the current motion vector. The is done offline and does not have to be applied during the prediciton process. The Linear regression function is then used to approximate the motion vector. The linear regression function has a little affect on the codec's time complexity.

Simulation results have demonstrated that the proposed method manages to outperform the H.264/AVC standard achieving a MSE reduction of 22%. This suggests that further compression can be achieved by the proposed system. This paper also shows that the Linear regression system outperfms both Multilayer Perceptron and SMO Regression schemes. Furthermore, it was demonstrated that the genetic search manages to reduce both complexity and prediction capabilities of the system.

## 8. REFERENCES


[1] ITU, *H.264: Advanced video coding for generic audiovisual services*, Nov. 2007.

[2] G. J. Sullivan and T. Wiegand, "Video Compression - From Concepts to the H.264/AVC Standard," *Proceedings of the IEEE*, vol. 93, no. 1, pp. 18–31, Jan. 2005.

[3] S. Wenger, "H.264/AVC over IP," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 7, pp. 645–656, July 2003.

[4] Guillaume Laroche, Joel Jung, and Beatrice Pesquet-Popescu, "RD Optimized Coding for Motion Vector Predictor Selection," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 18, no. 9, pp. 1247–1257, Sept. 2008.

[5] Juan Liu, Bin Feng, Zhan Ma, and Wen yu Liu, "Adaptive motion vector prediction based on spatiotemporal correlation," in *Wireless Communications, Networking and Mobile Computing, 2006. WiCOM 2006.International Conference on*, sept. 2006, pp. 1 –4.

[6] Sung Deuk Kim and Jong Beom Ra, "An efficient motion vector coding scheme based on minimum bitrate prediction," *Image Processing, IEEE Transactions on*, vol. 8, no. 8, pp. 1117 –1120, aug 1999.

[7] Wen Yang, O.C. Au, Chao Pang, Jingjing Dai, Feng Zou, Xing Wen, and Yu Liu, "An efficient motion vector coding algorithm based on adaptive predictor selection," in *Circuits and Systems (ISCAS), Proceedings of 2010 IEEE International Symposium on*, 30 2010-june 2 2010, pp. 2175 –2178.

[8] Hui Yuan, Yilin Chang, Zhaoyang Lu, and Yanzhuo Ma, "Model based motion vector predictor for zoom motion," *Signal Processing Letters, IEEE*, vol. 17, no. 9, pp. 787 –790, sept. 2010.

[9] Ron Kohavi and George H. John, "Wrappers for Feature Subset Selection," *Artificial Intelligence*, vol. 97, no. 1-2, pp. 273–324, 1997.

[10] James Dougherty, Ron Kohavi, and Mehran Sahami, "Supervised and Unsupervised Discretization of Continuous Features," in *Proceedings of the Twelfth International Conference on Machine Learning*, Armand Prieditis and Stuart Russell, Eds., 1995, pp. 194–202.

[11] David E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Addison-Wesley Professional, 1 edition, Jan. 1989.

[12] Christopher M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*, Springer, 1st ed. 2006. corr. 2nd printing edition, Oct. 2007.

[13] Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG, "H.264/AVC JM reference software," Jan. 2011.

[14] Richard O. Duda, Peter E. Hart, and David G. Stork, *Pattern Classification (2nd Edition)*, Wiley-Interscience, 2 edition, Nov. 2001.

[15] S.K. Shevade, S.S. Keerthi, C. Bhattacharyya, and K.R.K. Murthy, "Improvements to the smo algorithm for svm regression," *Neural Networks, IEEE Transactions on*, vol. 11, no. 5, pp. 1188 –1193, sep 2000.

[16] Weka Machine Learning Project, "Weka," URL http://www.cs.waikato.ac.nz/˜ml/weka.


View publication stats