

# DRIFT REDUCTION FOR A H.264/AVC FINE GRAIN SCALABILITY WITH MOTION COMPENSATION ARCHITECTURE

*João Ascenso*

joao.ascenso@lx.it.pt  
Escola Superior de Tecnologia de Setúbal  
Instituto das Telecomunicações

*Fernando Pereira*

fp@lx.it.pt  
Instituto Superior Técnico  
Instituto das Telecomunicações

## ABSTRACT

The recent advances in non-scalable video encoding brought by the H.264/AVC standard offered significant improvements in terms of rate-distortion performance. This paper proposes a H.264/AVC based fine grain scalable video encoder which also exploits the motion compensation tools of the H.264/AVC standard to explore the temporal redundancy in the enhancement layer. The enhancement layer is predicted from a high quality reference obtained from past information of the enhancement and base layers. One of the drawbacks of this architecture is the drift effect, which occurs when part of the enhancement layer used for prediction is not received by the decoder. The drift reduction approaches here proposed simultaneously allow improvements in the coding efficiency and a reduction of the drift effect. The experimental results show improvements up to 2 dB in coding efficiency in comparison to Intra coding (like used by the MPEG-4 FGS standard) using the MPEG-4 testing conditions.

## 1. INTRODUCTION

The H.264/AVC standard [1] represents an important step in the evolution of video coding standards since it offers a significant improvement in terms of rate-distortion performance – providing more than a factor of two in bitrate savings against the popular MPEG-2 Video standard [2]. As several experiments have shown, the currently available MPEG-4 fine grain granularity (FGS) coding scheme suffers from a loss of efficiency when compared with the corresponding non-scalable single layer encoding, for a wide range of bitrates and qualities [3]. Several recent efforts tried to overcome this problem by proposing new solutions that try to maintain the MPEG-4 FGS capabilities intact, e.g. fine grain adaptation to dynamic changes in network conditions, low complexity decoding and memory requirements, and packet loss resilience, while improving its coding efficiency performance.

In this paper, a new fine grain scalable video coding scheme, called AVC-MCFGS<sup>1</sup>, is proposed based on an architecture with two distinct encoding layers. The base layer is H.264/AVC conformant and the enhancement layer reuses the necessary tools from the base layer to perform motion compensation and encode a residue. In order to obtain a fine grain scalable bitstream, a bitplane coding technique (like the one used in the MPEG-4 FGS standard) is adopted, which guarantees a fine adaptation to changes in the available bandwidth. The AVC-MCFGS video encoder extends the work presented in [4] where a H.264/AVC scalable encoder with a structure similar to MPEG-4 FGS is proposed. This encoder, called AVC-FGS, reuses in the enhancement layer some of the H.264/AVC base layer tools,

<sup>1</sup> The AVC-MCFGS acronym expresses that the coding tools are those from the AVC standard and the fine grain scalability solution is the same as used in MPEG-4 FGS although now including motion compensation (MC).

namely the Integer transform and the UVLC entropy coder, benefiting in this way from the good H.264/AVC encoding performance. However, in the AVC-FGS scheme, the base and enhancement layers are only predicted from references in the base layer (no temporal prediction is used in the enhancement layer). Since predictions are always based in the lower quality base layer reference, the AVC-FGS coding efficiency suffers a loss when compared to the non-scalable H.264/AVC scheme. A different solution is here adopted for the AVC-MCFGS architecture where a high quality reference is used (HQR - High Quality Reference): this reference is obtained from the base layer and the most significant biplanes of the enhancement layer. This solution [5-7] leaves the AVC-FGS base layer untouched, i.e. with no drift problems, which means that the base layer syntax is still H.264/AVC conformant.

The AVC-MCFGS encoder here proposed has two main goals: a) To achieve better coding efficiency by exploiting the temporal redundancy in the enhancement layer by using the motion compensation tools already present in the base layer. These tools are responsible for the major coding efficiency gain of the H.264/AVC base layer in relation to previous standards. b) To manage the drift effect and the impact of transmission errors in the enhancement layer with techniques that limit the error propagation. Two techniques that rely on the principle of leaky prediction [8] are proposed: the Cyclic Weighted BitPlane coding (CWBP) and the Adaptive Weighted BitPlane coding (AWBP). They both try to achieve a trade-off between the coding efficiency, the drift effect and the propagation of transmission errors in the enhancement layer. This paper is organized as follows. First, Section 2 presents in detail the proposed AVC-MCFGS codec architecture. In Section 3, the two drift reduction techniques, CWBP and AWBP, are presented. In Section 4, the AVC-MCFGS performance with the drift reduction tools is evaluated, in comparison with the AVC-FGS solution. Finally, the conclusions and some future work topics are presented in Section 5.

## 2. AVC-MCFGS ARCHITECTURE

The architecture proposed for the AVC-MCFGS encoder is presented in Figure 1. The base layer is compliant with the H.264/AVC standard and the enhancement layer uses the H.264/AVC transform and entropy coding tools with the adaptations already defined for AVC-FGS [4]. The syntax of the enhancement layer does not suffer any modification in comparison with AVC-FGS and is similar to MPEG-4 FGS: this means a bitplane coding solution is adopted. In order to comply with the AVC-MCFGS architecture, the encoder includes two motion compensation modules and two frame stores. In comparison with AVC-FGS (and MPEG-4 FGS), motion compensation, inverse transform and a frame store (dark color in Figure 1) have been added in the enhancement layer. Moreover, in the enhancement layer, the motion compensation supports the following base layer tools: a) adaptive block size motion

compensation; b)  $\frac{1}{4}$  pixel motion compensation; and c) multiple reference pictures. The two frame stores are used to store:

- The **Low Quality Reference (LQR)** which is obtained from past base layer decoded images. If only this reference was used, AVC-MCFGS would be equivalent to AVC-FGS and no drift would occur in the enhancement layer, i.e. only an Intra refreshment (regarding the base layer would occur).
- The **High Quality Reference (HQR)** which is predicted from the base layer decoded images and some higher bitplanes ( $N_{ref}$ ) of the enhancement layer. This reference has a superior quality regarding LQR and is responsible for the coding efficiency gain in comparison to MPEG-4 FGS and AVC-FGS.

In order to be able to apply the drift reduction weight bitplane coding techniques proposed in the next Section, i.e. the weighting of the HQR and LQR references, all prediction operations are performed in the spatial domain. The two frame stores are synchronized, i.e. the reference images in frame store 1 (enhancement layer) and in frame store 0 (base layer) correspond to the same temporal instant. The motion vectors used for the motion compensation in the enhancement layer are the same as for the base layer. This avoids the estimation and transmission of new motion vectors in the enhancement layer; this characteristic is desirable, since errors in this type of information have a disastrous effect (MC mismatch).

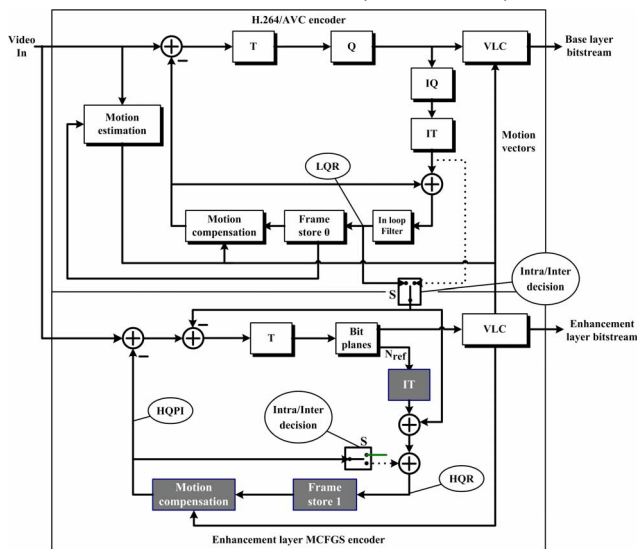


Figure 1 – AVC-MCFGS encoder architecture.

In the AVC-MCFGS enhancement layer, a frame can be encoded using one of two modes; this is controlled by the two switches in Figure 1. The two encoding modes are characterized as follows:

**a) Intra Mode:** In this mode, the enhancement layer is predicted and reconstructed only based on the LQR base layer reference. The AVC-MCFGS architecture is equivalent to MPEG-4 FGS and the encoding residue is obtained by calculating the difference between the original image and the decoded base layer image for the same time instant. In this mode, motion compensation is not performed in the enhancement layer. In order to use the Inter mode for the next frame, it is necessary to update the HQR reference by adding some part of the enhancement layer ( $N_{ref}$  bitplanes) to the decoded image of the base layer (solid line connected to the switches S in Figure 2).

**b) Inter Mode:** In this mode, the HQR reference is used by the motion compensation tool of the enhancement layer to obtain

the HQPI (*High Quality Predicted Image*) prediction. The final residue for transmission is determined by two predictions: one is the decoded base layer image and the other is the HQPI prediction which is based on previously enhancement layer decoded images (since multiple references are used). The images coded with this mode are referred as P frames and can also be used as references, i.e. as prediction for the next P frame. To obtain the updated HQR reference to be used for encoding the next P enhancement layer picture, it is necessary to add  $N_{ref}$  enhancement layer bitplanes to the base layer residual image and to the HQPI prediction (MC loop).

The AVC-MCFGS decoder is able to reconstruct the video from the base layer and the truncated enhancement layer bitstream. However, if only receives a part of the enhancement layer used for prediction at the encoder (less than  $N_{ref}$  bitplanes), the references at encoder and decoder will be different (MC mismatch) and the so called drift effect occurs. This only happens when fluctuations in the available bandwidth or transmission errors occur in the part of the enhancement layer used for prediction, i.e. the first  $N_{ref}$  bitplanes.

### 3. DRIFT REDUCTION BY WEIGHTED BITPLANE CODING

The motion compensation tools used in the AVC-MCFGS enhancement layer allow an improved coding efficiency in comparison with AVC-FGS (Intra mode) if all the bitplanes used as reference ( $N_{ref}$ ) are available at the decoder. Typically, this means high efficiency at high bitrates of the scalable coding range. However, if the reference is corrupted or truncated a loss of efficiency (and quality) occurs at the low bitrates of the scalable coding range, since the decoder does not have all the reference available. In this case, the HQR used by the decoder is different from the one used by the encoder and the propagation of reconstruction errors (drift effect) causes a loss of quality in the decoded image. The propagation of these errors can be stopped by using the Intra mode described in the previous Section, because it removes the prediction mismatch from the enhancement layer. However, if only the Intra mode is used by the encoder, the AVC-MCFGS scheme is conceptually similar to the FGS scheme (no MC in the enhancement layer) and thus the coding efficiency is limited in the same way.

In this context, it is necessary to develop tools which are able to reduce the drift effect present and simultaneously provide better coding efficiency than the MPEG-4 FGS standard and the AVC-FGS scheme. The drift reduction techniques here proposed are based on the Inter coding refreshment approach used in non-scalable coding to mitigate the propagation of transmission errors [8]. It was also used to improve the MPEG-4 FGS standard in [5], the starting point for the techniques here presented. This approach avoids error propagation but still makes use of past information in order to limit the reduction in terms of compression efficiency, typical of Intra refreshment schemes. In weighted bitplane coding, the enhancement layer reference used for prediction consists in a weighted combination of the base layer (LQR) and the enhancement layer (HQR) references. This type of techniques allows a reduction in the drift effect, assuming that the layer base is always available. In the AVC-MCFGS encoder, the principle of leaky prediction [8] is used to calculate the encoding residue for the enhancement layer. With the weighted prediction scheme, the prediction HQPI is obtained from the motion compensated version of the reference HQR (HQR[CM]) and the base layer reference LQR by:

$$HQPI = \alpha \times HQR[CM] + (\alpha-1) \times LQR \quad (1)$$

The selection of the parameter  $\alpha$ , where  $0 \leq \alpha \leq 1$ , is crucial for the global performance of the system since it mitigates the errors of past decoded frames in the current decoded frame with an exponential decay [8]. In order to choose an appropriate value for  $\alpha$ , the usual trade-off between coding efficiency and drift reduction has to be considered since as  $\alpha$  approaches 0, this scheme is closer to the MPEG-4 FGS solution; for  $\alpha = 0$ , all images are Intra coded. This allows a reduction in the drift effect but sacrifices the coding efficiency at the higher bitrates in the scalable coding range. On the contrary, for  $\alpha$  values closer to 1, the coding efficiency improves in comparison to the Intra only scheme (FGS), but introduces errors when part of the reference does not arrive to the decoder (drift effect). This is observed by a coding loss in PSNR values for the lower bitrates in the coding range. In order to limit the number of possible values for the parameter  $\alpha$  (and thus its decision and transmission costs), 3 additional coding modes have been defined: a) Mode Inter25:  $\alpha = 0.25$ ; b) Mode Inter50:  $\alpha = 0.5$  (average of the LQR and HQR references); c) Mode Inter75:  $\alpha = 0.75$ ; the Intra ( $\alpha = 0$ ) and Inter ( $\alpha = 1$ ) modes are still available. The selection of the AVC-MCFGS enhancement layer coding mode is essential to reach maximum coding efficiency for the relevant bitrate range. In the following, two drift reduction techniques are proposed which are able to stop the error propagation in the enhancement layer, by adjusting the impact of each layer prediction in the enhancement layer reference.

### 3.1 Cyclic Weight BitPlane Coding (CWBP)

CWBP selects the  $\alpha$  value through a simple cyclic function  $f(\alpha)$ , which allows to use both  $\alpha$  values close to 1 (more importance to HQR and thus potentially more drift) and  $\alpha$  values close to 0 (more importance to LQR and thus lower drift). The parameter  $\alpha$  changes for each frame in order to achieve a balance between the coding efficiency and the reduction of the drift effect. Figure 2 presents the function  $f(\alpha)$ ; this function shows two zones: one targeting a high coding efficiency (mode Inter is used,  $\alpha = 1$ ), and another zone where  $\alpha$  decreases and increases in order to reduce the drift effect without compromising too much the coding efficiency. The Intra mode is not used with this technique, to avoid too significant penalizations in terms of RD performance.

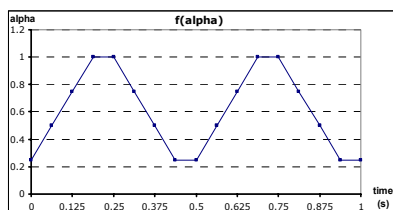


Figure 2 – Function  $f(\alpha)$  with  $P = 0.5s$  and frame rate of 15Hz.

The function  $f(\alpha)$  is mainly characterized by a parameter  $P$  which defines its period. This parameter has an important impact: it must allow a significant reduction of the drift effect after the time corresponding to one period, since some images were encoded with low  $\alpha$  values. The number of images to encode in each period of the function  $f(\alpha)$  depends on the parameter  $P$  and the video frame rate.

### 3.2 Adaptive Weight BitPlane Coding (AWBP)

AWBP selects the  $\alpha$  value and thus the contribution of each layer taking into account an estimation, performed at the encoder, of the drift effect. The choice of the  $\alpha$  parameter is performed in

an adaptive way depending on the amount of drift effect present at the decoder when some reference bitplanes are not received. The drift reduction technique here proposed has two major parts:

**a) Modeling of the drift effect at the encoder:** The modeling of the drift effect estimates the worst case drift error, i.e. the maximum drift introduced by the AVC-MCFGS scheme. This technique estimates that the worst case scenario corresponds approximately to the situation where the encoder uses as reference  $N_{ref}$  bitplanes (which establishes the quality of the reference HQR) and only half of them arrive to the decoder. Typically, the major loss of quality occurs around this situation; notice that, if all bitplanes are lost, only the drift free base layer image is decoded and thus no quality decay happens due to drift. With this approximate modeling of the drift, the encoder is able to estimate in advance the quality of the reference at the decoder and take actions accordingly to limit the negative effects, i.e. reduce the error propagation. To implement this modeling, it is necessary to introduce an additional decoding loop in the encoder, storing the so called Worst Quality Reference (WQR).

**b) Decision mechanism:** The decision mechanism selects the encoding mode based on the results of the modeling step. It chooses one of the five encoding modes, taking into account the drift effect present in the WQR reference without sacrificing the coding efficiency for the higher bitrates of the coding range. This is performed by a simple mechanism based on the average absolute error between the reference WQR and the original image for the five encoding modes available. In order to achieve a balance between coding efficiency and drift reduction, a parameter  $ld$  (where  $0 \leq ld \leq 1$ ) representing the maximum drift allowed with this technique is specified. The coding mode chosen corresponds to the largest prediction error (or drift error) without exceeding the  $ld$  value specified. For low  $ld$  values, the drift effect is completely removed at the cost of a worse performance for the higher bitrates of the scalable coding range; modes Intra, Inter25 are frequently selected. For high  $ld$  values, the RD performance improves, sacrificing the effective capability of drift reduction; modes Inter75 and Inter are frequently selected.

Both modules are related because the mode selection in the decision mechanism, for a given frame, influences the modeling of the drift effect for the next frame, i.e. the WQR reference depends on the decisions made in the past.

## 4. EXPERIMENTAL RESULTS

To evaluate the coding efficiency of the proposed AVC-MCFGS scheme, extensive experiments have been performed. In these experiments, two codecs were used: a) **Codec 1:** AVC-FGS (AVC-MCFS Intra mode always used) encoder proposed in [4], and b) **Codec 2:** AVC-MCFGS encoder proposed in Section 2 including the drift reduction techniques, CWBP and AWBP, described in Section 3. Both codecs use in their base layer the H.264/AVC Baseline profile as implemented by the H.264/AVC Joint Model 5.0 with rate control [9]; also the enhancement only used Baseline profile tools.

The test conditions used are those specified by MPEG [10], including three scenarios (S1 to S3) and a wide range of bitrates, and spatial and temporal resolutions. Each scenario corresponds to a different bitrate range  $[R_b, R_{max}]$ : scenario 1 for low bitrates - [32, 256] kbit/s; scenario 2 for intermediate bitrates - [128, 1024] kbit/s; and scenario 3 for high bitrates - [256, 2048] kbit/s. The test sequences chosen include the almost still “Boat” sequence, the fast “Stefan” sequence and the “Table” sequence with various scene cuts and different camera movements. In

order to evaluate the drift reduction capability, the encoding structure chosen in the enhancement and base layer is IPPP(P).

The two drift reduction techniques are evaluated using parameters which control the amount of drift introduced; these parameters were specified *a priori*, after conducting a series of experiments for a wide range of bitrates, sequences and spatial and temporal resolutions. For both techniques, the number of reference bitplanes ( $N_{ref}$ ) is always 3 and the encoding parameters of the drift reduction techniques are fixed and constant; in CWBP, the parameter P is 0.5 s; in AWBP, the parameter  $l_d$  is 0.75.

For each pair scenario/sequence, the corresponding scalable bitstream was truncated and decoded at several points ( $R_i$ ) in the specified bitrate range [ $R_b$ ,  $R_{max}$ ]; in these conditions, a PSNR versus bitrate curve was obtained. To evaluate the performance of each codec, the Bjontegaard measures [11] were used to express the average difference between two PSNR curves: the dPSNR value. In addition, other two measures were defined: a) **dMin**: minimum PSNR difference between AVC-MCFGS and AVC-FGS for any bitrate  $R_i$  and b) **dMax**: maximum PSNR difference between AVC-MCFGS and AVC-FGS for any bitrate  $R_i$ . While dMin evaluates the efficiency of each technique to reduce the drift effect that occurs in the lowest bitrates of each scenario, dMax measures the coding efficiency gain for the higher bitrates of each scenario. In Table 1, the results for each test configuration and sequence are presented when AVC-MCFGS is compared against AVC-FGS.

		Cyclic Weight Bitplane Coding			Adaptative Weight Bitplane Coding		
		dPSNR	dMin	dMax	dPSNR	dMin	dMax
Boat	S1	0.43	-0.12	1.12	0.3	-0.16	1.53
	S2	0.4	-0.03	1.14	0.57	0.02	1.28
	S3	0.29	-0.08	0.85	0.31	-0.02	1.03
Stefan	S1	0.56	0.01	1.32	0.53	-0.07	1.59
	S2	0.68	0.13	1.47	0.85	0.15	1.95
	S3	0.67	0.11	1.54	0.89	0.21	1.86
Table	S1	0.39	-0.05	0.91	0.46	-0.09	1.26
	S2	0.24	-0.12	0.84	0.4	-0.06	1.34
	S3	0.35	-0.07	0.87	0.53	-0.04	1.21

Table 1 – AVC-MCFGS versus AVC-FGS performance.

**1) AVC-MCFGS vs. AVC-FGS:** Table 1 shows that, for all sequences, the dPSNR (average dPSNR) was improved with the introduction of the motion compensation tools in the enhancement layer with both weighted bitplane coding techniques. Moreover, both techniques are able to reduce the drift effect to minimum values (dMin is less than 0.16 dB) while providing increased coding efficiency for the higher bitrates in each scenario (dMax goes up to 1.95 dB).

**2) Drift vs. Sequences:** On average, for both drift reduction techniques, the Stefan sequence has a larger coding gain compared to the other two sequences; this is due to the high correlation between frames in this sequence. The almost still Boat sequence comes second, mainly because the temporal redundancy is explored very efficiently in the base layer, i.e. the energy of the enhancement layer residue is very low.

**3) CWBP vs. AWBP:** For the AWBP technique, the dMax results are significantly improved in comparison with CWBP; moreover the drift introduced is still very low: dMin is less than 0.16 dB. The dPSNR AWBP values are also better than the CWBP values, with the exception of the Boat and Stefan sequences in scenario 1. The better coding efficiency of AWBP is due to the more complex decision mechanism for the prediction weighting factors in combination with the encoder modeling of the maximum drift. However, the CWBP results are rather good taking into account the very low complexity of the decision mechanism used.

## 5. FINAL REMARKS AND FUTURE WORK

In this paper, a new solution for a fine grain scalability scheme based on the H.264/AVC standard is proposed, called AVC-MCFGS. The AVC-MCFGS coding scheme uses motion compensation in the enhancement layer to improve the coding efficiency and includes drift reduction techniques to limit the error propagation problems. These techniques look for a balance between the coding efficiency and the amount of drift introduced when the encoder and decoder references are not the same, due to changes in the available bandwidth or transmission errors.

The experimental results show that AVC-MCFGS can simultaneously significantly improve the AVC-FGS coding efficiency (AVC-MCFGS Intra mode) and reduce the drift effect to insignificant values, with a low degradation at the lower bitrates. When the bitrate becomes moderate or high for the specified range, the coding efficiency improves comparing to the AVC-FGS and a coding gain up to 1.95 dB (AWBP) can be obtained. Regarding the drift reduction techniques proposed, the AWBP coding technique performs better due to the effective reduction in the drift effect while maintaining high coding gains. This way, it is possible to approximate the AVC-MCFGS performance to the non-scalable H.264/AVC performance, thus reducing the gap in quality between the two.

Further possibilities to continue the work are the selection of different prediction modes (weights) at the macroblock level (it was observed that the drift effect is more accentuated in some parts of the image) and the use of RD techniques to perform a more efficient control of the available prediction modes.

## 6. ACKNOWLEDGMENT

The work presented was developed within VISNET, a European Network of Excellence (<http://www.visnet-noe.org>).

## 7. REFERENCES

- [1] ISO/IEC International Standard 14496-10:2003, "Information Technology – Coding of Audio-visual Objects – Part 10: Advanced Video Coding".
- [2] T. Wiegand, H. Schwarz, A. Joch, F. Kossentini, G. J. Sullivan, "Rate-constrained Coder Control and Comparison of Video Coding Standards", IEEE Transactions on CSVT, Vol. 13, No. 7, July 2003.
- [3] H. M. Radha, M. van der Schaar, Y. Chen, "The MPEG-4 Fine-Grained Scalable Video Coding Method for Multimedia Over IP", IEEE Transactions on Multimedia, Vol. 3, No. 1, March 2001.
- [4] J. Ascenso, F. Pereira "H.264/AVC Fine Grain Scalability Using Bitplane Coding", 5th International Workshop on Image Analysis for Multimedia Interactive Services, Lisbon, April 2004.
- [5] H.-C. Huang, C.-N. Wang, T. Chiang, "A Robust Fine Granularity Using Trellis-Based Predictive Leak", IEEE Transactions on CSVT, Vol. 12, No. 6, June 2002.
- [6] A. Reibman, L. Bottou, "Managing Drift in DCT-based Scalable Video Coding", IEEE Data Compression Conference, April 2000.
- [7] F. Wu, S. Li, Y.-Q. Zhang, "A Framework for Efficient Progressive Fine Granularity Scalable Video Coding", IEEE Transactions on CSVT, Vol. 11, No. 3, March 2001.
- [8] P. Haskell, D. Messerschmitt, "Resynchronization of Motion Compensated Video Affected by ATM Cell Loss", IEEE ICASSP, San Francisco - EUA, March 1992.
- [9] K. Suehring, "H.264/AVC Reference Software", JM 5.0, <http://bs.hhi.de/~suehring/tml/>.
- [10] Video Group, "Common Conditions for Advanced FGS Experiments", ISO/IEC JTC1/SC29/WG11 N4748, Fairfax - EUA, May 2002.
- [11] G. Bjontegaard, "Calculation of Average PSNR Differences Between RD-curves", Doc. VCEG-M33, Austin - USA, April 2001.