2006 IEEE 24th Convention of Electrical and Electronics Engineers in Israel

Towards Model-based Transrating of H.264 coded video

Naama Hait and David Malah
Technion IIT, Haifa 32000, Israel
Department of Electrical Engineering
naamah@techunix.technion.ac.il, malah@ee.technion.ac.il

Abstract— **A common approach for video transrating (bit rate reduction) is to requantize the transform coefficients. Optimal requantization aims to find a set of new step-sizes that achieve the target bit rate while introducing minimal distortion. Since the state of the art H.264 standard coder constrains requantization by limiting the amount of change in the quantization step-size from one macroblock to the next, the common Lagrangian optimization approach cannot be applied. We propose a solution to this dependency problem by extending each Lagrangian iteration with a constrained dynamic programming stage. Further, in order to reduce the computational load of evaluating the rate and distortion at each macroblock for multiple step-sizes, we suggest analytic models that can be applied for this purpose. The developed models are suitable for requantization and are matched to the context-adaptive entropy coding used in H.264. The proposed algorithm performs the requantization in the compressed domain and currently supports inter coded frames only. It reduces the run-time by a factor of 4, as compared to the full exhaustive optimization, and achieves up to 1[dB] gain in PSNR, as compared to a simple one-pass algorithm.**

I. Introduction

In previous standards, like MPEG-2, the optimal requantization problem is defined as finding a set of optimal new step-sizes, where optimality is in the sense of minimizing the total distortion, subject to a given bit-rate constraint:

$$\min_{\{QP_i\}} \quad D, \quad subject \quad to \quad R \leq R_{target} \tag{1}$$

where $D = \sum_{i=1}^{N_B} d_i(QP_i)$ and $R = \sum_{i=1}^{N_B} r_i(QP_i)$,
N_B - number of macroblocks in the frame, QP_i - quantization parameter for the i-th macroblock, d_i - distortion caused to the i-th macroblock, r_i - number of bits produced by the i-th requantized macroblock.

A common approach [1] is to convert the constrained optimization problem to an unconstrained one:

$$\min_{\{QP_i\}} \quad J, \quad J = D + \lambda(R - R_{target}) \tag{2}$$

where λ is the Lagrangian parameter. The main advantage of solving the unconstrained problem is that the cost J can be broken into a sum of independent costs for each macroblock. Given a λ value, the set of quantization steps $\{QP_i^*\}_{i=1}^{N_B}$ that minimizes the set of independent costs is found and the corresponding average rate is calculated by $\sum_{i=1}^{N_B} r_i(QP_i^*)$. Then, the λ parameter is altered, using for instance, bisection iterations, until an average rate that is close enough to the target is obtained.

Unlike other standard coders, the H.264 standard limits the quantization step-size change from one macroblock to the next. Specifically, the quantization parameter at macroblock #i+1, QP_{i+1}, can only take the values
$QP_{i+1} \in \{QP_i{-}2, QP_i{-}1, QP_i, QP_i{+}1, QP_i{+}2\}$ (where an increase of 2 in QP corresponds to a step-size factor of about 1.26). We will denote this constraint as the ΔQP limitation. This constraint poses a problem for the common rate-distortion Lagrangian optimization algorithm, since the total cost cannot be broken into independent costs for each macroblock.

As a result, previous works on quantization (or requantization) in H.264 have chosen simple solutions, such as a uniform step-size for the whole frame [2], or a one-pass algorithm [3]. In section 2, we propose an algorithm for optimal requantization for inter coded frames that overcomes the dependency problem. At this point, intra coded frames weren't handled as the spatial prediction in H.264 introduces further macroblocks' dependencies. In section 3, we propose using new rate models at the macroblock level for H.264 to reduce the optimization computational load.

II. Optimal requantization

Due to the ΔQP limitation in H.264, our optimization problem has an extra constraint:

$$\min_{\{QP_i\}} D \quad subject \quad to \quad R \leq R_{target} \quad and \quad |\Delta QP| \leq 2 \tag{3}$$

Since the choices of quantization step sizes for different macroblocks are no longer independent, the whole set of quantization step-sizes $\{QP_i^*\}$ should be found at once, while keeping the ΔQP constraint. We propose to solve this problem by extending each Lagrangian iteration with a constrained dynamic programming stage. The external Lagrangian iterations change the Lagrangian parameter λ to improve the rate guess. At each examined value of λ, the constrained dynamic programming algorithm finds an optimal QP path by solving:

$$\min_{\{QP_i\}} J \quad subject \quad to \quad |\Delta QP| \leq 2 \tag{4}$$

where $J = D + \lambda(R - R_{target})$.

1-4244-0230-1/06/$20.00 ©2006 IEEE

The dynamic programming algorithm is defined over the set of states $\{(QP, i)\}$, where i is the macroblock index and QP is the quantization index, see Fig. 1. Each state (QP, i) has its cost-value $j_i(QP) = d_i(QP) + \lambda r_i(QP)$ and the total frame's cost along a path is $J = \sum_{i=1}^{N_B} j_i(QP)$.

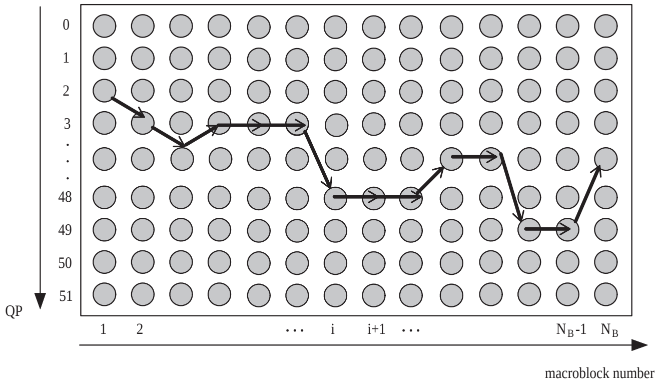

Fig. 1. Dynamic programming path illustration. Horizontal axis: macroblock number, vertical axis: the quantization parameter QP. Each circle denotes a state, and each column corresponds to a macroblock stage. The arrows show a path example, where the change in QP from one macroblock to the next is within ±2 units.

The optimal path up to the state (QP, i) is the path that has the minimal accumulated cost, $V_i(QP^*)$, over all possible paths that end at that state. There are at most 5 possible paths that end at the previous macroblock (#i-1) and that can be continued to the current state (QP, i), due to the ΔQP limitation. We choose among these by minimizing the value function of the current state:

$$V_i(QP) = V_{i-1}(QP_{Prev}) + j_i(QP) + cost(QP_{Prev}, QP) \tag{5}$$

$QP_{Prev} \in \{QP-2, QP-1, QP, QP+1, QP+2\}$.
It is the sum of the cost of the path until the previous macroblock $V_{i-1}(QP_{Prev})$, plus the cost of the current state $j_i(QP)$, plus the cost of moving from state $(QP_{Prev}, i-1)$ to (QP, i), where in our case the later is

$$cost(QP_{Prev}, QP) = \begin{cases} 0 & |QP - QP_{Prev}| \leq 2 \\ \infty & else \end{cases}$$

Or, in other words, the best path up to state (QP, i) is continued from state $(QP^*_{Prev}, i-1)$, where

$$QP^*_{Prev} = \arg\min_{QP_{Prev}} \{V_{i-1}(QP_{Prev}) + cost(QP_{Prev}, QP)\} \tag{6}$$

The corresponding value function update is then:
$V_i(QP) = V_{i-1}(QP^*_{Prev}) + j_i(QP)$.
At each stage i of the dynamic programming algorithm (from the first to the last macroblock), the best paths for all (QP, i) states are found and kept as lists of pointers, along with their values. When the algorithm reaches the last stage ($i = N_B$), the optimal path is the optimal path over the entire frame:

$$BestPathEnd = \underset{QP}{\operatorname{argmin}} V_{N_B}(QP) \tag{7}$$

The algorithm then traces back the optimal frame path using the chosen list of pointers, to obtain the optimal path: $\{QP^*_i\}_{i=1}^{N_B}$.

III. ρ DOMAIN RATE-DISTORTION MODELING

The optimization algorithm described above requires the evaluation of the rate and distortion obtained by requantizing each macroblock at multiple step-sizes. If no prior knowledge is used, such rate assessment involves the simulation of the actual requantization followed by entropy coding. As this procedure must be repeated multiple times, the optimization becomes computationally expensive. The computational complexity can be greatly reduced by using an analytic model for the relation between rate and quantization step-size, for each macroblock. In this section, we will elaborate on the model-based evaluation of the rate and the distortion.

A. Previous work

Different models in the literature suggest different relations for rate vs. quantization step size. In [4] [5], the ρ-domain source model is suggested, where ρ is the percentage of zero coefficients among the quantized transformed coefficients in a frame. The model states that there is a strong linear relation between ρ and the actual frame's bit rate: coarser quantization step-sizes generate more zero coefficients (and hence increase ρ) while decreasing the rate. Therefore, the suggested $rate - \rho$ relation is $r(\rho) = \theta \cdot (1 - \rho)$, where θ is the graph's slope. According to this equation, for $\rho = 1$ all the quantized coefficients are zeroed and thus the coding rate should approach zero. It is also argued that the $rate - \rho$ model is more robust than a rate− quantization-step model: the observed rate-ρ curves for both I and P frames share a very similar pattern, whereas the rate− quantization step-size curves change between different frame types.

The distortion too is more conveniently described in the ρ domain than in the quantization step-size domain as it's defined within the finite range of $0 \leq \rho \leq 1$ and follow a more robust and regular behavior. In [6], an exponential-linear model for the MSE distortion in the ρ domain was suggested as $d(\rho) = \sigma^2 \cdot e^{-\alpha \cdot (1-\rho)}$, where σ^2 is the variance and $\alpha > 0$ is a model parameter. Again, as $\rho \to 1$ and all the quantized coefficients are zeroed, the distortion approaches the σ^2 bound.

These models were derived for describing the rate and the distortion at the frame level, and were found quite accurate in [4], [5], [6], when tested for standards such as MPEG-2 and H.263 and were also used in [2], [3] for H.264. Since we aim to use the rate-ρ models for macroblock-level optimization as described in section 2, we suggest modified models for H.264 at the macroblock level to improve the accuracy.

B. H.264 context adaptive entropy coding

The H.264 context adaptive entropy coding with VLC tables (CAVLC), is designed to take advantage of the sparse (compact energy) characteristics of the quantized transform coefficients [7]. To this end, it uses a set of syntax elements, that includes both the customary run-level representation and additional overhead counts that mainly describe the zero valued coefficients distribution. On top of that, it switches between several VLC tables for each syntax element, in a context adaptive manner.

Though the run and level are encoded separately, their encoding is efficient due to the context based VLC tables switching. The additional overhead counts consist of two symbols. One describes the combination of the number of non-zero coefficients and the high-frequency trailing-ones (±1 at the end of the block). It's referred to as (TotalCoefficients, TrailingOnes). The other symbol, called TotalZeros, denotes the number of zeroed coefficients from the DC coefficient to the highest frequency non-zero coefficient. Both of which use multiple VLC tables. Fig. 2 shows an example for a 4x4 zig-zag scanned block, with 6 non-zero coefficients, 2 trailing-ones, and 2 TotalZeros (that are marked in gray).

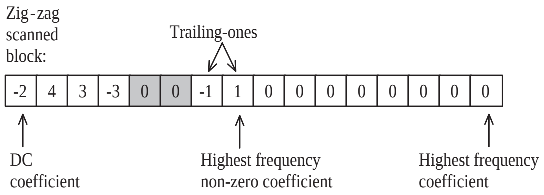

Fig. 2. An example of the additional overhead syntax elements in H.264.

C. Models for H.264 requantization

Examination of the $rate - \rho$ relation at the macroblock level has shown that a linear relation isn't a good descriptor of the empirical data. Therefore, and in light of the new entropy coding features, we suggest a different $rate - \rho$ model at the macroblock level. We decompose the rate into "*data*" and "*overhead*" components, where the "*data*" stands for the bits spent on coding the run-level, and the "*overhead*" designates the bits spent on coding the new syntax elements. For the model parameters estimation we use prior information, such as the original input quantized transform coefficients and their encoded rate.

"Data" Component

For the "data" component $rate - \rho$ relation, we suggest a closed-form model:

$$r^{data}(\rho) = \theta \cdot ln(1 + (1 - \rho)^{\eta}) \tag{8}$$

where $\theta \geq 0, \eta \geq 1$. The θ parameter controls the scale of the graph, whereas the η parameter changes its shape. Now, given this component's original input encoded rate of a macroblock, $r_{in}^{data}(\rho_{in})$, we can fit one of the parameters. Since this model requires fitting two parameters, we fit its shape parameter η using the input ensemble $\{r_{in}^{data}(\rho_{in})\}$ of all the frame macroblocks, while the scale parameter θ is matched to each macroblock separately. An example of normalized $r^{data}(\rho)$ relations of one frame's macroblocks is depicted in Fig. 3

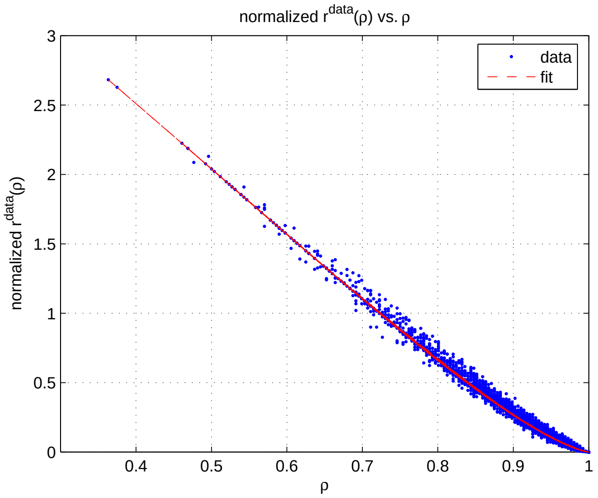

Fig. 3. Normalized $r^{data}(\rho)$ relation of one frame's macroblocks (blue dots) and its normalized fit with the common shape parameter η (red line).

"Overhead" Component

The "overhead" component $rate - \rho$ relation is very noisy due to two reasons. One is that the overhead syntax elements values (e.g. (TotalCoefficients, TrailingOnes)=(6,2) and TotalZeros=2 in the example of Fig. 2) aren't uniquely defined by the local block's ρ. The other is the use of multiple VLC tables for each syntax element, which means that the number of bits spent on coding the same syntax element value changes with the context. As a result, fitting a closed-form model for it becomes practically impossible. However, due to the partial dependency in the local ρ, we chose to use a statistical model to characterize the average code length at the 4x4 block level, and then average over the 16 blocks in the macroblock.

Each 4x4 block has a local percentage of zeroed coefficients, ρ_b, which is related to the local total non-zero coefficients count TC_b, by $\rho_b = 1 - \frac{TC_b}{16}$. The macroblock's level ρ is simply the average of these local ρ_b's: $\rho = \frac{1}{16}\sum_{b=1}^{16} \rho_b$. Using the statistical model that follows, we calculate once the average code lengths $\bar{c}_{(TC,Tr)}(\rho_b|context - prior)$ and $\bar{c}_{TZ}(\rho_b|input{-}prior)$ of the (TotalCoefficients, TrailingOnes) and TotalZeros syntax elements, respectively. These average lengths are kept in look-up tables and the rate "overhead" component is obtained by averaging over all the blocks in the macroblock:

$$r^{overhead}(\rho) = \frac{1}{16}\sum_{b=1}^{16} \overline{c}_{(TC,Tr)}(\rho_b|context - prior) + \frac{1}{16}\sum_{b=1}^{16} \overline{c}_{TZ}(\rho_b|input - prior) \tag{9}$$

We assume that the quantized transform coefficients are not correlated and follow a Laplacian distribution. Another assumption is that all ±1 quantized coefficients appearances occur at the highest nonzero frequencies, and are thus considered as high-frequency trailing-ones. Using the Laplacian distribution, the probability that the magnitude of a quantized transform coefficient will take the value k is:

$$Pr.(|l| = k) = \begin{cases} \rho & k = 0 \\ \frac{(1-\rho)^{2k}\rho(2-\rho)}{1-\rho} & k > 0 \end{cases} \tag{10}$$

and therefore the probability of a trailing-one coefficient, given that it's non-zero is:
$Pr.(TR) = Pr.(|l| = 1||l| > 0) = \rho(2 - \rho)$.

We define a binomial random variable that denotes the number of trailing-ones appearances given ρ_b and sum over the joint (TotalCoefficients, TrailingOnes) code length tables (there are 4 different tables) to obtain the average VLC tables $\overline{c}_{(TC,Tr)}(\rho_b|context - prior)$. We switch between these four average VLC tables by predicting the number of non-zero coefficients from the neighboring blocks, in accordance with the standard's context-based encoding.

Since the quantized blocks are typically sparse and most of the energy is concentrated at low frequencies, there is usually a tail of zeros at the end of the scanned block (see example in Fig. 4). So, instead of counting the TotalZeros syntax element, TZ, as the number of zeroed coefficients from the DC coefficient to the highest frequency non-zero coefficient, we can count its complement, the tail, since $TC + TZ + Ztail = 16$. As we increase the requantization step, the number of non-zero coefficients, TC, decreases, and the tail length monotonically increases. Therefore, $TC + TZ$ monotonically decreases.

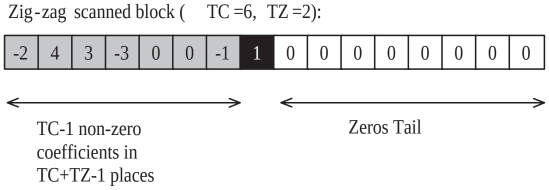
Zig-zag scanned block (TC =6, TZ =2):
-2
4
3
-3
0
0
-1
1
0
0
0
0
0
0
0
0
TC-1 non-zero coefficients in TC+TZ-1 places
Zeros Tail

Fig. 4. The example of Fig. 2 with TC, TZ and the zeros tail. There are TC=6 non-zero coefficients and TZ=2 zeros counted from the DC coefficient to the highest frequency non-zero coefficient (which is denoted in black).

Given the input prior information (TC_{in}, TZ_{in}), we find the probability of having TZ TotalZeros given ρ_b. The average code length for each of the 15 (TC_{in}, TZ_{in}) input priors is evaluated by summing over the joint (TotalCoefficients,TotalZeros) code length tables.
Finally, the total $rate - \rho$ relation is evaluated by:

$$r(\rho) = r^{data}(\rho) + r^{overhead}(\rho) \tag{11}$$

where $r^{data}(\rho)$ and $r^{overhead}(\rho)$ are evaluated from (8) and (9), respectively.

Distortion*$-\rho$ *model
According to the $distortion - \rho$ model suggested in [6], $ln(\overline{d}(\rho))$ should be linearly proportional to $1 - \rho$, where $\overline{d}(\rho) = \frac{d(\rho)}{\sigma^2}$ is the normalized distortion. Examining this relation at the macroblock level, we found that a linear model doesn't describe it with sufficient accuracy. We therefore suggest to extend the model to an exponential-quadratic relation:

$$d(\rho) = \sigma^2 \cdot e^{\alpha_1 \cdot (1-\rho)^2 + \alpha_2 \cdot (1-\rho)} \tag{12}$$

that better matches the empirical data. The model's accuracy in terms of its relative error distribution is depicted in Fig. 5.

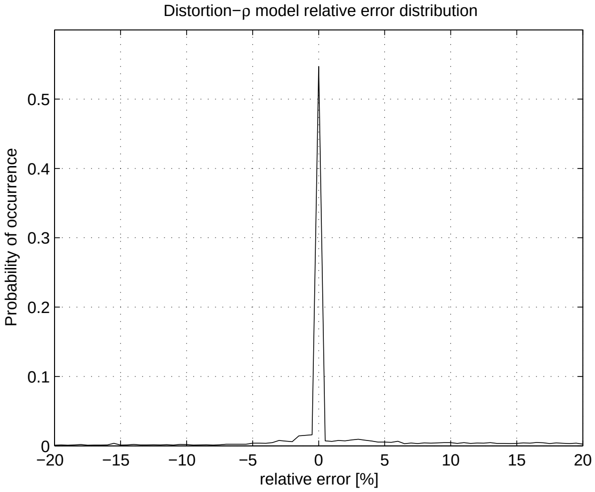

Fig. 5. Distortion$-\rho$ model relative error distribution.

IV. Results

Our suggested algorithm currently supports inter coded frames only, and its flow chart is depicted in Fig. 6. In order to compare its performance with other transrating schemes (including reencoding), all intra coded frames were fully reencoded in our simulations. A SIF 'football' sequence was initially encoded at 2[Mbps] and its bit rate was reduced using three different schemes for the inter coded frames: our suggested algorithm, a one-pass algorithm and full reencoding.

Fig. 7 depicts the quality vs. the transrated bit rate using two different quality measures. The upper graph shows the PSNR measure, whereas the lower graph shows the subjective VQM measure [8], which denotes the probability that a human observer will notice artifacts. For both measures, our suggested algorithm performance is consistently better than the one pass algorithm, with a PSNR gain of up to 1[dB]. The full reencoding scheme performs better than our algorithm, since it's free to make new coding decisions (such as motion vectors), more suitable for the lower rate so that it could allocate more bits for the transform coefficients coding.

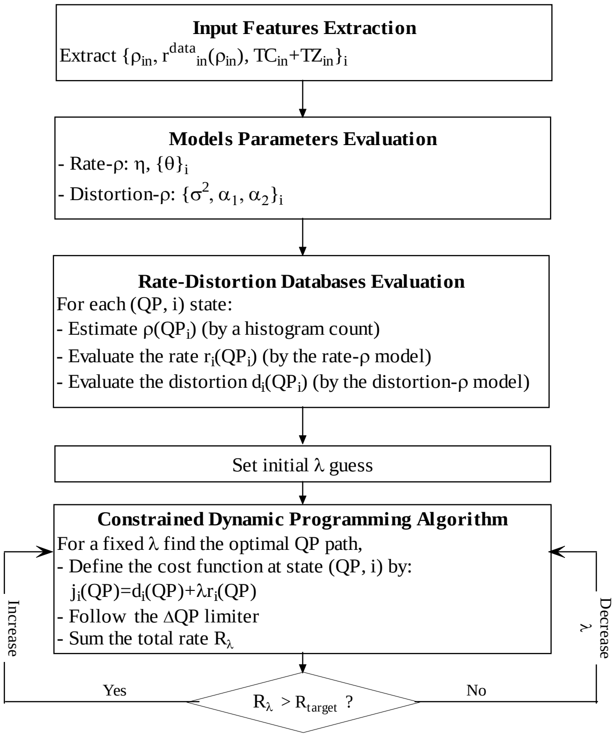
Input Features Extraction
Extract $\{\rho_{in}, r^{data}_{in}(\rho_{in}), TC_{in}+TZ_{in}\}_i$
Models Parameters Evaluation
- Rate-ρ: η, $\{\theta\}_i$
- Distortion-ρ: $\{\sigma^2, \alpha_1, \alpha_2\}_i$
Rate-Distortion Databases Evaluation
For each (QP, i) state:
- Estimate $\rho(QP_i)$ (by a histogram count)
- Evaluate the rate $r_i(QP_i)$ (by the rate-ρ model)
- Evaluate the distortion $d_i(QP_i)$ (by the distortion-ρ model)
Set initial λ guess
Constrained Dynamic Programming Algorithm
For a fixed λ find the optimal QP path,
- Define the cost function at state (QP, i) by:
$j_i(QP)=d_i(QP)+\lambda r_i(QP)$
- Follow the ΔQP limiter
- Sum the total rate R_λ
Increase λ
Decrease λ
Yes
No
$R_\lambda > R_{target}$?

Fig. 6. Model-based optimal requantization flow chart. The details of the first three building blocks ('Input Features Extraction', 'Models Parameters Evaluation', 'Rate-Distortion Databases Evaluation') are described in section III.C. The details of the 'Constrained Dynamic Programming Algorithm' building block are described in section II.

Another aspect is the computational complexity savings obtained using our suggested models. We evaluated this saving by comparing the run time of the optimization algorithm, once based on the full rate and distortion evaluation and once using our models. In our simulations, the incorporation of the new models reduces the run time by a factor of 4, as compared to a full exhaustive optimization.

V. Conclusion

A model-based optimal requantization algorithm for H.264 inter coded frames is proposed. It extends the common Lagrangian iterations with a constrained dynamic programming algorithm to account for the H.264 step-size change limitation. It achieves better performance compared to a simple one-pass requantization algorithm, both objectively and subjectively. Rate models suitable for H.264 requantization were developed, by which we reduced the computational complexity, as compared to a full exhaustive optimization, by a factor of 4. Future work will examine the selective change of coding decisions and extend the algorithm to intra coded frames.

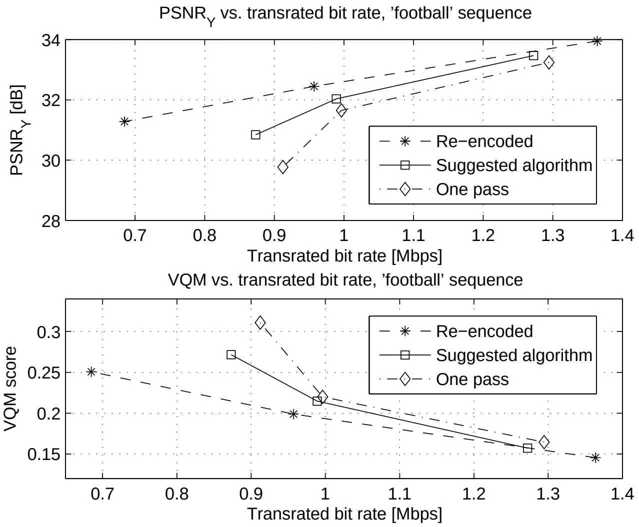

Fig. 7. Top: $PSNR_Y$ vs. transrated bit rate, bottom: VQM vs. transrated bit rate. Star: reencoded, square: suggested algorithm, diamond: one-pass algorithm.

VI. Acknowledgement

The authors acknowledge with thanks the support obtained from the STRIMM consortium under the MAGNET program of the Ministry of Trade and Industry via the Samuel Neaman Institute.

References

[1] P. A. A. Assuncao and M. Ghanbari, "A frequency-domain video transcoder for dynamic bit-rate reduction of MPEG-2 bit streams," *IEEE transactions on Circuits and Systems for Video Technology*, vol. 8, no. 8, pp. 953–967, Dec. 1998.

[2] I. Shin, Y. Lee, and H. Park, "Rate control using linear rate-ρ model for H.264," *Signal Processing: Image Communications*, vol. 19, no. 4, pp. 341–352, Apr. 2004.

[3] L. C. S. Milani and G. A. Mian, "A rate control algorithm for the H.264 encoder," in *Sixth Baiona workshop on Signal Processing in Communications*, 2003.

[4] Z. He and S. Mitra, "A linear source model and a unified rate control algorithm for DCT video coding," *IEEE transactions on Circuits and Systems for Video Technology*, vol. 12, no. 11, pp. 970–982, Nov. 2002.

[5] Y. K. Z. He and S. Mitra, "Low-delay rate control for DCT video coding via ρ-domain source modeling," *IEEE transactions on Circuits and Systems for Video Technology*, vol. 11, no. 8, pp. 928–940, Aug. 2001.

[6] Z. He and S. Mitra, "Optimum bit allocation and accurate rate control for video coding via ρ-domain source modeling," *IEEE transactions on Circuits and Systems for Video Technology*, vol. 12, no. 10, pp. 840–894, Oct. 2002.

[7] I. E. G. Richardson, "H.264/MPEG-4 part 10," in *H.264 and MPEG-4 Video Compression*. John Wiley, 2003, pp. 201–207.

[8] M. H. Pinson and S. Wolf, "A new standardized method for objectively measuring video quality," *IEEE Transactions on broadcasting*, vol. 50, no. 3, pp. 312–322, Sept. 2004.