Rate control strategies in H264

Simone Milani (simone.milani@dei.unipd.it)
Summary

- General scheme of the H.264 encoder
- Rate control problem over limited bandwidth channels
- The rate control algorithm JVT-D030
- The rate control algorithm JVT-D030
- Modelling of produced bit rate through the percentage of null quantized transform coefficients ("zeros")
- A rate control algorithm based on the percentage of "zeros"
- Parametric models for the coefficients statistics
- Future work
- Conclusions
The H.264 standard

H.264 standardizes a hybrid transform video coder with spatial and temporal prediction.

Input signal: YUV 4:2:0

8 bit/sample

Frame format | Size
---|---
QCIF | 176 X 144
CIF | 352 X 176

Macroblock: 16 x 16

4 blocks: 8 x 8

16 smaller blocks: 4 x 4
Spatial and temporal prediction

Intra

The current 4 x 4 block is predicted from the neighboring pixels according to different directions.

Inter (BMME)

Motion vector resolution: \( \frac{1}{4} \text{ pixel} \)

Image coding through motion vectors.
The prediction error, i.e. the difference between the original signal and the predicted one, is coded.

- The transform is applied on 4 x 4 pixel blocks.
- The transform is an accurate approximation of the Discrete Cosine Transform, implemented with sums and shifts, operating with integer values.
- The transform coefficients are quantized.

### Transform

\[
x: \text{prediction error signal} \\
\begin{array}{cccc}
1 & 1 & 1 & 1 \\
2 & 1 & -1 & -2 \\
1 & -1 & -1 & 1 \\
1 & -2 & 2 & -1 \\
\end{array}
\]

\[
y = [x_{11}, x_{12}, x_{13}, x_{14}] = [1, 2, 1, 1] \\
\begin{array}{cccc}
1 & 2 & 1 & 2 \\
1 & 1 & 2 & 1 \\
1 & -1 & -1 & 2 \\
1 & -2 & 1 & -1 \\
\end{array}
\]

An Hadamard Transform 4 x 4 can be optionally performed over the DC coefficients.

### Quantization

\[
\Delta = K \cdot 2^{QP/6} \quad QP \in [0, 51]
\]

- The coefficients are uniformly quantized
- 51 step sizes are defined (QP)
- There is an exponential relation between QP and \(\Delta\)
Bit stream coding in the H.264 standard

- The quantized transform coefficients are converted into a sequence of couples (run,length).
- The encoder syntax elements are converted into a binary stream with two possible coding engines
  - CAVLC
  - CABAC

CAVLC
Context-based Adaptive Variable Length Coding

CABAC
Context-based Adaptive Binary Arithmetic Coding

C: current block
A,B: previously coded blocks

Each syntax element is coded according to the information coded in the previous blocks.
A comparison between H.264 and MPEG4

A performing prediction scheme and an adaptive entropy coder allow the H.264 standard to provide better video quality than MPEG4.

<table>
<thead>
<tr>
<th>Standard</th>
<th>PSNR(dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MPEG-4</td>
<td>30.18</td>
</tr>
<tr>
<td>H.264</td>
<td>32.66</td>
</tr>
</tbody>
</table>

64 kbit/s @ 30 frames/s GOP IPPP 60 frames
MPEG4–TM5/H264–"zeros"

<table>
<thead>
<tr>
<th>Standard</th>
<th>PSNR(dB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MPEG-4</td>
<td>31.07</td>
</tr>
<tr>
<td>H.264</td>
<td>32.73</td>
</tr>
</tbody>
</table>
Rate Control

The transmission of a coded video sequence over a bandlimited channel with minimum distortion.

Target

Control parameters
QP: quantization parameter

Quality parameters
PSNR: reconstructed video quality
ε: relative bit rate error
B: buffer level

\[
PSNR = 10 \cdot \log_{10} \left( \frac{255 \cdot 255}{\frac{1}{N} \sum |P_{o,i} - P_{r,i}|^2} \right)
\]

The rate control algorithms are designed to keep the coded video bit stream among the bandwidth constraints modifying the quantization parameter QP.

The distortion is determined by the quantization parameter, which may vary according to the coded frame or macroblock.
Different kinds of rate control

- **Constant Bit Rate algorithms (CBR)**
  The produced bit rate is kept nearly constant according to the available bandwidth.

- **Variable Bit Rate algorithms (VBR)**
  Strong variations may affect the number of bits in the coded stream in order to keep a constant quality.

- **Average Bit Rate algorithms (ABR)**
  The bit rate fits the given constraints at constant time intervals

- **Maximum Bit Rate algorithms (MBR)**
  The produced bit stream is upper bounded.
Some rate control problems

The choice of a rate control algorithm is affected by different requirements

<table>
<thead>
<tr>
<th>Application</th>
<th>Storage</th>
<th>Videocall</th>
<th>Videoconferencing</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rate control precision</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>Video quality of reconstructed images</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>Variations in reconstructed video quality during the time</td>
<td>✔️</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>Coding latency due to pre-analysis</td>
<td>✗</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>Computational complexity</td>
<td>✗</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>Efficiency at low bit rates</td>
<td>✗</td>
<td>✔️</td>
<td>✔️</td>
</tr>
</tbody>
</table>

In video communication applications over mobile devices, rate control techniques that require low complexity and coding latency are required.
Rate control algorithms classification

Feed Forward algorithms
the input signal is pre-analyzed (pre-coded) in order to find the video coding parameter values that allow to keep the produced bit stream under the channel constraints.

Feed Backward algorithms
the input signal is coded without pre-analysis, using the knowledge of the past. E.g. the rate control algorithm based on “zeros”

Hybrid algorithms
they are based on a blending between the previous two techniques E.g. the algorithms JVT-D030, JVT-E069.
The JVT-D030 algorithm

- It is quite similar to the algorithm TMS of MPEG-2
- According to the available number of bits, a target bit size is assigned to the frame
- After each frame is coded, the transmission buffer and the parameters that characterize the frame statistics are updated.

\[ T_I = \max \left\{ \frac{R}{1 + \frac{N_R}{K_R X_I} + \frac{N_B}{K_B X_I}}, \frac{R_B}{8 \cdot F_R} \right\} \]

\[ T_P = \max \left\{ \frac{R}{N_P + \frac{N_R K_P X_I}{K_B X_P}}, \frac{R_B}{8 \cdot F_R} \right\} \]

\[ T_P = \max \left\{ \frac{R}{N_B + \frac{N_R K_B X_I}{K_R X_P}}, \frac{R_B}{8 \cdot F_R} \right\} \]
MB rate control for the JVT–D030 algorithm

At macroblock level, the quantization parameter is computed according to the buffer level and the image statistics.

- Each MB is precoded and an optimal QP is computed from the produced bit rate and the macroblock activity.
- The current macroblock is recoded in case the estimated optimal QP does not correspond to the one used in precoding.

\[
N_{act}(n) = \frac{2 \cdot act(n) + \overline{act}_{\{I,P\}}}{2 \cdot act_{\{I,P\}} + act(n)}
\]

\[
d_{\{I,P\}}(n) = d_{\{I,P\}}(0) + B \cdot \frac{T_{\{I,P\}}}{N_{MB} - n}
\]

\[
QP_n = \left( \frac{31 \cdot d_{\{I,P\}}(n)}{r} \right) \cdot N_{act}(n)
\]
Some comments about the JVT–D030 algorithm

**Advantages**
- Accurate rate control
- The macroblock precoding requires a computational effort lower than performing the precoding at frame level.

**Disadvantages**
- High computational load due to the preanalysis (in the worst case, each frame is coded twice)
- Great variation of QP (and, as a consequence, video quality) across the same image
- Quality losses and low performance with scene changes.
- Variable coding latency (not deterministic).

**Improvement**

The JVT–E069 algorithm
The JVT-E069 algorithm

- The algorithm structure is the same of the JVT-D030
- The quantization parameter computation and some parameter initialization are different

\[
QP_n = \left( \frac{31 \cdot d_{(l,P)}(n)}{r} \right) + dq
\]

\[
dq = \begin{cases} 
\frac{act(n)}{act} - 1 & \text{se } 0 < \frac{act(n)}{act} \leq \frac{1}{2} \\
0 & \text{se } \frac{1}{2} < \frac{act(n)}{act} \leq 2 \\
\frac{act(n)}{act} - 1 & \text{se } 2 < \frac{act(n)}{act} 
\end{cases}
\]

<table>
<thead>
<tr>
<th>Parameter \ Algorithm</th>
<th>JVT-D030</th>
<th>JVT-E069</th>
</tr>
</thead>
<tbody>
<tr>
<td>( X_I )</td>
<td>150 \cdot R_b/115</td>
<td>15 \cdot R_b/115</td>
</tr>
<tr>
<td>( X_P )</td>
<td>25 \cdot R_b/115</td>
<td>5 \cdot R_b/115</td>
</tr>
<tr>
<td>( X_B )</td>
<td>18 \cdot R_b/115</td>
<td>155 \cdot R_b/115</td>
</tr>
<tr>
<td>( d_0^l )</td>
<td>10 \cdot r_{D030}/31</td>
<td>20 \cdot r_{E069}/31</td>
</tr>
</tbody>
</table>
Comments about the JVT–E069 algorithm

**Advantages**

- Accurate rate control
- Smoothness of video quality among different area in the same frame
- Reduced QP variation at macroblock level

**Disadvantages**

- The complexity is not reduced
- Slower adaptation to channel bandwidth variations.
- Low quality at scene changes

<table>
<thead>
<tr>
<th>MB no.</th>
<th>$d_r(n)/r$</th>
<th>$a_{ct,m}(n)$</th>
<th>$a_{ct}$</th>
<th>$N_{act,m}(n)$</th>
<th>QP (JVT–D030)</th>
<th>QP (JVT–E069)</th>
</tr>
</thead>
<tbody>
<tr>
<td>23</td>
<td>0.513</td>
<td>1076</td>
<td>1000.0</td>
<td>1.025</td>
<td>16.30</td>
<td>15.90</td>
</tr>
<tr>
<td>49</td>
<td>0.551</td>
<td>2209</td>
<td>1000.0</td>
<td>1.287</td>
<td>21.98</td>
<td>18.08</td>
</tr>
<tr>
<td>96</td>
<td>0.489</td>
<td>429</td>
<td>1000.0</td>
<td>0.765</td>
<td>11.60</td>
<td>16.16</td>
</tr>
<tr>
<td>2</td>
<td>0.514</td>
<td>217</td>
<td>1044.6</td>
<td>0.641</td>
<td>10.21</td>
<td>18.93</td>
</tr>
<tr>
<td>36</td>
<td>0.617</td>
<td>1122</td>
<td>1044.6</td>
<td>1.024</td>
<td>19.59</td>
<td>19.13</td>
</tr>
<tr>
<td>63</td>
<td>0.631</td>
<td>2240</td>
<td>1044.6</td>
<td>1.276</td>
<td>24.96</td>
<td>20.56</td>
</tr>
<tr>
<td>96</td>
<td>0.559</td>
<td>323</td>
<td>1044.6</td>
<td>0.701</td>
<td>12.15</td>
<td>19.33</td>
</tr>
</tbody>
</table>

12/04/2007
Bit rate modelling through the percentage of null quantized transform coefficients ("zeros")

Many rate control techniques approximate the Rate-Distortion curves through a hyperbolic model.

It is possible to characterize the bit rate produced by the H.264 encoder through the percentage of null quantized transform coefficients ($\rho$), which are called “zeros”: the produced bit rate is a linear function of $\rho$.

\[ R(\Delta) = \gamma_0 + \frac{\gamma_1}{D(\Delta)} \]

where

$\Delta$: quantization step for transform coefficients

$P_x(a)$: coefficients distribution for the actual frame

\[ \rho = \sum_{|a|<\Delta} p_x(a) \]

the number of bits required to code the whole image can be related to $\rho$ through the equation

\[ R(\rho) = \lambda \cdot (1 - \rho) \]
Experimental results from the H.264 encoder

then it is possible to characterize $R(\rho)$ in a simple way:

$$R(\rho) = m \cdot \rho + q$$

[Plots from carphone, GOP IBBP 60 frames QP=cost. Frame 0(I), 18(P), 40(B)]
A rate control algorithm based on the percentage of “zeros”

The “zeros” model can be used to control the bit rate produced by the H.264 encoder.

The whole algorithm has different control levels:
- GOP level
- frame level
- macroblock level

As for the GOP and frame levels, a target number of bits is assigned to each frame according to the image statistics, the available bandwidth and the previous coding errors.

\[
T_I = K_{I,p}K_{P,B} \frac{G}{K_{I,p}K_{P,B} + K_{P,B}N_p + N_B}
\]

\[
T_P = K_{P,B} \frac{G}{K_{P,B}N_p + N_B}
\]

\[
T_B = \frac{G}{K_pN_p + N_B}
\]
A rate control algorithm based on the percentage of “zeros”

- After computing the target bit rate for the current frame, the algorithm has to find the percentage of zeros ($\rho$) that allows to fit the target number of bits.
- This value can be related to an average quantization parameter through the coefficients distribution.

From the equation

$$R(\rho) = m \cdot \rho + q$$

It is possible to relate the desired bit rate to a target percentage of “zeros” that can be related to a quantization step $\Delta$ or quantization parameter $QP$ through the equation

$$\rho = \sum_{|a|<\Delta} p_x(a)$$

The parameters $m$, $q$ and the coefficients distribution are found from the previously coded frames.

At macroblock level, the average $QP$ value is corrected according to the actual bit rate and percentage of “zeros”

$$k = \frac{\rho}{\rho^*} \quad \Rightarrow \quad QP = QP_{avg} - n \quad \text{if} \quad k < 1 - n \cdot \delta k$$
Comments about the “zeros” algorithm

**Advantages**
- Sufficiently accurate rate control.
- Quality smoothness between different areas of the same frame.
- Limited PSNR variations between different frames.
- Reduced and deterministic coding delay.
- Good performance at low bit rate.
- Robustness at scene changes.
- Fast adaptation to channel variations.
- Great reuse of H.264 encoder syntax parameters.

**Disadvantages**
- Less accurate than JVT algorithms.
- Requires a great memory area and a great number of memory accesses.
- At high bit rate, the performance are either equal or slightly worse.
A comparison between different algorithms

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>GOP</td>
<td>IPPP</td>
</tr>
<tr>
<td>Bit rate</td>
<td>64 kbit/s</td>
</tr>
<tr>
<td>Frame rate</td>
<td>30 frame/s</td>
</tr>
<tr>
<td>No. frames GOP</td>
<td>60</td>
</tr>
<tr>
<td>Coding engine</td>
<td>CAVLC</td>
</tr>
</tbody>
</table>

“Zeros”

JVT-E069

Scene change

Frame no.

PSNR (dB)
Introduction of parametric models in the rate control algorithm based on “zeros”

It is possible to reduce the hardware requirements of the “zeros” algorithm through the use of parametric models that approximate the coefficient distribution.

In our investigation we considered two approximations:

- a generalized gaussian p.d.f.
- a “laplacian+impulsive” p.d.f.

**Generalized gaussian p.d.f.**

\[
f_x(a) = \gamma \cdot e^{-(\beta |a|)^\alpha}
\]

with

\[
\beta = \sigma^{-1} \cdot \begin{bmatrix} \Gamma \left( \frac{3}{\alpha} \right) \\ \Gamma \left( \frac{1}{\alpha} \right) \end{bmatrix} \\
\gamma = \frac{\alpha \cdot \beta}{2 \cdot \Gamma(\alpha)} \\
\Gamma(p) = \int_0^\infty t^{\alpha-1} \cdot e^{-t} dt
\]

**“Laplacian+impulsive” p.d.f.**

\[
f_x(a) = b \cdot e^{-c|a|} + d \cdot \delta(a)
\]

where \(\delta(a)\) is the Dirac impulse.

The “l+i” model turns out to be the simplest one and provides the lower computational load.
A comparison between different algorithms

Sequence foreman R_b=64000 bit/s F_r=30 frame/s GOP IBBP 60 frame CAVLC
A comparison between different algorithms (2)

JVT-E069

Zeros

Sequenza foreman R_b=64000 bit/s F_r=30 frame/s GOP IBBP 60 frame CAVLC
A comparison between different algorithms (3)

Sequence foreman F_r=30 frame/s GOP IBBP 60 frame CAVLC
A comparison between different algorithms (4)

Sequenza container $R_b=64000$ bit/s $F_r=30$ frame/s GOP IBBP 60 frame CAVLC
A comparison between different algorithms (5)

Sequenza container $R_b=64000$ bit/s $F_r=30$ frame/s
GOP IBBP 60 frame CAVLC
A comparison between different algorithms (6)

Sequence foreman R_b=64000 bit/s F_r=30 frame/s GOP IBBP 60 frame CAVLC
A comparison between different algorithms (7)

Sequence table $R_x = 64000 \text{ bit/s}$ $F_r = 30 \text{ frame/s}$ GOP IBBP 60 frame CAVLC

JVT-E069

Zeros
A comparison between different algorithms (8)

Sequence table \( F_r = 30 \) frame/s GOP IBBP 60 frame CAVLC
A comparison between different algorithms (10)

JVT-E069

Zeros

Sequence table CIF $R_s=192000$ bit/s $F_r=30$ frame/s GOP IBBP 60 frame CAVLC
A comparison between different algorithms (9)
Conclusions

- The different algorithms presented have similar rate control accuracy.
- The “zeros” algorithms performances are better at low bit rates (higher PSNR and reduced quality variation) and comparable at high bit rates.
- The JVT-D030 and JVT-E069 algorithms requires a greater computational load.
- The “zeros” algorithms are more robust against scene changes than JVTs.
- The “zeros” algorithm is faster and provides an efficient technique for the implementation over time-varying channels.
- The coding latency of the “zeros” algorithms is constant.
- The “zeros” algorithms reuse many parameters belonging to the H.264 syntax.
Future work

- A VBR rate control algorithm.
- The adaptation of the rate control algorithm to the transmission over a packet switched network, taking into account the existing session layer protocols for audio/video transmission (e.g. RTP, ...).
- Increased robustness to errors and packet losses through a joint source–channel coding.
- A layered rate control implementation.
- Integration of the rate control into a multiple description scheme.