ROI based Computational Complexity Reduction Scheme for H.264/AVC Encoder

Tianruo ZHANG†, Xin JIN††, Chen LIU†, Minghui WANG†, Satoshi GOTO† (Member)
† Graduate School of Information, Production and Systems, Waseda University
†† Information Technology Research Organization, Waseda University

Summary This paper proposes a computational complexity reduction scheme for region-of-interest (ROI) based H.264/AVC encoding for videophone, video conferencing and surveillance systems. A proposed fast ROI detection algorithm which can detect face-like regions as ROI is applied to obtain accurate and small ROI so as to reduce the necessary coding effort of the encoder. The complexity reduction algorithm contains 3 methods: (1) the inter prediction mode selection based on quality difference, (2) unequal performance degradation based on unequal bits allocation, and (3) the ROI boundary enhancement to reduce the coding complexity of ROI. Experimental results show that proposed novel ROI detection and complexity reduction scheme can reduce 77.18% simulation time when QP difference between ROI and non-ROI is 20, with 1.02% bit-rate increase and 0.04 dB PSNR decrease. 18.17% of encoding time is further reduced compared with previous work with similar performance degradation.

Keywords: video compression, H.264/AVC, region-of-interest (ROI), computational complexity reduction

1. Introduction

The latest video coding standard H.264/AVC significantly increases compression efficiency while introducing heavy computation load. Reducing computational complexity is very critical for mobile devices. Lots of works have been done to reduce the computational complexity of H.264/AVC encoder. However, these algorithms give equal importance to the whole video sequences and equally introduce quality loss and bit-rate increase.

Region-of-interest coding strategy is proposed to give higher importance to some of the regions to enhance the coding performance and reduce computation load. These works apply simple ROI detection algorithms to avoid introducing heavy computation overhead. However, the detected ROIs are not accurate enough so that the encoders cannot achieve high performance based on them. Besides, ROI based computation power allocation only aims to allocate less computation power to the unimportant region to reduce the overall computational complexity. However, it simply used dynamic encoding parameter and mode clustering, regardless of the special motion estimation features introduced by unequal bits allocation. ROI based bits allocation only aims to allocate more bits to important region to increase its quality so as to enhance the subjective quality. It does not investigate the variation of the rate-distortion optimization (RDO) based mode decision while applying unequal bit allocation.

This paper proposes a low complexity H.264 encoder by improving both the ROI detection and the complexity reduction algorithm for from video phone over mobile network (CIF) to video conference over fixed high-speed network (720p). Faces can always be considered as ROIs in these applications. We propose a fast ROI detection algorithm which can detect...
accurate and small ROI, so that the H.264 encoder is able to achieve lower computational complexity. Proposed complexity reduction algorithm contains 3 parts. Firstly, the relation between quantization parameter (QP), ROI and the block size of motion estimation (ME) is investigated and a mode selection method is applied according to the ROI based quality assignment difference. Secondly, an unequal coding efficiency degradation method is introduced to significantly reduce the computational complexity of the non-ROI encoding. Although the coding efficiency of non-ROI encoding is also largely decreased, the coding efficiency of the whole encoder only slightly decreases due to the ROI based unequal bits assignment. Finally, the computational complexity of encoding the central of ROI is also reduced to achieve lower ROI encoding complexity.

The rest of this paper is organized as follows. Section 2 introduces the proposed ROI detection algorithm with the analysis on its accuracy and its influence on encoding complexity reduction. Section 3 describes the ROI based complexity reduction algorithm. Section 4 shows the experimental results for the improvements both in ROI detection and complexity reduction algorithm to guarantee the lower computational complexity. Conclusions are given in Section 5.

2. ROI Detection Algorithm

2.1 ROI Detection Algorithm for Video Encoding

A fast ROI detection algorithm is proposed. It has novel macroblock (MB) based data utilization and processing flow. The H.264 encoder can directly encode one MB after we detect whether this MB is one part of an ROI, without waiting for the detection results of the whole frame. It can detect face-like regions, which is always important in videophone, video-conferencing, and surveillance systems.

Proposed algorithm has a novel estimation-and-verification process to allow the detector firstly obtains rough detection result of each MB to catch up the MB-level pre-processing flow, and then accurately verifies the previous result by the following information. The flowchart of proposed algorithm is shown in Fig. 1.

In the estimation step, there are 3 main functions called continuity check, skin color detection, and contour decision. Based on the assumption that ROIs are always temporally continuous, continuity check searches the detection results of MBs in neighboring positions in previous frame, \( F(n-1) \), to estimate some of the MBs to be non-ROI MBs or ROI-like MBs. Skin color detection detects color distribution of each MB in pixel level. If most of the pixels in the MB are in skin color, the MB is estimated as an ROI-like MB. Otherwise, contour decision is performed to detect whether the MB contains ROI contour. Contour decision is based on the assumption that the shape of ROI as face is always an ellipse. An MB which is considered as a contour MB in certain position of an ellipse should satisfies 2 rules, the distribution of ROI status of neighboring MBs, and the distribution of skin-colored pixels in the MB. The rules of 8 different positions’ contour MBs are shown in Fig. 2. In order to fit the encoding process of H.264, only the up-left, upper, up-right, and left MBs are available as reference MBs. For example, if an MB contains most of the skin-colored pixels in the lower half of the MB, and its up-left, upper, and up-right neighboring MBs are non-ROI MBs while its left neighboring MB is an ROI-like MB, the MB will be estimated as an ROI-like MB containing the upper ROI contour.
The verification step is performed after estimation step for each MB to correct the false estimation. As an improvement to a previous work\(^9\), proposed detection algorithm can be performed in either single detection mode or multiple detection mode to detect one or multiple ROIs in each frame. All of the detected ROI MBs are categorized in ROI candidates. The 3 steps of combination check, generate a new candidate, and merge existing candidates are to assemble the ROI MBs of different ROIs to different candidates. Shape control is to delete the candidates which have unexpected height over width. When faces are defined as ROI, the ROI candidates should have the value of height over width between 1.0 and 2.0. Connection elimination is to delete the non-ROI skin-colored region as ties or hands which are connected to a detected ROI. Shape refinement is to set the MBs containing eyes or mouths which are not in skin color as ROI MBs by checking whether it is surrounded by ROI MBs of the same candidate. Size control is to delete the small candidates which are not possible to be further enlarged.

In the ROI detection and H.264 encoding process, the detection result of each MB is sent to the encoder after estimation and verification steps of the MB. It may not be the final detection result since it might be updated by the verification step of a following MB. This may causes the encoder to encode this MB by the wrong ROI status. Most of this situation happens in the 1st frame, in which the continuity check does not work. Because the 1st frame is I-frame in H.264 encoder and complexity reduction algorithm is not applied in I-frame, this wrong ROI status does not intensely influence the coding performance.

Proposed fast detection algorithm utilizes the skin color information and approximate ellipse shape of object to detect face-like ROIs. The estimation-and-verification process makes it suitable to be applied as an MB-level pre-processing function in an H.264 encoder.

### 2.2 ROI Detection Results

Detection results of multiple detection mode of CIF sequences are shown in Fig. 3. Fig. 3 (a) is the detection result sent to the encoder. This ROI status is adopted by the H.264 encoder. Fig. 3 (b) is the final detection result. If proposed detector is applied as a frame-level pre-processor, this will be the result. In MB-level pre-processing scheme, although this result is not sent to the encoder, it is saved in the ROI detector and increases the accuracy of the continuity check method when detecting the next frame. Fig. 3 (c) is the results of a fast ROI detection algorithm in 8). The algorithm in 8) is based on skin color, normalization and mean filter. Its weighted ROI are transformed to binary form by an optimized threshold value. Fig. 4 compares the detection results of single detection mode, multiple detection mode, and detection algorithm in 8). Fig. 5 shows the detection result of HD 720p sequences.

MB level hit rate and false alarm rate are proposed to evaluate the detection accuracy, as shown
Fig. 4  Comparison results of (a) single detection mode, (b) multiple detection mode, and (c) detection algorithm in 8)

\[
\text{False Alarmed MB Rate} = \frac{\text{MB Falsely Detected}}{\text{MB ROI}} \times 100\% \tag{1}
\]

\[
\text{Hit MB Rate} = \frac{\text{MB Detected} - \text{MB Falsely Detected}}{\text{MB ROI}} \times 100\% \tag{2}
\]

The ROI detection accuracy is compared in Table 1. It is tested by the first 200 frames of 4 CIF sequences foreman, carphone, headwithglasses, and silent. It can be found out that our proposed ROI achieves higher accuracy compared with another fast ROI detection algorithm in 8). Compared with conventional face detection algorithm based on haar-like feature\(^1\), proposed algorithm shows much higher hit rate which represents detecting much more face MBs that may at different poses and angles. Although the false alarmed MB rate of proposed algorithm is a little bit higher than that of 10), the computational complexity of proposed algorithm is much lower so that it is more suitable for a low power encoder.

2.3 Analysis of ROI and Encoding Complexity

The basic idea of ROI based complexity reduction approach is to apply encoding algorithms with different computational complexity and performance loss to ROI and non-ROI respectively, so as to achieve better trade-off between different algorithms. We assume that there are 2 different encoding algorithms, algo 1 and algo 2, which are 2 different trade-offs between complexity reduction rate and coding performance decrease. Compared with the full search algorithm with the highest computational complexity and coding efficiency, algo 1 reduces computation time by \(\Delta T^1\) and introduces bit-rate overhead by \(\Delta B^1\), while algo 2 reduces computation time by \(\Delta T^2\) and introduces bit-rate overhead by \(\Delta B^2\). When \(\Delta T^1 > \Delta T^2\), which means algo 1 is faster than algo 2, \(\Delta B^1 > \Delta B^2\) is always satisfied. In ROI based approach, algo 2 is applied in ROI while algo 1 is applied in non-ROI.

It is clear that the time reduction rate of this ROI based encoder is \(\Delta T \in (\Delta T^2, \Delta T^1)\), and the bit-rate overhead is \(\Delta B \in (\Delta B^2, \Delta B^1)\). Therefore, the target of designing an ROI based encoder with computational complexity reduction approximate to \(\Delta T^1\) and performance decrease approximate to \(\Delta B^2\).

The percentage of ROI in the video contents is denoted by \(p_{ROI}\). The video contents are encoded by full search style with encoding time \(T\) and the bit-rate \(B\). When algo 2 is applied in ROI and algo 1 is applied in non-ROI, the encoding time of ROI, time reduction of ROI, encoding time of non-ROI, time reduction of non-ROI, denoted by \(T_{ROI}, \Delta T_{ROI}, \)


where \( T_{ROI} \) and \( T_{NON} \) are calculated by Eq. 3.

\[
T_{ROI} = \text{proj}_1 T \\
\Delta T_{ROI} = \text{proj}_1 \Delta T^2 \\
T_{NON} = (1 - \text{proj}_1) T \\
\Delta T_{NON} = (1 - \text{proj}_1) \Delta T^1
\]  

By Eq. 3, the time reduction of the ROI based encoder is calculated by Eq. 4:

\[
\Delta T = \Delta T_{ROI} + \Delta T_{NON} \\
= \Delta T^1 - \text{proj}_1 (\Delta T^1 - \Delta T^2)
\]  

It can be seen from Eq. 4 that:

\[
\lim_{\text{proj}_1 \rightarrow 0} \Delta T = \Delta T^1
\]  

Larger \( \Delta T \) can be achieved by reducing \( \text{proj}_1 \). When the size of ROI is smaller, more video contents are encoded by the lower complexity algo 1.

The bit-rate of ROI, bit-rate overhead of ROI, bit-rate of non-ROI, bit-rate overhead of non-ROI, denoted by \( B_{ROI} \), \( \Delta B_{ROI} \), \( B_{NON} \), \( \Delta B_{NON} \), are calculated by Eq. 6:

\[
B_{ROI} = \text{proj}_1 b_{ROI} B \\
\Delta B_{ROI} = \text{proj}_1 b_{ROI} \Delta B^2 \\
B_{NON} = (1 - \text{proj}_1) b_{NON} B \\
\Delta B_{NON} = (1 - \text{proj}_1) b_{NON} \Delta B^1
\]  

in which, \( b \), \( b_{ROI} \), \( b_{NON} \) are the average bits per pixel (bpp) of the original encoded bit stream without ROI segmentation, ROI, and non-ROI. To compare the performance of video sequence under the same bit-rate, we assume that:

\[
b = \text{proj}_1 b_{ROI} + (1 - \text{proj}_1) b_{NON}
\]  

By Eq. 6, the bit-rate overhead of the ROI based encoder is calculated by Eq. 8:

\[
\Delta B = \Delta B_{ROI} + \Delta B_{NON} \\
= \text{proj}_1 b_{ROI} \Delta B^2 + (1 - \text{proj}_1) b_{NON} \Delta B^1
\]  

It can be seen from Eq. 8 that:

\[
\lim_{\text{proj}_1 \rightarrow 0} \Delta B = \Delta B^1 \\
\lim_{\text{proj}_1 \rightarrow 1} \Delta B = \Delta B^2
\]  

It means that when the size of ROI is smaller, the bit-rate overhead is larger. It means that if we simply reduce the size of ROI to reduce computation time, the coding performance will decrease. However, the bit-rate overhead is also an important index and has to be kept small.

It can also be seen from Eq. 7 and Eq. 8 that:

\[
\lim_{\text{proj}_1 \rightarrow 0} \Delta B = \Delta B^2
\]  

When the bpp of non-ROI is much smaller than bpp of ROI, the bit-rate overhead of the whole encoder is approximate to its minimum value. It is quite suitable for ROI based encoding, which always requires higher quality in ROI to enhance the subjective quality while requires lower quality in non-ROI to reduce the overall bit-rate.

In conclusion, to design an ROI based encoder with low computation time and low bit-rate overhead, which corresponds to performance loss, the size of ROI should be small and the bpp of non-ROI should be much lower than ROI.

As shown in Table 1, proposed detection algorithm can reduce 15.5% of detected ROI size by significantly reducing the false alarmed MB rate compared with detection algorithm in 8). It means that more computation time can be saved based on proposed ROI.

3. Complexity Reduction Algorithm

3.1 QP and ROI based Mode Selection

In the RDO based mode decision in variable block size motion estimation (VBSME) of H.264, smaller block type is more possible to be chosen under smaller quantization step when the residual bits are fewer. This work is for from video phone over mobile network (CIF) to video conference over fixed high-speed network (720p). Fig. 7 shows the relation between partition size of mode decision and QP in CIF and 720p sequences. Each row is the percentage of MBs using different modes. It can be found out that when QP is higher, larger modes are more possible to use.

In head-and-shoulder sequences which are widely used in videophone and video conferencing systems, the face ROI always has more global and local motion while the non-ROI background always has less motion. Therefore, smaller partitions are more likely to be used in ROI. Fig. 8 compares the difference of partition size of mode decision between ROI and non-ROI. Fig. 7 and 8 are simulated by the first 100 frames of 4 CIF and 2 720p sequences. RDO is on.
Motion estimation is fast full search (FFS). CABAC is used.

We can conclude that either lower QP or moving ROI leads to smaller mode partitions while either high QP or static non-ROI leads to larger mode partitions. In low QP ROI and high QP non-ROI scheme, the partition size preference becomes quite obvious. Therefore, in proposed algorithm, the modes which are no larger than P8\(\times\)8 can be skipped in non-ROI motion estimation.

3.2 Unequal Coding Efficiency Degradation Method

Traditional fast algorithms for ME\(^2\)\(^4\) use local information or spatial/temporal correlation to decide which mode is the most possible to select by RDO in an early stage. However, they do not consider the different importance to human attention in each part, so that they have to be equally performed on each MB and equally introduce performance loss while reducing computation time.

In ROI encoding scheme, the encoder enhances the quality of ROI while decreasing the quality of non-ROI to obtain high subjective visual quality under low bits budget. Therefore, the bit-rate of non-ROI influences the overall bit-rate less than the ROI. More computational complexity reduction should be assigned to non-ROI part to only reduce the coding efficiency of non-ROI, so as to achieve better trade-off between computation time and performance loss for the whole encoder.

Fig. 9 illustrates the unequal coding efficiency degradation method. A pair of typical rate-distortion (RD) curves are shown. Upper curve is the performance of a high complexity algorithm, e.g. full search, while the lower curve is encoded by a low complexity fast algorithm. Point A and C are the coding performance of high complexity algorithm under two QPs. Traditional fast algorithms for ME as 3) achieve the performance of point B and D respectively. Because these algorithms do not know which part of the video is more important, they have to keep the whole sequence in similar quality. Bit-rate increase is introduced as \(\Delta B_{\text{High}}\) and \(\Delta B_{\text{Low}}\) for high and low quality coding respectively. Because the slope of typical RD-curves for H.264 encoding always decreases when bit-rate increases, \(\Delta B_{\text{Low}}\) is always smaller than \(\Delta B_{\text{High}}\).

In proposed method, the ROI coding keeps high complexity so that it can achieve the performance of point A. The non-ROI coding has low complexity so
that it achieves the performance of point D. The bit-rate overhead is $\Delta B_{\text{Low}}$ while the overall bit-rate is larger than D. Therefore, the percentage of bit-rate increase is smaller than traditional fast algorithm and the quality of ROI keeps high.

In proposed algorithm, the coding efficiency is dynamically controlled by search range, the number of reference frames, and Intra prediction. In low quality non-ROI, smaller search range, less reference frames and Intra prediction skipping are applied to reduce the computational complexity. In high quality ROI, large search range, more reference frames and full Intra prediction are applied to achieve high coding efficiency. Search range is reduced to half and only 1 reference frames is used in non-ROI. Intra prediction in P slices is skipped in non-ROI.

### 3.3 ROI Boundary Enhancement

One problem of ROI based complexity reduction algorithms is that the complexity reduction rate is limited by the ratio of ROI in video contents as discussed in Section 2.3 because the computational complexity of ROI cannot be decreased\(^{11}\). Previous work, such as 8), uses weighted ROI to reduce the computational complexity of low weight ROI. Higher weights are assigned to the central MBs of ROI and lower weights are assigned to the boundary MBs so as to assign less computation load to the boundary MBs of ROI. However, assigning less computation load to the boundary of ROI is not suitable when an ROI is an object. The boundary MBs, which always contain parts of interested object and parts of background, always have more complicated motion features. Therefore, the boundary MBs are harder to encode by ME than the central MBs. In proposed algorithm, the computational complexity of ROI is reduced by reducing the encoding complexity of central MBs of ROI.

One of the main advantage of proposed face-oriented fast ROI detection algorithm is that it can detect clear and accurate boundary of ROI compared with other face-oriented fast ROI detection algorithms\(^{8,12}\). As shown in Fig. 3, 4 and 5, most of the MBs contain face contour are detected as the boundary MBs of ROI.

Therefore, in proposed algorithm, only the boundary MBs of ROI are encoded in highest complexity. The encoding complexity of the ROI central MBs are reduced by reducing search range and reference frames. The search range is reduced to 75% of original search range and 2 reference frames are used in the central MBs of ROI.

### 3.4 Working Flow of Computational Complexity Reduction Algorithm

A computational complexity reduction algorithm is proposed by combining method 1: QP and ROI based mode selection, method 2: unequal coding efficiency degradation, and method 3: ROI boundary enhancement. The flowchart of proposed complexity reduction scheme is shown in Fig. 10. Small partitions are ignored in non-ROI according to method 1. The non-ROI MBs are encoded with the smallest search range and the least reference frames while intra prediction is not performed in non-ROI MBs in P slices according to method 2. Only the boundary MBs in ROI are encoded with the largest search range and the most reference frames according to method 3.

### 4. Experimental Results

The number of MBs categorized as ROI central MBs, ROI boundary MBs, and non-ROI MBs are shown in Fig. 11. In 4 CIF and 2 720p test sequences, 11.32% MBs are in ROI, in which 6.50% and 4.82% are ROI central and boundary MBs respectively. ROI
Table 2  Comparison of performance degradation and encoding time reduction

<table>
<thead>
<tr>
<th>ΔQP</th>
<th>Sequence</th>
<th>ROI detection in 8) + complexity reduction algorithm in 8)</th>
<th>Proposed ROI detection + proposed complexity reduction algorithm</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>BDPSNR(dB)</td>
<td>BDBR (%)</td>
</tr>
<tr>
<td>4</td>
<td>Carphone (CIF)</td>
<td>-0.04</td>
<td>+0.93</td>
</tr>
<tr>
<td></td>
<td>Foreman (CIF)</td>
<td>-0.03</td>
<td>+0.68</td>
</tr>
<tr>
<td></td>
<td>Headwithglasses (CIF)</td>
<td>-0.18</td>
<td>+3.63</td>
</tr>
<tr>
<td></td>
<td>Silent (CIF)</td>
<td>-0.13</td>
<td>+3.57</td>
</tr>
<tr>
<td></td>
<td>Vidyo1 (720p)</td>
<td>-0.05</td>
<td>+2.12</td>
</tr>
<tr>
<td></td>
<td>Vidyo4 (720p)</td>
<td>-0.07</td>
<td>+1.90</td>
</tr>
<tr>
<td></td>
<td><strong>Average</strong></td>
<td><strong>-0.08</strong></td>
<td>+2.14</td>
</tr>
<tr>
<td>12</td>
<td>Carphone (CIF)</td>
<td>-0.03</td>
<td>+0.53</td>
</tr>
<tr>
<td></td>
<td>Foreman (CIF)</td>
<td>-0.02</td>
<td>+0.49</td>
</tr>
<tr>
<td></td>
<td>Headwithglasses (CIF)</td>
<td>-0.11</td>
<td>+2.76</td>
</tr>
<tr>
<td></td>
<td>Silent (CIF)</td>
<td>-0.11</td>
<td>+3.17</td>
</tr>
<tr>
<td></td>
<td>Vidyo1 (720p)</td>
<td>-0.04</td>
<td>+1.74</td>
</tr>
<tr>
<td></td>
<td>Vidyo4 (720p)</td>
<td>-0.07</td>
<td>+1.52</td>
</tr>
<tr>
<td></td>
<td><strong>Average</strong></td>
<td><strong>-0.06</strong></td>
<td>1.70</td>
</tr>
<tr>
<td>20</td>
<td>Carphone (CIF)</td>
<td>-0.02</td>
<td>+0.44</td>
</tr>
<tr>
<td></td>
<td>Foreman (CIF)</td>
<td>-0.01</td>
<td>+0.36</td>
</tr>
<tr>
<td></td>
<td>Headwithglasses (CIF)</td>
<td>-0.08</td>
<td>+1.26</td>
</tr>
<tr>
<td></td>
<td>Silent (CIF)</td>
<td>-0.07</td>
<td>+1.73</td>
</tr>
<tr>
<td></td>
<td>Vidyo1 (720p)</td>
<td>-0.03</td>
<td>+0.91</td>
</tr>
<tr>
<td></td>
<td>Vidyo4 (720p)</td>
<td>-0.05</td>
<td>+0.82</td>
</tr>
<tr>
<td></td>
<td><strong>Average</strong></td>
<td><strong>-0.04</strong></td>
<td>+0.92</td>
</tr>
</tbody>
</table>

boundary enhancement method can reduce the computation cost of ROI central MBs, which is the main part of ROI.

Performance of proposed ROI detection and complexity reduction algorithm is compared with the ROI detection and complexity reduction algorithm of 8) in Table 2, Fig. 12, 13, 14, and 15.

Table 2 compares the performance and encoding time reduction of H.264 encoder. Simulation is performed by JM12.4. Experimental conditions are listed in Table 3 and the ΔQP between ROI and non-ROI is set to 4, 12, and 20. BDPSNR and BDBR are calculated by the method in 14). The complexity reduction algorithm in 8) is based on weighted ROI and it is adjusted and normalized in our comparison experiments. When ΔQP is small, e.g. 4, the performance degradation is large because the QP based mode selection is not quite accurate. Algorithm in 8) adopts ROI based mode selection which shows the similar effect. When ΔQP is 20, 77.18% computation time is saved with 0.04 dB PSNR loss and 1.02% bit-rate increase. The performance degradation of proposed ROI based encoder is quite similar to 8), while 18.17% computation time is further reduced compared with 8).

Fig. 12 compares the rate-distortion (RD) curves of CIF sequence “Headwithglasses” of proposed en-

![Fig. 11 Percentage of ROI central MBs and ROI boundary MBs of CIF and 720p sequences](image-url)
ROI based Computational Complexity Reduction Scheme for H.264/AVC Encoder

Fig. 12 RD curve of H.264 encoder based on different ROI

Fig. 13 Performance comparison of different combination of ROI detection and complexity reduction algorithms

Fig. 14 SSIM Quality comparison of different ROI coder and the encoder in 8) including ROI detection and complexity reduction algorithm. It can be seen that performance is slightly decreased by either proposed fast algorithm or the fast algorithm in 8). It should also be noticed that the H.264 encoder with proposed ROI achieves higher RD-curves than the encoder with ROI in 8). When encoding by the same QP, the PSNR value of encoder with proposed ROI is a little bit lower while the bit-rate is much lower as shown as the spots in Fig.12. It means that higher average coding performance is achieved by accurately allocating QP according to proposed ROI.

Fig. 13 shows the effectiveness of either proposed ROI detection or complexity reduction algorithm by comparing the performance and encoding time of different combinations of ROI detection and fast algorithm proposed in this paper and 8). Implementing proposed ROI with the fast algorithm in 8) can reduce 69.29% computation time, while implementing proposed fast algorithm with ROI in 8) can reduce 68.37% computation time. We can conclude that the improvements of both the ROI detection and the complexity reduction algorithm guarantee the encoding time reduction of proposed H.264 encoder.

Fig. 15 Subjective quality comparison of Carphone in Fig. 14 (b). In this figure, (a) is encoded with 352 kbps bit-rate, 31.17 dB PSNR, and SSIM is 0.892. (b) is encoded with 345 kbps bit-rate, 33.81 dB PSNR, and SSIM is 0.927.

Table 3 Experimental conditions

<table>
<thead>
<tr>
<th>Frame Size</th>
<th>CIF (352×288) 720p (1280×720)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Frames</td>
<td>100</td>
</tr>
<tr>
<td>Test sequences</td>
<td>CIF: Carphone, Foreman</td>
</tr>
<tr>
<td></td>
<td>CIF: Headwithglasses, Silent</td>
</tr>
<tr>
<td></td>
<td>720p: Vidyo1, Vidyo4</td>
</tr>
<tr>
<td>Group of Pictures</td>
<td>PPP</td>
</tr>
<tr>
<td>RDO</td>
<td>ON</td>
</tr>
<tr>
<td>QP in non-ROI</td>
<td>32, 35, 40, 44</td>
</tr>
<tr>
<td>Search Range</td>
<td>32 for CIF, 64 for 720p</td>
</tr>
<tr>
<td>Number of Reference Frames</td>
<td>5</td>
</tr>
<tr>
<td>Search Mode of ME</td>
<td>Fast Full Search</td>
</tr>
<tr>
<td>Entropy Coding Method</td>
<td>CABAC</td>
</tr>
</tbody>
</table>

ROI: Region of Interest
which has stronger relation with perceptual visual quality of human eyes than PSNR. Each of the test sequences is encoded under 2 different bit-rate values with 2 kinds of ROI segmentation. The bit-rate difference of each pair of compared video sequence is less than 3%. $\Delta QP$ between ROI and non-ROI is 20. The left figure shows the comparison results of higher quality encoding, while the right figure shows the results of lower quality encoding. It can be seen that H.264 encoder with proposed ROI always obtains higher visual quality under the similar bit-rate budget. The subjective quality comparison is shown in Fig. 15. It can be seen that the skin of ROI in Fig. 15(a) is smoother than Fig. 15(b) which means the left ROI contains larger distortion. The non-ROI in Fig. 15(a) has lower bpp and shows obvious block effect because the larger ROI in 8) consumes larger ratio of bit-rate budget of the whole encoder.

Encoding time reduction and performance loss caused by 3 methods in proposed complexity reduction algorithm are shown in Fig. 16. Method 1, 2, and 3 are listed in Section 3.4. The test conditions are the same with Table 3. The $\Delta QP$ between ROI and non-ROI is set to 20. It can be seen that each method contributes to the overall computation time reduction. Method 2 leads to the highest time reduction rate and the least performance loss while method 3 introduces the heaviest performance loss but reduces computation time by 5.94%.

In Table 4, the performance of applying proposed computational complexity reduction algorithm in ROI and non-ROI is compared with full search style encoding respectively. Experimental conditions are the same with Table 3. The $\Delta QP$ between ROI and non-ROI is 20. The $\Delta$PSNR and $\Delta$bit-rate ($\Delta BR$) under the same QP are compared. Table 4 shows that the average visual quality of ROI is not reduced with slight increased bit-rate, while the visual quality and bit-rate of non-ROI is further reduced. This PSNR and bit-rate variation influenced by proposed fast algorithm is quite suitable for ROI based encoding, in which the quality of ROI and the bit-rate of the whole video stream are the most important.

### 5. Conclusion

This paper proposes a complexity reduction method for ROI based H.264 encoder in videophone, video conferencing and surveillance systems. A fast ROI detection algorithm is proposed to obtain small and accurate ROI for the encoder. The ROI size is reduced by 15.5% compared with previous work. Complexity reduction algorithm contains 3 main methods, the ROI and QP based mode selection, the unequal encoding parameter is assigned according to the unequal bits allocation in encoding. Only the ROI boundary MBs are encoded in highest complexity to further reduce the coding complexity of the central MBs of ROI since our proposed ROI detection algorithm can detect accurate ROI boundary. Mode selection is performed based on the partition size preference due to QP difference and ROI situation. Unequal encoding parameter is assigned according to the unequal bits allocation in encoding. Only the ROI boundary MBs are encoded in highest complexity to further reduce the coding complexity of the central MBs of ROI since our proposed ROI detection algorithm can detect accurate ROI boundary. Simulation results show that when QP difference between ROI and non-ROI is 20, 77.18% encoding time is reduced with 0.04 dB PSNR degradation and 1.02% bit-rate increase. 18.17% encoding time is further reduced compared with previous work by the improvements of both the ROI detection and
the complexity reduction algorithm.

References


(Received Aug. 24, 2010)
(Revised Feb. 9, 2010)
Minghui Wang  He received the B.E. degree in Department of Electronic Engineering from Tsinghua University in 2007. He is now Ph.D. candidate in Graduate School of Information, Production and Systems of Waseda University. His research interests are about algorithms of image and video processing.

Satoshi Goto (Member)  He was born on January 3rd, 1945 in Hiroshima, Japan. He received the B.E. degree and the M.E. degree in Electronics and Communication Engineering from Waseda University in 1968 and 1970, respectively. He also received the Dr. of Engineering from the same university in 1981. He is IEEE fellow, Member of Academy Engineering Society of Japan and professor of Waseda University. His research interests include LSI System and Multimedia System.