An Adaptive Video Sub-sampling Technique for the Conversion Between High and Low Resolution

Peter H. W. Wong*, Oscar C. Au**, Justy W.C. Wong⁺, A. Tourapis⁺⁺

Department of Electrical and Electronic Engineering
The Hong Kong University of Science and Technology
Clear Water Bay, Kowloon, Hong Kong.

Email: eepeter@ee.ust.hk*, eeau@ee.ust.hk**, eejusty@ee.ust.hk⁺, alexis@ee.ust.hk⁺⁺
Tel.: +852 2358-7053**

ABSTRACT

An adaptive sub-sampling technique is proposed for the conversion of high-resolution video to low-resolution video, which will eventually be converted back to high resolution. This technique can preserve high spatial frequency information by exploiting the temporal correlation of stationary regions without any extra storage or transmission bandwidth. Coding complexity can be reduced by a factor of four if the video is to be eventually displayed in high-resolution, with small degradation in visual quality and peak-signal-to-noise ratio. Most of the edge information can be preserved when the low-resolution video is converted back to high-resolution video and significant improvement can be obtained compared with conventional methods.

1. INTRODUCTION

Conversion between different resolutions of digital video is an important operation for digital video coding. Both the MPEG-2 [1] and MPEG-4 [2] support the scaling of video in spatial domain using additional storage or bandwidth to represent the error signal between the original high-resolution video and the up-sampled version of the low-resolution video. Conventional low-to-high resolution conversion methods such as bilinear interpolation and zero-order hold are simple but they exhibit blurring and blocking artifacts respectively. Many researches working on the extraction of high quality still image [3][4] or high-resolution video [5] using projections onto convex sets (POCS) based method applying on consecutive frames. All of them assumed that camera motions or motion blurring existing in the video sequence, but it is not always true. The visual quality of stationary background or object cannot be enhanced by these methods because there is no additional information in the temporal domain.

In order to preserve the high frequency content in the stationary regions, we propose an adaptive sub-sampling technique, which exploits the temporal correlation between consecutive frames. This technique consists of a pre-processing module prior to the video encoder and a post-processing module

Figure 1: Basic structure of proposed technique

after the decoder. The basic structure of the proposed technique is shown in figure 1. It should be noted that the video coder could be any standard coder provided that the bit rate for the video is high enough so that no temporal decimation would occur. As the video to be encoded and decoded is of low-resolution, the coding complexity is reduced by a factor of four when compared with coding the high-resolution video directly.

2. PRE-PROCESSING OF VIDEO

For the conventional sub-sampling approach, the high-resolution video is simply low-pass filtered and down-sampled. High spatial frequency information is filtered and cannot be recovered. Since the temporal correlation between consecutive frames for the stationary regions is very high, we exploit this correlation to transparently store the information so that the high spatial frequency information can be preserved. For the regions with motion, sub-sampling is carried out using conventional approach. The human visual system tends to be less sensitive to spatial resolution in these regions. Artifacts may be observed when the video is playback in lower speed. Here is the proposed algorithm

2.1. Motion detection

Assume that the high-resolution video sequence has spatial resolution M by N where M and N are even numbers. Assume also that the targeted low-

resolution video sequence should have the spatial resolution of $M/2$ by $N/2$. A motion map with the dimension of $M/2$ by $N/2$ is created between each pair of consecutive frames. Each entry of motion map is a binary number which represents whether the corresponding four pixels in the higher resolution frame are stationary or not. The sum of absolute difference (SAD) between two consecutive frames is compared with a pre-defined threshold for the creation of motion maps. The motion maps generation can be formulated by (1).

$$m(x, y, k) = \text{sign} \left\{ \sum_{i=0}^1 \sum_{j=0}^1 \text{abs} [F(2x+i, 2y+j, k+1) - F(2x+i, 2y+j, k)] - \text{threshold} \right\}$$

for $0 \leq x \leq M/2-1, 0 \leq y \leq N/2-1$ (1)

where $m(x, y, k)$ is the motion map between the k^{th} and $(k+1)^{\text{th}}$ frame, $F(x, y, k)$ is the xy^{th} pixel of the k^{th} original high-resolution frame. The *sign* function above is defined as:

$$\text{sign}(x) = \begin{cases} 1 & x > 0 \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

2.2. Decision map generation

A decision map for each frame is created which indicate that whether a group of four stationary pixels are stationary enough for four consecutive frames. The dimension of a decision is the same as the motion map and each entry consists of a binary number. The decision map for the xy^{th} pixel of the k^{th} frame, $d(x, y, k)$, is formulated as (3).

$$d(x, y, k) = \text{sign} \left[\sum_{i=0}^3 \prod_{j=0}^2 m(x, y, k+j-i) \right]$$

for $0 \leq x \leq M/2-1, 0 \leq y \leq N/2-1$ (3)

A decision value $d(x, y, k)$ of 0 means the xy^{th} pixel is stationary. On the other hand, a value of 1 means it is non-stationary.

2.3. Sub-sampling

An adaptive sub-sampling rule for the low-resolution frames $f(x, y, k)$ is formulated in (4) in order to preserve the information of the stationary regions as much as possible. For stationary pixels, sub-sampling is applied, whereas for non-stationary pixels, averaging is applied.

$$f(x, y, k) = \begin{cases} F(2x, 2y, k) \cdot d(x, y, k) \\ + A(x, y, k) \cdot [1 - d(x, y, k)] & \text{for } k \bmod 4 = 0 \\ F(2x, 2y+1, k) \cdot d(x, y, k) \\ + A(x, y, k) \cdot [1 - d(x, y, k)] & \text{for } k \bmod 4 = 1 \end{cases}$$

$$f(x, y, k) = \begin{cases} F(2x+1, 2y+1, k) \cdot d(x, y, k) \\ + A(x, y, k) \cdot [1 - d(x, y, k)] & \text{for } k \bmod 4 = 2 \\ F(2x+1, 2y, k) \cdot d(x, y, k) \\ + A(x, y, k) \cdot [1 - d(x, y, k)] & \text{for } k \bmod 4 = 3 \end{cases}$$

$$A(x, y, k) = \text{round} \left[\frac{1}{4} \cdot \sum_{i=0}^1 \sum_{j=0}^1 F(2x+i, 2y+j, k) \right]$$

for $0 \leq x \leq M/2-1, 0 \leq y \leq N/2-1$ (4)

where $(k \bmod 4)$ is the remainder when k is divided by 4.

2.4. Rearrangement of the sequence

As the highest correlation exists between frames that are four frames apart, the sub-sampled sequence is rearranged into four classes prior to any video encoder in order to obtain the highest coding efficiency. The relationship between the rearranged sequence and the original one as follows.

$$\begin{aligned} f_a(x, y, k) &= f(x, y, 4k) \\ f_b(x, y, k) &= f(x, y, 4k+1) \\ f_c(x, y, k) &= f(x, y, 4k+2) \\ f_d(x, y, k) &= f(x, y, 4k+3) \end{aligned} \quad (5)$$

3. POST-PROCESSING OF VIDEO

The rearranged video sequence can be encoded for storage or transmission. As the spatial information for the stationary regions is stored in consecutive frames, temporal resolution of the sequence should be maintained in order to recover the higher resolution sequence.

3.1. Rearrange the decoded sequence

The four classes of decoded sequence should be rearranged back to one sequence. This is the reverse process of section 2.4 described above.

$$g(x, y, k) = \begin{cases} g_a(x, y, k') & k \bmod 4 = 0 \\ g_b(x, y, k') & k \bmod 4 = 1 \\ g_c(x, y, k') & k \bmod 4 = 2 \\ g_d(x, y, k') & k \bmod 4 = 3 \end{cases} \quad (6)$$

$k' = \lfloor k/4 \rfloor$ is the largest integer less than or equal to $k/4$

3.2. Motion map estimation

The motion map estimation process consists of two stages. In the first stage, initial estimation is carried out based on the motion vectors information from the decoder. Stationary regions are estimated between any two consecutive frames. In the second stage, the absolute difference of each pixel between

two consecutive frames is compared with a pre-defined threshold with the stationary regions estimated in the first stage using a larger threshold value while the non-stationary regions using a smaller one.

Five decision maps for each frame are determined from the estimated motion maps $d(x, y, k)$, which determine where the spatial information can be recovered from the temporal neighbours for the stationary regions of each frame. The decision maps are created according to (7)

$$\begin{aligned}
d_a(x, y, k) &= \prod_{i=-2}^0 d(x, y, k+i) \\
d_b(x, y, k) &= \prod_{i=-1}^1 d(x, y, k+i) \cdot [1 - d_a(x, y, k)] \\
d_c(x, y, k) &= \prod_{i=-3}^{-1} d(x, y, k+i) \cdot [1 - d_a(x, y, k)] \\
&\quad \cdot [1 - d_b(x, y, k)] \\
d_d(x, y, k) &= \prod_{i=0}^2 d(x, y, k+i) \cdot [1 - d_a(x, y, k)] \\
&\quad \cdot [1 - d_b(x, y, k)] \cdot [1 - d_c(x, y, k)] \\
d_e(x, y, k) &= 1 - d_a(x, y, k) - d_b(x, y, k) \\
&\quad - d_c(x, y, k) - d_d(x, y, k) \quad (7)
\end{aligned}$$

3.3. Generation of low-resolution sequence

If it is targeted to display low-resolution video $h(x, y, k)$, the intensity of each stationary pixel takes the temporal average of corresponding temporal neighbours based on the decision maps. This process is described in (8).

$$\begin{aligned}
h(x, y, k) &= \text{round}\{[1 - d_e(x, y, k)] \cdot g(x, y, k) \\
&\quad + 1/4 \cdot d_e(x, y, k) [g(x, y, k-1) \cdot [d_a(x, y, k) \\
&\quad + d_b(x, y, k) + d_c(x, y, k)] + g(x, y, k-2) \\
&\quad \cdot [d_a(x, y, k) + d_c(x, y, k)] + g(x, y, k-3) \\
&\quad \cdot d_e(x, y, k) + g(x, y, k+1) \cdot [d_a(x, y, k) \\
&\quad + d_b(x, y, k) + d_d(x, y, k)] + g(x, y, k+2) \\
&\quad \cdot [d_b(x, y, k) + d_d(x, y, k)] + g(x, y, k+3) \\
&\quad \cdot d_d(x, y, k) + g(x, y, k)]\} \quad (8)
\end{aligned}$$

3.4. Reconstruction of high-resolution video

The reconstruction of high-resolution video is also based on the use of temporal information and the decision map. The up-sampled sequence is then passed through a spatial low pass filter for better visual quality and peak-signal-to-noise ratio (PSNR). The up-sampling processing is formulated in (9) and the impulse response of the low pass filter is shown in figure 2.

$$G(x, y, k) = \begin{cases} g(x', y', k) & \text{for } X \bmod 2 = k \bmod 2, \\ & Y \bmod 2 = k \bmod 2 \end{cases}$$

$$G(x, y, k) = \begin{cases} g(x', y', k-1) \cdot [d_a(x', y', k) + d_b(x', y', k) \\ + d_c(x', y', k)] + g(x', y', k+3) \cdot d_d(x', y', k) \\ + g(x', y', k) \cdot d_e(x', y', k) \\ \text{for } x \bmod 2 = k \bmod 2, \\ y \bmod 2 \neq k \bmod 2 \\ g(x', y', k-2) \cdot [d_a(x', y', k) + d_c(x', y', k)] \\ + g(x', y', k+2) \cdot [d_b(x', y', k)] + d_d(x', y', k) \\ + g(x', y', k) \cdot d_e(x', y', k) \\ \text{for } x \bmod 2 \neq k \bmod 2, \\ y \bmod 2 \neq k \bmod 2 \\ g(x', y', k-3) \cdot d_c(x', y', k) \\ + g(x', y', k+1) \cdot [d_a(x', y', k)] + d_b(x', y', k) \\ + d_d(x', y', k) + g(x', y', k) \cdot d_e(x', y', k) \\ \text{for } x \bmod 2 \neq k \bmod 2, \\ y \bmod 2 = k \bmod 2 \end{cases}$$

$$\begin{aligned}
x' &= \lfloor x/2 \rfloor \\
y' &= \lfloor y/2 \rfloor \quad (9)
\end{aligned}$$

0	1/18	0
1/18	7/9	1/18
0	1/18	0

Figure 2 Impulse response of the spatial filter

4. SIMULATION AND RESULTS

The luminous component of CIF (352 × 288) format “salesman” is used as the original high resolution. It is sub-sampled in the straightforward way to form a QCIF (176 × 144) sequence, which we will call QCIF2. The CIF sequence is also adaptively sub-sampled using the proposed scheme in section 2 to form a sequence QCIF1. The QCIF1 is adaptively up-sampled using the proposed scheme in section 3 to form a CIF sequence called “CIF1”. The QCIF2 is up-sampled by zero-order hold to form a CIF sequence called “CIF2”. Both QCIF1 and QCIF2 are compressed and decompressed by MPEG1 at 1300kbps/s. The decompressed QCIF2 is up-sampled using zero-order hold to form a CIF sequence called “CIF2”. The decompressed QCIF1 is up-sampled using the proposed scheme in section 3 to form a CIF sequence called “Coded CIF1 without filtering”. When the proposed filtering is applied, the resulting sequence is called “Coded CIF1 with filtering”.

Sequence	Average PSNR
CIF1	36.28 dB
Coded CIF1 without filtering	35.32 dB
Coded CIF1 with filtering	35.66 dB
Coded CIF2	29.12 dB

Table 1: Average PSNR of reconstructed video by

different low-resolution sequences

Figure 3: Original frame 80 of 'salesman'

Figure 4: Reconstructed frame 80 by coded QCIF2

Figure 5: Reconstructed frame 80 by coded QCIF1

after filtering

Figure 6: Average PSNR of reconstructed video

5. CONCLUSIONS

The proposed technique provides a novel way of video sequence sub-sampling, by exploiting the temporal correlation of stationary regions. High spatial frequency information such as edges can be well preserved in the stationary regions. If the display is targeted at low-resolution, it is very suitable for real-time zoom in operation because the complexity of the technique is small enough for software based implementation. Furthermore, if the display is targeted at high-resolution, this technique can reduce the encoding complexity by a factor of four with small loss in visual quality. Further investigations can be focused on improving the visual quality of non-stationary region or efficient coding of error images.

REFERENCE

- [1] "Generic Coding of Moving Pictures and Associated Audio," (MPEG-2), *ISO/IEC 13818-2*, Part 2: Video, November 1993.
- [2] J. Brailean, "MPEG-4 Universal Accessibility and Object-based Functionality," *ISCAS Tutorial on MPEG-4*, Chapter 3.3, June, 1997.
- [3] A.J. Patti, M.I. Sezan, and A.M. Rekalp, "Robust Methods for High-Quality Stills from Interlaced Video in the Presence of Dominant Motion," *IEEE Trans. on Circuits System for Video Technology*, vol. 7, No 2, pp. 328-342, April 1997.
- [4] R.R. Schultz. and R.L. Stevenson, "Extraction of High-Resolution Frames from Video Sequences" *IEEE Trans. on Image Processing*, vol. 5, No. 6, pp. 996-1011, June 1996.
- [5] Andrew J. Patti, M. Ibrahim Sezan and A. Murat Tekalp, "High Resolution Standards Conversion of Low Resolution Video," *Proceeding ICASSP 95*, vol. 4, pp. 2197-2200, May 1995.