Fast multiple reference FRAME motion estimation For h.264

Yeping Su and Ming-Ting Sun
Department of Electrical Engineering
University of Washington, Seattle, WA 98195

Abstract

Multiple reference frame motion compensation is a new feature introduced in H.264/MPEG-4 AVC to improve video coding performance. However, the computational cost of Multiple Reference Frame Motion Estimation (MRF-ME) is very high. In this paper, we propose an algorithm that takes into account the correlation/continuity of motion vectors among different reference frames. We also show that the algorithm effectively reduces the computations of MRF-ME, and achieves similar coding gain compared to the full-search approach.

1. Introduction

H.264/MPEG-4 AVC is the newest international video coding standard of the ITU-T Video Coding Experts Group and the ISO/IEC Moving Picture Experts Group [1]. It represents the state-of-the-art video compression technology and addresses the full range of video applications including low bit-rate wireless video applications, standard-definition & high-definition broadcast television, and video streaming over the Internet. In terms of compression performance, it provides more than 50% bit-rate savings for equivalent video quality relative to the performance of MPEG-2 video coding standard. To achieve such a high coding efficiency, H.264 includes many new features including variable blocksize motion compensation, quarter-pixel accuracy motion compensation, and multiple reference frame motion compensation.

In the variable blocksize motion compensation, H.264 supports luma block-sizes of 16x16, 16x8, 8x16, and 8x8 in the inter-frame prediction. In case 8x8 is chosen, further smaller block-sizes of 8x4, 4x8, and 4x4 can be used.

In the multiple reference frame motion compensation, a picture can be predicted from many reference pictures. A scenario of Multiple Reference Frame Motion Estimation (MRF-ME) is shown in Figure 1. It is an effective technique to improve the coding efficiency [3]. However, MRF-ME dramatically increases the computational complexity of the encoders because the Motion Estimation (ME) process needs to be performed for each of the reference frames. Considering motion estimation is the most computationally intensive function block in the video codec, this increased complexity penalizes the benefit gained from the better coding efficiency, and thus may restrict its applicability.

Figure 1. Multiple Reference Frame ME

The reference software of H.264/AVC, JM7.3 [2], performs the full-search motion estimation for all block-sizes across all reference frames in the encoder. In [4], a fast algorithm is proposed to speed-up the MRF-ME by considering the different sub-pixel sampling position of each block, and performing ME on the selected reference frames with similarly sampled contents. In [5], several heuristics are used to decide whether it is necessary to search more than the most recent reference frame, and hence reduce the computations.

In this paper, we first investigate why multiple reference frames provide better predictions, based on the observations from experiments on standard test sequences. We then propose a fast MRF-ME algorithm, which can achieve nearly the same coding efficiency as the full-search approach, but cut down the computations by roughly three fold.

The paper is organized as follows. In Section 2, we first introduce the MRF-ME in H.264 and then give some analyses to reveal why MRF-ME enhances the prediction. Section 3 details the proposed algorithm. Simulation results are shown in Section 4. Finally, conclusions are presented in Section 5.
2. Analyses of multiple reference FRAME Motion Estimation

2.1. Multiple Reference Frame Motion Estimation

In the reference software JM7.3, the Motion Vector (MV) search returns the MV that minimizes
\[ J(m, \lambda_{\text{MOTION}}) = \text{SAD}(s, c(m)) + \lambda_{\text{MOTION}} \cdot R(m - p) \]
with \( m = [m_x, m_y]^T \) being the MV, \( p = [p_x, p_y]^T \) being the prediction for the MV, and \( \lambda \) being the Lagrange multiplier. In the process, the MV search is done for all reference pictures, so the computational cost increases linearly with the number of reference pictures.

2.2. Why Multiple Reference Frame Help Predictions?

There are many reasons for the MRF-ME to achieve better predictions than those using just single reference picture. Often cited reasons include [5]:

a) Repetitive motions. Due to the repetitive nature of the motion, there are better appearances of the same object/texture several frames ago.

b) Uncovered background. Some parts of the picture may originally be covered by a moving object. As the object moves, the uncovered backgrounds may not find a good match from the previous frame, but may be able to find a good match from several frames ago when they were also uncovered. Falling leaves in the Tempete sequence falls into this category. The uncovered backgrounds occur next to the moving object boundaries.

c) Alternating camera angles that switch back and forth between two different scenes.

Besides these often cited reasons, based on our observations from experiments on standard video sequences, there are several other reasons why MRF-ME performs better than the motion estimation using single reference frame:

d) Camera shaking, such as the last part of the Foreman sequence. When a camera is moving up and down, the current frame may better resemble a frame appeared several frames ago. Those parts moving into the picture may also be better predicted from a frame appeared several frames ago.

e) Sampling. When an object moves with a non-integer pixel displacement, the sampling positions of the object in different frames may be different. Due to this different sampling, the current block may get a better match to a block in more previous reference frames. An example is the moving calendar in the MobileCalendar sequence, with a zoom-in view of the current block to be coded and the best matched blocks in the previous two reference frames shown in Figure 2. The ResidualBlock(N-1) has MSE=203.8, versus 32.4 for ResidualBlock(N-2).

f) Shadow changes. As an object moves, the un-shadowed backgrounds may not find a good match from the previous frame, but may be able to find a good match from several frames ago when they were also un-shadowed. The un-shadowed backgrounds also often occur next to the moving object boundaries.

g) Lighting changes. An area or a moving object may not have exactly the same pixel values as those at the previous locations in the previous frame since they may have different lighting conditions or reflections.

h) Noises in the source signal produced by the camera and other factors. Even in the stationary areas of the picture, some blocks may find a better match in more previous reference pictures, which happens in many sequences.

i) Other reasons. Such as new contents on picture boundary or occluded areas which can only find a good reference by chances, and sometimes a better chance is found on more previous reference pictures.

The situations of e), f), g), and h) actually occur quite often. Under these situations, there are strong correlations among the motion vector fields with multiple reference frames. Our proposed MRF-ME algorithm is based on the intuition that due to the strong correlations of the MVs with multiple reference frames, we can find a better reference by tracing along the motion trajectories, instead of performing the full-search on every reference frame.

2.3. Correlations in Multiple Reference Picture Motion Vectors
The correlation among the motion vector fields in multiple reference frames can be utilized to compose motion vectors:

\[ MV_{n-k} = MV_{n-k_1} + MV_{n-k_1} \]

where \( MV_{n-k} \) represent the motion vectors of Frame \( n \) referring to Frame \((n-k)\). We try to use the correlation to save the computation of the MRF-ME process, i.e., to compose the motion vector \( MV_{n-k} \) by combining \( MV_{n-k_1} + MV_{n-k_1} \), or to perform ME only along the motion trajectories.

In order to verify this idea of the MV correlation, some simple data analyses are performed on the MVs from the full-search MRF-ME in JM7.3. For those blocks with motion vectors decided to be \( k = 2 \), the Mean Square Error (MSE) between \( MV_{n-2} \) and the composed \( MV_{n-1} + MV_{n-1} \) are collected into a histogram shown in Figure 3. Note that in the composing of the motion vectors, all MV’s are represented in the 4x4 blocksize which is the smallest unit in the H.264 variable block-size definition.

Since in general the referred block in Frame \((n-1)\) does not fall onto block boundaries, we end up having multiple possibly different composed MV’s, and in those cases the average MV is used.

One typical histogram of the MSE’s between the composed MV and the MV from the full-search collected from coding one frame of the foreman sequence is plotted in the Figure 3. As we can see, the majority of composed MV’s are very close to the ones resulted from the full-search, which justifies the motion concatenation in the fast motion search algorithm elaborated in the following section.

3. proposed algorithm

Based on the idea of back tracing the motion trajectory, the proposed fast MRF-ME algorithm is described in this section.

When coding Frame \( n \), only a full-search referenced to Frame \((n-1)\) is performed, which results in \( ME_{n-1} \). For \( ME_{n-k} \) with \( k \) starts from 2 to \#ref (the number of the multiple reference frame), motion composition is performed as:

\[ ME_{n-k} = ME_{n-k}^{-1} + ME_{n-k-1}^{-1} \]

In this composition, the MV estimation \( ME_{n-k} \) is formed by:

- For each Macroblock (MB), in the results of \( ME_{n-k}^{-1} \), make a decision among the 7 block patterns defined in H.264 using motion search costs \( J’s \), and a 4x4 based MV field \( MV_{n-k}^{-1} \) is obtained.

- For a specific partition in a specific MB pattern, collect 4x4 block MV’s in \( MV_{n-k}^{-1} \). Going along these MV’s, and combining them with \( MV_{n-k-1}^{-1} \), we have many candidate MV’s with different overlapping areas. Figure 5 shows this process pictorially: there are possibly six overlapping partitions involved, and only one concatenation is shown.

- Try each one of the candidate MV’s, and pick the final estimate based on the minimum cost.

- Perform refinement within \( \pm 1 \) pixel around the final estimate.

4. simulation results

The proposed MRF-ME algorithm was tested on four sequences which show significant gains when
MRF-ME is enabled. The scheme is implemented based on the reference software JM7.3. The Computation Reduction in Table 1 is the averaged speedup of the motion estimation processes: $\frac{T_{\text{FullSearch}}}{T_{\text{FastSearch}}}$, where $T$ denotes the total runtime of the motion estimation subroutine. The experiments are conducted on a PC with 2 GHz P4 CPU, with respective search range listed in Table 1. Rate-distortion curves are shown in Figure 5. The proposed scheme performs almost the same as that using the full-search on each reference frame.

![Figure 5. Coding results. (a) Foreman, (b) MobileCalendar, (c) Tempete, (d) Carphone](image)

<table>
<thead>
<tr>
<th>Sequence</th>
<th>Resolution</th>
<th>Search Range</th>
<th>Computation Reduction</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mobile</td>
<td>CIF, 300frame</td>
<td>±32</td>
<td>2.85</td>
</tr>
<tr>
<td>Tempete</td>
<td>CIF, 260frame</td>
<td>±32</td>
<td>2.77</td>
</tr>
<tr>
<td>Foreman</td>
<td>CIF, 300frame</td>
<td>±32</td>
<td>2.83</td>
</tr>
<tr>
<td>Carphone</td>
<td>QCIF, 382frame</td>
<td>±16</td>
<td>3.07</td>
</tr>
</tbody>
</table>

5. conclusion

In this paper, a novel multiple reference frame motion estimation algorithm is proposed. In the MRF-ME process, initial MVs are formed based on the motion trajectories and refinements are used to improve the accuracy of the motion vector composition. Results show the scheme is very effective in reducing the computational cost while keeping good coding efficiency.

References

[2] Joint Video Team software, JM7.3