New frame rate up-conversion using bi-directional motion estimation

Author: Byung-Taw Choi, Student Member, IEEE, Sung-Hee 2 and Sung-Jea Ko 1, Senior Member, IEEE
1 Department of Electronics Engineering, Korea University, Korea
   E-mail: sjko@dali.korea.ac.kr
2 Digital Media System Lab, Samsung Electronics Co, Korea

Publisher: IEEE Transactions on Consumer Electronics, Vol. 46, No. 3, August 2000

谷爱国

School of electronic engineering, XiDian University
Introduction

- What is frame rate up-conversion
Why frame rate up-conversion is required?

- Different display devices, different frame rate.
- Frame rate up-conversion technique can be used for video compression and slow motion replay.
Review on Conventional MCI’s

(1) Moving objects: Bi-directional MCI

\[ f_{ti}(p) = W_b * f_{t1}(p-W_f * D_{1,2}(p)) + W_f * f_{t2}(p+W_b * D_{1,2}(p)) \]

(2) Covered background: Forward prediction

\[ f_{ti}(p) = f_{t1}(p) \]

(3) Uncovered background: Backward prediction

\[ f_{ti}(p) = f_{t2}(p) \]

(4) Static background: Linear interpolation

\[ f_{ti}(p) = W_b * f_{t1}(p) + W_f * f_{t2}(p) \]

\( f_{ti}(p) \): pixel intensity
\( D_{1,2}(p) \): the motion vector at pixel \( p \) from the previous frame \( f_{t1} \)
to the current frame \( f_{t2} \)
\( W_f = (t_i-t_1)/(t_2-t_1) \); \( W_b = (t_2-t_i)/(t_2-t_1) \)

(Fig.1)
The proposed algorithm

Initial motion vector estimation

Refinement of $V_{oi}$

Frame interpolation using overlapped MCI

Spatio-temporal smoothing

$V_{oi}$

$V_i$
1. Initial motion vector estimation

- We get the initial motion vector with the full search block-matching algorithm. The search region is usually symmetric with respect to the current block, up to $R_x$ pixels to the left and right, and up to $R_y$ pixels above and below.

- The initial motion vector is used to initialize the initial value of the bi-directional motion vector without any modification.

\[
MAD[B_{i, j}, \vec{D}] = \sum_{p \in B_i, j} \left| f_{n-1}(p) - f_{n+1}(p + \vec{D}) \right|
\]

\[
\vec{D}_s(B) = \arg \min \text{MAD}[B_{i, j}, \vec{D}](\vec{D}_s(B) \in \vec{D})
\]
2. Refinement of initial motion vector

- Using bi-directional motion vector estimation
- To solve the problem of overlapped pixels and the hole

\[ (x_1, y_2) = \vec{p} - \vec{D} \]
\[ (x_2, y_2) = \vec{p} + \vec{D} \]

\[ \text{MAD}[B_i, j, \vec{D}] = \sum_{p \in B_i, j} | f_{n-1}(p - \vec{D}) - f_{n+1}(p + \vec{D}) | \]

\[ \vec{D}_s(B) = \arg \min \text{MAD}[B_i, j, \vec{D}](\vec{D}_s(B) \in \vec{D}) \]
3. Spatio-temporal smoothing

\[
MAD[B_i, j, \vec{D}] = \sum_{p \in B_i, j} | f_{n-1}(p - \vec{D}) - f_{n+1}(p + \vec{D}) |
\]

\[
\vec{D}_s(B) = \arg \min \, MAD[B_i, j, \vec{D}] (\vec{D}_s(B) \in \vec{D})
\]

(Fig.3) interpolated frame

\( \vec{D}(B) \) and \( \vec{D}(N_i) \) denote the corresponding bidirectional motion vector of B and N\(_i\).

\( \vec{D}(B), \vec{D}(N_i) \subset \vec{D} \)
4. Overlapped block MCI

➢ To eliminate the blocking artifact

(Fig. 4 Block overlapping pattern in the overlapped block MCI.)

\[ R_1 : f_0(\overrightarrow{p} \in R_1, \overrightarrow{D}(B)) \]

\[ R_2 : \frac{1}{2} \{ f_0(\overrightarrow{p} \in R_2, \overrightarrow{D}(B)) + f_0(\overrightarrow{p} \in R_2, \overrightarrow{D}(N_i)) \} \]

\[ N_i \in \{ N_2, N_4, N_5, N_7 \} \]

\[ R_3 : \frac{1}{4} \{ f_0(\overrightarrow{p} \in R_3, \overrightarrow{D}(B)) + S_k \} \]

\[ S_1 = f_0(\overrightarrow{p}, \overrightarrow{D}(N_1)) + f_0(\overrightarrow{p}, \overrightarrow{D}(N_2)) + f_0(\overrightarrow{p}, \overrightarrow{D}(N_4)) \]

\[ S_2 = f_0(\overrightarrow{p}, \overrightarrow{D}(N_2)) + f_0(\overrightarrow{p}, \overrightarrow{D}(N_3)) + f_0(\overrightarrow{p}, \overrightarrow{D}(N_5)) \]

\[ S_3 = f_0(\overrightarrow{p}, \overrightarrow{D}(N_4)) + f_0(\overrightarrow{p}, \overrightarrow{D}(N_6)) + f_0(\overrightarrow{p}, \overrightarrow{D}(N_7)) \]

\[ S_4 = f_0(\overrightarrow{p}, \overrightarrow{D}(N_5)) + f_0(\overrightarrow{p}, \overrightarrow{D}(N_7)) + f_0(\overrightarrow{p}, \overrightarrow{D}(N_8)) \]
In this experiment four test sequences are used and each sequence contains frames with a specific camera motion.

### Experimental Design

<table>
<thead>
<tr>
<th>sequence</th>
<th>Frame size</th>
<th>Typical camera motion</th>
</tr>
</thead>
<tbody>
<tr>
<td>Foreman</td>
<td>176*144</td>
<td>complicated movement</td>
</tr>
<tr>
<td>Interview</td>
<td>720*576</td>
<td>large displacement</td>
</tr>
<tr>
<td>Inition-2d3d-Showreel</td>
<td>960*544</td>
<td>zoom in/out</td>
</tr>
<tr>
<td>InnerGate</td>
<td>640*384</td>
<td>large displacement, zoom in / out</td>
</tr>
</tbody>
</table>
Experimental Design

- For each sequence, we perform 1:2 frame rate up-conversion. The block size is 16*16 and the search range is ±16. The search range is from -2 to +2 in motion estimation for the refinement of the initial motion vector. In overlapped block MCI, we use the overlapping width w=2.

- Evaluative criteria

  - PSNR
  - Visual performance
Experimental Results

(Fig 6. (a): the original image; (b): after the first step; (c): after the second step; (d): after the third step; (e): after the fourth step)
Experimental Results

(Fig 7. (a): the original image; (b): after the first step; (c): after the second step; (d): after the third step; (e): after the fourth step)
Experimental Results

(Fig 8. (a): the original image; (b): after the first step; (c): after the second step; (d): after the third step; (e): after the fourth step)
Experimental Results

(Fig.9. (a): the original image; (b): after the first step; (c): after the second step; (d): after the third step; (e): after the fourth step)
Experimental Results

(Fig10. (a): the original image; (b): after the first step; (c): after the second step; (d): after the third step; (e): after the fourth step)
Conclusions

The main feature of the proposed motion-compensated frame rate up-conversion scheme is that, unlike conventional MCI algorithm, the proposed technique does not produce any overlapped pixels and hole region in the interpolated frame. It has better performance than the conventional MCI algorithm and is very robust especially in sequences with camera motions like panning and zoom.
Thank you