Adaptive Bidirectional Time-Recursive Interpolation for Deinterlacing

Jelena Kovačević, Robert J. Safranek and Edmund M. Yeh
Signal Processing Research Department
600 Mountain Avenue
AT&T Bell Laboratories
Murray Hill, NJ 07974

There exists a need for finding a good deinterlacing (scan format conversion) system, since, for example, current available cameras are interlaced and the US HDTV Grand alliance has put forward a proposal containing both interlaced and progressive scanning formats. On the other hand, over the next few years, the local broadcasting stations will find themselves in the position of receiving video material that could be HDTV quality, progressively scanned, while their news/commercials are still NTSC produced (interlaced scanning). Yet another application springs out of the integration of multimedia services on desktops and workstations, where one needs progressive video for display.

In the current literature, there is no systematic way of dealing with this problem. We have developed a new algorithm for deinterlacing based on the algorithm in [1]. It interpolates the missing pixels using a weighted combination of spatial and temporal methods. The algorithm is self-adaptive, since it weights various processing blocks based on the error they introduce. Thus, in parts with high motion content, for example, the algorithm will weigh in heavily the motion-based blocks of the system. Spatial interpolation methods include vertical and orientational interpolation, while the temporal ones are linear temporal interpolation and motion-compensated interpolation along the trajectory of the estimated motion. The algorithm is embedded; it vertically interpolates first, followed by temporal interpolation. This result is used to orientationally interpolate the field which is then in turn used to perform motion-compensated interpolation. Previously interpolated field is used in motion estimation, making the algorithm time recursive. Moreover, motion compensation is performed bidirectionally to allow for better performance in case of scene changes, or covered (uncovered) objects.

Experiments were run on both “real-world” and computer generated video sequences. The results were compared to the “original” obtained as an output of the ray-tracer, as well as to the reference algorithm provided by the AT&T HDTV group. A pilot subjective test was run and the overall conclusion was that the algorithm performs very well. Since the current version of motion compensation involves block matching, improvements could be obtained by using a pixel-based motion estimation algorithm.

References