MULTI-RESOLUTION MOTION ESTIMATION

Gregory J. Conklin, Sheila S. Hemami

Visual Communications Lab
School of Electrical Engineering
Cornell University, Ithaca, NY 14853
gjc2@cornell.edu hemami@ee.cornell.edu

ABSTRACT

Spatial multi-resolution video sequences provide video at multiple frame sizes, allowing extraction of only the resolution or bit rate required by the user. This paper proposes fine-to-coarse motion estimation (ME) for multi-resolution video coding. While coarse-to-fine ME, used in previously proposed coding schemes, can provide a better estimate at the coarsest resolution, it is outperformed by fine-to-coarse ME at finer resolutions due to the inability of coarse-to-fine ME to accurately track motion at finer resolutions. At the finest resolution, fine-to-coarse ME provides a PSNR improvement of up to 1 dB, for the sequences tested, and better visual quality at all resolutions. In addition, fine-to-coarse ME provides more accurate and thus more compressible motion estimates.

1. INTRODUCTION — COARSE-TO-FINE VS. FINE-TO-COARSE

Spatial multi-resolution video sequences provide video at multiple frame sizes, allowing extraction of only the resolution or bit rate required by the user. Generating such a coded video sequence presents a challenge. Temporal coding such as motion estimation (ME) is needed to best exploit frame-to-frame correlation. However, this coding must be combined with multi-resolution spatial coding. Such coding should minimize both memory requirements and coding delay. Additionally, the ability to easily decode any given frame is beneficial to allow fast, random access to any point in the sequence.

Several video coding algorithms have been developed that combine spatial multi-resolution and temporal coding [1-6]. The MPEG-2 standard provides two frame sizes by upsampling and prediction [1], while [2-6] employ spatial subband coding to provide several frame sizes. [2-4] combine temporal subband coding and motion compensation, producing good results but blurring motion in lower-frame-rate video [4] and making individual frame access difficult due to the recursive temporal decomposition. [5,6] employ temporal coding using motion vectors (MVs) and prediction residuals, facilitating random-frame decoding, reducing memory requirements and coding delay over temporal subband coding, and producing clear low-frame-rate video.

Using forward motion estimation, in [5,6], MVs are first computed for non-overlapping blocks of the current frame at the coarsest resolution using a reference frame at the coarsest resolution. These MVs are then coded and transmitted along with a prediction residual. MVs are then refined using the next finer resolution. The refinements are coded and transmitted along with the necessary prediction residual information. This algorithm iterates until all resolutions have been coded. In this scheme, motion is estimated in a *coarse-to-fine* manner. A drawback of coarse-to-fine ME is the potential for inaccurate ME at the coarsest resolution, due to lack to detail and aliasing effects. These inaccuracies, while providing a good prediction at the coarsest resolution, result in suboptimal ME at finer resolutions.

To alleviate this problem, this paper proposes *fine-to-coarse* ME with spatial subband coding to provide multi-resolution video. Accurate motion estimates are formed at the finest resolution and then scaled to coarser resolutions in the encoding process. These motion estimates better track the true motion and exhibit lower entropy than coarse-to-fine estimates, providing higher quality, both visually and quantitatively, at the same bit rates. Fine-to-coarse ME is a general technique that can be easily incorporated into video coding algorithms providing multiple frame rates, frame sizes and visual qualities, and hence can improve both compression performance and visual results over a broad range of coded video.

2. MULTI-RESOLUTION MOTION ESTIMATION ALGORITHMS

Two general coding algorithms have been used to compare fine-to-coarse with coarse-to-fine ME. The first is based on [5,6]; the second is the proposed algorithm. The only difference between these algorithms is the frame resolution at which motion is initially estimated. Both algorithms code the same information: A set of MVs for the coarsest resolution, a set of MV refinements for each additional frame resolution, and a multi-resolution representation of the prediction residual. For the following, $x(n)$ is the two dimensional signal representing a frame of the original sequence at time index n.

The coarse-to-fine ME algorithm is as follows.

1) From $x(n)$ obtain coarser resolution sequences, $x_k(n)$, using an N level multi-resolution decomposition, where $x_0(n)$ is the coarsest representation and $x_N(n)$ is the finest representation.
2) Set n=0. Code the multi-resolution representation of $x(0)$.

3) Increment n. Form the best set of MVs for non-overlapping blocks of $x_0(n)$ from $x_0(n\text{-}1)$. Set k=0. From the MVs for $x_0(n)$, form a motion compensated prediction, $\hat{x}_0(n)$, and a prediction residual, $\tilde{x}_0(n) = x_0(n) - \hat{x}_0(n)$. Code the MV information, and the prediction residual, $\tilde{x}_0(n)$.
4) Increment k. Scale and refine MVs found for $x_{k-1}(n)$ so they describe the motion for non-overlapping blocks of $x_k(n)$ from $x_k(n\text{-}1)$. From these MVs, form a motion compensated prediction, $\hat{x}_k(n)$, and a prediction residual, $\tilde{x}_k(n) = x_k(n) - \hat{x}_k(n)$. Code the MV refinement information, and the three high-pass subbands of the multi-resolution decomposition of the prediction residual, $\tilde{x}_k(n)$.
5) If k=N, and there are more frames to code, return to step (3). If $k \neq N$, return to step (4).

The fine-to-coarse ME algorithm is as follows.
1) From $x(n)$ obtain coarser resolution sequences, $x_k(n)$, using an N level multi-resolution decomposition, where $x_0(n)$ is the coarsest representation and $x_N(n)$ is the finest representation.
2) Set n=0. Code the multi-resolution representation of $x(0)$.
3) Increment n. Form the best set of MVs for non-overlapping blocks of $x_N(n)$ from $x_N(n\text{-}1)$. For k=N-1$\rightarrow$0, scale and refine MVs found for $x_{k+1}(n)$ so they describe the motion of non-overlapping blocks of $x_k(n)$ from $x_k(n\text{-}1)$. Set k=0. From the MVs for $x_0(n)$, form a motion compensated prediction, $\hat{x}_0(n)$, and a prediction residual, $\tilde{x}_0(n) = x_0(n) - \hat{x}_0(n)$. Code the MV information, and the prediction residual, $\tilde{x}_0(n)$.
4) Increment k. From the MVs for $x_k(n)$ (found in step(3)), form a motion compensated prediction, $\hat{x}_k(n)$, and a prediction residual, $\tilde{x}_k(n) = x_k(n) - \hat{x}_k(n)$. Code the MV refinement information, and the three high-pass subbands of the multi-resolution decomposition of the prediction residual, $\tilde{x}_k(n)$.
5) If k=N, and there are more frames to code return to step (3). If $k \neq N$, increment k, return to step (4).

From these two general algorithms, five specific algorithms have been simulated — two using coarse-to-fine ME (CtF, CtF-FBS), two using fine-to-coarse ME (FtC-T, FtC-R) and one without ME (NMC).

CtF. The first coarse-to-fine ME algorithm uses a block size of 2x2 pixels at the coarsest resolution, doubling in size both horizontally and vertically with each increasing frame resolution. Motion is initially estimated at half-pel resolution within a search window of 2 pixels at the coarsest resolution, and refined to half-pel resolution within a range of 1 pixel with each increasing frame resolution.

CtF-FBS. The second coarse-to-fine ME algorithm is very similar to an algorithm presented in [5]. It estimates motion in the same manner as CtF but uses a fixed block size of 11x10 pixels at all resolutions.

FtC-T. The first fine-to-coarse ME algorithm uses a block size of 16x16 pixels at the finest resolution, halving in size both horizontally and vertically with each decreasing frame resolution. These 16x16 blocks correspond to the same spatial locations as the 2x2 blocks in the coarsest resolution frames using the CtF algorithm. Motion estimates are initially formed at half-pel resolution within a search window of 16 pixels at the finest resolution, and are halved and truncated to half-pel resolution with each decreasing frame resolution.

FtC-R. The second fine-to-coarse ME algorithm estimates motion in the same manner as FtC-T, but refines estimates to half-pel resolution within a range of 1 pixel with each decreasing frame resolution.

Quarter size chrominance frames are coded in the same way as luminance frames, using scaled MVs.

3. IMPLEMENTATION ASPECTS AND SIMULATION RESULTS

Simulations have been done on 352x240, 4:2:0 *Football*, *Table Tennis*, *Flower Garden* and *Mobile and Calendar* sequences using a three level wavelet decomposition with biorthogonal wavelet filters of lengths 7 and 9. MVs were formed by finding the relative location of the best matched blocks, in terms of squared difference, in the previous frame. Compression results are obtained by calculating the entropy of the uniformly quantized prediction residuals and independently DPCM coded MVs and refinements.

3.1 Motion Estimates and MV Data

In general, coarse-to-fine motion estimates exhibit much larger entropy than fine-to-coarse estimates. For the finest resolution *Flower Garden* sequence, CtF motion estimates require 930 KB for 150 frames, CtF-FBS motion estimates require 550 KB, while FtC-T motion estimates require 333 KB. For the *Table Tennis* sequence, CtF motion estimates require 946 KB, CtF-FBS estimates require 700 KB, while FtC-T estimates require 499 KB. Fine-to-coarse ME is better able to track motion and is thus able to take advantage of uniform motion over several pixel blocks using DPCM. Coarse-to-fine ME can give inaccurate motion estimates and thus may detect more random motion than is present, resulting in inefficient coding.

Inaccurate estimates at the coarsest resolution are caused by three effects. Since finite-extent subband filters are used, coarser resolution frames contain aliasing components, preventing translational motion at one resolution to result in true translational motion at other resolutions. A good mathematical explanation of this is given in [5]. Furthermore, motion is best estimated through the translation of edges and textures of objects in the scene. Thus, ME at coarser resolutions is more difficult since edge, texture, and detail information is lost. In addition, when ME is done using a coded reference frame, quantization noise can confuse ME algorithms. This effect is made worse by using smaller pixel blocks for estimation.

Examples of inaccurate ME are shown in Fig. 1. The MVs at the finest resolution for the second frame of the *Table Tennis* sequence, using each algorithm, are shown. The estimates from the CtF algorithm seem very random. Since they are initially estimated at the coarsest resolution on 2x2 pixel blocks, quantization noise and lack of detail hinder the algorithm. The CtF-FBS algorithm produces less random MVs, yet gives inaccurate estimates. Here, motion is initially estimated using 11x10 pixel blocks at the coarsest resolution, giving a total of twelve MVs. However, the algorithm is unable to accurately refine these

estimates at finer resolutions. The FtC-T and FtC-R algorithms give the best results, producing the same MVs. Some inaccuracies occur due to quantization noise, but give far better estimates that exhibit less entropy.

3.2 PSNR and Compression Results

Fig. 2 gives PSNR-rate curves for the *Football* and *Table Tennis* sequences at each resolution using each algorithm (except FtC-R). Comparing the FtC-T and the FtC-R algorithms at any resolution, we find that FtC-R provides a smaller residual at coarser resolutions, but requires more bits to refine motion estimates, such that both algorithms perform comparably. While a single simulation generates a single PSNR-rate point for each resolution on these curves, using progressive refinement techniques, as in [2,6,8], any rate (and thus quality) for any desired resolution can be achieved from one coded stream.

At the finest resolution FtC-T performs as well as or better than all other algorithms. In sequences with active motion, like *Football* and *Table Tennis*, fine-to-coarse ME improves performance by about 0.5 dB over its nearest competitor, CtF-FBS, and 1 dB over CtF, at a bit rate of 1.5 Mbits/sec. For the other sequences tested, FtC-T performs as well as the other algorithms. In addition, the NMC algorithm performs poorly at full resolution, as expected. The CtF algorithm also gives poor results.

Below the finest two resolutions, CtF-FBS outperforms other algorithms for all sequences. However, this is not necessarily due to improved motion compensation, but because at coarser resolutions CtF-FBS has fewer MVs to code. For example, because CtF-FBS forms less accurate motion estimates, coding the coarsest resolution *Football* sequence using CtF-FBS, at a PSNR of 51.1 dB, results in a prediction residual that is 26% larger than the residual generated by FtC-T, and 230% larger than that generated by CtF, yet yields a lower overall bit rate.

At the coarsest resolution, the NMC algorithm performs best for the *Football* and *Table Tennis* at higher rates, but falls just below CtF-FBS at lower rates. At this resolution, FtC-T, FtC-R and CtF inefficiently code motion information since a single MV is used for every 2x2 pixel block. Likewise, the amount of motion information coded in the CtF-FBS algorithm can reduce its performance to below that of NMC. Due to this inefficiency, eliminating motion compensation at coarser resolutions has been proposed [5]. Employing this strategy with fine-to-coarse ME, the performance of the NMC algorithm can be obtained at the coarser resolutions, while maintaining superior performance at finer resolutions. For sequences like *Flower Garden* and *Mobile and Calendar*, some motion compensation at the coarsest resolution will improve performance since these sequences consist of a relatively swift camera translation. For these sequences CtF-FBS outperforms NMC on average by 1 dB.

3.3 Visual Results

Visually, fine-to-coarse ME gives better results at the two finest resolutions, for the same bit rate. Coarse-to-fine ME not only generates larger distortion, but this distortion tends to be more apparent in the active regions of the scene, where motion has been inaccurately estimated. In addition,

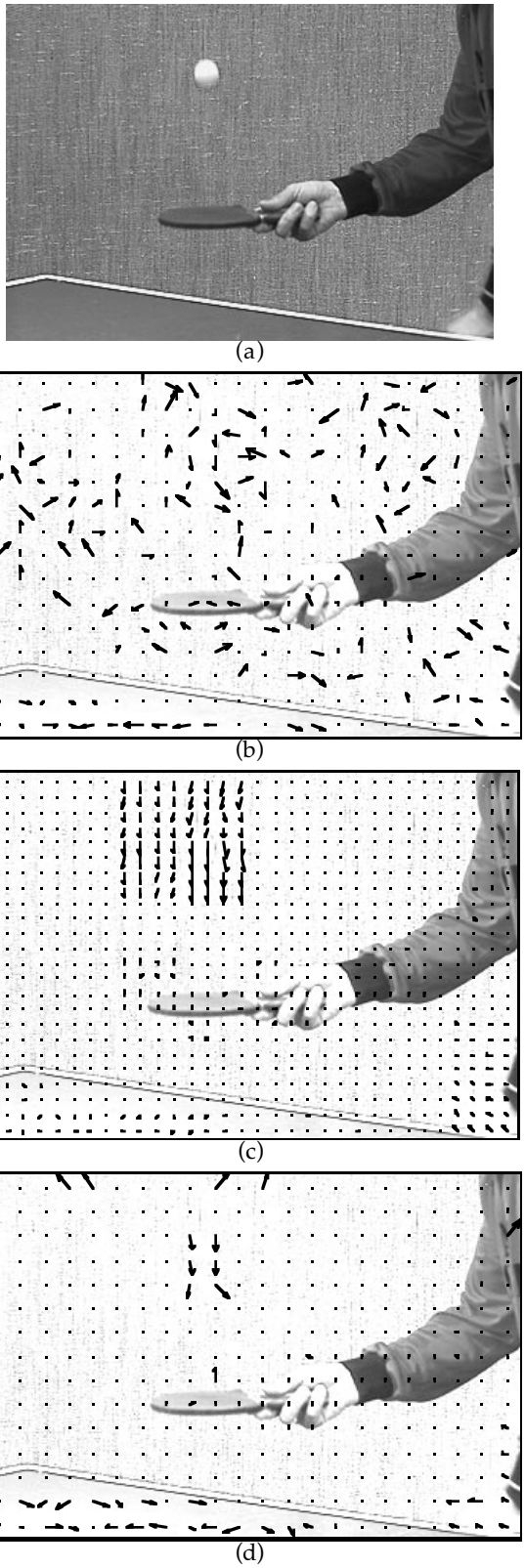

Fig 1: Motion estimates for frame 2 of the *Table Tennis* sequence using different algorithms. (a) original frame (b) CtF (b) CtF-FBS (c) FtC-T(R)

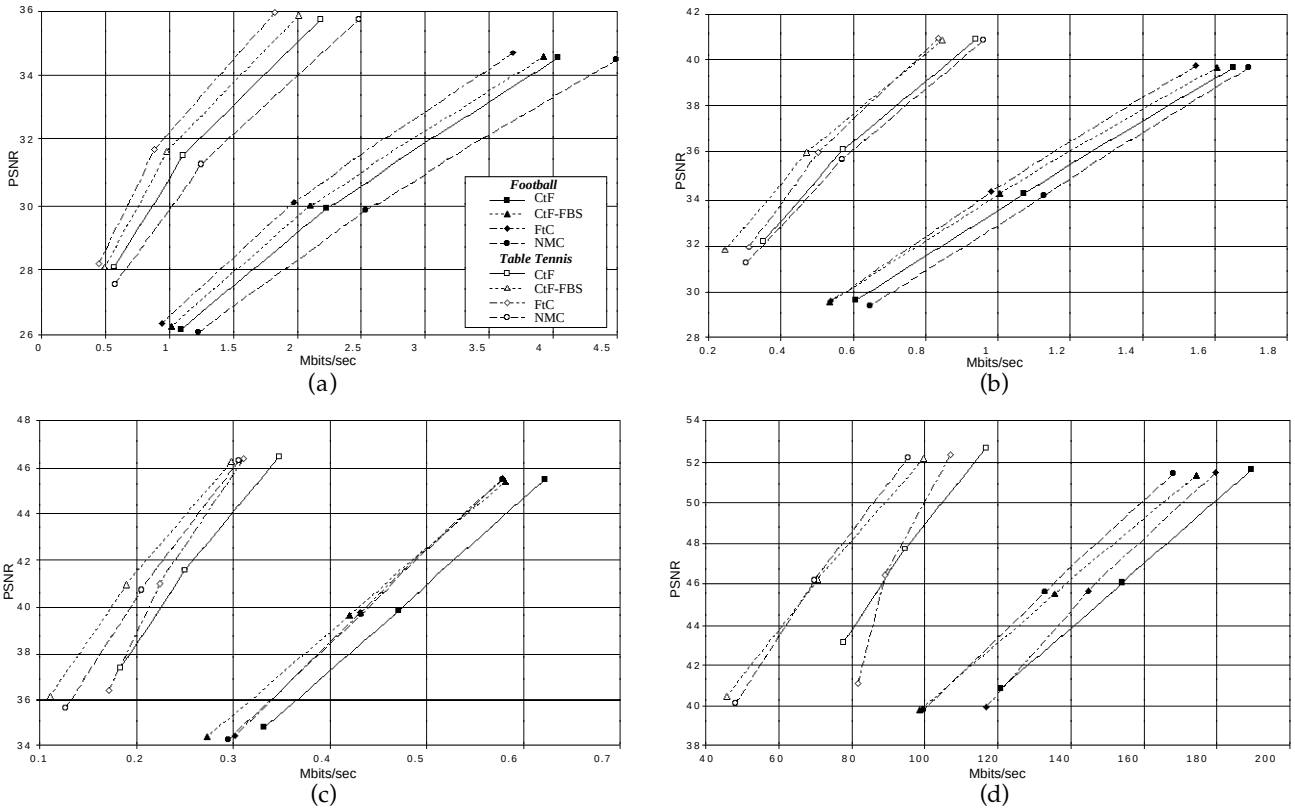

Fig 2: PSNR-rate curves for *Football* and *Table Tennis* for various resolutions.
(a) 352x240 (b) 176x120 (c) 88x60 (d) 44x30

fine-to-coarse ME forms better estimates at finer resolutions, resulting in smaller variances for the high-pass subbands of the finer resolution residuals. Thus, this high frequency information is coded more accurately than algorithms using coarse-to-fine ME.

As a result, flickering in textured regions, such as the flower bed and tree tops in *Flower Garden* and the textured background in *Table Tennis*, is significantly reduced using fine-to-coarse ME. Furthermore, much more detail is preserved, particularly in moving regions such as the roof top in *Flower Garden*, and the net as the camera zooms back in *Table Tennis*. At the coarsest resolution, the two algorithms give visually identical results, due mostly to the reduced frame resolution.

4. CONCLUSION

This paper proposes fine-to-coarse ME for multi-resolution video coding. While coarse-to-fine ME provides better estimates at the coarsest resolution, it is outperformed by fine-to-coarse ME at finer resolutions due to the inability of coarse-to-fine ME to accurately track motion at finer resolutions. At the finest resolution, fine-to-coarse ME provides both higher PSNRs and better visual quality. The performance of fine-to-coarse ME at coarser resolution can be improved in some cases by eliminating ME at coarser resolutions. In addition, fine-to-coarse ME provides more accurate and thus more compressible motion estimates.

5. REFERENCES

[1] ISO/IEC JTC1/SC29/WA11/602 13818-2 Committee Draft, November 1993.

[2] D. Taubman and A. Zakhor, "Multirate 3-D Subband Coding of Video," *IEEE Trans. Image Processing*, Vol. 3, No. 5, pp. 572-588, Sept. 1994.

[3] J.-R. Ohm, "Advanced Packet-Video Coding Based on Layered VQ and SBC Techniques," *IEEE Trans. Circuits and Systems for Video Technology*, Vol. 3, No. 3, pp. 208-221, June 1993.

[4] C. Podilchuk, N. Jayant, N. Farvardin, "Three dimensional subband coding of video," *IEEE Trans. on Image Processing*, Vol. 4, No. 2, pp. 125-139, Feb. 1995.

[5] T. Naveen and J. W. Woods, "Motion Compensated Multiresolution Transmission of High Definition Video," *IEEE Trans. Circuits and Systems for Video Technology*, Vol. 4, No. 1, pp. 29-41, February 1994.

[6] T. Naveen and J. W. Wood, "Rate Constrained Multiresolution Transmission of Video," *IEEE Trans. Circuits and Systems for Video Technology*, Vol. 5, No. 3, pp. 193-206, June 1995.

[7] J. M. Shapiro, "Embedded Image Coding Using Zerotrees of Wavelet Coefficients," *IEEE Trans. Signal Processing*, Vol. 41, No. 12, Dec. 1993