THE UNIVERSITY OF
WARWICK

**Original citation:**
Yu, Andy C., Park, Heechan and Martin, Graham R. (2006) Fast mesh-based motion estimation employing an embedded block model. In: IEEE International Symposium on Circuits and Systems, Kos Isl, Greece, 21-24 May 2006. Published in: 2006 IEEE International Symposium on Circuits and Systems Proceedings, Volume 1-11 pp. 4703-4706.

**Permanent WRAP url:**
http://wrap.warwick.ac.uk/31042

warwick**publications**wrap
highlight your research

**http://wrap.warwick.ac.uk**

# FAST MESH-BASED MOTION ESTIMATION EMPLOYING AN EMBEDDED BLOCK MODEL

*Andy C. Yu, Heechan Park, and Graham R. Martin*

Department of Computer Science
University of Warwick, Coventry CV4 7AL, United Kingdom
email: {andycyu, heechan, grm}@dcs.warwick.ac.uk

## ABSTRACT

A fast algorithm for mesh-based motion estimation employing uniform triangular patches is proposed. The technique utilises an embedded block model to estimate the motion of the mesh grid points. Without the need for time-consuming evaluation, the algorithm reduces the number of search iterations according to the inherent motion. A block-wise coding approach is taken for the motion information, permitting any picture degradation caused by the fast algorithm to be successfully compensated by the residue coding. Simulations on three classes of test sequence show that the proposed algorithm results in a better PSNR-rate performance than the hexagonal matching algorithm. Moreover, a reduction of up to 91% in mesh iterations is obtained.

## 1. INTRODUCTION

The use of mesh-based triangular patches provides an alternative approach to the estimation of motion. The procedure is to divide the current frame into a number of triangular patches, and to find the best matching corresponding patch in the reference frame when deformed by affine transformation. The mean absolute difference is commonly used as the matching criterion, and the motion is defined by the change in position of the corresponding vertices [1][2]. Compared to the conventional block-matching algorithm (BMA), mesh-based motion models provide more visually acceptable results in the predicted frame. This is credited to the fact that connectivity of the triangular patches is maintained, compared with a disjointed collection of blocks in the BMA. Furthermore, motion is estimated more accurately due to the utilisation of an affine transformation, which supports various spatial deformations such as translation, rotation and zoom [3]. The deformed triangular patches are described by the motion displacement of the grid points (the points shared by the vertices of the triangles) for motion coding purposes.

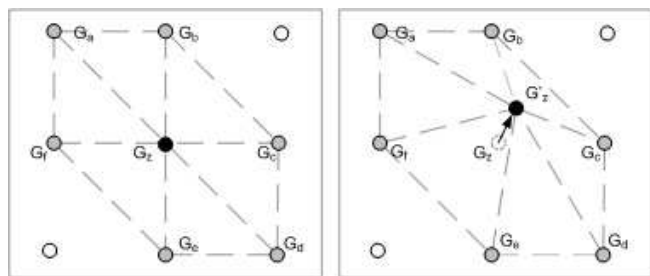In mesh-based motion estimation, the computation of the motion vector for a grid point is affected by its neigh-



**Fig. 1**. Search scheme for the hexagon matching algorithm: (1) Fix the position of the six vertices, $G_a$ to $G_f$; (2) Search for the best position of $G_z$.

bours. The interdependence requires a costly computation, performed in a recursive manner. A scheme based on a hexagonal matching algorithm was proposed by [4] to reduce the number of computational iterations. It basically relies on a hexagon formed by seven grid points as shown in Fig. 1. By fixing the position of the six peripheral vertices, the central grid point, $G_z$, is allowed to acquire an optimal motion displacement within the hexagon. Estimation of the motion for the other vertices forming the hexagon is performed in a similar manner. Subsequently, a refinement process is activated for the central grid point if the displacement of any grid point in the hexagon has been modified. The iterative refinement process is applied to all grid points until convergence to local or global minima.

The hexagon matching algorithm provides a successful approach to mesh-based motion estimation. However, it is computationally expensive as the recursive refinement process is unavoidable. The motivation of this paper is to develop a fast algorithm that is compatible with the hexagon-based scheme. In the next section, an algorithm incorporating an embedded block model is proposed to achieve a reduction in computation. In section 3, a block-wise approach to motion coding is presented, to improve the PSNR-rate performance of the encoded test sequences. Simulation results using a number of test sequences are included in section 4. Finally, conclusions are drawn.

## 2. MESH-BASED MOTION ESTIMATION INCORPORATING AN EMBEDDED BLOCK MODEL

The popularity of block-based motion compensation is evidenced by the MPEG-x and H.26x video coding standards. The reasons are that (a) a significant proportion of the motion trajectories found in natural video can be described with a rigid translational motion model; (b) fewer bits are required to describe simple translational motion; (c) implementation is relatively straightforward and amenable to hardware solutions. However, in spite of its relatively low computational requirement, the block-matching algorithm can provide visually unacceptable results when motion trajectories exhibit non-translational motion. The difficulties stem from the fact that the connectivity of vertices of adjacent blocks is not preserved [5].

In contrast, mesh-based motion models deform interconnected triangular patches by computing the displacement of the grid points. These have been shown to be a good alternative to block-based models when coping with non-rigid translational motion. As mentioned above, one method of evaluating motion displacement is to use the hexagonal matching algorithm which recursively computes each grid point. We propose an embedded block model to reduce the computational requirement.

It is observed in mesh-based models that a significant proportion of the grid points do not change from frame to frame. This is normally attributed to a stationary intensity field that exists in consecutive frames. This is particularly prevalent in scenes comprising a static background. It is natural to ask whether it is possible to detect the location of these areas and to know in advance that estimation of new grid point locations is not required. Richardson and Zhao [6] advocate an error metric based on mean absolute difference (MAD) to detect skipped macroblocks (macroblocks that are directly copied from the previous frame at the corresponding location). The MAD evaluation for a block-based model is defined as:

$$MAD_{B(i,j)} = \frac{1}{M \times N} \sum_{M} \sum_{N} |B(i,j,t) - B(i,j,t-1)|$$
(1)

where M×N is the size of the block. $B(i,j,t)$ and $B(i,j,t-1)$ represent corresponding blocks at location of $(i,j)$ in frames $t$ and $t-1$ respectively.

The question arises whether (1) can be applied to a mesh-based model. In a block-based model the vertices of a block are the four adjacent grid points as shown in Fig.2(a), and the individual grid points have little influence on the MAD evaluation for a block. However in a mesh-based scheme, the displacement of each grid point is dependent on its adjacent neighbours. This is particularly true at the boundaries of a static intensity field. Thus, a modification to (1) is
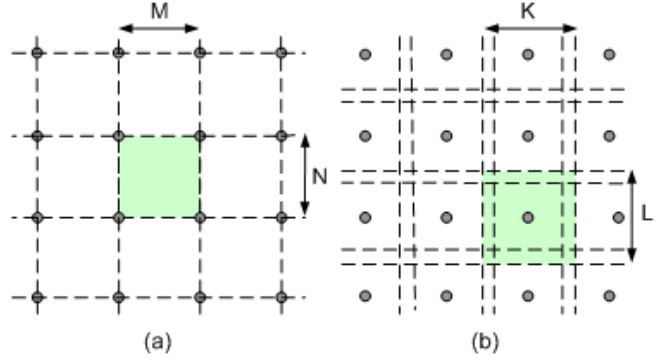


**Fig. 2**. (a) Conventional representation of a block (size M×N) containing four grid points; (b)Embedded blocks (of size K×L) centred at each grid point.

necessary. A MAD evaluation incorporating an embedded block model can solve the above problem. In Fig.2(b), an embedded block is formed at the centre of each individual grid point, and a MAD is calculated from the effective area within the embedded block. Note that the size and geometry of the embedded blocks can affect the evaluation. In this paper, we discuss only rectangular embedded blocks which are allowed to overlap each other. The MAD evaluation for an embedded block of size K×L is then defined as

$$MAD_{\widetilde{B}(G(i,j))} = \frac{1}{K \times L} \sum_{K} \sum_{L} |\widetilde{B}(G(i,j),t) -$$

$$\widetilde{B}(G(i,j),t-1)|$$
(2)

where $\widetilde{B}(G(i,j),t)$ and $\widetilde{B}(G(i,j),t-1)$ are the embedded blocks centred at grid point $G(i,j)$ in frames t and t-1, respectively.

A decision process is then used to decide whether the current grid point is to be exempt from motion estimation. The process contains a threshold, $\Delta_{Th}$, to provide a boolean outcome:

$$Decision_{G(i,j)} = \begin{cases} true & \text{if } MAD_{\widetilde{B}(G(i,j))} \leq \Delta_{Th}, \\ false & \text{otherwise.} \end{cases}$$
(3)

A positive outcome in (3) results in the exemption of motion estimation for the current grid point, $G(i,j)$. Eventually, all the grid points are assigned a boolean value. In the decision making process, the algorithm ensures that motion is estimated for those grid points located at static intensity field boundaries.

## 3. BLOCK-WISE APPROACH FOR MOTION INFORMATION CODING

As mentioned, clusters of stationary grid points are observed when coding natural video. Intuitively, much of the redundancy in motion vector coding can be eliminated by group-
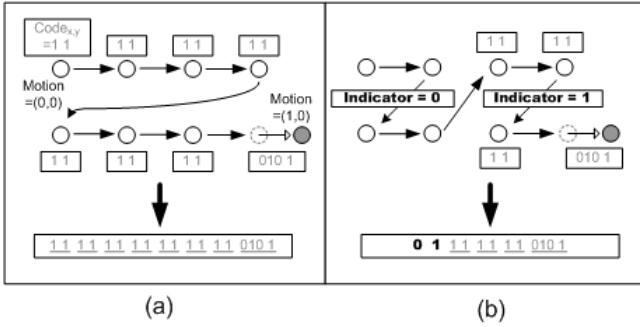
**Fig. 3**. The motion information coding process. (a) Conventional approach using raster scan; (b) the proposed approach incorporating indicator bits and a block-wise scan that results in fewer bits to encode the same amount of motion information.

ing grid points with zero displacement. A block-wise approach is proposed to achieve this, without specifying the location of the groups. Fig. 3 illustrates an example of two different schemes utilising an Exp-Golomb lookup table for motion information coding. In Fig. 3(a) motion information is encoded in the conventional raster manner without considering the correlation of adjacent motion displacements. In contrast, the proposed algorithm employs a block-wise scan considering groups of four grid points at a time, as shown in Fig. 3(b). An indicator is assigned to each group to indicate whether all four grid points have zero motion (indicator = '0'). For a group in which there is at least one grid point with non-zero motion the indicator is set to '1' and is followed by the motion vector difference (MVD) code for each member of the group. The proposed motion coding algorithms for encoder and decoder are summarised as follows:

Encoder:

1. *Consider a group of four grid points and check for consistency of zero motion displacement.*
2. *If motion is consistently zero, assign indicator bit '0' and proceed to the next group.*
3. *Otherwise, assign indicator bit '1' and code individual members. Proceed to the next group.*

Decoder:

1. *Examine the indicator bit (the first bit of the bitstream)*
2. *If the indicator bit is '0', assign all members of the group as having zero motion displacement. Return to 1.*
3. *Otherwise, decode the next four codewords for individual members of the group, and return to 1.*

Table 1 lists simulation results of employing the conventional and proposed approaches for motion information coding. It is observed that reductions of up to 78% are achieved.

| Sequences | Exp-Golomb | Proposed | Diff. |
|---|---|---|---|
| Carphone | 343.3 | 305.2 | **11.1%** |
| Miss A | 299.2 | 49.0 | **78.6%** |
| Silent | 227.8 | 86.5 | **62.0%** |
| Table Tennis | 276.9 | 186.6 | **32.6%** |

**Table 1**. Reduction in motion coding (average bits per frame) achieved by the proposed block-wise approach

| Sequences | HMA | Proposed | Diff. |
|---|---|---|---|
| Carphone | 32.44 | 32.41 | **-0.03dB** |
| Miss A | 42.55 | 42.53 | **-0.02dB** |
| Silent | 37.31 | 37.31 | **0.00dB** |
| Suzie | 38.49 | 38.49 | **0.00dB** |
| Table Tennis | 30.03 | 30.03 | **0.00dB** |

**Table 2**. Average PSNR performance of the hexagon algorithm and the proposed algorithm prior to residue coding with JPEG2000

## 4. SIMULATION RESULTS

This section discusses the performance of the proposed algorithm in a complete codec. Results are presented as improvements over the hexagon matching algorithm (HMA) employing $16 \times 16$ triangular meshes. The selected video sequences were of QCIF resolution ($176 \times 144$) and 30 frames of each sequence were processed. Except for the first frame, all the frames are encoded with mesh-based coding. The search range of the motion estimation is set to $\pm 8$ pixels. Finally, the motion information was encoded using the Exp-Golomb method and the residue data using JPEG2000 at fixed bit rates.

Table 2 shows the average picture degradation for the two motion estimation algorithms. The PSNR measurements are made prior to residue coding with JPEG2000. Significantly, the sequences employing the proposed algorithm achieve almost the same PSNR values as those by the hexagon matching algorithm. A maximum of 0.03dB in picture degradation is observed in the Carphone sequence (Class C) which demands a detailed motion search.

Table 3 shows the reduction in number of mesh-based grid point iterations. Generally, the reduction is dependent on the amount of motion within the sequence. For the class C sequences, Carphone and Table Tennis, an average speedup of 30% is obtained, and a speedup of over 40 % is achieved for the other classes.

Fig.4 illustrates the PSNR v. bit-rate performance for the three different classes of test sequences. The results are obtained after residue coding with JPEG2000 at fixed bit rates. For Class A and Class B sequences, the proposed algorithm benefits from PSNR gains at low bit rates. This can be attributed to the bit reduction obtained from the proposed block-wise approach. A reduction in motion infor-

| Sequences | HMA | Proposed | Speed up |
|---|---|---|---|
| Carphone | 263.2 | 185.7 | **29.4%** |
| Miss A | 195.8 | 17.4 | **91.1%** |
| Silent | 141.3 | 42.2 | **70.1%** |
| Suzie | 149.6 | 84.5 | **43.5%** |
| Table Tennis | 186.9 | 124.6 | **33.3%** |

**Table 3**. Reduction in mesh grid point iterations over the hexagon matching algorithm

mation coding allows more bits to be employed for residue coding. However Class C sequences do not reflect the same advantage as the bit reduction is less significant.

## 5. CONCLUSION

In this paper, we present a fast algorithm based on an embedded block model for mesh-based motion estimation. It is decided whether grid points should undergo further motion searches depending on a MAD evaluation of the embedded blocks. Based on the similar concept of assessing whether motion is present, a block-wise approach for motion information coding is proposed. Simulation results show that the proposed algorithm achieves a better PSNR-rate performance than the hexagon matching algorithm. Furthermore, a reduction of up to 91% in the number of mesh grid-point iterations is obtained.

## 6. REFERENCES

[1] A. Nosratinia, "New kernels for fast mesh-based motion estimation," *IEEE trans. on Circuit and System for Video Technology,*, vol. 11, no. 1, pp. 40–51, Jan. 2001.

[2] H. Park, A. Yu, and G. Martin, "Progressive mesh-based motion estimation using partial refinement," in *Proc. of International Workshop on Very Low Bit-rate Video (VLBV) 2005*, Sep. 2005, p. 4 pp.

[3] Y. Altunbasak and M. Tekalp, "A hybrid video codec with block-based and mesh-based motion compensation modes," *Special Issue of Int. Journal of Imaging System and Techology,*, vol. 9, no. 4, pp. 248–256, Aug. 1998.

[4] Y. Nakaya and H. Harashima, "Motion compensation based on spatial transformations," *IEEE trans. on Circuit and System for Video Technology,*, vol. 4, no. 3, pp. 339–367, Jun. 1994.

[5] N. Božinović, J. Konrad, T. André, M. Antonini, and M. Barlaud, "Motion-compensated lifted wavelet video coding: toward optimal motion/transform configuration," in *Proc. of European Signal Process Conference (EUSIPCO) 2004*, Sep. 2004, pp. 1975–1978.

[6] I. Richardson and Y. Zhao, "Video encoder complexity reduction by estimating skip mode distortion," in *Proc. of International Conference on Image Processing (ICIP) 2005*, Sep. 2004, pp. 103–106.
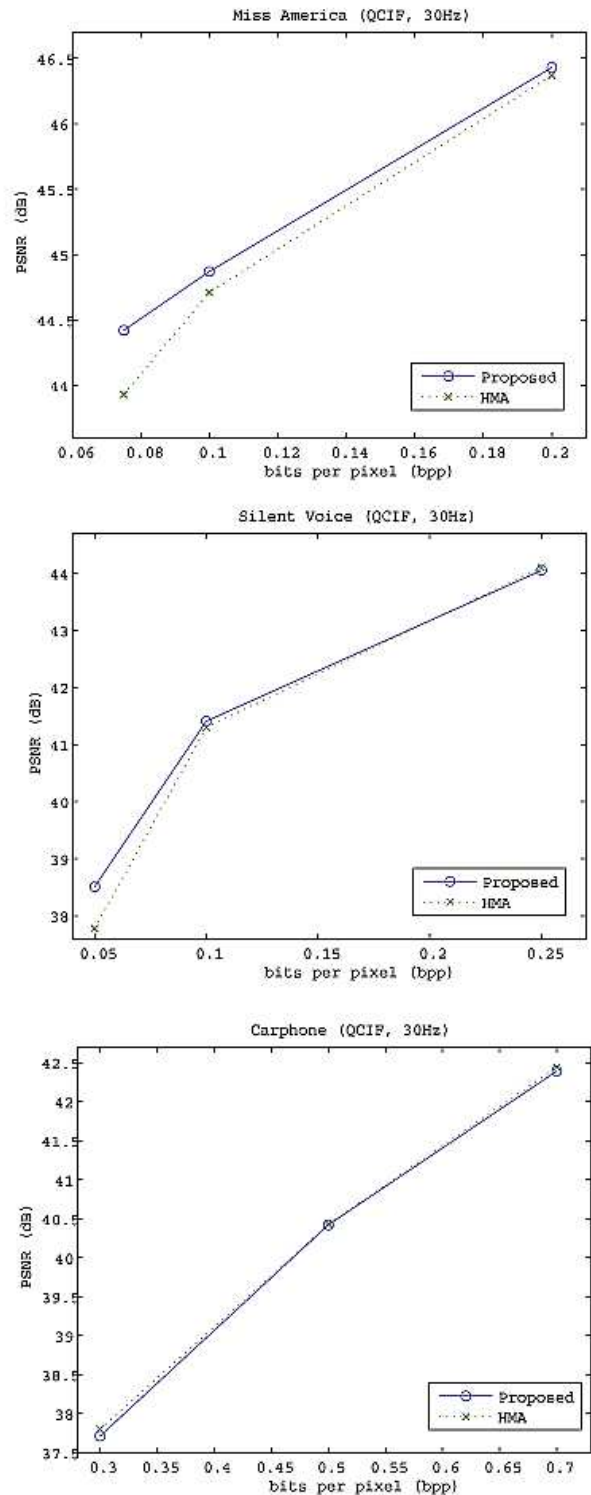
**Fig. 4**. The PSNR v. bit-rate diagrams (top to bottom) for Miss America (Class A), Silent Voice (Class B) and Car Phone (Class C) employing the proposed algorithm (solid line) and hexagon matching algorithm (dotted line)