Mesh-Based Video Coding For Low Bit-rate Communications

P. Kocharoen, *Student Member*, IEEE, K. M. Ahmed, R.M.A.P. Rajatheva,
and W.A.C. Fernando, *Member*, IEEE

Abstract — *In this paper, a new method for low bit-rate content-adaptive mesh-based video coding is proposed. Intra-frame coding of this method employs feature map extraction for node distribution at specific threshold levels to achieve higher density placement of initial nodes for regions that contain high frequency features and conversely sparse placement of initial nodes for smooth regions. Insignificant nodes are largely removed using a subsequent node elimination scheme. The Hilbert scan is then applied before quantization and entropy coding to reduce amount of transmitted information. For moving images, both node position and color parameters of only a subset of nodes may change from frame to frame. It is sufficient to transmit only these changed parameters. The proposed method is well-suited for video coding at very low bit rates, as processing results demonstrate that it provides good subjective and objective image quality at a lower number of required bits[1].*

Index Terms — Video Coding, Mesh Generation, Multimedia Communications.

I. INTRODUCTION

Today, there is an ever increasing demand for multimedia consumer devices for mobile and home use. Due to the rapid advances in video products such as digital cameras, camcorders, storage devices (DVDs) and the explosion of the internet, the digital video "for every one" is now becoming a reality. Many consumers want to have digital video where ever they go and where ever they are. The main business motivation of broadcasting and telecommunications organizations is to achieve the highest possible user-perceived added value to terminals with the lowest possible resources, particularly the bandwidth. Therefore, video compression has become an important area of research and enabled a variety of consumer applications including video storage on DVD and video CD's, digital video broadcast over cable, satellite and terrestrial digital television and video conferencing and videophone.

Over the last ten years, mesh-based coding schemes have achieved excellent results in image coding [1]-[9]. Mesh-based coding of image deals with dividing the image domain into a total of **M** non-overlapping mesh elements and then using a two-dimensional interpolation to reconstruct the intensity over each element. Mesh generation is a significant component of mesh-based coding. Available methods such as [5] provide an accurate node placement mechanism to represent high quality reconstructed images, but the required number of nodes is still high. If this number is reduced, the quality deteriorates rapidly with these techniques.

For moving images, the waveform-based coding reduces the redundant information between inter-frames by using motion estimation and compensation. Many motion estimation techniques are introduced. The commonly used motion estimation technique in all standard video codecs is the block matching algorithm (BMA) where an MxN pixels sized block of the reference frame is searched in the current frame by minimizing the mean square error (MSE) over all positions within the search range [12]. Tekalp [1], [10] introduced a mesh-based motion model which is used the polygonal patches, triangles or quadrangles, for inter-frame coding instead of block matching model. Mesh-based modeling can deform the polygon patches by the movements of the nodes from the current frame into the reference frame and the texture inside each patch is warped using the affine transform.

On the other hand, a picture in the mesh-based interpolative image coding is represented by a number of nodes with parameters of position and color [2]-[9]. The basic principle of mesh-based interpolative coding is different from the well-known waveform-based coding techniques, i.e. DCT and wavelet transform which form the most image and video coding standards [11]. For moving pictures, both node position and color parameters of only a subset of nodes may change from frame to frame. Therefore the coder transmits only these changes to the decoder. At the receiver, the received node movement and color modification parameters are employed then the images can be reconstructed. There are many publications in mesh-based image coding, e.g. [2]-[9] but only a few present the method of mesh-based inter-frame video coding. Baum presents an optimization method for inter-frame coding which is called node tracking [2]. The node tracking method optimizes the mesh structure of consequent frames by adding, removing and insertion of nodes in the reference frame. The node tracking method can be used in the sequences with moderate object movement but if new objects occur or if there is any foreground change, this method could face difficulties.

In this paper, we propose a low bit-rate mesh-based video coding method. This method reduces the required number of nodes in intra-frame coding while maintaining image quality. In addition, the proposed inter-frame coding method is able to deal with the occurrence of new objects and also with major foreground scene changes. The paper is organized as follows. The proposed methods and the system are described in Section II, III. Processing results of intra- and inter-frame are presented in Section IV.

[1] P. Kocharoen, K. M. Ahmed and R.M.A.P. Rajatheva are with the Telecommunications Program, Asian Institute of Technology, Thailand. (e-mail: preecha.kocharoen@ait.ac.th).

W.A.C. Fernando is with the Department of Electronic and Computer Engineering, Brunel University, UK. (e-mail: anil.fernando@brunel.ac.uk).

Contributed Paper
Manuscript received February 21, 2006

II. Intra-frame Coding

We use the same image function and notations as given in [6]. Let **D** denote the non-overlapping Delaunay triangles divided into **M** elements $\mathbf{D}_m$, m=1,2,3,…,M over the image area. The represented image over each $\mathbf{D}_m$ is approximated by two-dimensional linear interpolation function $\boldsymbol{T}$ over $\mathbf{D}_m$. The image function over each $\mathbf{D}_m$ is defined as follows:

$$g_m(x) = \boldsymbol{T}(f(x_n), \mathbf{D}_m) \tag{1}$$

where $f(x_n)$ is location and intensity of x_n.
x_n are vertices n=1,2,3 of $\mathbf{D}_m$.
$g_m(x)$ is the interpolated intensity value of the missing pixels within the triangle $\mathbf{D}_m$.

The reconstructed image is :

$$\hat{f}(x) = \sum_{m=1}^{M} g_m(x) \tag{2}$$

In natural pictures, many areas show a flat color or smooth texture known as a low-frequency feature [6]. The intensity of pixels in such areas is similar, and thus only a few sparsely placed nodes are sufficient to construct the mesh elements in these regions. On the other hand, in areas containing such high-frequency features as object edges, a dense distribution of nodes is required. The proposed method also applies a Laplacian filter [13] to extract a feature map from the image space. A novel node elimination technique [4] is then applied to remove the nodes without significantly affecting image quality.

The proposed method is described briefly as follows. The initial nodes are located from the crowded sampling points by intensity feature filtering method which is based on theoretical error bound of a mesh representation [5]. Insignificant nodes are then removed using a node elimination technique, which is described in detail in section II-B. As a result of node elimination, the number of nodes is drastically reduced and there remain only a few samples to be coded. Next, Hilbert scan, node color quantization and Huffman encoding are applied to reduce the amount of transmitted information [14]. After which, all nodes are triangulated in order to be connected by the meshes using a number of non-overlapping triangles. The reproduced image is constructed from the interpolated intensity of the pixels within a triangle by a linear combination of the three vertices of the triangle [15]. In order to achieve a unique triangular mesh without transmitting side information about the mesh topology, the Delaunay mesh is used [16], [17].

A. Image Intensity Feature-Map Extraction

The second derivative of the intensity distribution of pixels provides a zero value in the middle of a double, bi-directional peak for either type of transition - light to dark or dark to light. These can be used to identify the position of the edge [13], [18]. The usual way of looking for intensity discontinuities is to run a special mask or window over the image. The Laplacian operator is a second-order derivative that can also be used for detecting intensity changes. This operator is defined as follows:

$$\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} \tag{3}$$

The mask of the two-dimensional Laplacian operator at each pixel *(i,j)* is computed as follows:

$$\begin{aligned} \nabla^2 fxx(i,j) &= \tfrac{1}{2} f(i+1,j) - \tfrac{3}{2} f(i,j) + \tfrac{1}{2} f(i-1,j), \\ \nabla^2 fxy(i,j) &= \tfrac{1}{4} \begin{Bmatrix} f(i+1,j+1) + f(i+1,j-1) \\ + f(i-1,j+1) + f(i-1,j-1) \end{Bmatrix}, \\ \nabla^2 fyy(i,j) &= \tfrac{1}{2} f(i,j+1) - \tfrac{3}{2} f(i,j) + \tfrac{1}{2} f(i,j-1). \end{aligned} \tag{4}$$

In order to extract the intensity feature from the image, we adopted the Laplacian filter in [5], [13]. For some specific image, such as a high contrast image or a bi-level intensity image, a single threshold level may not give a satisfying result. The bi-level intensity image gives ambiguous significant nodes when the intensity feature filtering with a single threshold is applied. This leads to ineffective node removal in the node elimination procedure. The method could remove incorrect nodes and resulting in low quality reconstructed image. Therefore, a more complex threshold is introduced where the intensity feature-map $\Phi_\beta(x_i, y_i)$can be expressed as:

$$\Phi_\alpha(x_i, y_i) = \frac{\left|\nabla^2 \mathbf{I}(x_i,y_i)\right|^\gamma}{\max\left|\nabla^2 \mathbf{I}(x_i,y_i)\right|} > \tau_0 \tag{5}$$

$$\Phi_\beta(x_i, y_i) = \tau_1 < \Phi_\alpha(x_i, y_i) < \tau_2 \tag{6}$$

where
∇ = the Laplacian operator,
$\mathbf{I}(x_i,y_i)$ = the image intensity at (x_i,y_i),
γ = the parameter for adjust the sensitivity of the image feature-map. Normally, $\gamma = 1$,
τ_0, τ_1, τ_2 = the predefined threshold level, normally

$$\tau_0 = \frac{3}{2^8 - 1}, \tau_1 = \frac{2^8/2 - 2}{2^8 - 1}, \tau_1 = \frac{2^8/2 + 2}{2^8 - 1}.$$

B. Node Elimination Procedure

Node elimination is the major component of adaptive mesh generation [4]**.** The concept of a node elimination procedure is simple and straightforward. First, we calculate the mean square error (MSE) between the values of intensity within the convex hull of the original image and the reconstructed image after removal of a node. If the MSE exceeds a predefined threshold, the removed node is recovered; otherwise, the node remains eliminated. Every node, from first to last, is examined.

The initial node locations $\mathbf{X_N}$ are identified using the result of (6). Let $\mathbf{x_i} = (x_i, y_i) \in \mathbf{X_N}$ where the set of node points $\mathbf{X_N}=[\mathrm{x}_1, \mathrm{x}_2,...,\mathrm{x}_n]$. For any $\mathbf{x_i} \in \mathbf{X_N}$, $\mathbf{X_N}\backslash \mathrm{x_i}$ indicates removing node $\mathbf{x_i}$ from the total set of nodes $\mathbf{X_N}$. After that the Delaunay triangles have to be 'locally' re-triangulated or updated. The node elimination procedure can be represented as follows:

$$\mathbf{X_{N\text{-}e}} = \mathbf{X}_{\tilde{\mathbf{N}}} \setminus \boldsymbol{x_i}\Big|_{\bar{\eta}^2\left(\mathbf{D_m},\tilde{\mathbf{D}}_\mathbf{m}\right)<\mu} ; i = 1,2,\ldots,\mathbf{N} \tag{7}$$

where

Delaunay triangles $\mathbf{D_m}$=$\mathbf{D}(\mathbf{X})$,
the eliminated $\boldsymbol{x_i}$ triangles = $\mathbf{D}(\mathbf{X}\backslash \boldsymbol{x_i})$,
μ = the pre-defined threshold level,
e = number of removed nodes.

The reconstructed image can be represented using (1) as:

$$\tilde{g}_m(x) = \boldsymbol{T}\left(f(\boldsymbol{x_n}), \tilde{\mathbf{D}}_{\boldsymbol{m}}\right) \tag{8}$$

The difference between the reconstructed and the original image is measured using the mean square error $\left(\mathrm{MSE} : \bar{\eta}^2\left(\mathbf{D}_{\boldsymbol{m}}, \tilde{\mathbf{D}}_{\boldsymbol{m}}\right)\right)$. The measurement is formulated as:

$$\bar{\eta}^2(\mathbf{D}_{\boldsymbol{m}}, \tilde{\mathbf{D}}_{\boldsymbol{m}}) = \frac{\sum_{x_i \in \mathbf{X}} \left|\tilde{\mathbf{I}}(\mathbf{X} \setminus \mathbf{x_i}) - \mathbf{I}(\mathbf{X})\right|^2}{\mathbf{N}} \tag{9}$$

where

$\tilde{\mathbf{I}}(\mathbf{X} \setminus \mathbf{x_i})$ = the reconstructed image,
$\mathbf{I(X)}$ = the original image,
$\mathbf{N}$ = total number of pixel within the convex hull.

The peak signal to noise ratio (PSNR) can be calculated using the result of (8):

$$\mathrm{PSNR} = 10 \log_{10}\left(\frac{2^8 - 1}{\bar{\eta}^2(\mathbf{D}_{\boldsymbol{m}}, \tilde{\mathbf{D}}_{\boldsymbol{m}})}\right) \tag{10}$$

If either the number of existing nodes or the PSNR of the image is not satisfied by the first removal pass, more loops are executed with an adjustable error threshold in order to reach the desired number of nodes or the PSNR.

C. Delaunay Triangulation

Delaunay triangulation, **D**, exists and is unique for **V**, if **V** is a set of vertices (nodes) in a general position, i.e., no three nodes are on the same line and no extra node is on the circle which is constructed from the three vertices, for a two-dimensional set of nodes. Delaunay triangulation **D** of **V**, was first introduced by Delaunay [19]. Any circle in the plane is said to be empty if it contains no extra vertex of V in its interior (Vertices are permitted on the circle). A triangle is said to be Delaunay if and only if its circumcircle is empty. The Delaunay mesh is used in order to achieve a unique triangular mesh without transmitting side information about the mesh topology [17].

D. Interpolation

The values for color intensity, which are within the triangle obtained by the three nodes, are reconstructed by interpolation [20]. Interpolation is one of the most important aspects of the mesh-based coding as it is used both in the node elimination procedure and in the image reconstruction procedure. Let $\boldsymbol{T}(x_i, y_i)$ be an interpolated function and n_s be the number of node points where $\boldsymbol{s}_i = (\boldsymbol{p}_i, \boldsymbol{c}_i)$, $i \in \{1, 2, \ldots, n_s\}$, with positions of nodes $\boldsymbol{p}_i = (x_i, y_i)$ and nodes colors $\boldsymbol{c}_i$. The color intensity inside the triangular region can be calculated by the equations introduced in [2].

E. Hilbert Scanning

Hilbert curve scanning is used because the path scan order of this method starts from one local area to another local area until a two-dimensional space is filled up. This has an advantage if the neighboring nodes have similar positions and similar colors, as it can be performed effectively when differential coding is applied. An example with a description of Hilbert scanning can be found in [21].

F. Quantization and Entropy Coding

It was demonstrated in [22], that the color quantization of nodes is less critical than the quantization of the node path scan order. Small modifications in the node locations can have a significant effect on the mesh structure and this results in a degradation of image quality. Therefore, only the node colors are quantized in order to increase the coding efficiency. After quantization of the node colors, the differential encoding is applied followed by Huffman encoding in the final step to compact the information before transmission.

G. Computational Complexity

In this section, the computational complexity of the proposed node elimination procedure is analyzed. The computational complexity can be determined from several parts. The construction of the Delaunay triangle from initial nodes $\mathbf{X_N}$ results in a complexity of $O(\mathbf{X_N}\ log\ \mathbf{X_N})$. Node elimination requires updating only the local triangles, thus only local nodes $\mathbf{x}_{\mathrm{local}}$ are used for calculation where $\mathbf{x}_{\mathrm{local}} << \mathbf{X_N}$. Then the complexity of local retriangulation is $O(\mathbf{x}_{\mathrm{local}}\ log\ \mathbf{x}_{\mathrm{local}})$ where $O(\mathbf{x}_{\mathrm{local}}\ log\ \mathbf{x}_{\mathrm{local}}) << O(\mathbf{X_N}\ log\ \mathbf{X_N})$. The nodes are executed from the first to the last node, hence the complexity becomes $\mathbf{X_N}\ O(\mathbf{x}_{\mathrm{local}}\ log\ \mathbf{x}_{\mathrm{local}})$.

Altogether, this shows that the asymptotic complexity is at most:

$$\mathbf{X_N}\ O(\mathbf{x}_{\mathrm{local}}\ log\ \mathbf{x}_{\mathrm{local}}) + O(\mathbf{X_N}\ log\ \mathbf{X_N}) = O(\mathbf{X_N}\ log\ \mathbf{X_N}) \tag{11}$$

Finally, the node elimination loop is executed *R* times in order to reach the desired number of nodes or PSNR therefore the asymptotic complexity becomes at most:

$$R(O(\mathbf{X}_{\mathrm{N}}\ log\ \mathbf{X}_{\mathrm{N}})) = O(\mathbf{X}_{\mathrm{N}}\ log\ \mathbf{X}_{\mathrm{N}}) \tag{12}$$

III. Inter-frame Coding

A. Inter-frame Mesh Generation

Two methods of the inter-frame mesh generation, brute force and feature filtering method are described next. Brute force method is based on the node tracking method in [2] whereas the feature filtering method is based on the adaptive mesh generation in [4].

1) Brute Force Method

Here the node generation is carried out for inter-frames in two steps. First, motion estimation determines a coarse motion vector for each node of the reference frame, using a block matching algorithm [12], [23]. A 15x15 pixel sized block of the reference frame, with the node as the center, is searched in the current frame by minimizing the mean square error (MSE) over all positions within the search range. Applying these motion vectors to the node positions of the previous frame, the node generator creates an initial mesh. Starting from this mesh, the final node positions are determined by an iterative node movement procedure. The nodes are repeatedly moved by one pixel in any direction, at a time, to find the optimum position. The steps for these methods are shown next.

a) Block Matching Motion Estimation (BMA)

The principle of the BMA is to apply a translational motion model to subblocks of the image. For every block, the matching measure is based on the Displaced Frame Difference (DFD) [24]. The DFD expresses the difference between the luminance of the image at current frame (*t*) and the luminance of the image at reference frame (*t-dt*). The displacement vector in *x* and *y* direction are *dx* and *dy* respectively.

$$\mathrm{DFD}(x,y,dx,dy,t) = \mathbf{I}(x+dx,y+dy,t-dt) - \mathbf{I}(x,y,t) \tag{13}$$

where $\mathbf{I}(x+dx,y+dy,t-dt)$ and $\mathbf{I}(x,y,t)$ describe the reference and current image plane respectively.

The aim of the motion estimation is to determine the vector that minimizes the DFD. The criteria that is used in this context to compute the value of the DFD over a precise region is called the mean square error $\left(\mathrm{MSE} : \overline{\eta}^2(\mathrm{DFD})\right)$. Equation (8) can be used for calculation the MSE.

The implementation of block matching motion estimation to the mesh-based video coding is described below:

i. The image $\mathbf{I}(t-dt)$ is partitioned to a set of subblocks of size (*i* by *j*) with nodes as the center of the subblocks.

ii. All subblocks in the current image $\mathbf{I}(t)$ are moved to all possible locations and MSE within the blocks is calculated.

iii. The motion vectors corresponding to the positions with the lowest MSE are obtained.

b) Brute-force inter-frame mesh generation method

The procedure of brute-force inter-frame mesh generation method is described below:

i. Determine a coarse motion vector of each node using a 15x15 pixel sized block matching whose center is the position of the node.

ii. Applying the motion vectors to the node positions of the previous frame to generate initial nodes.

iii. Let $\rho_r = \rho_1, \rho_2, \ldots, \rho_R$ where ρ_r= iteration loop number

iv. For r=1,…,R

- Let $\boldsymbol{x_i} = (x_i,\ y_i) \in \mathbf{X}_{\mathrm{N}}$ and $\boldsymbol{x}_{\boldsymbol{i},non\text{-}zero\ vector} = (x_i,y_i)|_{\text{non-zero vector nodes}} \in \mathbf{X}_{\mathrm{N}}$ where set of node points $\mathbf{X}_{\mathrm{N}} = [\boldsymbol{x_1}, \boldsymbol{x_2}, \ldots, \boldsymbol{x_n}]$.
- Move $\boldsymbol{x}_{\boldsymbol{i},non\text{-}zero\ vector}$ by one pixel in any direction, at a time.
- Locally update triangle and check whether PSNR is equal a target PSNR. The condition is then satisfied and the procedure ends.

This method generates high correlation among new nodes and reference nodes and thus can achieve efficient entropy coding in the following step. On the other hand, the technique also needs several node movement iterations, which causes the method to be slow.

2) Feature Filtering Method

Feature filtering method obtains new node locations for inter frames in three steps. First, motion estimation determines coarse static nodes by using a block matching algorithm. Applying these results, i.e., motion vectors from the block matching algorithm to the node positions of the previous frame, initial coarse nodes of the current frame are created. The new nodes which have no motion are then classified to be static nodes. Second, a triangular mesh and image are constructed from new nodes and MSE of each local triangle within the image is calculated. The nodes of the triangles which have MSE above the predefined threshold are then removed from the static nodes list. Then triangular mesh from the static nodes is constructed and the sampling points that are located inside the static triangle areas are filtered out. The sampling points which are located outside the static triangle are then defined to be dynamic nodes. Finally, the node elimination method [4] is applied to the dynamic nodes to remove the insignificant sampling points.

Furthermore, this method locates new nodes from the crowded sampling points by intensity feature filtering method [14] which is based on theoretical error bound of a mesh representation [5]. Hence the unpredicted objects in the image can be handled automatically. The details of this procedure are given below.

a) Static and Dynamic nodes Registration

Let $\boldsymbol{x_i}$=(x_i,y_i) ∈ $\mathbf{X_N}$ where set of node points $\mathbf{X_N}$ = [$\boldsymbol{x_1}$, $\boldsymbol{x_2}$,…, $\boldsymbol{x_n}$] and motion vector $\boldsymbol{mv}$=(dx_i,dy_i). In addition, **D** denotes non-overlapping Delaunay triangles which are divided into **M** elements $\mathbf{D}_m$, m=1,2,3,…, **M** over image area. Two types of nodes registration are described next.

Static nodes registration:

The initial static nodes can be obtained using the following equation.

$$\mathbf{X}_{\text{n,init.static}} = \mathbf{X}_{\text{N}}\left(\boldsymbol{x_i}, mv\right)\Big|_{mv=0} \tag{14}$$

The initial static nodes are used to construct the triangular mesh and an initial image. The nodes within the triangles which have MSE above the predefined threshold are then removed from the static nodes list.

$$\mathbf{X}_{\text{n,over}} = \mathbf{X}_{\text{N}}\left(\boldsymbol{x_i}, \overline{\eta}^2(\mathbf{D}_{\boldsymbol{m}}, \widetilde{\mathbf{D}}_{\boldsymbol{m}})\right)\Big|_{\overline{\eta}^2(\mathbf{D}_{\boldsymbol{m}}, \widetilde{\mathbf{D}}_{\boldsymbol{m}}) > \mu_S} \tag{15}$$

$$\mathbf{X}_{\text{n,static}} = \mathbf{X}_{\text{n,init.static}} \setminus \mathbf{X}_{\text{n,over}} \tag{16}$$

where

μ_{s}= the predefined threshold, normally μ_{s}=30

Equation (9) is used to calculate MSE of each local triangle within the image.

Dynamic nodes registration:

Equation (6) is used to extract the feature map from the image and to locate initial sampling points. After that, triangular mesh from the static nodes using (2) is constructed and the sampling points that are located inside the static triangle areas are filtered out. Then, the sampling points which are located outside the static triangle are defined to be dynamic nodes.

$$\mathbf{X}_{\text{n,dynamic}} = \Phi_{\beta(x_i, y_i)}\Big|_{\Phi_{\beta(x_i, y_i)} \notin \text{triangle area of } \mathbf{X}_{\text{n,static}}} \tag{17}$$

b) Node Elimination

The node elimination method is finally applied to the dynamic nodes to remove the insignificant sampling points. ($\text{X}_{\text{n,dynamic}} \backslash x_i$) is described as removing node x_i from the total set of nodes $\text{X}_{\text{n,dynamic}}$ where the Delaunay triangles have to be 'locally' updated. The node elimination procedure can be represented as follows.

$$\mathbf{X}_{(\text{n,dynamic})\text{ - e}} = \mathbf{X}_{\text{n,dynamic}} \setminus \boldsymbol{x_i}\Big|_{\overline{\eta}^2\left(\mathbf{D_m}, \widetilde{\mathbf{D}}_{\mathbf{m}}\right) < \mu_d} \tag{18}$$

where

Delaunay triangules $\mathbf{D_m}$=**D**($\mathbf{X}_{\mathbf{n,}\text{dynamic}}$),

the eliminated $\boldsymbol{x_i}$ triangles = **D**($\mathbf{X}_{\mathbf{n,}\text{dynamic}}$ \\$\boldsymbol{x_i}$),

μ_d = the pre-defined threshold level, normally μ_d = 30,

i=1,2,..., max($\mathbf{X}_{\mathbf{n,}\text{dynamic}}$),

e = number of removed nodes.

This technique produces low correlation among new nodes and reference nodes and thus a higher number bits are required for entropy coding in the next step. On the other hand, the unpredicted objects in the image can be handled by using this method.

B. Mesh Based Inter-Frame Coding

Based on the results of the previous section, we introduce a technique that can benefit from the correlation of successive frames in the image sequences. In mesh based video coding, the difference of image frames is represented by the number of changed nodes with node position and color parameters. Only the changes in partial nodes from frame to frame are coded in order to benefit from the interframe coding. The characteristic of scene transition affects the method of coding the interframe. In low bit-rate video communication such as videoconference or video on mobile phone, the scene is mostly on human face and shoulders with only some motion. Therefore, the proposed method is limited to the scenes that have modulated object movements located in the center of the scene. We classify types of the limited condition scene change into four categories which are: 1) High DFD motion, 2) Low DFD motion, 3) Occurrence of new objects, and 4) Major foreground change. Different types of interframes are coded by different techniques with respect to rate and distortion.

1) High DFD motion

High DFD motion vector is coded by brute force method.

2) Low DFD motion

Low DFD motion vector is coded by modifying the brute force method. The block-matching algorithm is applied to all nodes in the reference frame in order to find the initial motion vectors. After that an iterative node movement procedure is applied on the nodes which contain non-zero motion vector until the optimum node positions are determined.

3) Occurrence of new objects

Occurrence of new objects is handled by inserting extra nodes. First, modified brute-force or brute-force method is applied. Then, the image is reconstructed with some areas containing high error values. These areas correspond to the occurrence of new objects. Extra nodes are then inserted to handle the new object areas.

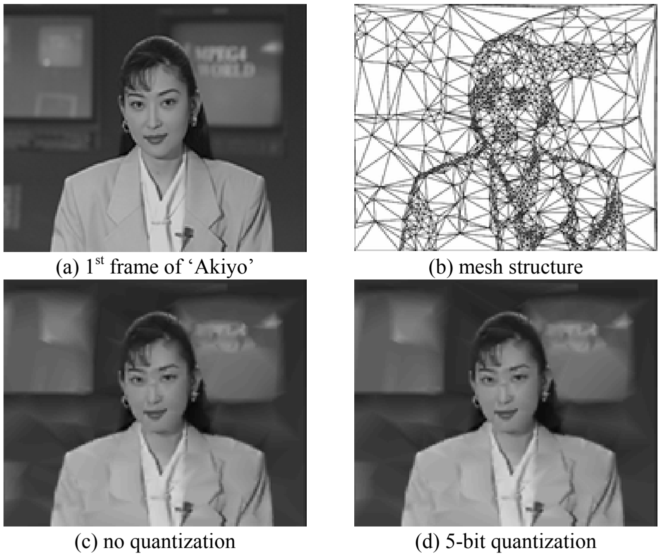
4) Major foreground change

Major foreground change is coded by using the intensity feature filtering method. The feature filtering method achieves new node locations for inter frames in three steps. First, motion estimation determines coarse static nodes by using a block-matching algorithm for generating the static nodes. Second, the dynamic nodes located within the high MSE triangles are identified. Finally, the node elimination method is applied to the dynamic nodes to remove the insignificant nodes.

IV. Results of Video Processing

A. Results of Intra-frame Mesh Generation

First frame of many QCIF video sequences was used as test images. The mesh generation method was applied to the first frames of many standard test QCIF video sequences –'Akiyo', 'Carphone', 'Claire', 'Foreman', 'Grandma', 'Miss America', 'Mother and Daughter', 'Salesman', and 'Suzie'. These sequences are in format YUV 4:2:0 in which Y, U, and V are processed separately. Most significant information is in the Luminance (Y) part thus the results focus only on this part. The first step in the proposed method is to create the image feature maps, which are obtained from the first frame of each sequence. The standard mask, which is used in this operation, was implemented in (4). Image feature map are extracted and the initial nodes are placed by screening the feature map with a specific threshold level (i.e. $\tau = 3$). Then the node elimination procedure is applied. Using three-passes of node elimination with the reference MSE set to 30, the remaining nodes are obtained (see Table I). These nodes are used to construct the image mesh structures. An example of a mesh structure is shown in Fig. 1(b). A two-dimensional linear interpolation is used to produce the reconstructed images. An example of the first frame of Akiyo reconstructed image is illustrated in Fig. 1(d) with PSNR=33.07dB.



(a) 1st frame of 'Akiyo'

(b) mesh structure

(c) no quantization

(d) 5-bit quantization

Fig. 1. The subjective results of the test image

Table I
The Numerical Results Of QCIF Test Images

Image	No. of nodes	No. of bits	PSNR(dB)
Akiyo	1330	12624	33.07
Carphone	2065	18776	32.72
Claire	809	8224	34.96
Foreman	2063	18792	32.64
Grandma	1711	15448	33.03
Miss America	552	5560	35.30
Mother	1234	11848	33.44
Salesman	3104	26424	31.58
Suzie	1266	12048	33.03

B. Comparison with the results from other mesh generation methods

The proposed method is compared with the results of three sub-optimum methods taken from [5] which are: 1) adaptive sampling by Yang's method; 2) adaptive sampling by Garcia's method; 3) a quadtree method. In addition, it is also compared to a mesh optimization by Baum's method [2]. The comparison is based on a 128x128 'Peppers' standard test image where the mesh generation methods are implemented on a 2.8GHz Pentium-4 computer. The numerical results are shown in Table II. The Yang's method performs best among the existing sub-optimum methods, which uses 4081 nodes to reconstruct the image with PSNR=28.94dB. However, the results as presented in Table II show that the proposed method is better than all existing sub-optimum schemes. Even though the numbers of nodes is half that of Yang's method, PSNR of the proposed method is 31.70 dB, which is 2.76dB more than the Yang's method. In addition, as the PSNR values of the reconstructed images are approximately the same, the proposed method performs well even when it uses only 1302 nodes.

Table II
The Numerical Results Comparison With Other Methods

Method	PSNR(dB)	No. of nodes	Processing time (s)
The proposed method	31.70	2005	78.29
	28.45	1302	187.46
Yang's	28.94	4081	< 1
Garcia's	26.88	4152	< 1
Quadtree	26.16	4037	10.50
Optimization	32.11	1302	4905

The mesh optimization method in [2] performs well but the processing time is very long because each node is moved, deleted and inserted in all directions in many iteration steps in order to obtain the optimum solution. The number of operations in [2] at each node point can be found as follows: move 8 times, add 8 times, and remove 1 times. The total number of operations for a single iteration is 17 times with around 10 such loops required. Therefore, the number of operations is at most 170N times, where N = total number of nodes.

The number of operations in our method at each node point is only removing 1 times and requiring 3 loops. Thus, the number of operations is at most 3N times.

The proposed method reduces the processing steps from 170N times to 3N times. Thus the processing time is expected to be much less than that in [2]. Simulation results in table I show that this method can reduce the processing time from more than an hour to only 3 minutes. On the other hand, the reconstructed image quality of the proposed method is worse than the optimum method in [2], but it still performs the best among all sub-optimum methods as shown in Table II.

The results show that the proposed mesh generation method outperforms the existing sub-optimum methods. It works well for all kinds of natural images, i.e., textures, smooth background, sharp edges, and smooth/sharp transitional regions. The proposed scheme is suitable for videophone applications, e.g. 'Claire', 'Miss America', and 'Mother', whose scenes contain the human face with a smooth background. In these cases the reconstructed images require a small number of nodes while preserving good PSNR values. However, for images which have finer details or a complicated background, the proposed scheme requires a higher number of nodes for the reconstruction of the images, e.g., 'Baboon' and 'Barbara'. Therefore, the mesh-based image coding is excellent for applications which have objects with smooth background.

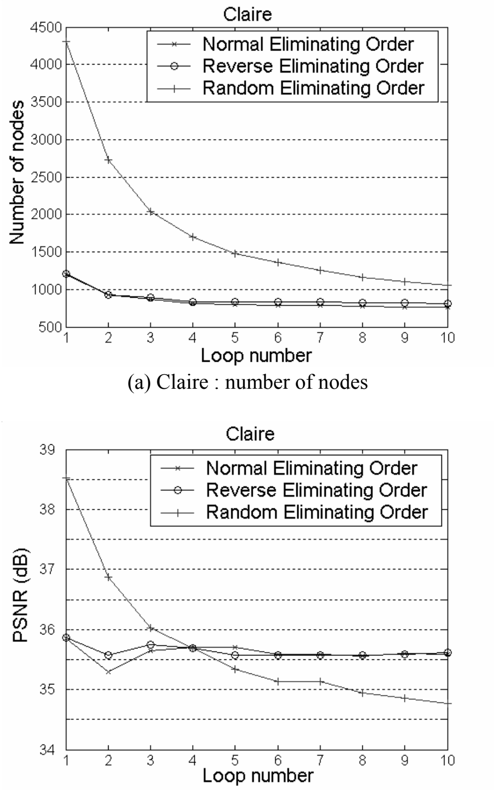
C. Results of Intra-frame Coding

A test of image compression was carried out for the first frame of many grayscale standard test QCIF video sequences. Some of the test images have the property of a smooth background, e.g. 'Claire', 'Mother', 'Miss America', and 'Suzie'. The others have more complex information in the background, e.g., 'Akiyo', 'Carphone', 'Foreman', 'Grandma', and 'Salesman'.

The first step of the proposed image compression method is to generate adaptive meshes which are obtained from the first frame of the test sequences. The remaining node positions are described in path scan order using Hilbert scan. Node colors are quantized. Then both node positions and colors are differentially encoded. The information is then entropy encoded using the Huffman code. The numerical results of encoding images are given in Table I. The encoding is based on 5-bit quantization so the quantized symbols are {0, 1,...,31}. In the image reconstruction, the received node positions and colors are used to produce the mesh structure. The image is then reconstructed from a two-dimensional linear interpolation.

D. The Effect of Node Eliminating Order

Node elimination is a local procedure. Once a node is eliminated, the following examinations take the previous removals into account. Therefore, the order of examination may affect the performance of the algorithm. Three types of node eliminating order; normal, reverse, and random, are tested. The normal node eliminating order is processed start from the first to last node whereas the reverse node elimination order is processed from the last to first node. In addition, a binary pseudonoise sequence (or maximum length sequence) [25] is used to generate the random node eliminating order. The results are plotted and shown in Fig. 2. The results show that the order of examinations affects the performance of the algorithm and the sequential eliminating order, normal and reverse, performs better than the random eliminating order.



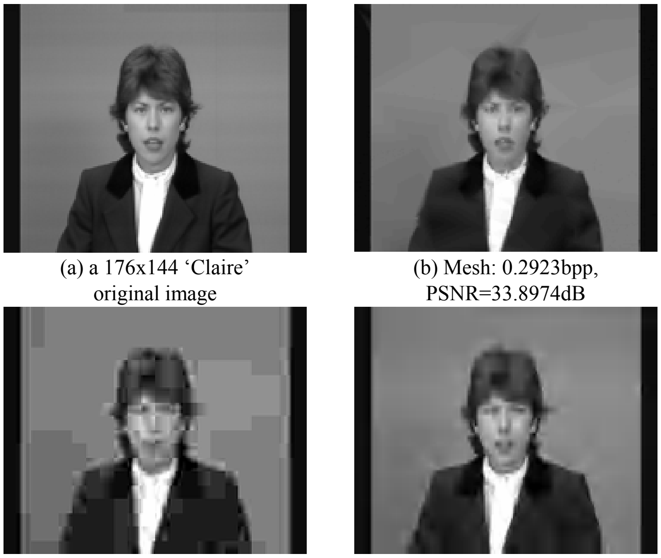
(a) Claire : number of nodes

(b) Claire : PSNR (dB)

Fig. 2. The effect of node eliminating order

E. Comparison with Other Image Compression Schemes

The proposed method is compared here with DCT and Wavelet compression scheme which are implemented by JPEG and JPEG2000 image compression standards respectively. The comparison is based on the first frame of the grayscale 'Claire' standard test sequence and implemented on a 2.8GHz Pentium-4 computer. The 'nconvert' [26] and 'jasper' [27] software are used to convert the test images into JPEG and JPEG2000 format correspondingly. The subjective results are shown in Fig.3 and the numerical results are presented in Fig.4. The subjective results show that the proposed method performs the best among the existing methods, representing sharp edges without ringing and blocking artifacts. The numerical results presented in Fig.4 show that the proposed method is slightly worse than the Wavelet schemes in terms of the higher number of bits used per picture. However, both the visual quality and the object quality of the proposed method are much better than those of the others without producing ringing and blocking artifacts.



(a) a 176x144 'Claire' original image

(b) Mesh: 0.2923bpp, PSNR=33.8974dB

(c)JPEG: 0.2958bpp, PSNR=28.6485dB

(d)JPEG2000: 0.2914bpp, PSNR=33.1224dB

Fig. 3. The subjective results comparison of the test images with other methods

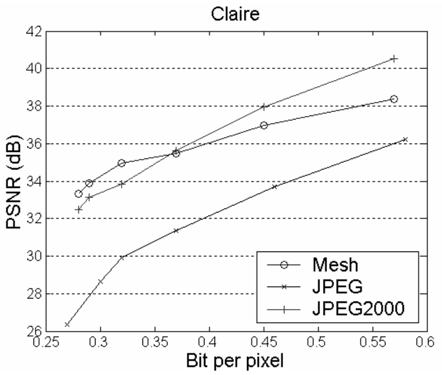


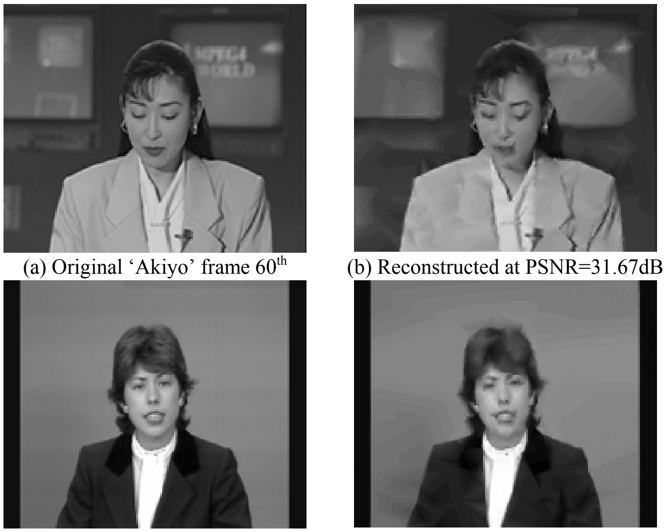
Fig. 4. The numerical results comparison with other image compression standards

F. Results of Inter-Frame Processing

This section implements a mesh based interframe coding method to compress video sequences. Standard QCIF sequences 'Akiyo', 'Claire', 'Grandma', 'Miss America', and 'Mother' are used as test sequences. The test sequences are coded at 10 frames per second, i.e. each third frame of the original sequence is processed and only the Luminance parts are shown. The interframe coder is run without rate control.

The initial task of the proposed method is to create a coarse motion vector for each node of the reference frame, using a block matching algorithm and to check the types of scene transition. A suitable method is selected for each type of the scene change with respect to rate and distortion. Only the node color is quantized before applying differential and Huffman coding, in order to increase the coding efficiency. The quantization is based on 5 bits so the quantized symbols are {0, 1,…, 31}. The proposed method compresses the node parameters which are locations, motion vectors and colors, with differential and Huffman coding. The merit factor is measured by their PSNR values, while the compression rate is kilo bits per second (kbps).

The sample frames of the 'Akiyo' and 'Claire' test sequence are displayed in Fig.5. The numerical results of the test sequences are shown in Table III. From Table III, the average PSNR of the 'Claire' reconstructed sequence is 32.96dB. It can be seen that, the number of average nodes changed per frame for interframe coding is only 50.70 nodes, which is 6.22% of the number of the nodes in the intra frame.



(a) Original 'Akiyo' frame 60th

(b) Reconstructed at PSNR=31.67dB

(c) Original 'Claire' frame 27th

(b) Reconstructed at PSNR=32.94dB

Fig. 5. The samples subjective results of the QCIF test sequences

TABLE III
THE NUMERICAL RESULTS OF QCIF TEST SEQUENCES

Image	PSNR (dB)	Bit rate (kbps)	Total nodes	Moved nodes	Avg. bits per node
Akiyo	31.52	7.61	1329	47.36	16.08
Claire	32.96	7.89	814	50.70	15.56
Grandma	31.50	5.49	1663	33.30	16.50
Miss Am.	32.43	9.63	565	65.52	14.70
Mother	31.17	7.10	1221	44.33	16.01

G. Comparison to the Existing Low Bit-rate Video Coding Standard

This section compares the proposed method with the existing low bit-rate video coding standard. The color QCIF-sized of Claire sequence is used as a test sequence. The test sequence is coded at 10 frames per second, i.e. Each third frame of the original sequence is processed. The interframe coders are run without rate control. The PSNR of each frames are calculated by average all PSNR in R, G, and B color space. Figure 6 shows results of the comparison. The average PSNR of the test sequence from frame 0 to 99 of mesh-based coding, H.263 and H.264 are 32.20, 31.77 and 34.49 dB at bit-rate 6.29 kbps, 6.30 kbps, and 6.36 kbps respectively.

For the comparison results, the performance of the proposed mesh-based coding is better than the H.263 video coding standard whereas it is worse than the H.264 video coding standard by about 2 dB. The Figure 7 shows the sample pictures of the test sequence encoded with Mesh-based coding, H.263, and H.264 respectively. Comparing the objective quality of the proposed mesh-based coding method with the H.263 and H.264 standard, it is observed that the proposed mesh-based method provides sharp edges without ringing and blocking artifacts whereas the H.263 provides both ringing and blocking artifacts.

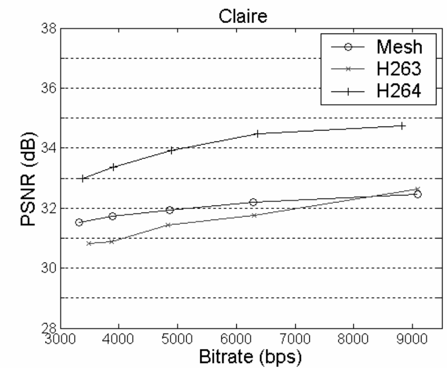
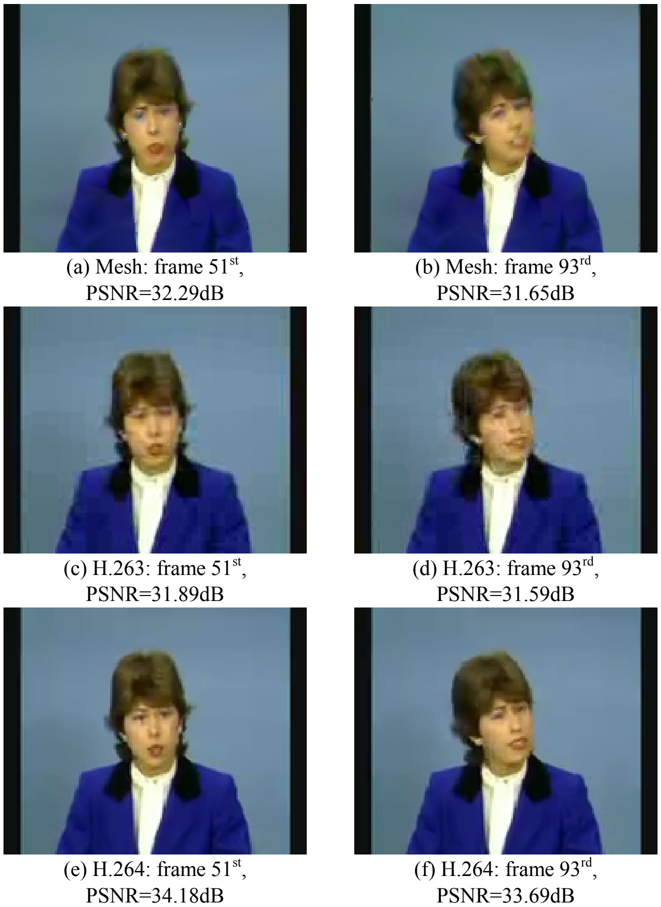


Fig. 6. The comparison between the proposed method and the standard video coding of the Claire test sequences



(a) Mesh: frame 51st, PSNR=32.29dB

(b) Mesh: frame 93rd, PSNR=31.65dB

(c) H.263: frame 51st, PSNR=31.89dB

(d) H.263: frame 93rd, PSNR=31.59dB

(e) H.264: frame 51st, PSNR=34.18dB

(f) H.264: frame 93rd, PSNR=33.69dB

Fig. 7. The samples subjective results of the Claire test sequence

V. Conclusions

Mesh based coding for image and video compression have been previously presented by many researchers, e.g. [1]-[9]. The proposed node elimination method with a feature map extraction for intra-frame coding aims to improve the mesh generation efficiency which constitute:

- Fast generating mesh structure.
- Adaptive mesh structure on the video scene contents.
- Constructing the triangular mesh with low number of nodes.

Three techniques are presented to code difference scene changes in inter-frame coding which are the brute-force, the modified brute-force and the feature filtering method. The brute-force and the modified brute-force inter frame coding performs well but the processing takes long time. On the other hand, the feature filtering method can reduce the processing steps for constructing interframes. It is also capable to handling the new objects in the scene or a significant change in foreground. However, the correlation between the reference nodes and the new nodes of the feature filtering method is lower than the brute-force method which leads to inefficient entropy encoding. Therefore, different types of interframes are coded by different techniques or they can be combined with respect to rate and distortion.

The processing results show that in the scenes containing objects with smooth background, the proposed method provides a good subjective and objective image quality. It also provides objective image quality comparable to the DCT and wavelet schemes in intra-frame coding. In addition, it provides much better visual quality than the other methods without producing ringing and blocking artifacts. However, in the images that have fine details or complicated background, the proposed method requires a higher number of nodes to reconstruct the images. Therefore the wavelet scheme is better than the mesh-based coding in this aspect. For moving images, the results show that the proposed method is better than the H.263 low bit-rate video coding whereas it is worse than that H.264 standard. The processing results show that the proposed method is suitable for low bit-rate video applications. For example, the results of the human face with smooth background, i.e. Claire, Miss America, and Mother preserve good PSNR values while requiring a small amount of bits, less than 10kbps.

The major drawback of the proposed method is that it requires iterations in the encoder side hence the increase in coding time is significant. Because of this reason, the proposed technique is not suitable for real-time applications. However at the decoder side, the picture can be reconstructed simply by constructing the mesh structure and interpolating their node colors. This could be applied to mobile video broadcasting for low-cost mobile handsets as the proposed method has a low computation complexity at the decoder which leads to low battery consumption.

References

[1] A.M. Tekalp, P. Van Beek, C. Toklu and G. Unsel, "Two-Dimensional Mesh-Based Visual-Object Representation for Interactive Synthetic/Natural Digital Video," *Proceedings of the IEEE*, Vol.86, No.6, pp. 1029-1051, June 1998.

[2] E. Baum and J. Speidel, "Novel Video Coding Scheme Using Adaptive Mesh-Based Interpolation And Node Tracking," *Proc. SPIE, Visual Communications and Image Processing 2000*, Vol. 4067, pp.200-208, 2000.

[3] C.L. Huang and C.Y. Hsu, "A New Motion Compensation Method For Image Sequence Coding Using Hierarchical Grid Interpolation," *IEEE Trans. Circuits Syst. Video Technol.*, Vol.4, No.1 pp.42-52, 1994.

[4] P. Kocharoen, K. M. Ahmed, R.M.A.P. Rajatheva and W.A.C. Fernando, "Adaptive Mesh Generation For Mesh-Based Image Coding Using Node Elimination Approach," *The IEEE International Conference on Communications, ICC05*, Vol.3, Seoul, Korea, pp.2052-2056, 16-20 May 2005.

[5] Y. Yang, J.G. Brankov and M.N. Wernick, "A Fast Approach for Accurate Content-Adaptive Mesh Generation," *IEEE Trans. on Image processing*, Vol.12, No.8, pp.866-881, August 2003.

[6] Y. Yang, J.G. Brankov and M.N. Wernick, "Tomographic Image Reconstruction Based on a Content-Adaptive Mesh Model," *IEEE Trans. on Medical Imaging*, Vol.23, No.2, pp.202-212, February 2004.

[7] L. Demaret, N. Dyn, A. Iske and M. Floater, "Adaptive Thinning with Application to Terrain Modelling and Image Compression," *Advances in Multiresolution for Geometric Modelling*, N.A. Dodgson, M.S. Floater, and M.A. Sabin(eds.), Spinger-Verlag, Heidelberg, pp.321-340, 2004.

[8] M.A. Garcia, B. Vintimilla and A. Sappa, "Efficient Approximation of Gray-Scale Images through Bounded Error Triangular Meshes," *IEEE Int. Conf. on Image Processing*, Kobe, Japan, pp.168-170, October 1999.

[9] M.A. Garcia and B. Vintimilla, "Acceleration of Filtering and Enhancement Operations Through Geometric Processing of Gray-Level Images," *IEEE Int. Conf. on Image Processing*, September 2000, Vancouver, Canada, Vol.1, pp.97-100.

[10] A. Altunbasak and A.M. Tekalp, "Occlusion-Adaptive, Content-Based Mesh Design and Forward Tracking," *IEEE Transactions on Image Processing*, Vol. 6, No. 9, pp.1270-1280, September 1997.

[11] H. Li, A. Lundmark and R. Forchheimer, "Image Sequence Coding At Very Low Bitrates: A Review," *IEEE Transactions on Image Processing*, Vol. 3, No. 5, pp. 589-609, September 1994.

[12] M. Ghanbari, *Video Coding: an Introduction to Standard Codecs*, The Institution of Electrical Engineers, London,pp.27-32, UK, 1999.

[13] C.G. Rafael, E.W. Richard, *Digital Image Processing*, Addison-Wesley Publishing Company, New Jersey, pp.147-214, 1993.

[14] P. Kocharoen, K. M. Ahmed, R.M.A.P. Rajatheva and W.A.C. Fernando, "Intensity Feature Filtering With Node Elimination Approach For Low Bit-Rate Mesh-Based Image Coding," *The IASTED International Conference on Networks and Communication Systems, NCS2005*, Krabi, Thailand, pp. 161-166, 18-20 April 2005.

[15] R.J. Renka, "Triangulation and Interpolation at Arbitrarily Distributed Points in the Plane," *ACM Transactions on Mathematical Software(TOMS)*, Vol. 10, No. 4, pp. 440-442, Dec 1984.

[16] A. Bowyer, "Computing Dirichlet tessellations," *The Computer Journal*, Vol.24, No.2, pp. 162-166, 1981.

[17] S. Rebay, "Efficient Unstructured Mesh Generation By Means Of Delaunay Triangulation And Bowyer-Watson Algorithm," *Journal of Computational Physics*, Vol.106, pp.125-138, 1993.

[18] G.J. Awcock and R. Thomas, *Applied Image Processing*, McGraw-Hill, Inc., pp.126-146, 1996.

[19] J.O. Rourke, *Computational Geometry in C*, 2nd edition, Cambridge University Press, pp.155-192, 1998.

[20] I. Amidror, "Scattered Data Interpolation Methods for Electronic Imaging Systems: A Survey," *Journal of Electronic Imaging*, Vol. 11 No. 2, pp. 157-176, April 2002.

[21] Z. Song and N. Roussopoulos, "Using Hilbert Curve In Image Storing And Retrieving," *International Multimedia Conference Proceedings of the 2000 ACM workshops on Multimedia*, California, USA, pp. 167-170, 2000.

[22] P. Kocharoen, *Internal project report*, Telecommunication Program, Asian Institute of Technology, Jan. 2004, unpublished.

[23] J.R. Jain and A.K. Jain, "Displacement Measurement and Its Application in Interframe Image Coding," *IEEE Trans. Comm.*, Vol. COM-29, no. 12, pp.1799-1808, December 1981.

[24] C.S. Fuh and P. Maragos, "Affine Models for Image Matching and Motion Detection," *Proc. IEEE Int. Conf. on ASSP*, Toronto, pp. 2409-2412, 14-17 May 1991.

[25] W.W. Peterson and E.J. Weldon, Jr., *Error-Correcting Codes*, 2nd edition, Halliday Lithograph Corporation, tenth printing, pp.170-203, 1990.

[26] P.E. Gougelet, The XnView website. [Online].
Available: http://www.xnview.com

[27] M.D. Adams, The JasPer Project Home Page, Department of Electrical and Computer Engineering, University of Victoria, [Online]. Available: http://www.ece.uvic.ca/~mdadams/jasper/

Preecha Kocharoen received the Bachelor degree in Electrical Engineering from Sripatum University, Thailand, in 1997, and Master degree in Communication Engineering from University of Manchester Institute of Science and Technology (UMIST), UK, in 1998. Since 1999 he has been with Sripatum University, Thailand. His current position is Lecturer in Department of Electrical Engineering, Faculty of Engineering. He is currently a doctoral student in the Telecommunications program School of Advanced Technologies, Asian Institute of Technology (AIT). His current research interests include: digital image and video processing and intelligent video encoding.

Dr. Kazi M. Ahmed is currently a Professor at Asian Institute of Technology (AIT). He received the M.Sc. Eng. Degree in Electrical Engineering from Institute of Communications, Leningrad, USSR; 1978. He has completed his Ph.D. at the University of Newcastle, NSW, Australia; 1983. His current research interests include: digital signal processing, antenna array processing, tropospheric and ionospheric propagation studies for microwave, VHF-UHF communications, satellite communication and radar problems.

Dr. R.M.A.P. Rajatheva is with Telecommunications program School of Advanced Technologies, Asian Institute of Technology (AIT) from Dec. 2004 as an Assosiate Professor. He was with the University of Moratuwa, Sri Lanka before joining AIT where he was promoted to Professor in Electronic & Telecommunication Engineering in June 2003. He also had been previously with TC-SAT from May 1996 to Dec.2001, as an Associate Professor from January 2001. He is a Senior Member of the IEEE. Dr. Rajatheva graduated with the first class honors in B.Sc.(Eng.) from the University of Moratuwa in 1987 and obtained both M.Sc., Ph.D. from the University of Manitoba, Canada in 1991 and 1995 respectively. His research interests include: Mobile and Wireless Communications, Coding and Modulation Techniques, Space Time Processing for MIMO Systems and Communication Theory.

Dr. W.A.C. Fernando is currently a Lecturer in signal processing at the Brunel University, U.K. Prior to that, he was an Assistant Professor in AIT. He received the B.Sc. Engineering degree (First class) in Electronic and Telecommunications Engineering from the University of Moratuwa in 1995 and the M.Eng. degree (Distinction) in Telecommunications from AIT in 1997. He has completed his Ph.D. at the Department of Electrical and Electronic Engineering, University of Bristol, UK. His current research includes: digital image and video processing, stereo and multi-view video coding, rate controlling, OFDM and CDMA for wireless channels, combined source and channel coding for video communications over wireless channels.