UDC 621.397

E.V. OSHAROVSKAYA
Odessa National O.S. Popov's Academy of Telecommunications, Ukraine

CODING TECHNIQUES OF DYNAMIC 3D MESHIS

Abstract. This paper provides an overview of the state-of-the-art techniques recently developed within the emerging field of dynamic mesh compression. Static encoders, wavelet-based schemes, a compression approach based on principal component analysis differential temporal and spatio-temporal predictive techniques, and clustering-based representations are considered.

Key words: image, 3D mesh, dynamic mesh compression, segmentation, PCA.

Е.В. ОШАРОВСКАЯ
Одесская национальная академия связи им. А.С. Попова

МЕТОДЫ КОДИРОВАНИЯ ДИНАМИЧЕСКИХ 3D СЕТОК

Аннотация – Эта статья представляет собой обзор современных методов в области сжатия динамических сеток. Рассматриваются статические кодеры, схемы на основе вейвлет – разложения, анализа главных компонент, временных и пространственно – временных методов предсказания и кластеризации.

Ключевые слова: изображение, 3D сетка, сжатие динамических сеток, кластеризация, PCA.

Introduction

Three dimensional (3D) meshes are emerging multimedia contents to represent realistic visual data. They consist of geometrical positions of sampled points (or vertices) and topological relations among the vertices. The huge size of a typical 3D mesh has necessitated the development of mesh compression technology. This paper provides an overview of the state-of-the-art techniques recently developed within the emerging field of dynamic mesh compression. Static encoders, wavelet-based schemes, a compression approach based on principal component analysis PCA -based approaches, differential temporal and spatio-temporal predictive techniques, and clustering-based representations are considered, presented, analyzed, and objectively compared in terms of compression efficiency, algorithmic and computational aspects and offered functionalities (such as progressive transmission, scalable rendering, computational and algorithmic aspects, field of applicability).

Compression of dynamic 3-D meshes is a young research area. First compression approaches were presented 1999, and in the following years, only a few papers dealing with this topic appeared. Since 2004, along with emerging research efforts in 3DTV, a high level of research activity can be noticed in this prospective area, which is reflected in many publications. Dynamic 3-D meshes, i.e., sequences of static meshes with no changing connectivity, show not only geometrical dependencies in spatial direction but also dependencies in temporal direction. Compression approaches exploiting these dependencies can be classified in two types: transform dominated and prediction dominated. [1]

However, there are some limitations in the existing algorithms. First, most algorithms can encode only isomorphic sequences, in which the number of vertices and the topological relations among vertices are invariant over all the frames. In the case when the mesh structure changes substantially between frames, their algorithm spends many bits to encode the topological changes and yields even worse performance than static mesh coders. Second, the existing algorithms do not support progressive coding. In progressive coding, a coarse mesh is first transmitted and rendered, and additional data are then transmitted to refine the mesh. Therefore, progressive coding is desirable for transmission of complex mesh sequences over networks with limited bandwidth. [2]

The area of compression of static 3-D meshes has already reached a high level of maturity, which is reflected in many publications and an already existing MPEG standard. Compression of dynamic 3-D meshes is a young but nevertheless active and prospective research area, which receives many impulses from static mesh compression. Recently, MPEG started a new activity on the specification of an improved standard for compression of dynamic 3-D meshes. Textured dynamic meshes, i.e., dynamic meshes with varying texture information at each time-instance, are used for 3-D video object representations. They allow a free-viewpoint representation, as it is required for interactive 3DTV applications. Only little attention has been paid so far to compression of dynamic meshes with time-varying texture information, pointing out promising directions for future research. [3]

Vertex-based predictive techniques

Here, the animation sequence is processed locally in space and time. At each stage of the compression process, only a limited number of previous frames (generally the last encoded frame) are involved when encoding the current frame, within the framework of a predictive scheme. A traversal order of the mesh vertices (e.g. such as the ones considered by mono-resolution connectivity encoders) is supposed to be available.

The Interpolation Compression (IC) scheme recently adopted by the MPEG-4 AFX proposes an interpolation procedure aiming at exploiting redundancies between successive key-frames. The principle consists of reducing the amount of data by sub-sampling the initial sequence of key-frames that are subsequently differentially encoded. Three simple prediction modes are currently supported, which involves spatial, temporal, and spatial-temporal predictors respectively described in [2], [3] and [4]. The decoder restores the key-frames and generates the intermediate frames by applying a linear interpolation procedure. Different strategies can be applied for selecting the key-frames, starting from simple uniform sampling techniques, and up to more sophisticated procedure aiming to determine an optimal set of key frames, minimizing an error criterion over the whole set of frames in the sequence. Such an approach is discussed in, where starting from the initial and final frames, a refinement procedure is iteratively applied while the error score becomes lower than a given threshold.

$$v_i^j = v_i^k + r_i^j ,\ v_i^j = v_{i-1}^k + r_i^j ,\ v_i^j = v_{i-1}^j + (v_i^k - v_{i-1}^k) + r_i^j \qquad (1)$$

Here, v_i^j is the position of vertex j at the instant *i*, r_i^j its prediction error, and k is the index of a previously decoded vertex. Such prediction rules are obviously too elementary, but offers the advantage of a low computational complexity.

Predictive approaches offer the advantage of simplicity and low computational cost, which makes them adapted for real time decoding applications. However, being based on a deterministic traversal of mesh vertices, such mono-resolution approaches do not support more advanced functionalities such as progressive transmission and scalable rendering. This drawback is overcome by the multi-resolution representations, based on the wavelets techniques described in the next section.

2.2 Wavelet-based approaches

Wavelets techniques have been extensively used for still image and video progressive compression applications with impressive compression performances. However, the extension of wavelet basis functions, usually defined on regular lattices, to 3D meshes of arbitrary topology is not straightforward. [5]

For this reason, the first wavelet-based 3D mesh representations modifies the original irregular topology in order to obtain regular or semi-regular connectivities adapted for wavelet construction. The re-meshing based techniques lead to the best compression performances for static mesh encoding. Such approaches discard totally the original irregular connectivity and re-sample the mesh surface in order to obtain a semiregular or regular topology well-suited to a wavelet construction. Beside the high compression performances, the wavelet techniques naturally support progressivity and scalability.

The approach presented in [6] includes the following steps: construction of a mesh parameterization, defined on the unit square; uniform re-sampling of the parametric domain and construction of so-called geometry images which store the *x, y* and *z* vertex coordinates as the R, G and B planes of a color image; encoding of the geometry image with a 2D traditional wavelet encoder. The extension of this technique to animated meshes is straightforward: a geometry image is derived for each frame. A video sequence is thus constructed, and wavelet encoded. Two encoding modes intra and predicted are defined and applied for each frame. In the intra-mode, the frame (so called I-frame) is encoded independently, as a still image, without any reference to other frames. In the predicted mode, the prediction errors between current (so called P-frame) and previous frame are feeding the wavelet encoder. The I-frame enables a random access to the compressed animation stream while the P-frames ensure the temporal decorrelation and thus the compression efficiency. The semi-regular mesh sequence is decomposed in the spatiotemporal domain and compressed with a progressive coder. Each frame in the semi-regular mesh sequence is decomposed into base mesh points and wavelet coefficients for subdivision points. Each subdivision point $\mathbf{p}$ is predicted as $\mathbf{p}_b$ from the butterfly prediction using lower level points to yield a wavelet coefficient $\psi\mathbf{p}$, given by

$$\psi\mathrm{p} = \mathrm{p} - \mathrm{p}_b. \qquad (2)$$

The wavelet coefficients are further decomposed in the temporal domain along motion. Suppose that a temporal sequence of wavelet coefficients along a motion trajectory, $\{\psi\mathbf{p}_n\}$, is divided into the even subsequence $\{\psi\mathbf{p}_{2k}\}$ and the odd subsequence $\{\psi\mathbf{p}_{2k+1}\}$, where *n*, 2*k* and 2*k* + 1 are time indices. Then, the temporal lifting decomposition can be written as

$$\mathrm{h}_k = \psi\mathrm{p}_{2k+1} - P(\{\psi\mathrm{p}_{2k}\}),\quad \mathrm{l}_k = \psi\mathrm{p}_{2k} + U(\{\mathrm{h}_k\}), \qquad (3)$$

where h_k and l_k are the highpass and the lowpass terms, respectively. *P* is the prediction function using the even subsequence. For example, when $P(\{\psi\mathrm{p}_{2k}\}) = \psi\mathrm{p}_{2k}$ and $U(\{\mathrm{h}_k\}) = 0.5\mathrm{h}_k$, (3) becomes the Haar decomposition.

In order to deal with meshes of arbitrary topologies, an optimized cutting procedure is applied. However, in the case of animated meshes, the parameterization information is shared by all the meshes in the sequence and can be transmitted only once for the first frame. Guskov and Khodakovsky exploit this idea in order to compress mesh sequences with irregular and anisotropic wavelet transform. [3]

The Wavelets Animation Compression (AWC) approach constructs first a Progressive Mesh (PM) hierarchy, defined on the first mesh M_0 of the sequence. Then, an anisotropic wavelet transform is constructed on the top of this PM structure. Starting from the base mesh, a refinement process is applied which consists of re-inserting the previously eliminated vertices until the initial connectivity is completely recovered. For each newly inserted vertex *v* at LODj, an optimized, anisotropic wavelet filter Φ_j is computed. The so-determined PM hierarchy and wavelets filters are finally used for computing the wavelets detail coefficients d_i^j for all the frames in the sequence (*n* stands for the number of LODs). Each detail d_i^j records the difference between the actual vertex positions at frame *i* and its anisotropically predicted position from the LOD $_{j-1}$.

The animation stream is encoded as two types of frames: I-frames and P-frames. The I-frames are encoded using only spatial wavelet transform and thus enable random access to the animation sequence. The P-frames linearly predict the wavelets details from the previous frames in order to capture the temporal correlations. The so-obtained coefficients are quantized and progressively sent to the decoder using a zero-tree-like algorithm. [4]

Clustering based approaches

The mesh splits into sub-parts, so-called clusters, whose motion can be accurately described by rigid transform, defined with respect to a reference frame and determined using an heuristic approach. Such a transform allows to apply a motion compensation procedure and to predict the position of vertices in the current frame from the reference one. The object’s motion is finally described by the set of rigid motion parameters associated with each cluster and the prediction residuals associated with each vertex. However, such segmentation-based approaches lead to visually unpleasant motion discontinuities at the level of frontiers between clusters. In order to reduce this effect, a spatio-temporal smoothing scheme, based on a post processing, low-pass filtering of vertex displacements is applied. A different clustering-based representation, based on an octree decomposition of the object is proposed. [7]

The principal component analysis

The principal component analysis-is one of the main ways to reduce the dimensionality of the data, having lost the least amount of information. It is used in image processing, data compression; Evaluation of principal component reduces to the calculation of the eigenvectors and eigenvalues of the covariance matrix of the original data. Sometimes the principal component transform is referred to as Karhunen - Loeve transform, or Hotelling transform. The task of principal component analysis has at least four basic versions: approximated data linear manifolds of lower dimension: to find a subspace of lower dimension in the orthogonal projection on which scatter in the data (i.e., the standard deviation from the mean) is maximal; to find a subspace of lower dimension in the orthogonal projection on which the mean square distance between the points as possible; to construct an orthogonal coordinate transformation for the multivariate random variable, which resulted in a correlation between the individual coordinates will turn to zero. [8]

All the tasks of the main components analysis are connected to the problem of diagonalization of the covariance matrix or the sample covariance matrix. Empirical or sample covariance matrix is

$$C = [c_{ij}], \mathrm{c_{ij}} = \frac{1}{m}\sum_{l=1}^{m}(x_{li} - \overline{X}_i)(x_{lj} - \overline{X}_j) \qquad (4)$$

Covariance matrix of the multivariate random variable X is

$$\Sigma\Sigma = [\sigma_{ij}], \sigma_{\mathrm{ij}} = \mathrm{cov}(x_i x_j) = E[(x_i - E\overline{X}_i)(x_j - E\overline{X}_j) \qquad (5)$$

In the general, multidimensional case process of allocation of the main components occurs so:

1. We look for cloud data center, and are moved to the new origin – this is zero principal components (PC0)
2. The direction of maximum change is selected in the data - this is the first principal component (PC1)
3. If the data is not fully covered (large noise), selects another direction (PC2) - perpendicular to the first so as to describe a change in the remaining data, etc.

Conclusion

The suggested methods show improvement in reducing the data rate and amount of data required to code dynamic mesh. The best result is obtained during irregular meshes processing. Although this methods are appropriate for 3D animation, further study on other methods that can improve the existing approaches in increasing the compression efficiency should be conducted.

References

1. K. Mamou, T. Zaharia and F. Preteux, “FAMC: The MPEG-4 Standard for Animated Mesh Compression," Proc. Image Processing (ICIP ’08), pp. 2676-2679, 2008, doi:10.1109/ICIP.2008.4712345.

2. Z. Karni, C. Gotsman, “Compression of soft-body animation sequences”, Computer Graphics, Vol. 28, pp. 25-34, 2004.

3. Guskov, A. Khodakovsky, “Wavelet Compression of Parametrically Coherent Mesh Sequences”, ACM Siggraph Symposium on Computer Animation, pp. 183-192, Grenoble, France, 2004.

4. O. Petrik and L. Vasa, “Improvements of MPEG-4 Standard FAMC for Efficient 3D Animation Compression,” Proc. Fifth The True Vision Capture, Transmission and Display of 3D Video (3DTV), 2011.

4.1. Maglo, C. Courbet, P. Alliez and C. Hudelot, “Progressive Compression of Manifold Polygon Meshes,” Computers and Graphics, vol. 36, no. 5, pp. 349-359, 2012.

5. K. Sakuragi, B. Zhou, Y. Fujita and A. Kawanaka, "Efficient Coding for Dynamic 3-D Mesh with Vertices Structuring and Connectivity-oriented Multiresolution Decomposition," Digital Signal Processing, vol. 23, no. 1, pp. 176-183, Jan 2013, doi:10.1016/j.dsp.2012.09.003.

6. K. Müller, A. Smolic, M. Kautzner, P. Eisert, T. Wiegand, “Predictive compression of dynamic 3D meshes”, ISO/IEC JTC1/SC29/WG11, MPEG04/M11240, Palma de Mallorca, Spain, October 2004.

7. V´aˇsa and V. Skala, “A perception correlated comparison method for dynamic meshes,” IEEE Trns. on Visualization and Computer Graphics, vol. 17, no. 2, pp. 220–230, 2011.

Рецензія/Peer review : 27.12.2014 р. Надрукована/Printed :14.1.2015 р.