[3-23]

Progressive Compression of PointTexture Images

*In-Wook Song, **Chang-Su Kim and *Sang-Uk Lee
*Seoul National University, **The Chinese University of Hong Kong

iwook@ipl.snu.ac.kr, cskim@ieee.org, sanguk@ipl.snu.ac.kr

Abstract

In this paper, we develop a tree-structured predictive partial matching (PPM) scheme for progressive compression of PointTexture images. By incorporating PPM with tree-structured coding, the proposed algorithm can compress 3D depth information progressively into a single bitstream. Also, the proposed algorithm compresses color information using a differential pulse coding modulation (DPCM) coder and interweaves the compressed depth and color information efficiently. Thus, the decoder can reconstruct 3D models from the coarsest resolution to the highest resolution from a single bitstream. Simulation results demonstrate that the proposed algorithm provides much better compression performance than a universal Lempel-Ziv coder, WinZip.

I. Introduction

Depth image-based representation (DIBR) is a new method to represent and render 3D objects with complex geometries [1] and has been adopted into MPEG-4 Animation Framework eXtension (AFX) [2]. Instead of representing objects with polygonal meshes as done typically in computer graphics, DIBR represents a 3D object with a set of reference images covering its visible surface. Each reference image comes with a depth map, which is an array of distances from the pixels in the image plane to the object surface. One of the advantages of DIBR is that reference images can provide high quality visualization of the object without using complex polygonal models. In addition, the complexity of rendering a DIBR view is only related to the number of pixels in the view (*i.e.* the resolution of the view) regardless of the scene complexity.

DIBR has three major formats: SimpleTexture, PointTexture, and OctreeImage. Among them, PointTexture represents an object with an array of pixels viewed from a single camera location. Each PointTexture pixel is represented by its color, depth (the distance from the corresponding object point to the camera), and a few other properties. There can be multiple pixels along each line of sight, and thus a PointTexture usually consists of multiple layers. Complex PointTexture images, which represent realistic images with high sampling density, typically require a huge amount of data. Therefore, the compression of PointTexture images should be performed efficiently.

Duan and Li proposed an algorithm for compressing PointTexture images [3], which are called layered depth images (LDIs). Their algorithm uses JPEG-LS algorithm [4] to compress depth data. Also, color data are compressed by using existing coding standards MPEG-4 SA-DCT [5] and JPEG-2000 [6]. The direct application of these methods is inefficient, since LDI data are sparser than 2D images. Thus, they proposed a data aggregation method to improve the coding gain. However, the aggregation method does not fully exploit the characteristics of 3D surfaces and the resulting data may not exhibit a sufficiently high correlation. Moreover, their algorithm does not support progressive compression and transmission. To transmit 3D objects over Internet or wireless channels, it is desirable to develop progressive coding schemes, which facilitate the adaptation of transmission bitrate and interactive 3D model browsing.

In this paper, we develop a tree-structured predictive partial matching (PPM) scheme for progressive compression of PointTexture images. PPM was originally proposed to compress facsimile data in a lossless manner [7]. By incorporating PPM with tree-structured coding, the proposed algorithm can compress 3D depth information progressively into a single bitstream. Also, the proposed algorithm compresses color information by using a differential pulse coding modulation (DPCM) coder and interweaves the depth and color information effectively.

More specifically, a PointTexture image is first converted into volume data and then expressed by an octree, called an initial tree. The initial tree is merged iteratively to approximate the original information at several refinement levels. The merging order is selected to minimize a pre-defined criterion. After the merging, the bitstream is encoded in the reverse order, *i.e.* the lastly merged is encoded first. Therefore, the decoder can reconstruct the 3D model from the coarsest model (single root node) to the most refined model (original model).

This paper is organized as follows. Section II briefly describes the data structure of DIBR. Section III describes the octree generation algorithm. Sections IV and V propose the encoding and decoding algorithms, respectively. Section VI provides simulation results. Finally, conclusions are drawn in Section VII.

II. Depth Image-based Representation

There are many different methods to represent and render depth images or images with depth information, and DIBR is designed to standardize such representation methods. DIBR has three main formats: SimpleTexture, PointTexture, and OctreeImage.

SimpleTexture is a data structure that consists of an image, corresponding depth map, and camera description (its position, orientation and type). Although it enables the reconstruction of object views at a substantial range of angles, the representation capability of a single SimpleTexture image is restricted to simple objects such as facade of a building. Therefore, a collection of SimpleTextures should be acquired with properly positioned cameras to represent the whole building.