Activity-Aware Adaptive Compression

A Morphing-Based Frame Synthesis Application in 3DTI

Shannon Chen, Pengye Xia, and Klara Nahrstedt
Department of Computer Science, University of Illinois at Urbana-Champaign
{cchen116, pxia3, klara}@illinois.edu

ABSTRACT

In view of the different demands on quality of service of different user activities in the 3D Tele-immersive (3DTI) environment, we combine activity recognition and real-time morphing-based compression and present the Activity-Aware Adaptive Compression. We implement this scheme on our 3DTI platform: the TEEVE Endpoint, which is a runtime engine to handle the creation, transmission and rendering of 3DTI data. User study as well as objective evaluation of the scheme show that it can achieve 25% more bandwidth saving compared to conventional 3D data compression as *zlib* without perceptible degradation in the user experience.

Categories and Subject Descriptors

H.1.2 [**Information Systems**]: Models and Principles – *human factors;* H.4.3 [**Information Systems Applications**]: Communications Applications – *Computer conferencing, teleconferencing, and videoconferencing;* I.4.2 [**Image Processing and Computer Vision**]: Compression (Coding) – *Approximate methods*

Keywords

3D Tele-Immersion; Compression; Morphing; Frame Rate; Adaptation

1. INTRODUCTION

During the past decade, the potential of 3D Tele-immersive (3DTI) services has gained its attention from both academia and industry. While most commercial 3D systems are specialized for sole purpose, the trend of development of 3DTI platforms is towards multi-purpose, multi-sites, and multi-modal platforms [1-2] to enable a variety of user activities including e-learning, collaborative art performance, and exergaming [4].

Inevitably, with its great potential, the resource demand of 3DTI rises due to its interactive characteristic, complexity of 3D rendering, and delivery of media-rich content [2]. From previous observations [5], we see that 3DTI users can have very different tolerance to quality degradation when participating in different types of user activities. Thus, we present the Activity-Aware Adaptive Compression (A3C), which is a real-time compression scheme that combines activity recognition and morphing-based compression on the 3DTI content.

MM'13, October 21–25, 2013, Barcelona, Spain.

ACM 978-1-4503-2404-5/13/10…$15.00.
http://dx.doi.org/10.1145/2502081.2508116

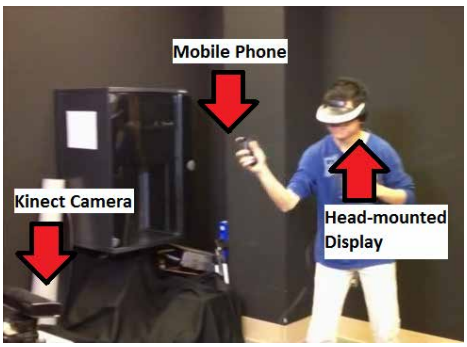

Figure 1. User interface of TEEVE Endpoint.

A3C adopts the Morphing-based Frame Synthesis (MBFS) as its compression scheme. Taking advantage of the unique properties of 3DTI scenes, the MBFS technique allows the content receiver to boost up the video frame rate by injecting synthesized frames into the received content. Thus, the transmission frame rate of the content producer can be reduced to save substantial amount of bandwidth for content delivery. With MBFS, the deducted frames will be synthesized and restored at the receiver side, hence the perceived quality can be preserved.

To maximize the effectiveness of MBFS compression, we apply activity recognition to identify the motion characteristics of current user activity and to assign suitable compression parameters (e.g., the Synthesized Frames per Second, which will be introduced later in Section 3.2.) The visual quality of synthesized frames depends heavily on the motion level of the content. For high motion content, the graphical difference between captured frames is large and the sharpness degrades due to motion blur. These effects increase the artifacts introduced by feature-based morphing, making the synthesis unnatural and more detectable to viewers. Therefore, we combine the compression scheme with user activity recognition to dynamically fine-tune the compression scheme and achieve high visual quality and optimum resource saving.

We implement the A3C scheme on our 3DTI platform: the TEEVE Endpoint, which is a runtime engine to handle the creation, transmission and rendering of 3DTI data. The user interface of the TEEVE Endpoint includes Kinect camera, head-mounted display, and the user's mobile phone (Figure 1.) The Kinect camera and the accelerometers embedded in the display capture the head position and the view direction of user. To provide a vivid immersive experience, the information is used in the construction of the 3DTI scene to synchronize the view point and view direction of user in the virtual space with those in the physical user space. The user's mobile phone serves both purposes of a control console and an auxiliary on-body motion sensor for activity recognition. With a downloadable app, the accelerometer in the phone would be connected to the endpoint and provide sensing data to aid the immersive experience.

We evaluate our total solution via objective compression ratio as well as subjective user study. Using four common user activities with distinctive motion characteristics, the former shows a bandwidth saving up to 25% more than conventional compression scheme adopted by existed 3DTI platforms. The user study, on the other hand, is conducted via interviewing the players of a 3DTI virtual fencing game built upon the TEEVE Endpoint. Results show that A3C does not introduce perceptible degradation to the gameplay experience with its substantial resource saving.

The remainder of this paper is organized as follows. In the next section, an overview of the TEEVE Endpoint is provided as a background for the introduction of A3C in Section 3. Section 4 covers the evaluation of A3C, which includes both objective and subjective aspects. Finally, in Section 5 we conclude.

2. SYSTEM COMPONENTS

2.1 TEEVE Endpoint

The TEEVE Endpoint is our runtime engine to handle the 3DTI data and it provides application programming interfaces (APIs) to developers to easily create 3DTI applications. As shown in Figure 2, the architecture of the runtime engine contains several modules. The *capturing module* takes the raw data from 3D camera and generates 3DTI data in real-time. The *resource management module* takes the 3DTI data to compose 3DTI frames which are disseminated over the network to the remote sites through the *transmission module*. The 3DTI data is passed to the *rendering module* for displaying the visual feedback. The *3DTI game engine* sits on top of the rendering module to control the rendering behavior in response to the events detected within the system. These events include system state changes, collision detection, and user-interaction input. The 3DTI game engine further provides API to the developers to register callback functions for those events to define the rendering behavior.

2.2 Tele-immersive User Interface

The user interface of our solution includes Kinect camera, mobile phone and head-mounted display. The three devices collaborate with each other as part of the UI module to provide 3D visual data input, user perspective input, user control input and user activity input, which are coordinated by the game engine. The *3D visual data input* is captured by the Kinect camera. Each 3DTI site is equipped with one Kinect camera. With the depth information, the scene is captured and represented as a point cloud. The *User perspective input* represents the view point and view direction of the user. The position and rotation of the head are tracked by the Kinect and the motion sensor embedded in the head-mounted display, respectively. The *User control input* is retrieved from the translational movement and rotational movement of the user's hand. The former is tracked by the Kinect camera while the latter is tracked by user's mobile phone when being held as a console. The *User activity input* is continuously fetched by the mobile phone when it is not being used as a console. The triaxial accelerometer embedded in the phone provides motion data to the endpoint for activity recognition.

3. ACTIVITY-AWARE ADAPTIVE COMPRESSION

Our total solution of A3C can be broken down into three major components: activity recognition, frame deduction, and frame amendment (Figure 3.) The first two components reside in the resource management module at the producer endpoint, while the last component resides in the rendering module at the receiver endpoint. In the following sections, we introduce the design of these components.

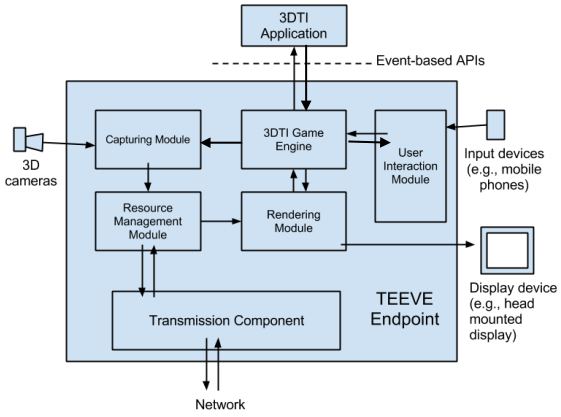

Figure 2. TEEVE Endpoint Architecture.

3.1 Activity Recognition

As the first component of our system, we build a SVM-based [6] mapping from motion data acquired from user's mobile phone to user activities. Each common user activity in 3DTI environment has its motional and postural uniqueness. Thus, by monitoring the motion data from the on-body mobile phone, the activities can be classified in real-time with a machine leaning approach. The activities we are targeting and their motional/postural characteristics are listed as follows.

- Storytelling: User is *sitting* in the center of the 3DTI environment with most of her action concentrate on facial area. Occasional gesture is expected.
- Speech: User is *standing* in the center of the 3DTI environment. Facial movement is expected along with occasional gestural and body movement.
- Exercise learning: Slow and gross-motor movements of all body parts are expected from the user.
- Gaming: Both posture and position of user change rapidly. Fast and gross-motor movements are expected.

To build up a mapping from sensor data to user activity for real-time classification, we apply the Support Vector Machine [6]. The features (e.g., positions, speed, changing frequency,) fed into the machine, are deducted from the variation of acceleration in the time domain and the power spectrum in the frequency domain. The former is acquired by sliding window analysis on the data compiled while the latter is acquired by Fast Fourier Transform.

We recruited 10 participants to perform all four activities in the 3DTI environment. Due to the size of regular smart phones, all participants naturally place the phone in their pants pocket when asked to carry it on-body. 200-minute record containing all four activities in equal lengths was compiled. With this training data, the SVM is able to generate a user activity classifier for our system which takes current motion data as input, and output the type of user activity in real-time. The accuracy of the trained machine is 91.5% (10-fold cross-validation,) which provides a solid foundation to the consecutive procedures of A3C.

3.2 Frame Deduction and Amendment

The design of the morphing-based compression mechanism is to deduct a certain amount of frames at the resource management module of the producer site before they are sent to the transmission module. The deduction will then be amended at the

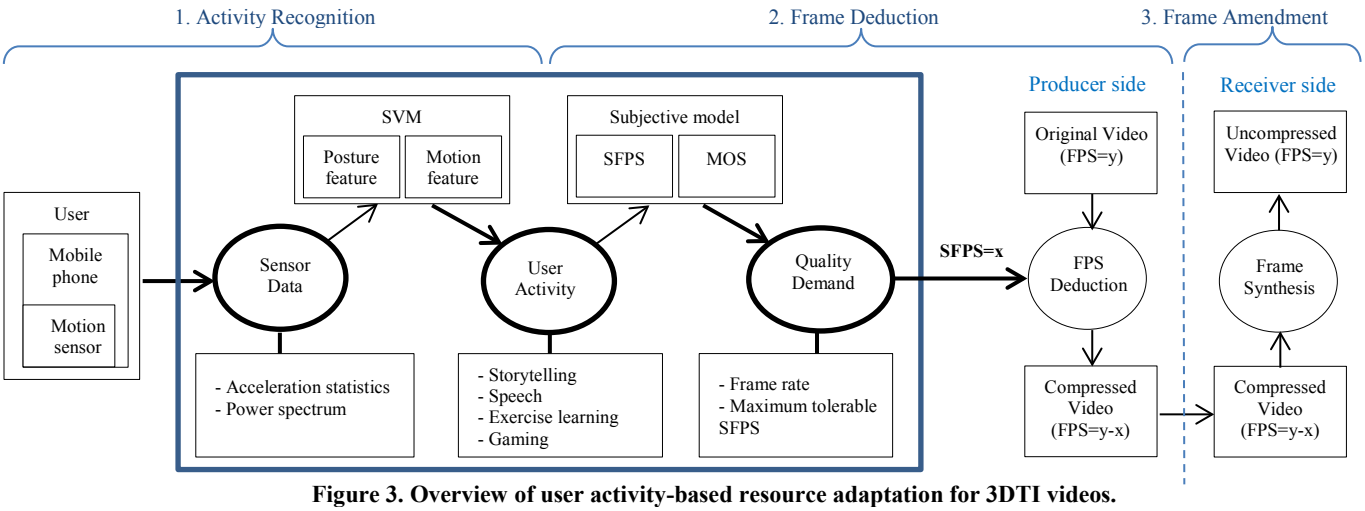

Figure 3. Overview of user activity-based resource adaptation for 3DTI videos.

receiver site in its rendering module. Synthesized frames will be created based on the received content and injected into the video. These synthesized frames replace the deducted one and restore the frame rate. Figure 3 shows an example where *x* frames are deducted/amended in the compression/decompression of a video which originally contains *y* frames per second.

The amount of frames deducted is described by a compression parameter: the Synthesized Frames per Second (SFPS), which also stands for the number of synthesized frames injected in the final uncompressed result. In the following, we first introduce the technique of frame amendment: the Morphing-Based Frame Synthesis (MBFS) before we detail the decision of SFPS adjustment based on user perception and activity characteristics.

3.2.1 Morphing-Based Frame Synthesis

With the frame synthesis technique, 3DTI visual content has more flexibility on frame rate adaptation compared to conventional 2D videos. 3DTI bears some properties that make MBFS possible:

- With the depth information, only the subjects are captured to be put into the virtual space. The background is discarded.
- Common subjects in 3DTI are human bodies, which have fair sizes that take major portion of the scene.
- Number of subjects is limited by the interactive characteristic and the size of the display. This lowers the graphical complexity of each scene.

Combining these properties, we propose to create synthesized frames by graphic morphing [7], which is a special effect in motion pictures that transit one image to another based on predefined feature pairs. Figure 4 shows an example of applying morphing technique for frame synthesis. The first and the last frames in the figure are the only real frames captured by a camera, while the ten frames in-between are synthesized.

We use graphical feature matching [10-11] and skeleton detection to mark the feature pairs in two captured frames. Due to the limited number of subjects and their fair sizes, the graphical feature matching can provide a meaningful number of matching pairs. In addition, human bodies are the most common subjects in the scene. Thus, we also include the joints detected by skeleton detection of Kinect as matching feature pairs. For image processing after the feature pairs are acquired, the paper of Beier and Neely [7] provides extensive details. As a result, with the morphing technique we can insert extra synthesized frames to boost up the original frame rate.

3.2.2 Activity-Aware Frame Deduction

Due to different motion characteristics, for different user activities, the optimum adjustment of SFPS is very different. The prominence of synthesized frames depends heavily on the motion of the visual content. For relatively static activities, the difference between frames is small, which makes the morphing result very similar to the deducted frame. On the other hand, for activities that contain intense body movements, the high motion introduces motion blur to the captured frames, which raises the bar of graphical feature matching and skeleton detection. These disadvantages increase the possibility of feature mismatch, which renders the synthesis unnatural and more detectable by the viewer.

To explore this complication, we conduct user tests and ask 15 participants to rate sequences of pre-recorded 3DTI visual content of the four types of 3DTI user activities (Section 3.1.) Each sequence contains video clips of the same activity compressed with different SPFS settings. The participants are requested to view these clips and give a Mean Opinion Score (MOS) to represent the Quality of Experience (QoE.) From the feedback of the participants, we are able to pinpoint the Just Noticeable Difference (JND) perceptual thresholds [8] of SFPS for all activities. The thresholds represent the SPFS that separate noticeable degradation from unnoticeable ones for each activity. These thresholds are built into the frame deduction component of A3C so that the system can adaptively compress the visual content without affecting its perceptual quality.

4. EVALUATION

4.1 Objective Metric: Compression Ratio

The purpose of adaptive compression is to find the lowest compression ratio for the 3DTI activity without comparable degradation of the QoE. Thus, we setup two resource saving modes with different levels of QoE degradations: imperceptible mode (targeting 90% of QoE of the original reference clip) and acceptable mode (targeting 80% QoE.) Table 1 shows the adjustment of SPFS in both modes and the resulting compression ratio (CR) of each scenario (due to space limit, detailed analysis of these results can be found in [1]). We also include the compression ratio of *zlib* [9], which is a popular compression scheme adopted by previous 3DTI systems. The result shows that the proposed A3C scheme can achieve up to 25% more bandwidth saving comparing to *zlib*.

Figure 4. An example of frame rate boosting. [1]

Note that Table 1 only shows the results of each activity separately. To achieve adaptive compression, A3C utilizes the activity recognition component (Section 3.1) to classify current movement and reconfigure the SFPS according to Table 1. For example, when user leaves her chair and starts to exercise, A3C automatically tunes the SFPS from 2.5 to 1.0 (assuming acceptable mode.) In practice, A3C reduces the 4.1 Mbps bandwidth consumption of 3DTI visual stream by 2.0~2.5 Mbps.

Table 1. Activity-based resource adaptation

User Activity			Story	Speech	Exercise	Game
A3C	Imper. mode	SFPS	1.4	0.0	0.0	2.5
		CR	2.1:1	1.8:1	1.9:1	2.4:1
	Accept. mode	SFPS	2.5	2.5	1.0	2.5
		CR	2.4:1	2.4:1	2.2:1	2.4:1
zlib		CR	1.8:1	1.8:1	1.9:1	1.9:1

4.2 Subjective Metric: Gameplay Experience

In order to evaluate the perceptual effect introduced by A3C, we choose the activity with the highest motion level: exergaming, as a medium of the subjective test. As we mentioned previously, high motion content is more likely to introduce artifacts under morphing-based compression. Thus, the quality of exergaming is the most vulnerable one to compression among other activities.

We implement a virtual fencing game on the TEEVE Endpoint as an experiment testbed. The game includes two sites connected within the campus network. Each site contains one Kinect camera to capture the 3D scene. The 3D data of each player is transmitted to the other site and rendered in the virtual world. Player in each site wears a head-mounted display embedded with accelerometers and sees from her first person's perspective in the virtual world. The players hold their mobile phone as game consoles. When the phone is used as a game console, the activity recognition of A3C component classifies current activity as "gaming." The SFPS is adaptively tuned to the value stated in Table 1. A sword is rendered in the virtual world at the hand of the player with the sword direction synchronized with the rotational movement of the phone. The objective of the game is to hit the remote player (the opponent) with your weapon. Player's score decreases as hit by one another. A player wins when her opponent's score decreases to zero. Figure 5 shows a sample game screen.

Table 2. Interview Questions and Average Scores

	A3C Enabled	A3C Disabled
Q1: Are you satisfied with the graphical resolution of the game?	3.3	3.3
Q2: How is the responsiveness of the game control?	4.4	4.1
Q3: Do you consider the game realistic?	3.4	3.7
Q4: Do you enjoy the immersive experience?	4.6	4.6

(Score: 5 being the most and 1 being the least)

Among the two 3DTI sites, only one of them has the A3C scheme enabled. We recruit seven real players to participate in the game in pairs. After a five-minute gameplay, we interview each player about their gameplay experience. After that, the players are asked to switch sites, play the game for another five minutes, and being interviewed again. The interview questions are listed in Table 2,

Figure 5. Game screen of the virtual fencing.

which focus on the sensory immersion of gameplay experience [3]. The players are asked to answer the questions on a 5-point Likert scale. The result suggests that the experience within both sites do not have significant difference. The similarity of the two sites is supported by two-way ANOVA test ($F<0.01$, $p=1$).

5. CONCLUSION

In this work, we propose a solution for 3DTI applications to reduce its bandwidth consumption without degrading the user experience. We present the Activity-Aware Adaptive Compression (A3C) scheme, which combines activity recognition and morphing-based compression. The scheme is implemented on the TEEVE Endpoint and evaluated via both objective and subjective metrics. Result shows that A3C brings 25% more bandwidth saving comparing to existed compression scheme of 3DTI without compromising the user experience.

6. ACKNOWLEDGMENTS

This material is based upon work supported by NSF Grant CNS 10-12194, CNS09-64081KN.

7. REFERENCES

[1] S. Chen et al. Activity-based synthesized frame generation in 3DTI Video. ICME, 2013.
[2] Z. Yang et al. Enabling multi-party 3D tele-immersive environments with ViewCast. TOMCCAP, 2009.
[3] L. Ermi et al. Fundamental components of the gameplay experience: analyzing immersion. DIGRA, 2005.
[4] M. Renata et al. Advancing interactive collaborative mediums through tele-immersive dance (TED): a symbiotic creativity and design environment for art and computer science. MM, 2008.
[5] Z. Huang et al. Towards the understanding of human perceptual quality in tele-immersive shared activity. MMSys, 2012.
[6] C.-C. Chang et al. LIBSVM: a library for support vector machines. TIST, 2011.
[7] T. Beier et al. Feature-based image metamorphosis. SIGGRAPH, 1992.
[8] G. Gescheider. Psychophysics: the fundamentals. Psychology Press, 3/e, 1997.
[9] zlib. http://www.zlib.net/
[10] M. Muja et al., Fast approximate nearest neighbors with automatic algorithm configuration, VISAPP , 2009.
[11] H. Bay et al., SURF: speeded up robust features, CVIU, 2008.