A COMPARISON OF DIFFERENT HAPTIC COMPRESSION TECHNIQUES

Cyrus Shahabi, Antonio Ortega, Mohammad R. Kolahdouzan

Integrated Media Systems Center
University of Southern California
Los Angeles, CA, 90089

shahabi@usc.edu ortega@sipi.usc.edu kolahdoz@usc.edu

ABSTRACT

Immersive environments provide an artificial world to surround users. These environments consist of a composition of various types of immersidata: unique data types that are combined to render a virtual experience. To construct such an environment, immersidata acquisition is indispensable for storage and future query. However, this is challenging because of the real time demands and sizeable amounts of data to be managed. In this research, we propose and evaluate alternative techniques for achieving efficient sampling and compression of one immersidata type, the haptic data, which describes the movement, rotation, and force associated with user-directed objects in an immersive environment. Our experiments identify the benefits and limitations of various techniques in terms of their data storage, bandwidth and accuracy.

1. INTRODUCTION

Immersive environments are those that aim to surround users in a combination of real and artificial worlds. They connect persons with other people, objects, places and databases through an augmented or virtual reality experience. These environments are compelling because they enable applications that simulate reality. For certain application domains, such as virtual training, an immersive environment dramatically improves the practicality and cost-effectiveness of those applications. Immersive environments consist of a composition of various types of immersidata [5], unique data types that are combined to render a virtual experience. One such immersidata type is the haptic data, information that describes the movement, rotation, and force associated with user-directed objects in an immersive environment. An example of a haptic interface is the sensing chair developed at MIT Media Lab and Purdue University. This chair is mounted with contact sensors, which monitor the posture and position of someone sitting in the chair [11]. Many potential applications of haptic interfaces involve training and simulation. The Virtual Reality Dental Training System Dental simulator was developed using a PHANToM with four tips which simulates actual dental instruments [1]. Training students to perform certain brush strokes or molding a sculpture a certain way was also demonstrated using haptic interfaces [9], [10].

Haptic data acquisition is critical because it allows the data to be stored and thus enables several important modes of operation on that stored data. However, haptic data acquisition is particularly challenging because of the real-time demands of simultaneously recording multiple sensors, maintaining the accuracy of recording, and handling the voluminous data produced during a session of reasonable length. One approach we are exploring deals with reducing the sampling rate for a given sensor or group of sensors to a level that reduces the amount of data we need to collect without sacrificing accuracy [7]. The general intuition is: for certain sessions or parts of sessions, many sensor values may not change or may not exhibit a spectrum of values that demands a high sampling rate. For these cases, it would seem useful to reduce the sampling rate to a much lower level. The problem is how to identify that level.

In this paper, we introduce run-time and off-line approaches that identify sampling rates based on the sensor or sensor group and the semantics of the session. In addition, We investigate the impact of an orthogonal approach to compress the haptic data through quantization techniques (e.g., Adaptive DPCM). We also combine sampling approaches with ADPCM technique and conduct several experiments to compare the accuracy and efficiency of our different haptic sampling and compression techniques.

The remainder of this paper is organized as follows. In Section 2, we describe haptic data and the device we used in our experiments in more detail. In Section 3 we introduce three sampling techniques, discuss their relative merits, and explain the adaptive DPCM technique for haptic data compression. Section 4 presents our experimental results and finally, Section 5 concludes and presents some future research directions.

2. HAPTIC DATA

Haptic data consists of a series of sensory values measured at some time point. Today, there are commonly three basic types of sensors associated with haptic devices: those that measure position, those that measure rotation, and those that measure force levels. However, there are still many other types of sensors. For example, a haptic glove or exoskeletal device that slips onto the hand might measure finger force. Also, the number and placement of sensors can affect the accuracy of representing haptic sessions. Acquiring

THIS RESEARCH HAS BEEN FUNDED IN PART BY NSF GRANTS EEC-9529152 (IMSC ERC) AND IIS-0082826, NIH-NLM GRANT NR. R01-LM07061, NASA/JPL CONTRACT NR. 961518, DARPA AND USAF UNDER AGREEMENT NR. F30602-99-1-0524, AND UNRESTRICTED CASH GIFTS FROM NCR, MICROSOFT AND OKAWA FOUNDATION.

haptic data involves sampling it at various time points.

We focus our study of haptic data acquisition on one set of the available devices: the CyberGrasp exoskeletal interface and accompanying CyberGlove, together with a Polhemus Fastrak position tracking device, that include 33 sensores. CyberGrasp interfaces two subclasses of haptic data: grasping and kinesthetic data. Grasping data is a set of 28 floating point values corresponding to 22 angles for the 22 degrees of freedom of the hand, 3 values for hand coordinates in 3D Cartesian space, and 3 angles for hand orientation. These values can be captured by monitoring the amount of stretching, bending, and euclidean distances of the fibers within the CyberGrasp fingers as well as the tracking device located on the center of the palm. Finally, the kinesthetic data is a set of 5 values representing the force applied to each finger, captured through monitoring CyberGrasp tendons.

3. DATA ACQUISITION

A major challenge in haptic data acquisition is deciding how often to acquire the sensory values. A naive approach may involve sampling the sensor values as fast as the acquisition system (both hardware and software) can operate. The intuition is that the more samples we collect, the higher the accuracy. But due to device limitations and the nature of human motion, the value of a sensor might not change as fast as the system samples. Hence, the sampling rate might be much higher than necessary, which will result in wasting storage space and bandwidth. Identifying a lower sampling rate will enable haptic data acquisition to scale-up to many concurrent users and sensors, thus enabling integration with low bandwidth network environments. Acquiring haptic data becomes even more complicated if we consider that the optimal sampling rate also depends on both the sensor being sampled and the immersive session being recorded.

3.1. Sampling Methodology

Our sampling methodology is based on Nyquist sampling techniques[1]. Based on the Nyquist theory, a signal should be sampled with a rate twice as fast as its maximum frequency f_{max} in order to be able to fully reconstruct it. Because f_{max} is usually a large number for random signals, which leads to high sampling rates, a more practical approach is to introduce a cut-off frequency, $f_{cut-off}$, where the signal has a negligible amount of energy (e.g., 1%) for frequencies greater than that, and use the $f_{cut-off}$ instead of f_{max}. A discrete signal with rate r can be down-sampled with the ratio of $r/f_{cut-off}$, which produces a *reduced rate* signal with rate $f_{cut-off}$. The reduced rate signal should be up-sampled with the ratio $f_{cut-off}/r$ to produce a *reconstructed signal* with rate r.

3.2. Fixed Sampling (FS)

We define *fixed sampling* as the approach that a constant sampling rate is applied to all haptic device sensors. One approach is to use the maximum sampling rate, r_{max}, allowable by the SDK (as a function of the CPU speed). It is easy to implement, but it is also the most ineffective sampling technique described because it records data for each sensor at every opportunity, regardless of sensor type and the semantics of the session. A more efficient form of fixed sampling, *modified fixed sampling* (MFS), involves finding the minimum sampling rate r_0 required by all haptic device sensors, which is then used as the sampling rate. There is one disadvantage to this approach: we need to identify r_0 before we can start sampling at that rate. This means that we must have *training sessions* for the haptic device we are interested in recording. Training sessions may cause over-sampling or under-sampling due to the possible differences between real and training sessions.

[1]The Nyquist Theorem is a fundamental signal processing technique described by [3] and [8].

3.3. Grouped Sampling (GS)

We introduce the *grouped sampling* approach, which takes the modified fixed sampling approach one step further and identifies the minimum sampling rate for different groups of sensors on an immersidata device. The intuition here is that haptic devices have several sensors, and in many cases these sensors can be mapped to distinct groups. Thus, we can isolate the sampling rates for each group and acquire data at different rates, based on group membership. The advantage here is an improvement over the basic modified fixed sampling by reducing storage and bandwidth requirements further while maintaining accuracy. However, it still retains the disadvantage of requiring training sessions in order to identify sampling rates for each group. Using human intelligence, we can easily distinguish between finger force sensors and tracker sensors and understand why they are different. However, for an arbitrary set of immersidata sensors, it may not be easy to identify these groups. In addition, our own intuition about natural groups in some cases may not be correct or meaningful. With these issues in mind, we consider automating grouped sampling as an approach to perform standard cluster analysis techniques over the set of optimal rates, obtained during the training session for each sensor, and form groups based on those results.

3.4. Adaptive Sampling (AS)

We define *adaptive sampling* as a dynamic form of sampling that is applied to each sensor. With adaptive sampling, an optimum rate r_{ij} is dynamically identified for each sensor i during a given window j of the session. This approach uses the sampling methodology described in Section 3.1, identifies the Nyquist sampling rates r_{ij} for each sensor periodically, down-samples, transmits, and stores sampled data at those rates. Adaptive sampling could potentially reduce bandwidth and storage requirements to far lower levels than what fixed or grouped sampling did, enabling much more sensors to participate in cooperative immersipresence. Furthermore, unlike grouped sampling, a training session does not determine the lifelong sampling rate of a particular sensor group. Rather, the sampling rates change as the nature of the sessions change. This makes the approach more effective than grouped and modified fixed sampling. The disadvantage here is the delay (specifically the length of the sampling window) introduced by this process. To do efficient haptic data acquisition, it is useful to buffer a fairly large amount of data before performing calculations on that data towards identifying Nyquist rates. This is mainly because of the random nature of the data. The buffering process means that some degree of the real-time acquisition is sacrificed.

3.5. Adaptive DPCM(ADPCM)

The intuition in DPCM is that the consecutive samples taken from a haptic device are very similar and the differences between them

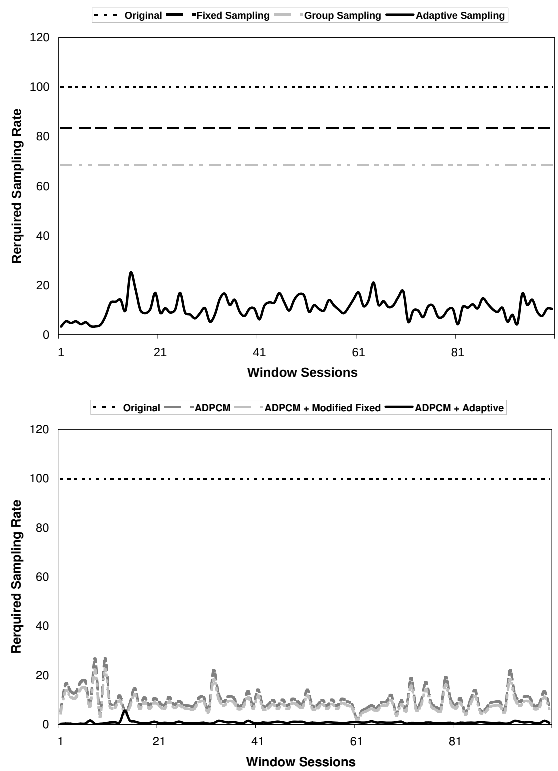

Figure 1: Reduction in required sampling rate for different sampling approaches.

is usually very close to zero. Hence, *predictive coding* [2] techniques can be used to compress the samples of the haptic devices. The basic principle of predictive coding is to use previously quantized samples to generate a predicted values for the current sample. This follows by quantizing the difference between the predicted and the actual values of the current sample. During a slow motion period, most consecutive samples will be similar, whereas the difference will be larger when the motion increases. This motives us to use an Adaptive DPCM (ADPCM) [4] scheme, where the quantization step-size is increased during periods of fast motion and decreased in period of slow motion. Large quantization step-size can accommodate fast changes, which would lead to inefficiency in slow motion situations. The adaptation mechanism works as follows. A circular buffer stores the most recent quantized differences. If the maximum value in this buffer is larger than a threshold, that means high motion is being observed and the step-size is increased by a factor; otherwise, the step-size is decreased by another factor. These factors and thresholds are characterized by a *max-level value*. The basic trend is that the larger this value, the less the compression and error distortion. The ADPCM approach is orthogonal to Fixed, Group and Adaptive Sampling techniques and can be combined with them.

4. PERFORMANCE EVALUATION

We conducted several experiments to identify the optimal sampling rates for MFS and GS for CyberGlove device; and to compare the required storage, bandwidth and the accuracy of the three proposed sampling techniques and their combinations with ADPCM approach. We recorded a total of 100 window sessions: 10 people genarting 10 color signs using American Sign Language (ASL) [6]. The duration of each session was 2 to 3 seconds. The original sampling rate was 100 Hz with memory requirement of approximately 30 to 60 KB for the input data and the bandwidth requirement of 180 Kb/s. Note that this leads to a 2 MB/s database population rate for 100 concurrent sessions.

Our experimental process was: Step 1, we grouped CyberGlove's sensores to 6 different groups, found the minimum sampling rate r_0 (as described in Section 3.2) for MFS and GS, resampled the original signals, and computed the reduced storage and bandwidth and the average error distortion. Step 2, for each window session *i* and sensor *j*, we applied the sampling methodology described in Section 3.1 and fixed CT=99% to compute the sampling rate r_{ij}. Subsequently, for each window session *i*, we averaged the error distortion for all sensors to compute the accuracy of that window session. The reduced storage for window session *i* is also computed by using sampling rate r_{ij}. Step 3, we directly applied adaptive DPCM methodology to the original signals. And finally, we investigated the effect of combining ADPCM with MFS and AS on bandwidth requirement and distortion.

4.1. Results

Figure 1 depicts the required sampling rates for different sampling techniques, when CT is 99% and the maximum level value for ADPCM is 39.

Even though FS is the easiest approach to implement, the figure shows that it is also the most ineffective method, since the sampling rates are usually unnecessarily high. MFS is an optimized fixed sampling, with a $(\frac{r_{max}-r_0}{r_{max}})$ (17% in our experiment) reduction in required sampling rate (and bandwidth/storage). GS is a more intelligent form of MFS, since it targets groups of related sensors. Our results show that GS led to sampling rate reductions of $(\frac{\sum_{i=1}^{m}(r_{max}-r_{g_i})}{m.r_{max}})$ (35% overall in our setup), where *m* is number of groups. AS, which continuously identifies optimal sampling rates for each sensor on a haptic device, was very efficient and substantially reduced the rate by $(\frac{\sum_{i=1}^{n}(r_{max}-r_{ij})}{n.r_{max}})$ for session *j* (75% to 95% in our scenarios), where *n* is the total number of sensors participated in experiments. However, it is also the most complicated one among all the sampling techniques because it requires concurrent, periodic analysis of each sensor output for a given window session before the optimal rate for that sensor can be identified. ADPCM approach also reduced the sampling rate by a factor of 74% to 98%. The most efficient approach is when ADPCM is combined with AS, that reduces the sampling rate by factor of 94%-99.9%.

We quantified the accuracy by using the *error distortion* function (Equation 1) applied to the original and reconstructed signals.

$$ErrorDistortion(x,y) = \sqrt{\frac{\sum_{i=1}^{N}(x_i - y_i)^2}{N}} \quad (1)$$

where x_0 ... x_N and y_0 ... y_N are the values of the original and reconstructed signals. The optimal value for the error distortion function is *0*, when $x_i = y_i$. Figure 2 shows the error distortion value for ADPCM and ADPCM+AS for one of the sessions. The average error distortion for all sensors is less than 0.5, which shows

[2] *Predictive coding* is a well known technique in coding of speech and audio.

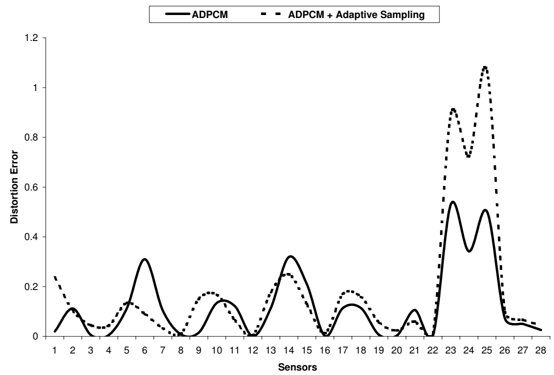

Figure 2: Error distortion.

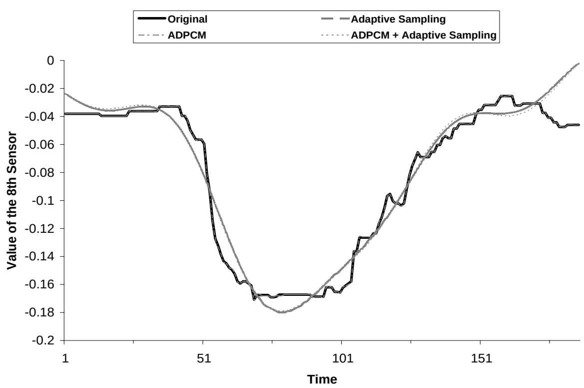

Figure 3: Comparison of the quality of the original vs. reconstructed signals

a high accuracy. The error distortion for other approaches is even less, between 0 to 0.2, which shows the less the compression provided by a technique, the less the distortion. Figure 3 shows the original and reconstructed signals for the 8th sensor of the haptic device when AS, ADPCM and ADPCM+AS approaches were used to sample the original signal.

5. CONCLUSION AND FUTURE WORK

We identified and introduced several methods for real-time compression of haptic data, and compared the data storage/bandwidth requirements and accuracy of them. A combination of ADPCM and AS approaches shows substantial reduction in storage and bandwidth requirements while maintaining a low error distortion. A proper selection of the configurable parameters, e.g., *max-level value* for ADPCM, *window-session size* and CT, is dependent on the type of the haptic device and application requirements, which needs further investigation. We plan to continue our work to extract semantic information from the sampled haptic data (e.g., recognizing formation of fist in a long session). The implication is that semantic information can reduce the amount of data required to describe a session, thus enabling us to more efficiently transmit and store haptic sessions (see [2] for primary attempts). Overall, our continual goal is to achieve a better level of manageability for immersidata such as haptic data. In the future, these environments will likely consist of many sensors and many participants, placing even higher demands on data compression.

6. REFERENCES

[1] W. Aviles and J. Ranta. “A Brief Presentation on the VRDTS - Virtual Reality Dental Training System”, *In Proc. Fourth PHANToM Users Group Workshop, MIT,* 1999.

[2] J. Eisenstein, S. Ghandeharizadeh, C. Shahabi, G. Shanbhag, and R. Zimmermann. “Alternative Representations and Abstractions for Moving Sensores Databases”, *in International Symposium on Intelligent Multimedia, Video and Speech Processing,* Hong Kong, 2001.

[3] H. Nyquist“Certain Factors Affecting Telegraph Speed”, *Bell System Technical Journal* April 1924.

[4] A. Ortega, Y. Lin, and M. Mclaughlin et al. “Lossy Compression of Haptic Data, in Toush in Virtual environments”, *IMSC Press, Prentice Hall,* 2001.

[5] C. Shahabi, G. Barsih, B. Ellenberger, N. Jiang, M. Kolahdouzan, A. Nam, and R. Zimmermann. Immersidara Management: Challenges in Management of Data Generated within an Immersive Environment. In *Proceedinds of the International Workshop on Multimedia Information Systems*, October 1999.

[6] C. Shahabi, L. Kaghazian, S. Mehta, A. Ghoting, G. Shanbhag, and M. McLaughlin. “Analysis of Haptic Data for Sign Language Recognition.”, *in the International conference on Universal Access in Human-Computer Interaction,* August 2001

[7] C. Shahabi, M. Kolahdouzan, G. Barsih, R. Zimmermann, D. Yao, K. Fu, and L. Zhang. “Alternative Techniques for the Efficient Acquistion of the Haptic Data”, *in ACM SIGMETRICS/Performance 2001,* Cambridge, August 2001

[8] C.E. Shannon and W. Weaver“The Mathematical Theory of Communication”, *The University of Illinois Press, Urbana, Illinois,* 1948.

[9] C. Sjostrom. “The Phantasticon: The PHANToM for Disabled Children”, *Center of Rehabilitation Engineering Research, Lund University.* 1997 URL: http://www.certec.lth.se

[10] S.S. Snibbe, Anderson and B. Verplank. “Springs and Constraints for 3D Drawing”, *In Proc. PHANToM Users Group, PUG98* 1998

[11] Hong Z. Tan. “Haptic Interfaces”, *Communications of the ACM* March 2000.