MOCAP DATA CODING WITH UNRESTRICTED QUANTIZATION AND RATE CONTROL

Choong-Hoon Kwak and Ivan V. Bajić *

School of Engineering Science, Simon Fraser University, Burnaby, BC, Canada

ABSTRACT

Motion Capture (MoCap) technology is becoming increasingly popular in gaming, entertainment, and multimedia industries. Interactive systems using MoCap technology require low-delay MoCap data compression. In this paper, we extend previous work on low-delay MoCap compression by introducing several useful features, such as unrestricted quantization, more efficient entropy coding, as well as encoder rate control. Experimental results show that the proposed rate control provides better than 99% accuracy in controlling encoder's output bitrate. At the same time, improvements in quantization and entropy coding provide over 20% reduction in bit rate for the same reconstruction quality, compared to the current state of the art.

Index Terms— Motion capture, low-delay compression, quantization, rate control

1. INTRODUCTION

Recently, Motion Capture (MoCap) has been playing an important role in a variety of fields, such as performing arts [1], entertainment [2], automotive industry [3], and rehabilitation [4]. Some of these applications involve interactive motion capture. For example, in [5], a system is proposed that enables the actor to control virtual characters in real time, possibly allowing their interaction with remote audience. Such applications face many challenges in view of limited resources and communication channel impairments, especially if the parties involved utilize wireless devices.

Early MoCap data compression methods were targeted at storage applications, and did not have low-delay requirements like the more recent interactive applications. As such, most of them employed temporal transforms to reduce temporal redundancy. For example, Karni and Gotsman [6] used the Principal Component Analysis (PCA) over the entire MoCap sequence. Liu and McMillan [7] proposed a piecewise PCA over shorter segments of the MoCap sequence. Significant change in motion determined segment boundaries. PCA was applied to each segment, combined with spline interpolation for improved performance. In [8], Principal Geodesic Analysis (PGA) was proposed as an extension of PCA, and applied to articulated human skeleton motion.

*This work was supported in part by NSERC grant SPG/430592-2012.

In addition to PCA, other transforms, such as wavelets and Discrete Cosine Transform (DCT), were considered in MoCap compression. In [9] and [10], wavelet transform is applied to each joint temporally. Preda *et al*. in [11] and [12] proposed two methods for MPEG-4 Bone Based Animation (BBA): one relying on temporal differencing, the other based on applying DCT temporally across 16 consecutive frames.

A state of the art method for low-delay MoCap coding was presented in [13]. This method employs a "hybrid" coding loop borrowed from video coding, where Differential Pulse Code Modulation (DPCM) is used to reduce temporal redundancy, while a spatial transform reduces spatial redundancy. Unlike many of the above methods, this method does not rely on human skeletal constraints, which makes it applicable to arbitrary (not just human) MoCap data. Further, it has zero encoding delay, meaning that the new data can be encoded immediately after they are captured. This is invaluable for interactive MoCap applications over a network; an error concealment method for the data encoded by this coder was developed in [14]. It was shown in [13] that this coder offered better compression performance at zero encoding delay compared to the methods in [6, 7, 11, 12].

In the present paper we improve the coder from [13] in several ways. First, we remove some restrictions on quantization, allowing arbitrarily large input samples to be encoded without overload distortion. Second, we improve entropy coding by introducing adaptive Golomb-Rice coding. Finally, we develop a rate control mechanism to provide control over the encoder's output bitrate, which minimizes the chance of buffer overflow or underflow in an end-to-end communication system. The paper is organized as follows. Starting with a brief review of the coder from [13], we describe unconstrained quantization and improved entropy coding in Section 2. Statistical analysis of data frames and the proposed rate control mechanism are presented in Section 3. Experimental results are presented in Section 4, followed by conclusions in Section 5.

2. IMPROVEMENTS IN CODING EFFICIENCY

2.1. Hybrid MoCap coding

Fig. 1 shows a block diagram of the hybrid low-delay MoCap encoder from [13]. The input to the encoder is a sequence

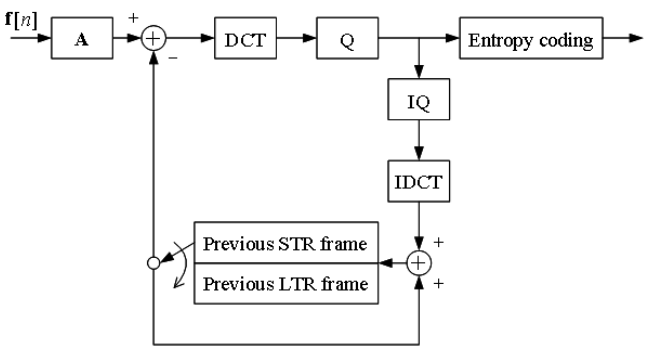

Fig. 1: Block diagram of the MoCap encoder from [13].

MoCap data frames $\mathbf{f}[n]$ consisting of (x, y, z) coordinates of motion markers. Data are linearly reordered via matrix $\mathbf{A}$ before entering the DPCM loop. In the loop, two types of frames are distinguished: Short Term Reference (STR) frames, which are predicted from the immediately preceding frame, and Long Term Reference (LTR) frames, which are predicted from the previous LTR frame. The spacing between LTR frames can be chosen in a variety of ways, possibly adaptively, but in [13] it was fixed at 30. The set of frames consisting of an LTR frame and subsequent STR frames until the next LTR frame is called a Group of Frames (GOF). Prediction residuals are transformed using DCT, quantized using a uniform midtread quantizer, and entropy coded using a version of adaptive arithmetic coding called range coding [15].

2.2. Unrestricted quantization

In order to define the symbol frequency table for range coding, the range of quantizer symbols should be known in advance. To fulfill this requirement, in [13], the quantization input range was limited, so that the number of possible symbols was simply equal to the ratio of this range and the quantizer step size. This meant that input values outside of this pre-set quantization range would have been quantized to one of the two outermost bins, and potentially suffer large quantization error known as overload distortion [16]. While this occurs infrequently enough not to affect average distortion, in practice, instantaneous large distortion is also undesirable.

To overcome this problem, in this work we replace the quantizer from [13] by an unrestricted midtread quantizer whose input range covers the entire real line. An input sample x is quantized to a signed integer index q given by

$$q = sign(x)\,\lfloor |x|/\Delta + 0.5 \rfloor\,, \tag{1}$$

where Δ is the quantizer step size. Dequantization is performed as $\widehat{x} = q\Delta$. While unrestricted quantization eliminates overload distortion, it also results in an unbounded range of quantizer symbols, which necessitates modifications to the entropy coder.

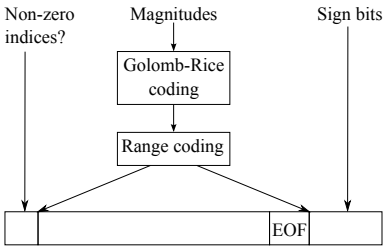

Fig. 2: Structure of the encoded bitstream.

2.3. Improved entropy coding

In order to handle the unbounded range of quantizer indices, we employ Golomb-Rice coding [17]. Given a quantizer index q, its magnitude $|q|$ is divided by the number $m = 2^k$, and represented as $|q| = mn + r$. The integer quotient n is encoded by a unary code (n ones followed by a zero), while the remainder $r \in \{0, 1, ..., 2^{k-1}\}$ is encoded by a fixed-length k-bit codeword. The Golomb-Rice bitstream of all quantized DCT coefficient magnitudes in the frame is passed to a ternary range coder, which accepts three symbols - binary symbols from the Golomb-Rice encoder plus an End-of-Frame (EOF) symbol. Sign bits of DCT coefficients are stored uncoded.

The overall structure of the bitstream for one MoCap frame is shown in Fig. 2. The first bit indicates whether there are any non-zero coefficients in the frame. If this bit is 0, which happens occasionally at low bitrates and/or when there is no motion in the sequence, the remainder of the bitstream for that frame is skipped; at the decoder, all DCT coefficients are decoded as zero. Otherwise, if the initial bit is 1, the range-coded output of the Golomb-Rice coder follows. This part of the bitstream encodes DCT coefficient magnitudes starting from DC towards high frequencies. In the C3D MoCap data format with 41 markers [13], 123 DCT coefficients are produced in each frame. However, at low bitrates and/or slow motion, many of the trailing high-frequency coefficients are zero. To gain efficiency, an EOF symbol is encoded after the last non-zero coefficient, and the remaining zero coefficients are skipped. At the decoder, upon decoding the EOF symbol, all remaining DCT coefficients are set to zero. Finally, after EOF, sign bits for non-zero coefficients are stored in the same order as their corresponding coefficients.

Coding parameters are adapted throughout encoding. When encoding the current frame, parameter k (used in Golomb-Rice coding for the divisor $m = 2^k$) is set as [18]

$$k = \max\left\{0, \lceil \log_2\left(0.5 \cdot \mathrm{avg}(\widehat{\mathbf{x}_r})\right)\rceil\right\}, \tag{2}$$

where $\widehat{\mathbf{x}_r}$ is the vector of decoded DCT coefficients in the reference frame. Values of k do not need to be encoded, since the decoder has access to $\widehat{\mathbf{x}_r}$ and is able to compute k as in (2). Range coding symbol frequencies are set to uniform at the start of encoding, and are updated thereafter. Since the number of range coding symbols is small (i.e., 3), their frequencies update quickly.

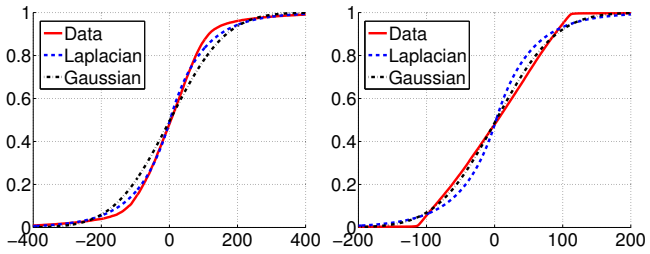

Fig. 3: CDF of LTR (left) and STR (right) residuals.

Sequence	LTR residuals		STR residuals	
	Gaussian	Laplacian	Gaussian	Laplacian
13_29	0.1113	**0.0654**	**0.0704**	0.0821
85_12	0.1192	**0.0598**	**0.0660**	0.0690
86_02	0.1069	**0.0782**	**0.1053**	0.1077
86_08	0.0811	**0.0617**	**0.0684**	0.0897

Table 1: KS statistics for transformed LTR and STR residual data fitted to Gaussian and Laplacian distributions.

3. RATE CONTROL

3.1. Modeling LTR and STR frames

Fig. 3 shows the cumulative distribution function (CDF) of transformed LTR and STR residuals for one popular MoCap sequence 13_29 from [19], along with Gausian and Laplacian CDFs fitted to the data. From the figure, Laplacian distribution seems a better model for LTR residuals, whereas Gaussian seems a better model for STR residuals. This is confirmed in Table 1, where we show Kolmogorov-Smirnov (KS) statistic [20] for goodness-of-fit of data to the distribution, for four test sequences from [19] that we later use in the experiments. Lower statistic (indicating better fit) is shown in boldface in each case. It is not surprising that that Laplacian is a better model for LTR residual data, whereas Gaussian is a better model for STR residual data; this is because LTR frames are further away from each other, which leads to larger residuals and heavier distribution tail compared to STR data.

3.2. Low-rate quantization

In this work, we aim to cover both high-rate quantization [21], where the distribution of the data in each quantization bin is approximately uniform, as well as low-rate quantization, where this is not the case. We therefore use relationships between rate R and uniform midtread quantizer step size Δ from low-rate quantization theory, which converge to high-rate results as $\Delta \to 0$. For a Laplacian random variable, this relationship is given by [22]

$$R(\Delta) = H\left(\sqrt{\theta}\right) + \sqrt{\theta}\left(1 - \log_2\left(1-\theta\right)\right) - \sqrt[3]{\theta}\frac{\log_2\theta}{1-\theta}, \quad (3)$$

where $\theta = e^{-\lambda\Delta}$, $\lambda = \sqrt{2}/\sigma_r$, and $H(\cdot)$ is the binary entropy function. For a Gaussian random variable, an approximate relationship is [23]

$$\begin{aligned} R(\Delta) &\approx -\sum_{k=-1}^{1} p_k \log_2 p_k, \\ p_k &= \frac{1}{2}\left(\operatorname{erf}\left(\frac{a\,(2k+1)}{2}\right) - \operatorname{erf}\left(\frac{a\,(2k-1)}{2}\right)\right), \end{aligned} \quad (4)$$

where $a = \Delta/(\sqrt{2}\sigma_r)$ and $\operatorname{erf}(\cdot)$ is the error function. In (3) and (4), σ_r is the standard deviation of the corresponding random variable. In our codec, we set $\sigma_r = \max\{T_\sigma, \operatorname{std}(\widehat{\mathbf{x}_r})\}$, where, as before, $\widehat{\mathbf{x}_r}$ is the vector of decoded DCT coefficients in the reference frame and T_σ is a small threshold to avoid setting $\sigma_r = 0$ when there is no motion in the sequence. This allows the encoder to adapt to changing signal statistics, while no extra information needs to be transmitted because the decoder has access to $\widehat{\mathbf{x}_r}$ and is able to update its model accordingly.

3.3. Bit budget management

Let r_t be the target bitrate in bits per second, N_{GOF} be the number of frames in a GOF, and F be the frame rate in frames per second (fps). Then the target number of bits per GOF is given by $R^t_{\mathrm{GOF}} = r_t N_{\mathrm{GOF}}/F$. If N_c is the number of DCT coefficients per frame ($N_c = 123$ for C3D MoCap data with 41 markers), then there are $N_c N_{\mathrm{GOF}}$ coefficients in a GOF. Of these, N_c belong to the LTR frame, and $N_c(N_{\mathrm{GOF}} - 1)$ belong to STR frames. Let R^t_{LTR} and R^t_{STR} be the target bits per sample for LTR and STR frames, respectively. Then

$$R^t_{\mathrm{GOF}} = N_c R^t_{\mathrm{LTR}} + N_c(N_{\mathrm{GOF}} - 1) R^t_{\mathrm{STR}}. \quad (5)$$

We assign more bits to LTR frame coefficients, since each LTR frame affects all subsequent frames, while each STR frame only affects subsequent STR frames in the same GOF. Let $R^t_{\mathrm{LTR}}/R^t_{\mathrm{STR}} = \alpha > 1$. Using this in (5), we obtain

$$R^t_{\mathrm{STR}} = \frac{R^t_{\mathrm{GOF}}}{N_c(\alpha + N_{\mathrm{GOF}} - 1)}, \quad R^t_{\mathrm{LTR}} = \alpha R^t_{\mathrm{STR}}. \quad (6)$$

In both (3) and (4), $R(\Delta)$ is monotonically decreasing in Δ. Hence, solving $R(\Delta) = R^t_{\mathrm{LTR}}$ and $R(\Delta) = R^t_{\mathrm{STR}}$ for Δ amounts to a simple line search.

The rate control scheme is presented in Algorithm 1. In the first GOF, R^t_{GOF} is obtained as $R^t_{\mathrm{GOF}} = r_t N_{\mathrm{GOF}}/F$. Subsequently, R^t_{GOF} is assigned based on the target bitrate (r_t) and the actual number of bits spent. If there is an overshoot or undershoot in any GOF, the encoder tries to compensate for it in subsequent GOFs. The encoder obtains quantizer step size Δ from (3) or (4), depending on the type of the frame. Upon encoding each frame, the bit budget is updated. Target bitrate r_t is encoded in the header as an 8-byte floating point number, so that decoder can initialize the quantizers for the first GOF. Subsequently, the decoder tracks quantizer changes by running the same algorithm as the encoder.

Algorithm 1 Encoder rate control for each GOF

Input: R^t_{GOF} ▷ target bits for current GOF
Output: $R^t_{\mathrm{GOF,next}}$ ▷ target bits for next GOF
$R^a_{\mathrm{GOF}} \leftarrow 0$ ▷ actual bits in current GOF
Compute R^t_{LTR} from (6)
Solve $R(\Delta) = R^t_{\mathrm{LTR}}$ for Δ using (3)
Encode LTR frame; R^a_{LTR} is the number of bits spent
$R^a_{\mathrm{GOF}} \leftarrow R^a_{\mathrm{GOF}} + R^a_{\mathrm{LTR}}$
for $n = 2$ to N_{GOF} **do**
    $R^t_{\mathrm{STR}} \leftarrow \frac{R^t_{\mathrm{GOF}} - R^a_{\mathrm{GOF}}}{N_c(N_{\mathrm{GOF}} - n + 1)}$
    Solve $R(\Delta) = R^t_{\mathrm{STR}}$ for Δ using (4)
    Encode STR frame; R^a_{STR} is the number of bits spent
    $R^a_{\mathrm{GOF}} \leftarrow R^a_{\mathrm{GOF}} + R^a_{\mathrm{STR}}$
end for
return $R^t_{\mathrm{GOF,next}} = r_t N_{\mathrm{GOF}}/F + R^t_{\mathrm{GOF}} - R^a_{\mathrm{GOF}}$

4. EXPERIMENTAL RESULTS AND DISCUSSION

Four MoCap test sequences from [19], 13_29, 85_12, 86_02, 86_08, containing diverse human motion, were used in the experiments. We first examine the coding efficiency of the new approach compared to the reference method in [13]. Fig. 4 shows codec performance in terms of the Signal-to-Quantization-Noise Ratio (SQNR) [21] vs. bitrate on two of the test sequences (results for the other sequences were similar and omitted due to space constraints). REF is the benchmark codec in [13], MOD is the proposed codec without rate control, and RC is the proposed codec with rate control and bit allocation ratio $\alpha = 10$ (see Section 3.3). All coders use $N_{GOF} = 30$. MOD incorporates unrestricted quantization and improved entropy coding from Section 2, and uses a fixed quantizer step size for all frames.

As seen in Fig. 4, at lower bitrates, MOD provides over 20% bitrate reduction for the same quality compared to REF. RC provides additional gain due to higher bit allocation to LTR frames ($\alpha = 10$). However, the gain of RC relative to MOD reduces as the bitrate increases. This suggests that a fixed bit allocation strategy is not equally appropriate across different bitrates, and suggests possible future improvements by making the bit allocation adaptive.

Next, we turn to rate control assessment. We encoded each of the four test sequences with the target bitrates r_t (in kbps) given as $r_t \in \{3, 4, 5, 6, 7\}$. The percentage error between the actual bitrate and the target bitrate was measured in each case. The average error was 0.21%, while the maximum error was 0.67%. Hence, the proposed rate control is able to control the total bitrate fairly precisely, with accuraccy of over 99% (i.e., error less than 1%).

To illustrate instantaneous bitrate, in Fig. 5 we plot the actual number of bytes produced per GOF by MOD and RC coders at the target bitrate of 6 kbps for two sequences (other cases were similar and omitted due to space constraints). For

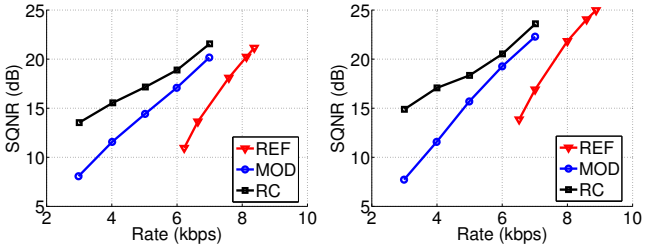

Fig. 4: SQNR vs. rate for 13_29 (left) and 86_08 (right).

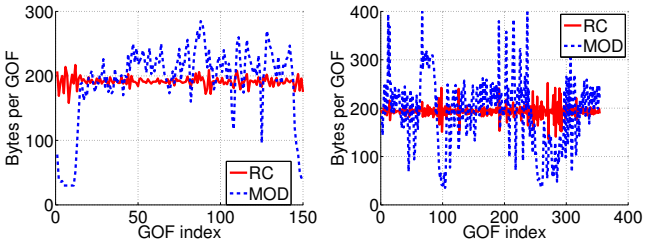

Fig. 5: Bytes per GOF with and without rate control at 6 kbps for 85_12 (left) and 86_02 (right).

this test, the quantizer step size in MOD was iteratively adjusted until near-target bitrate was reached. As seen in the figure, RC achieves much lower deviation from the target compared to MOD. This is further quantified in Table 2, where we show the average standard deviation of bytes per GOF produced by MOD and RC coders across the five target bitrates listed above. Based on Table 2, RC achieves 3-5 times lower standard deviation of bytes per GOF compared to MOD.

5. CONCLUSIONS

In this paper, we presented several improvements to a state-of-the-art low-delay hybrid MoCap encoder. Specifically, we incorporated unrestricted quantization and improved entropy coding, which eliminated instantaneous overload noise and improved the coding efficiency by over 20% at low bitrates. We also developed a rate control scheme to constrain instantaneous bitrate, based on low-rate quantization theory. By adapting quantizer step sizes according to the rate control algorithm, the encoder is able to reduce the fluctuation of bytes generated per GOF by 3-5 times. One possibility for improvement is adaptive bit allocation among LTR and STR frames.

Sequence	MOD	RC
13_29	42.5	12.5
85_12	48.0	8.8
86_02	60.3	10.9
86_08	51.5	9.3

Table 2: Average standard deviation of bytes per GOF.

6. REFERENCES

[1] A. Andreadis, A. Hemery, A. Antonakakis, G. Gourdoglou, P. Mauridis, D. Christopoulos, and J. N. Karigiannis, "Real-time motion capture technology on a live theatrical performance with computer generated scenery," in *Proc. IEEE PCI'10*, Tripolis, Greece, 2010, pp. 148–152.

[2] M. Z. Patoli, M. Gkion, P. Newbury, and M. White, "Real time online motion capture for entertainment applications," in *Proc. DIGITEL 2010*. IEEE, 2010, pp. 139–145.

[3] M. Jung, H. Cho, T. Roh, and K. Lee, "Integrated framework for vehicle interior design using digital human model," *J. Computer Science and Technology*, vol. 24, no. 6, pp. 1149–1161, 2009.

[4] S. Subramanian, L. A. Knaut, C. Beaudoin, B. J. McFadyen, A. G. Feldman, and M.F. Levin, "Virtual reality environments for post-stroke arm rehabilitation," *Journal of neuroengineering and rehabilitation*, vol. 4, no. 1, pp. 20, 2007.

[5] Q. Wu, P. Boulanger, M. Kazakevich, and R. Taylor, "A real-time performance system for virtual theater," in *Proc. SMVC 2010*. ACM, 2010, pp. 3–8.

[6] Z. Karni and C. Gotsman, "Compression of soft-body animation sequences," *Computers & Graphics*, vol. 28, no. 1, pp. 25–34, 2004.

[7] G. Liu and L. McMillan, "Uniform threshold scalar quantizer performance in Wyner-Ziv coding with memoryless, additive Laplacian correlation channel," in *Proc. SCA'06*, Sep. 2006, pp. 127–135.

[8] M. Tournier, X. Wu, N. Courty, E. Arnaud, and L. Reveret, "Motion compression using principal geodesics analysis," *Computer Graphics Forum*, vol. 28, no. 2, pp. 355–364, 2009.

[9] P. Beaudoin, P. Poulin, and M. van de Panne, "Adapting wavelet compression to human motion capture clips," in *Proc. Graphics Interface (GI'07)*, 2007, pp. 313–318.

[10] A. Firouzmanesh, I. Cheng, and A. Basu, "Perceptually guided fast compression of 3-D motion capture data," *IEEE Trans. Multimedia*, vol. 13, no. 4, pp. 829–834, 2011.

[11] M. Preda, B. Jovanova, I. Arsov, and F. Prêteux, "Optimized MPEG-4 animation encoder for motion capture data," in *Proc. Web 3D'07*. ACM, May 2007, pp. 181–190.

[12] B. Jovanova, M. Preda, and F. Preteux, "MPEG-4 part 25: A generic model for 3D graphics compression," in *Proc. 3DTV Conference (3DTV-CO)*. IEEE, 2008, pp. 101–104.

[13] C. H. Kwak and I. V. Bajić, "Hybrid low-delay compression of motion capture data," in *Proc. IEEE ICME'11*, 2011.

[14] C. H. Kwak and I. V. Bajić, "Error concealment strategies for motion capture data streaming," in *Proc. IEEE StreamComm - ICME'11*, 2011.

[15] M. Servais, *Range Coding in MATLAB*, [Online]. Available: http://www.ee.surrey.ac.uk/CVSSP/VMRG/hdtv/code.htm.

[16] N. S. Jayant and P. Noll, *Digital Coding of Waveforms*, Prentice Hall, 1984.

[17] N. Memon, "Adaptive coding of DCT coefficients by Golomb-Rice codes," in *Proc. IEEE ICIP'98*, Chicago, Illinois, 1998, vol. 1, pp. 516–520.

[18] D. S. Taubman and M. W. Marcellin, *JPEG2000: Image compression fundamentals, standards, and practice*, Kluwer Academic Publishers, 2002.

[19] *The CMU Motion of Body (MoBo) Database*, [Online]. Available: http://mocap.cs.cmu.edu.

[20] M. M. Desu and D. Raghavarao, *Nonparametric Statistical Methods for Complete and Censored Data*, Chapman and Hall/CRC, 2004.

[21] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*, Kluwer Academic Publishers, 1992.

[22] V. Sheinin, A. Jagmohan, and D. He, "Uniform threshold scalar quantizer performance in Wyner-Ziv coding with memoryless, additive Laplacian correlation channel," in *Proc. IEEE ICASSP'06*, Toulouse, France, May 2006, vol. 4, pp. 217–220.

[23] V. Sheinin and A. Jagmohan, "Low rate uniform scalar quantization of memoryless Gaussian sources," in *Proc. IEEE ICIP'06*, Atlanta, GA, 2006, pp. 793–796.