# Lossless Compression of Human Movement IMU Signals

David Chiasson[1], Junkai Xu[1], and Peter Shull[1]

[1]State Key Laboratory of Mechanical System and Vibration, School of Mechanical Engineering, Shanghai Jiao Tong University

***Abstract*—Real-time human movement inertial measurement unit (IMU) signals are central to many emerging medical and technological applications, yet few techniques have been proposed to process and represent this information modality in an efficient manner. In this paper, we explore methods for the lossless compression of human movement IMU data and compute compression ratios as compared with traditional representation formats on a public corpus of human movement IMU signals for walking, running, sitting, standing, and biking human movement activities. Delta coding was the highest performing compression method, which compressed walking, running and biking data approximately 10 times smaller and sitting and standing data 18 times smaller than the original CSV formats. Furthermore, delta encoding is shown to approach the a posteriori optimal linear compression level. All methods are implemented and released as open source C code using fixed point computation which can be integrated into a variety of computational platforms. These results could serve to inform and enable human movement data compression in a variety of emerging medical and technological applications.**

***Index Terms*—kinematic data, human movement, compression, codec**

## I. Introduction

MANY emerging technological and medical applications rely on real-time human movement inertial measurement unit (IMU) signals as a crucial component, including virtual reality [1], autonomous navigation [2], internet of things [3], activity monitoring [4] [5], physical therapy [6] [7], and human performance [8] [9] among others. These applications have been enabled by the recent explosion of inexpensive inertial based sensors (IMUs) along with mobile computation power to process this data in real time. The human movement IMU signal represents a nascent field of multimedia processing which is starkly under-developed compared to the existing maturity of text, audio, and visual type signal processing methods. This is evidenced by the lack of standards or tools for handling movement data. To enable these emerging applications, efficient and standard methods for representing and processing movement signals are needed. Compression can be a crucial component of this missing toolset as it improves situations of limited transmission bandwidth and limited storage space.

Compression seeks to represent information in a space efficient manner. This is generally done by exploiting spatio-temporal redundancy, correlation, and smoothness [10]. A lossless compression algorithm can be divided into two components, modeling and coding/decoding (Figure 1) [11]. The model incorporates prior understanding of the signal class to be compressed. It estimates a probability mass function which represents the likelihood of occurrence for each possible input symbols. A dynamic model is one which changes its probability estimates after new input symbols are received. The model is sometimes described as a transformation, which refers to some reversible operation which changes the signal into a lower entropy or easier to predict form. The Coder uses the probability mass function produced by the model to compute a unique variable length encoding for each possible symbol. Short encodings are assigned to likely symbols, and long encodings are assigned to unlikely symbols such that the average length of the compressed signal is minimized. The decompressor uses an identical model to provide the same probability mass function to the decoder which is used to revert each code back into the original symbol. If the model is dynamic, this output is used to update the model for future predictions.

The coding component is well understood. The theoretical limit of coding performance on a signal with a know probability distribution is given by Shannon's noiseless coding theorem [12] as first order entropy:

$$H = -\sum_i P_i log_2 P_i$$

While Shannon developed several efficient coding techniques, the first optimal technique was developed in 1952 by Huffman [13]. Huffman's technique was proven optimal, but it made the assumption that output codes must be an integer number of bits, which prohibited it from reaching Shannon's limit under certain conditions. Arithmetic coding was developed to address this deficiency, showing better performance particularly in high compression situations [14]. Another technique, Golomb coding and the Rice special case, has experienced significant adoption in industry because of the efficient binary implementation, and the ability to encode online without needing a preliminary pass through the data to compute the probability distribution [15] [16]. Because of the many efficient and optimal techniques available, some consider coding to be a solved problem [17].

Modeling on the other hand must be revisited for each new signal class and application. Due to the pigeon hole principal, no algorithm can compress every possible input [18]. Each model must make an implicit decision about the class and scope of signals that will be compressed. To the

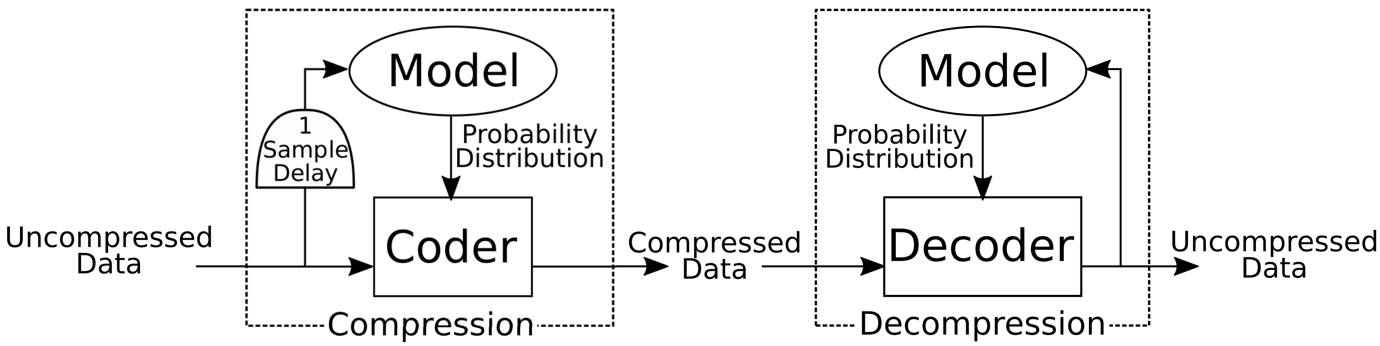


Fig. 1. Generalized compression and decompression. Modeling and coding are independent. The model is mathematically identical for compression and decompression. Coder and decoder perform inverse operations parameterized by the probability distribution.

authors' knowledge, no previous work has directly addressed the modeling problem for the human movement IMU signal. This provides the motivation for the current work.

This study explores a range of linear predictive models applied to the human movement signal as quantified by 6-axis IMUs. A corpus of representative human movement IMU signals was selected to demonstrate the compression performance of each model. To put the performance in context, several traditional representation formats are selected. The compression performance of these traditional formats provides a lower bound for the performance of a useful compression method. Finally, the optimal linear predictive models are computed numerically utilizing full knowledge of the signals in the corpus. The compression performance of these optimal models provide an upper bound on the performance of our proposed methods.

## II. Methods

The Human Gait Database (HuGaDB) [19] was selected as a corpus to meaningfully and repeatable demonstrate the performance of various compression methods. HuGaDb is a public dataset of six-axis IMU signals collected from six different body segments (right and left foot, right and left shin, right and left thigh) of 18 healthy subjects performing various movement activities (including walking, running, sitting, standing, and biking) sampled at 60Hz. This database was selected because it allows the comparison of compression methods for a variety of movement activities. Signals are sampled as 16 bit signed integers. The authors of HuGaDB indicate in their online repository that some of their collected data contains corrupted gyroscope signals. Such data has been excluded from this study. The remaining data is separated by subject, movement activity, trial, and body segment, to produce a total of 1626 separate test cases to compare the performance of proposed compression methods. Each test case contains six time series signals.

With the exception of traditional data representation formats, all compression methods in this work will be presented as predictive autoregressive linear models. The predictive model will estimate the current sample given past input. The difference between the model prediction and observed sample will be referred to as the residual signal. All compression methods will encode this residual signal using Golomb-Rice coding to produce the final compressed data. Golomb-Rice coding is computationally efficient on binary base computational platforms, and has been shown to approach optimal coding if the input signal is geometrically distributed [15] [16].

### A. *Proposed Compression Methods*

Several restrictions are placed on the methods considered in this study. First, a viable algorithm must be causal. This is a basic requirement for an algorithm to be implemented in a real-time application. We also chose to only consider algorithms with zero filter delay. Since our sensors operate at a relatively low sampling frequency of 60Hz, a delay of one sample would be 16ms which is significant for modern information networks.

All algorithms considered in this paper are node independent. The model for each signal considers at most the information from the six co-located signals including it's own past input. While it is likely that utilizing inter-node correlation could produce better compression ratios, especially if a human biomechanics model were introduced, such an approach would limit the usefulness of said algorithm to a specific placement of nodes on the body.

Only lossless compression methods are considered in this study. The reason for this is that designing a lossy compression method requires a well defined distortion criteria [10]. This criteria is a value judgment about what type and magnitude of distortion is acceptable for the compressed signal. Defining a distortion criteria for audio and visual signals, while not trivial, is certainly tractable as there is generally a single well-defined application of the signals, namely consumption by the human ear and eye [20] [21]. Other signal classes such as text and binary data have sensitive applications that cannot tolerate any error in the signal, and thus lossy compression methods are not considered. The human movement IMU signal on the other hand has an array of applications, each of which is likely to have different requirements for acceptable and unacceptable distortions of the movement signal. In the absence of a specific application, no justification can be made for discarding any portion of information.

In practice, the implementation of a strictly lossless algorithm turns out to be non-trivial. This is because the standard for floating point computation IEEE 754 [22] is not sufficiently stringent to guarantee identical results on various implementations [23] [24]. For example, rounding is permitted to differ slightly between two computation platforms, or the same computation platform at two points in time. Additionally, a compiler or interpreter which processes the source code for an algorithm implementation will often utilize mathematical properties such as commutativity to optimize computation. This may result in different machines performing floating point operations in a different order which could lead to differing results even if each rounding operation were well defined by IEEE 754. To avoid both of these scenarios and guarantee identical results across diverse computational platforms, all algorithms in this study were implemented using integer operations and fixed-point 16.16 precision. There is significant cost associated with this technical decision. Namely that quantization error is independent of magnitude. This can lead to significant numerical issues for some algorithms.

All compression methods in this study were implemented in the C programming language using fixed point 16.16 computation. The source code has been released at https://github.com/dchiasson/kinetic_codec. The choice of programming language as well as the restriction to use integer arithmetic allow this code to be incorporated into programs on a diverse array of computation platforms, even those without a floating-point unit. The hope of the authors is that this code can be a starting point for academic or industry developers implementing some human movement application.

The following lossless compression methods are proposed in this study:

- **Delta encoding** Current sample is predicted to be equivalent to the previous sample so that the difference between the two is encoded: $\hat{x}[n] = x[n-1]$. If a signal varies slowly with time, this signal will be smaller than the original signal. This method can be considered 0-order polynomial regression.
- **Linear extrapolation** Current sample is estimated as a linear extrapolation from previous samples. Also known as first order polynomial regression. Linear extrapolation from a regression of the past two samples results in the estimator: $\hat{x}[n] = 2x[n-a] - 1x[n-2]$
- **$2^{nd}$ to $5^{th}$ order polynomial regression** These methods assumes that the signal is a polynomial which is estimated from a least squares regression of past samples. This polynomial is then extended to get a prediction of the current sample.

$$\hat{x}[n] = \sum_{i=0}^{d} b_i (p+1)^i$$

  Where $d$ is the polynomial order and $p > d$ is the number of past samples included in the regression. The coefficients $\mathbf{b}$ may be found via least squares regression.

$$\underset{\mathbf{b}}{\text{minimize}} \quad \sum_{j=1}^{p} \left( x[n-j] - \sum_{i=0}^{d} b_i (p-j+1)^i \right)^2$$

  The resulting prediction $\hat{x}[n]$ is a linear combination of past samples or an autoregressive model of order $p$.
- **Spline extrapolation** A spline is the minimum curvature piece-wise polynomial which connects a set of points. It is commonly used for interpolation, namely computer graphics smoothing. This method was selected as splines are known to avoid Runge's phenomenon which is witnessed when extrapolating higher order polynomials. Results from the cubic spline with natural boundary conditions are presented in this paper. The cubic spline is a piecewise cubic function

$$f_i(n) = a_i n^3 + b_i n^2 + c_i n + d_i \quad n \in (n_i, n_{i+1})$$

  which is smooth

$$\frac{d}{dn} f_i(n_{i+1}) = \frac{d}{dn} f_{i+1}(n_{i+1})$$

  and passes through the set of $p$ past samples.

$$f(n) = x[n]$$

  Choosing natural boundary conditions,

$$\frac{d^2}{dn^2} f_1(n_1) = \frac{d^2}{dn^2} f_{p-1}(n_p) = 0$$

  we can extrapolate the spline to predict the next sample.

$$\hat{x}[n_{p+1}] = f_{p-1}[n_{p+1}]$$

  Like polynomial regression, spline extrapolation is also an auto-regressive linear model.

## B. Traditional Representation Formats

To provide a lower bound for the useful performance of compression methods, several traditional data representation formats are chosen as reference. In this study, the baseline data format (exhibiting a compression ratio of one) is comma separated value (CSV), a simple text based format which is the de-facto standard for storing sensor information. Performance of each compression method is assessed by computing the compression ratio CR relative to CSV via the following formula:

$$CR = \frac{\text{size of CSV file}}{\text{size of compressed file}}$$

In total, the following three traditional data representation formats were chosen to provide context for our results:

- **CSV** Text based format considered the de facto standard. CSV files are ANSI encoded and formatted to have a constant length sample format to eliminate a source of randomness in our CR computation. Due to this decision, Binary format will have the same CR regardless of data properties.
- **Binary** The optimal fixed size format. In our corpus, every sample is two bytes. This would be the optimal compression if each sample were an IID random variable uniformly distributed across the sample space.
- **ZIP compression of CSV** ZIP is a general purpose file compression format integrated into all major computer systems. ZIP was executed using the DEFLATE method [25] and a compression level of 6 (Linux command: *zip -6 –compression-method deflate*).

### C. Optimal Linear Compression

To complete the context for our proposed compression methods and provide an upper limit on compression performance, we numerically compute the optimal linear predictive model for our data. To do so, we will formulate our model as an auto-regressive process of order $p$ and define the prediction error or residual signal as:

$$e[n] = x[n] - \sum_{k=1}^{p} a_k x[n-k] \tag{1}$$

where $a_k$ is the linear contribution of sample $k$ in the past to our current prediction. Equation (1) can expressed be in concise matrix notation as:

$$\mathbf{e} = \mathbf{x} - \mathbf{X}\mathbf{a}$$

For our application, we are interested in the residual signal $\mathbf{e}$ which can be encoded into the minimum number of bits. To compute this, we consider the effect of Rice-Golomb coding on our residual signal. A Rice-Golomb encoding first splits each residual $e[n]$ into a quotient and remainder portion:

$$q[n] = \lfloor \frac{e[n]}{2^m} \rfloor \quad \text{and} \quad r[n] = e[n] - q[n] * 2^m$$

$\lfloor \cdot \rfloor$ denotes the floor operation and $m \in \mathbb{N}_0$ is the Rice-Golomb order which is selected based on signal statistics [26] [27]. The remainder $r[n]$ is truncated binary encoded at a fixed size of $m$ bytes, while the quotient $q[n]$ is unary encoded, requiring $q[n] + 1$ bits. The size in bits of each Rice-Golomb encoded element of the residual signal is thus:

$$m + \lfloor \frac{e[n]}{2^m} \rfloor + 1$$

If we relax our rounding operation, the size of each element can be approximated as a affine function of $e[n]$. The total compressed size for a signal of length $l$ has an approximate size:

$$l + lm + 2^{-m} \sum_{n=0}^{l} e[n]$$

Minimizing this value with $l$-1 normalization is equivalent to the optimization problem:

$$\underset{\mathbf{a}}{\text{minimize}} \quad \|\mathbf{x} - \mathbf{X}\mathbf{a}\|_1 + \lambda \|\mathbf{a}\|_1 \tag{2}$$

The key takeaway from this result, is that compression is proportional to the total *absolute prediction error* of the model, and not the *squared error*. $l$-1 normalization is used since it encourages sparsity in the model. Sparsity is desirable in this application as it reduces quantization error of the model coefficients and reduces the fixed point arithmetic error during execution. Since this problem is convex, it can readily be solved with a variety of numerical solvers. For this study, python bindings for the Splitting Conic Solver were used [28] [29] [30].

If problem (2) is solved considering past history of each stream, then the model for each axis of accelerometer and gyroscope can be computed independently. This model will be referred to as the optimal auto-regressive model (AR). However, if we also take into account the past history of other axes and sensors, then the model can account for any cross-correlation which may result from the interrelated nature of rotation and orientation information. The residual signal (1) can be rewritten using this expanded model for stream $i$ as:

$$e_i[n] = x_i[n] - \sum_{j=1}^{s} \sum_{k=1}^{p} a_{i,j,k} x_j[n-k]$$

where $a_{i,j,k}$ is now the linear contribution of the sample $k$ in the past of stream $j$ to our current prediction of stream $i$. Solving problem (2) with this expanded system will be referred to as the optimal multivariate autoregressive model (MVAR). Note that both AR and optimal MVAR models are non-causal and expensive to compute, and are thus disqualified as a proposed method. Instead they serve as upper-limit reference points for the evaluation of proposed methods.

### D. Data Analysis

To determine the statistical significance of our results, we follow the recommendations of [31]. To compare the CRs of our proposed compression methods, we begin with the Friedman Test [32] which is a non-parametric test for significant differences between algorithm performance which considers the ranks of each algorithm's performance per test case. A non-parametric test is necessary since our corpus does not meet the normal distribution assumption of many parametric tests. Figure 3 confirms that our data is indeed bi-modal. If the null-hypothesis is rejected, we proceed with the Nemenyi post-hoc test [33] to determine which pairs of methods differ significantly (at $p = .05$). Statistical analysis of performance is presented for proposed method results over the entire corpus.

Additionally, we wish to determine if a proposed method has significantly different performance on different classes of data. The Friedman test is not appropriate for this scenario since we are comparing the performance of one algorithm across different groups of test cases, and rank is undefined. However, test cases that are in the same class are more likely to follow a normal distribution, so the parametric ANOVA test is used [34]. If the null-hypothesis is rejected, we proceed with the Tukey HSD post-hoc test [35] to determine which pairs of data classes experience significantly different compression. This statistical analysis is presented for the highest performing proposed method.

## III. Results

All proposed compression methods outperformed all traditional methods in size efficiency for every data class and nearly every test case (Table I). Figure 4 shows delta encoding consistently ($p < 0.001$) achieved the highest compression of the proposed methods (CR=12.75), and each higher degree polynomial performed progressively worse with 5th degree polynomial consistently ($p < 0.001$) the worst performing (CR=11.25). The spline method was not found to be significantly different than linear or 2nd degree polynomial ($p > .5$). Significant differences were also not found between 3rd and

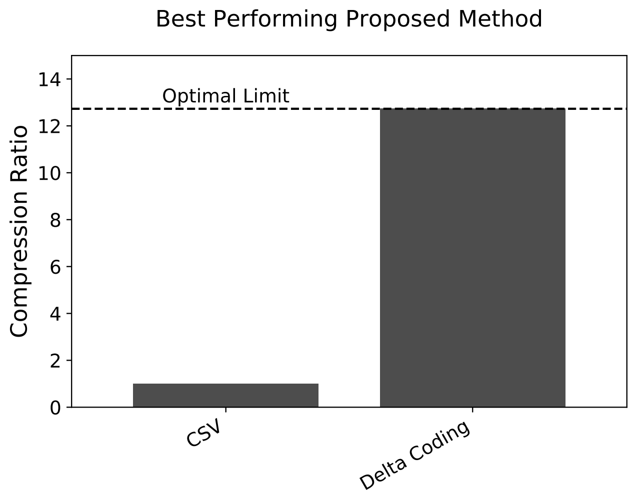


Fig. 2. Mean compression ratio (original size / compressed size) of the best performing proposed technique, delta encoding, compared with traditional CSV. Delta encoding approaches the optimal auto-regressive compression level.

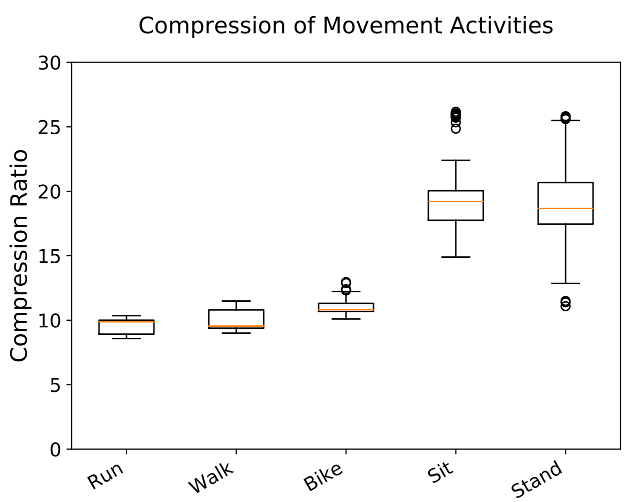


Fig. 3. Distribution of compression ratio (original size / compressed size) of each movement activity using delta encoding. Activities with less movement intensity experience greater compression.

4th degree polynomials ($p = .25$) or between 2nd and 3rd degree polynomials ($p = .19$) The CR of delta encoding approached optimal AR and MVAR model compression for all data class.

Low movement activities such as sitting (CR=17.87) and standing (CR=18.69) were compressed significantly more ($p < 0.001$) than high movement activities running (CR=9.52) walking (CR=9.95) and biking (CR=10.96). Significant differences were not found within the low movement activity group ($p = .9$) or the high movement activity group ($p > .13$). (Figure 3). Body segments did not show significant variation in CR ($p > .05$).

## IV. Discussion

These results show that many methods result in a significant compression over the traditional representation formats of CSV, ZIP, and binary. Using any of the proposed methods for a human movement IMU application represented in our corpus will likely result in significant space improvement.

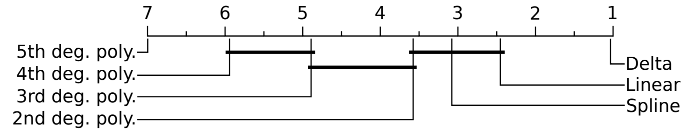


Fig. 4. Mean ranks of proposed methods compared using the Nemenyi test. Groups of methods that are not significantly different (at $p = 0.05$) are connected.

Delta encoding mathematically dominated all proposed compression methods by providing greater compression for nearly all test-cases (CR=12.75, $p < 0.001$) while having lower computation requirements. This is not entirely surprising as delta encoding as often chosen as the lossless compression model in other domains [36] [37]. Each increasing order of polynomial regression produced worse compression (CR=12.22, 12.0, 11.8, 11.57, 11.25). This is likely because higher degree polynomial predictors suffer from poor white noise attenuation [38] causing an effect known as Runge's phenomenon. Higher degree polynomial predictors are seen to require disproportionately higher computation as a regression over more past samples were found to lead to better predictions. This is in contrast to zero-degree polynomial regression, where all orders of simple moving averages were found to under-perform delta encoding. Only the best performing predictor of each type is presented in the results.

On several test cases, delta encoding slightly exceeds the compression of the optimal reference models. In theory, the optimal linear models should perform as well or better than delta encoding, since delta encoding is within the class of models that were optimized over in equation (2) . The occasional inferior performance of the optimal linear models might be due to quantization error of the model coefficients and increased fixed point error from computational complexity. This explanation is supported by the observation that the MVAR model, which has many more coefficients, often underperforms relative to the AR model while the opposite would be true in the absence of numeric error. Investigation of the optimal models' coefficients showed that they were similar to those of delta encoding.

The compression level achieved on each movement activity varied significantly ($p < .001$) and separated into two distinct groups. The first group consiting of running, walking, and biking (high intensity movement, CR$\approx$10) achieving much less compression than the second group sitting and standing (low intensity movement, CR$\approx$10) (Figure 3). This matches our expectations as higher intensity of movement will intuitively have more information content.

The MVAR optimal model and the AR optimal model achieved similar performance. This fails to suggest a linear relationship between the various axes of accelerometer and gyroscope information. While we would expect a relationship between rotation as measured by the gyroscope and orientation of the gravity vector as measured by the accelerometer, this is not a linear relationship and thus was not captured by our

TABLE I
COMPRESSION RATIOS FOR ALL METHODS ACROSS MOVEMENT ACTIVITY AND BODY SEGMENT

| | O | All | Activity | | | | | Body Segment | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | | | | Foot | | Shin | | Thigh | |
| | | | Run | Walk | Bike | Sit | Stand | R | L | R | L | R | L |
| **Traditional Formats** | | | | | | | | | | | | | |
| **CSV** | | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 | 1.00 |
| **ZIP** | | 3.97 | 3.26 | 3.06 | 3.09 | 6.07 | 6.18 | 3.99 | 4.10 | 3.95 | 3.98 | 3.98 | 3.93 |
| **Binary** | | 8.32 | 8.32 | 8.32 | 8.32 | 8.32 | 8.32 | 8.32 | 8.32 | 8.32 | 8.32 | 8.32 | 8.32 |
| **Proposed Methods** | | | | | | | | | | | | | |
| **5th deg. poly.** | (19*$n$) | 11.25 | 8.59 | 8.92 | 9.87 | 15.28 | 15.78 | 11.24 | 11.16 | 11.10 | 11.20 | 11.49 | 11.34 |
| **4th deg. poly.** | (19*$n$) | 11.57 | 8.73 | 9.09 | 10.15 | 15.93 | 16.39 | 11.54 | 11.47 | 11.42 | 11.54 | 11.81 | 11.65 |
| **3rd deg. poly.** | (19*$n$) | 11.80 | 8.84 | 9.22 | 10.30 | 16.44 | 16.80 | 11.74 | 11.67 | 11.66 | 11.78 | 12.05 | 11.88 |
| **2nd deg. poly.** | (13*$n$) | 12.00 | 8.97 | 9.35 | 10.44 | 16.77 | 17.29 | 11.95 | 11.87 | 11.85 | 12.00 | 12.25 | 12.07 |
| **Spline** | (3*$n$) | 12.20 | 9.23 | 9.63 | 10.74 | 16.61 | 17.24 | 12.04 | 11.99 | 11.98 | 12.12 | 12.60 | 12.48 |
| **Linear** | (8*$n$) | 12.22 | 9.11 | 9.52 | 10.52 | 17.19 | 17.81 | 12.21 | 12.11 | 12.07 | 12.20 | 12.45 | 12.26 |
| **Delta** | (1*$n$) | 12.75 | 9.52 | 9.95 | 10.96 | 17.87 | 18.69 | 12.68 | 12.61 | 12.54 | 12.71 | 13.07 | 12.87 |
| **Optimal Methods** | | | | | | | | | | | | | |
| **Optimal AR** | | 12.73 | 9.54 | 9.93 | 11.12 | 17.87 | 18.76 | 12.61 | 12.55 | 12.48 | 12.68 | 13.12 | 12.92 |
| **Optimal MVAR** | | 12.70 | 9.56 | 9.93 | 11.12 | 17.83 | 18.67 | 12.50 | 12.26 | 12.34 | 12.60 | 12.90 | 12.70 |

Groups of data classes and methods without significantly different compression ($p \geq 0.05$) are indicated by bars below and to the right of the table respectively. Specifically, horizontal bars represent statistical analysis of the Delta compression method applied to each data class, while vertical bars represent statistical analysis of each proposed methods applied to all data. Computational complexity of implementation is shown after each proposed method.

models. This study does not preclude the possibility of a non-linear model to successfully exploit such a relationship.

The compression ratios presented in this paper are intended to demonstrate the relative difference between compression methods and may not be representative of the absolute CR experienced in other applications. There are many other factors which can affect the compression ratio which are not explored in this paper. Namely, sensor differences of precision, noise, bias, and sampling rate are expected to have a large impact on the CR achieved. That being said, our corpus consisted of low-cost consumer grade IMUs at a low sampling rate, and the authors would expect many applications to experience significantly higher compression than presented here if higher sampling rate or higher quality sensors are used.

The corpus chosen for demonstrating performance in this work allowed us to explore the effect of movement activity and body segment on compression. However, it could be improved for this purpose by including diverse hardware and higher sampling rates which would be more representative of applications. Signals from more body segments and magnetometer signals should also be included in an ideal corpus.

In future work, the the methods described in this paper can be improved to dynamically detect the optimal Golomb coding order and to recover from dropped packets. Dynamic linear models have shown promising results in similar applications [36] and non-linear models can also be explored. The community would also benefit from a standardized format for representing IMU data, as well as distortion criteria for the various applications of the human movement IMU signal. This would pave the way for the development of lossy compression methods.

## V. CONCLUSION

This work explores methods for the compression of human movement IMU signal. Greater compression will improve data transmission throughput and storage efficiency, potentially enabling new technology applications which were previously infeasible. For the corpus selected, delta encoding provided near-optimal linear compression with low computational cost.

## ACKNOWLEDGMENT

This work was supported by the National Natural Science Foundation of China (51875347).

## REFERENCES

[1] N. Ahmad, R. A. R. Ghazilla, N. M. Khairi, and V. Kasi, "Reviews on Various Inertial Measurement Unit (IMU) Sensor Applications," *International Journal of Signal Processing Systems*, vol. 1, no. 2, pp. 256–262, 2013.

[2] S. Campbell, N. O'Mahony, L. Krpalcova, D. Riordan, J. Walsh, A. Murphy, and C. Ryan, "Sensor Technology in Autonomous Vehicles : A review," *29th Irish Signals and Systems Conference, ISSC 2018*, pp. 1–4, 2018.

[3] T. M. Fernández-Caramés and P. Fraga-Lamas, "Towards the internet-of-smart-clothing: A review on IoT wearables and garments for creating intelligent connected E-textiles," *Electronics (Switzerland)*, vol. 7, no. 12, 2018.

[4] C. C. Yang and Y. L. Hsu, "A review of accelerometry-based wearable motion detectors for physical activity monitoring," *Sensors*, vol. 10, no. 8, pp. 7772–7788, 2010.

[5] A. Filippeschi, N. Schmitz, M. Miezal, G. Bleser, E. Ruffaldi, and D. Stricker, "Survey of motion tracking methods based on inertial sensors: A focus on upper limb human motion," *Sensors (Switzerland)*, vol. 17, no. 6, pp. 1–40, 2017.

[6] S. Patel, H. Park, P. Bonato, L. Chan, and M. Rodgers, "A review of wearable sensors and systems with application in rehabilitation," *Journal of NeuroEngineering and Rehabilitation*, vol. 9, no. 1, p. 21, 2012.

[7] P. B. Shull, W. Jirattigalachote, M. A. Hunt, M. R. Cutkosky, and S. L. Delp, "Quantified self and human movement: A review on the clinical impact of wearable sensing and feedback for gait analysis and intervention," *Gait and Posture*, vol. 40, no. 1, pp. 11–19, 2014.

[8] V. Camomilla, E. Bergamini, S. Fantozzi, and G. Vannozzi, "Trends supporting the in-field use of wearable inertial sensors for sport performance evaluation: A systematic review," *Sensors (Switzerland)*, vol. 18, no. 3, 2018.

[9] N. M. Gobinath Aroganam and D. H. *, "Review on Wearable Technology Sensors Used in Consumer Sport Applications," *Sensors*, vol. 19, no. 1983, 2019.

[10] K. Sayood, *Introduction to Data Compression*. Elsevier, third ed., 2006.

[11] M. Nelson and J.-l. Gailly, *The Data Compression Book*. MIS:Press, 2nd ed., 1995.

[12] C. E. Shannon, "A Mathematical Theory of Communication," *The Bell System Technical Journal*, vol. 27, no. 3, pp. 212–214, 1948.

[13] D. A. Huffman, "A Method for the Construction of Minimum-Redundancy Codes," *Proceedings of the IRE*, vol. 40, no. 9, pp. 1098–1101, 1952.

[14] I. H. Witten, R. M. Neal, and J. G. Cleary, "Arithmetic coding for data compression," *Communications of the ACM*, vol. 30, pp. 520–540, jun 1987.

[15] S. W. GOLOMB, "Run-Length Encodings," *IEEE Transactions on Information Theory*, vol. IT-12, no. 3, pp. 399–401, 1966.

[16] R. F. Rice, "Practical universal noiseless coding.," *Applications of digital image processing III*, vol. 79-22, pp. 247–267, 1979.

[17] M. Mahoney, *Data Compression Explained*. Dell, Inc, 2013.

[18] A. N. Kolmogorov, "Three approaches to the definition of the concept quantity of information," *Probl. Peredachi Inf.*, vol. 1, no. 1, pp. 3–11, 1965.

[19] R. Chereshnev and A. Kertész-Farkas, "HuGaDB: Human gait database for activity recognition from wearable inertial sensor networks," *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, vol. 10716 LNCS, pp. 131–141, 2018.

[20] J. O. Limb, "Distortion Criteria of the Human Viewer," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 9, no. 12, pp. 778–793, 1979.

[21] T. Painter and A. Spanias, "Perceptual coding of high-quality digital audio," *Proceedings of the IEEE*, vol. 88, no. 4, pp. 451–513, 2000.

[22] I. C. Society, *IEEE Std 754-2008 (Revision of IEEE Std 754-1985), IEEE Standard for Floating-Point Arithmetic*, vol. 2008. 2008.

[23] D. Monniaux, "The pitfalls of verifying floating-point computations," *ACM Transactions on Pro- gramming Languages and Systems (TOPLAS)*, vol. 30, no. 3, 2008.

[24] D. Goldberg, *What every computer scientist should know about floating-point arithmetic*, vol. 23. 1991.

[25] N. W. Group, P. Deutsch, and A. Enterprises, "DEFLATE Compressed Data Format Specification version 1.3," 1996.

[26] a. Kiely, "Selecting the Golomb Parameter in Rice Coding," *Analysis*, vol. 2, pp. 1–18, 2004.

[27] N. Merhav, G. Seroussi, and M. J. Weinberger, "Coding of Sources with Two-Sided Geometric Distributions and Unkown Parameters," *IEEE Transactions on Information Theory*, vol. 46, no. 1, pp. 229–236, 2000.

[28] B. O'Donoghue, E. Chu, N. Parikh, and S. Boyd, "Conic optimization via operator splitting and homogeneous self-dual embedding," *Journal of Optimization Theory and Applications*, vol. 169, pp. 1042–1068, June 2016.

[29] B. O'Donoghue, E. Chu, N. Parikh, and S. Boyd, "SCS: Splitting conic solver, version 2.1.2." https://github.com/cvxgrp/scs, Nov. 2019.

[30] S. Diamond and S. Boyd, "CVXPY: A Python-embedded modeling language for convex optimization," *Journal of Machine Learning Research*, vol. 17, no. 83, pp. 1–5, 2016.

[31] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *Journal of Machine Learning Research*, vol. 7, pp. 1–30, 2006.

[32] M. Friedman, "The Use of Ranks to Avoid the Assumption of Normality Implicit in the Analysis of Variance," *Journal of the American Statistical Association*, vol. 32, no. 200, pp. 675–701, 1937.

[33] P. B. Nemenyi, *Distribution-free multiple comparisons*. PhD thesis, Princeton University, 1963.

[34] R. A. Fisher, *Statistical methods and scientific inference (2nd edition)*. Hafner Publishing Co., 1959.

[35] J. W. Tukey, "Comparing Individual Means in the Analysis of Variance," *Biometrics*, vol. 5, no. 2, pp. 99–114, 1949.

[36] D. Blalock, S. Madden, and J. Guttag, "Sprintz: Time Series Compression for the Internet of Things," vol. 2, no. 3, 2018.

[37] J. Coalson, "Flac-free lossles audio codec." http://flac.sourceforge.net, 2008.

[38] J. M. A. Tanskanen, *Polynomial Predictive Filters : Implementation and Applications*. No. November, 2000.

## Scratch Section

Lambda discussion Compare study of other signals talking about other signal studies

TODO: rewrite paragraphs below, maybe drop or move to method

The goal of the model is to produce an accurate PMF with minimum entropy. Intuitively this means that compression is higher the more "sure" the model is about some occurrences over other ones. If an optimal encoder is used, the compression ratio can be regarded as a quantitative measurement of the model's understanding of the underlying signal. Because of this, we expect the performance of various signal models to give us insight into the content of the kinematic signal.

The same coder and decoder is employed by all compression methods explored in this work, and methods are thus identified by the model used. Models are presented as predictive algorithms. Using this paradigm, the residual is the difference between each input symbol and what the model predicted the symbol to be. The compressed size is the first order entropy of the residual signal.

Consider deleting below:

TODO: why do we assume our signal is geometrically distributed about the mean!? TODO: relationship for natural signals of residual signal and compression ratio

Linearized rotating gravity - A Newtonian physics based model in which a constant magnitude acceleration vector rotates according to gyroscope readings.

In the context of compression, complex or numerically unstable algorithms may produce compression ratios which underestimate their understanding of the underlying signal since their implementation is not equivalent to their mathematical derivation.

TODO: handle negatives in optimal filter derivation!

TODO: Discussion of cross stream FIR coefficients, pole zero plots, precision effect

TODO: Medical names for body parts?

TODO: can I prove that the optimal IIR linear filter cannot exceed that of the optimal FIR linear filter?

TODO: audio compression comparison of methods and performance TODO: demonstrate long implementation of compression technique to show that error is from quantization and fixed point error TODO: re clarify hypothesis, discuss what was verified TODO[change the name of the repo, clean up docs, and release].

TODO: can I reference solved problem Mahoney2013? [TODO remove?] We hypothesized that compression ratios would be comparable to those achieved in lossless audio compression. We also hypothesized that the best performing method will utilize some cross axis correlation and be informed by physics based models. [TODO] find better pidgeon hole citation TODO fix citations (some seem wrong or in the wrong format)

Future work includes: -arithmetic coding -inter sensor correlation -sensors with fixed point, homogeneous sample space

akward second paragraph first sentence

state of modeling in biomed and audio how it is the same and/or different what techniques are used [TODO: consider computation time?]