On Network-Efficient Multimodal Multi-Vantage Foundation Models for Distributed Sensing

Tianchen Wang*, Yizhuo Chen*, Hongjue Zhao, You Lyu, Jinyang Li, Tomoyoshi Kimura, Yigong Hu, Denizhan Kara, Maggie Wigness†, Jeffrey Twigg†, Tarek F. Abdelzaher
*Equal Contribution
University of Illinois Urbana-Champaign
†DEVCOM Army Research Labs

***Abstract*—The rise of multi-modal, multi-node foundation models has revolutionized intelligent IoT sensing systems by enabling general-purpose inference from distributed sensing sources to support diverse downstream applications. However, the high communication cost of transmitting raw sensor data from distributed nodes to a central inference model remains a critical bottleneck, particularly in bandwidth- or energy-constrained environments. While existing compression methods can reduce data volume, they often lack the adaptability needed to handle variations in data relevance and redundancy across sources, modalities, and time. To address this challenge, we introduce ZipFM, a lightweight, plug-and-play middleware that dynamically configures sensor data compression strategies on a per-node, per-modality, and per-time-step basis to minimize network traffic while preventing model degradation, taking model sensitivity to different data sources into account. ZipFM is (i) compatible with different pre-trained foundation models without requiring access to their internal mechanisms or retraining, (ii) agnostic to the underlying tools available for data compression, and (iii) independent of the specific downstream inference tasks performed. At its core, ZipFM uses the compression-induced latent representation shift, produced by the foundation model's backbone, as a proxy for downstream accuracy degradation, and enforces a system-wide optimal representation shift (in the sense of minimizing compression-related degradation) through a lightweight feedback control mechanism. Experiments on three real-world IoT sensing datasets demonstrate that ZipFM significantly reduces communication costs while preserving model performance.**

I. Introduction

The concept of **multi-modal, multiview foundation models** has recently been proposed [1] to catalyze a new era in intelligent Internet of Things (IoT) sensing systems [2], [3]. In these models, the model's backbone inherently recognizes not only *multiple data modalities* but also *multiple sensor vantage points*. The models learn the association of signals and locations to help best interpret the *multiview* (also called *multi-vantage*) sensing data. By understanding the impact of sensing vantage points on received signals, these models are better equipped to reason about spatial-temporal properties of the observed environment, given the set of data streams emanating from sparsely and irregularly deployed sensors used for data collection. As such, they are poised to achieve advances in inference across many IoT applications, ranging from earthquake localization [1] to military target tracking [4]. However, multiview foundation models introduce a critical systems challenge. Multi-modal, multi-dimensional, and often redundant data are transmitted from sensing devices to the central model. This creates a communication bottleneck, particularly in bandwidth-limited or energy-constrained environments. Naively transmitting all raw data is unsustainable for large-scale deployments.

Prior work has explored a range of (lossy) compression techniques[1] to reduce communication overhead [5], [6]. While these approaches can be effective in specific scenarios, they fall short in multi-modal multiview foundation model-based IoT systems due to a fundamental issue: the relevance and redundancy of data across different nodes, modalities, and samples vary significantly over time, causing substantial time-varying differences in model sensitivity to the degree of compression of individual data streams. A fixed or global compression policy inevitably leads to inefficiencies and/or loss of fidelity.

To address this challenge, we propose ZipFM, a lightweight, plug-and-play middleware designed for multi-modal, multi-node IoT sensing systems powered by foundation models. Instead of enforcing a one-size-fits-all policy, ZipFM dynamically configures *the choice of compression algorithm and its parameters* per node, per modality, and per time step, to optimize the trade-off between communication cost and sensing fidelity. ZipFM is designed to satisfy three criteria:

1) **Plug-and-Play Model Compatibility**: ZipFM is agnostic to the structure, parameters, and training procedure of the underlying foundation model. It requires no model retraining or architectural modification.
2) **Plug-and-Play Compression Algorithm Compatibility**: ZipFM can operate with any set of lossy (configurable) user-defined compression tools (e.g., JPEG, ZFP), making it easily extensible and domain-adaptable.
3) **Task-Independence**: ZipFM supports multiple concurrent downstream inference tasks without access to task-specific labels or task logic, making it robust to a wide range of workloads.

ZipFM is built on two key insights: (1) **Latent Representation Shift as a Proxy for Accuracy Loss**: Foundation models use a backbone-head architecture, where a shared backbone extracts semantic latent representations used across tasks. We

[1]While lossless compression has also been used, it is not adaptive to the amount of resources available, which is a challenge in IoT environments where available network bandwidth can dynamically change. Thus, when we refer to *compression* in this paper, we mean *lossy compression* by default.

show that the shift in these latent representations strongly correlates with downstream performance degradation. This enables task-agnostic, label-free performance monitoring. (2) **Convexity of Latent Shift with Respect to Compression**: Empirically, we find that the latent representation shift behaves as a convex function of communication bitrate. This observation enables principled, convex optimization for per-source bitrate allocation.

Building on these insights, ZipFM casts bitrate assignment as a constrained convex optimization problem: minimize total communication while ensuring the latent representation shift remains within a tolerable range. From this formulation, we derive a system-wide optimality principle: *the marginal decrease in distortion per bit should be equal across all data sources*. ZipFM enforces this principle using a lightweight feedback-control loop, which estimates the derivative via runtime probing and adjusts compression settings accordingly.

We evaluate ZipFM on three diverse, real-world IoT sensing datasets: **M3N-VC** (vehicle classification) [7], **RealWorld HAR** (human activity recognition) [8], and the **Ridgecrest Seismicity Dataset** (earthquake localization) [9]. Results show that ZipFM can achieve significant communication traffic reduction, while preserving the original model's sensing accuracy.

To summarize, the paper tackles the challenging problem of lossy data compression to support network-efficient multi-modal multiview foundation model inference, while minimizing quality degradation. Our contributions are recapped as follows:

- We propose a novel inference-task-independent measure of quality degradation based on compression-induced latent representation shift.
- We propose a novel system-wide optimality principle that minimizes degradation by equalizing marginal shift in distortion per bit.
- We develop ZipFM, a new middleware that implements the above principle to attain adaptive, task-agnostic optimal data compression, independent of model architecture and compression specifics, that minimizes degradation.
- We demonstrate its effectiveness using multiple real-world datasets, achieving large communication savings while preserving quality of inference.

The rest of this paper is organized as follows. Section II reviews related work. Section III describes the analytical foundations behind ZipFM and presents its design. Section IV describes the evaluation results. A discussion is presented in Section V, followed by conclusions in Section VI.

II. Related Work

Foundation Models for Multi-Modal, Multi-Node IoT Systems. Recent advances in pretraining for IoT have produced foundation models that generalize across sensing tasks using unlabeled, heterogeneous data from multiple modalities. These include contrastive approaches such as Cosmo [10], Cocoa [11], and FOCAL [12], which align modality-specific views in a shared space, and more recent systems like ImageBind [13], MMBind [14], and InfoMAE [15], which extend to loosely paired or unpaired modalities. Generative methods, including Ti-MAE [16], MOMENT [17], TS-MAE [18], FreqMAE [4], PhyMask [19], and SPAR [1], apply masked reconstruction to time-series and frequency-domain data for robust, transferable features. LLM-inspired approaches such as LIMU-BERT [20], IoT-LM [21], LLMSense [22], and Penetrative AI [23] enable zero-/few-shot inference via prompting, summarization, or adapters. While effective, these works assume full-resolution sensor inputs and do not address the communication bottlenecks in real IoT deployments. In contrast, our middleware enables scalable use of such models under bandwidth constraints.

Communication Traffic Reduction Methods for IoT. To address bandwidth and energy limitations in IoT deployments, data reduction strategies generally fall into three categories: Data Compression (DC), Data Prediction (DP), and Data Aggregation (DA) [5]. DC methods encode raw sensor data before transmission. Lossless techniques like Huffman coding [24] or LZW [25] preserve full fidelity but offer modest compression ratios. Lossy methods [6] achieve higher reductions by discarding redundant information, though they often apply fixed policies without considering data relevance to downstream models. DP methods build models to predict sensor readings and send only unexpected or divergent measurements [26], [27]. Single and dual prediction frameworks [28], often based on lightweight ML models, can greatly reduce volume but are sensitive to distribution shifts and noisy environments. DA techniques reduce redundancy by summarizing data across spatial or temporal dimensions, often at edge gateways [29], [30]. Aggregation functions like mean or min/max are computationally cheap and energy-efficient [31], but discard fine-grained information critical to high-resolution sensing tasks. While each class of methods offers benefits, none dynamically adapts compression to heterogeneous, time-varying model sensitivities as in our work.

Lossy Data Compression Methods for IoT. Among the above strategies, lossy compression offers particularly strong potential for large reductions in communication traffic by selectively removing redundancy. Existing approaches can be grouped into four categories: lossy-lossless hybrids, AI-based encoders, interpolation-based methods, and transform-based methods. Hybrid schemes [32]–[34] switch between lossy and lossless modes depending on factors such as energy availability or data criticality. AI-based methods [35], [36] employ models like autoencoders to learn compact representations, excelling on high-dimensional or nonlinear data but incurring training and inference costs unsuitable for constrained devices. Interpolation-based techniques [37], [38] use linear or nonlinear approximations to summarize signals under error bounds; they are lightweight but struggle with abrupt changes or high-frequency content. Transform-based methods [39] leverage domain conversions (e.g., wavelet or cosine transforms) to sparsely represent structured signals, but often generalize poorly across modalities. While effective in specific contexts, these methods typically lack per-node, per-modality adaptivity to foundation model sensitivities—a gap addressed by our

proposed middleware.

III. Algorithm Design

In this section, we present the design and implementation of ZipFM, a lightweight middleware that enables efficient, adaptive compression for multi-modal, multi-node IoT sensing systems built on foundation models. ZipFM dynamically configures compression strategies—per node, per modality, per time step—to minimize communication cost without sacrificing sensing accuracy. Crucially, ZipFM satisfies the three criteria described in Section I: it is *Agnostic to Models*, *Agnostic to Compression Algorithms*, and *Agnostic to Tasks*. These properties make ZipFM broadly applicable, plug-and-play, and robust in real-world deployments.

The core idea behind ZipFM is to monitor and control the **latent representation shift**—a measure of how much the latent representation of the input data (as produced by the foundation model's backbone) changes when the input is compressed. We show that this shift is a good indicator of inference performance degradation, and that it behaves as a convex function with respect to the compression rate. By modeling this relationship and controlling it through feedback, ZipFM is able to intelligently balance communication efficiency and model fidelity.

For clarity, we adopt the following notation convention throughout the paper: scalars are denoted by lowercase or uppercase letters (e.g., t, T), vectors by bold lowercase letters (e.g., $\boldsymbol{x}$), and sets by calligraphic uppercase letters (e.g., $\mathcal{X}$).

A. Latent Representation Shift as an Effective Proxy for Sensing Accuracy Degradation

Consider a system with N edge nodes, each collecting data from M different modalities over T time steps. Let $\boldsymbol{x}_{i,j,t}$ denote the raw, uncompressed data from node i, modality j, at time t, and let the complete multi-node, multi-modal input at time t be

$$\mathcal{X}_t = \{\boldsymbol{x}_{i,j,t}\}_{i=1,\dots,N;\, j=1,\dots,M}.$$

In a sensing system based on a multi-modal, multi-node foundation model, the input data $\mathcal{X}_t$ is gathered from edge nodes into a central server and then processed by a pre-trained backbone network f_θ, which extracts a joint latent representation $f_\theta(\mathcal{X}_t)$. This latent representation captures high-level semantic information and is then passed to multiple task-specific heads to generate outputs $\boldsymbol{y}_{k,t}$, where $k \in \{1, \dots, K\}$ indexes different sensing tasks. For instance, in a vehicle monitoring system, seismic signals and acoustic signals collected from various roadside nodes may be fused into a single latent representation. And one task head may then perform vehicle type classification, while another localizes the vehicle spatially.

To reduce bandwidth usage, each node compresses its data before transmission. Let $\boldsymbol{x}'_{i,j,t}$ denote the compressed version of $\boldsymbol{x}_{i,j,t}$, and let the compressed system input be:

$$\mathcal{X}'_t = \{\boldsymbol{x}'_{i,j,t}\}_{i=1,\dots,N;\, j=1,\dots,M}.$$

When $\mathcal{X}_t$ is replaced with $\mathcal{X}'_t$, the backbone's output $f_\theta(\mathcal{X}'_t)$ may differ from the original one, potentially harming task performance.

However, directly measuring the drop in task accuracy requires labels and task-specific logic, which is not practical in real-time deployments. Instead, we propose a task-agnostic, label-free proxy: *the latent representation shift*, defined as the squared Euclidean distance between the latent representation computed from the original and the compressed inputs:

$$d_t = \|f_\theta(\mathcal{X}_t) - f_\theta(\mathcal{X}'_t)\|_2^2.$$

Empirical evidence, as shown in Fig. 1, Fig. 2, and Fig. 3 in Section IV-A, reveals that d_t strongly correlates with actual drops in task performance across datasets and sensing tasks. This is because the backbone's latent representation encodes general, task-agnostic features that are sensitive to information loss.

Why Correlation is Expected: A Rate-Distortion Theory Perspective. This approach can be grounded in rate-distortion theory [40], which formalizes the trade-off between compression efficiency (rate) and retained task-relevant information (distortion). In our setting, the goal is to minimize communication bandwidth, measured by the mutual information between the original and compressed inputs $I(\mathcal{X}_t; \mathcal{X}'_t)$, while ensuring that compression does not excessively degrade the information needed for downstream tasks. Formally, we pose this as:

$$\begin{aligned} \min \quad & I(\mathcal{X}_t; \mathcal{X}'_t) \\ \text{s.t.} \quad & I(\boldsymbol{y}_{k,t}; \mathcal{X}_t \mid \mathcal{X}'_t) \le c \quad \text{for all } k. \end{aligned} \tag{1}$$

Directly computing the distortion $I(\boldsymbol{y}_{k,t}; \mathcal{X}_t \mid \mathcal{X}'_t)$ for all tasks is impractical, as it would require full access to each task head's structure. To resolve this, we observe that in foundation model-based systems, the backbone output $f_\theta(\mathcal{X}_t)$ serves as a task-agnostic, sufficient statistic for all downstream tasks. Therefore, we can obtain the following bound by applying the Data Processing Inequality:

$$I(\boldsymbol{y}_{k,t}; \mathcal{X}_t \mid \mathcal{X}'_t) \le I(f_\theta(\mathcal{X}_t); \mathcal{X}_t \mid \mathcal{X}'_t).$$

Thus, we can ensure that the distortion with respect to *any* downstream task is constrained by constraining the distortion on the backbone's latent representation. Formally, we can reframe the optimization problem in (1) as:

$$\min \quad I(\mathcal{X}_t; \mathcal{X}'_t) \quad \text{s.t.} \quad I(f_\theta(\mathcal{X}_t); \mathcal{X}_t \mid \mathcal{X}'_t) \le c.$$

In practice, ZipFM approximates the $I(f_\theta(\mathcal{X}_t); \mathcal{X}_t \mid \mathcal{X}'_t)$ term using the latent representation shift d_t as we described above. This approximation is efficient and deployable, and still conforms to rate-distortion theoretical principles.

B. Convexity of Latent Representation Shift Enables Optimal Compression Budget Allocation

A key insight enabling ZipFM's adaptive control is that, in most cases, the latent shift d_t behaves *convexly* with respect to communication bitrate. This empirical observation is supported by the results shown in Fig. 1, Fig. 2, and Fig. 3 in Section IV-A.

For each node i, modality j, and time t, let $b_{i,j,t}$ be the number of bits used to transmit $\boldsymbol{x}'_{i,j,t}$. We observe empirically that, as compression is reduced (i.e., $b_{i,j,t}$ increases), the marginal improvement in d_t diminishes.

This insight enables us to formulate a principled optimization problem: how should we assign bit budgets to each data source—i.e., each combination of node i and modality j—to minimize total communication cost, while keeping the overall distortion (as measured by latent representation shift) under a desired threshold?

To express this formally, let $\mathcal{B}_t = \{b_{i,j,t}\}_{i=1,\ldots,N;\, j=1,\ldots,M}$ denote the set of bitrates used to encode each data stream at time t, and let $d_t(\mathcal{B}_t)$ denote the resulting latent shift under this allocation. Our objective is to find the bitrate allocation that minimizes total bandwidth while ensuring the induced distortion does not exceed a predefined threshold c:

$$\min_{\mathcal{B}_t} \quad \sum_{i,j} b_{i,j,t} \quad \text{s.t.} \quad d_t(\mathcal{B}_t) \leq c.$$

As we describe above, our systems satisfy the **Convexity:** The distortion function $d_t(\mathcal{B}_t)$ is convex with respect to each bitrate $b_{i,j,t}$. Besides, they also satisfy the **Feasibility:** When there is no compression, $d_t(\mathcal{B}_t) = 0$, which means Slater's condition holds.

Therefore, standard results from Karush-Kuhn-Tucker (KKT) conditions [41] tell us that a solution $\mathcal{B}_t^*$ is optimal in our systems if and only if the following conditions hold for all i and j:

$$\frac{\partial d_t}{\partial b^*_{i,j,t}} = -\frac{1}{\mu} \quad \text{and} \quad d_t(\mathcal{B}_t^*) = c,$$

where $\mu > 0$ is the Lagrange multiplier associated with the distortion constraint. This result has an elegant interpretation: at the optimal allocation, the marginal decrease in distortion per additional bit should be equal across all data sources. This principle underpins the adaptive compression mechanism implemented in ZipFM.

Importantly, in our setting, the distortion threshold c and the multiplier μ are mutually dependent. As a result, we propose to set μ as a hyperparameter in ZipFM, leaving the corresponding c automatically induced. This simplifies the deployment and tuning of ZipFM.

Why Convexity is Expected: A Rate-Distortion Theory Perspective. We further provide a theoretical argument, within the same rate–distortion framework mentioned above, to explain why the latent shift d_t often exhibits *convex* behavior with respect to communication bitrates. Specifically, the distortion, approximated by d_t, can be expressed as:

$$\begin{aligned} d_t \approx & I(f_\theta(\mathcal{X}_t); \mathcal{X}_t \mid \mathcal{X}'_t) \\ = & I(f_\theta(\mathcal{X}_t); \mathcal{X}_t, \mathcal{X}'_t) - I(\mathcal{X}'_t; f_\theta(\mathcal{X}_t)) \\ = & I(f_\theta(\mathcal{X}_t); \mathcal{X}_t) - I(\mathcal{X}'_t; f_\theta(\mathcal{X}_t)) \end{aligned} \tag{2}$$

Here, $I(f_\theta(\mathcal{X}_t); \mathcal{X}_t)$ is constant with respect to compression, so the bitrate dependence lies entirely in $I(\mathcal{X}'_t; f_\theta(\mathcal{X}_t))$. Modern compression methods are deterministic and designed to preserve semantic content, so we adopt a simplifying and idealized assumption that $\mathcal{X}'_t$ retains the most information of $f_\theta(\mathcal{X}_t)$. Under this assumption, we can have:

$$\begin{aligned} I(\mathcal{X}'_t; f_\theta(\mathcal{X}_t)) &= \min(H(f_\theta(\mathcal{X}_t)), H(\mathcal{X}'_t)) \\ &= \min(H(f_\theta(\mathcal{X}_t)), \sum_{i,j} b_{i,j,t}) \end{aligned} \tag{3}$$

Substituting this into (2) shows that d_t becomes a piecewise linear, convex function of each $b_{i,j,t}$, providing a theoretical rationale for the convex patterns observed in our experiments.

C. Enforcing Optimal Compression Budget Allocation with Feedback Control

To enforce the principle of equalizing the marginal decrease in distortion per additional bit across all data sources, ZipFM implements a *feedback-control mechanism* that adjusts the compression algorithm and the associated compression levels based on how far each data source is from the ideal point $-\frac{1}{\mu}$.

Let $u_{i,j,t} \in \{1, \ldots, U\}$ represent the compression level applied to node i, modality j, at time t, where higher values of u correspond to more aggressive compression (i.e., fewer bits transmitted). Given a compression algorithm, $u_{i,j,t}$ maps to a specific bitrate $b_{i,j,t}$. Periodically, ZipFM updates $u_{i,j,t}$ using the following rule:

$$u_{i,j,t+1} = \begin{cases} u_{i,j,t} - 1 & \text{if } \frac{\partial d_t}{\partial b_{i,j,t}} + \frac{1}{\mu} \leq -h, \quad u_{i,j,t} > 1 \\ u_{i,j,t} + 1 & \text{if } \frac{\partial d_t}{\partial b_{i,j,t}} + \frac{1}{\mu} \geq h, \quad u_{i,j,t} < U \\ u_{i,j,t} & \text{otherwise} \end{cases}$$

Here, $h > 0$ is a tolerance margin that prevents unnecessary oscillations. This simple control rule nudges each stream's compression toward the system-wide ideal point $-\frac{1}{\mu}$.

Estimating the Derivative in Practice. To compute the required derivative $\frac{\partial d_t}{\partial b_{i,j,t}}$ at runtime, ZipFM periodically instructs a node to send two versions of the same data—using compression levels $u_{i,j,t}$ and $u_{i,j,t}+1$ ($u_{i,j,t}-1$ if $u_{i,j,t} = U$), respectively. The rest of the system operates normally, so any change in d_t is attributable solely to that source. The central server measures the square Euclidean distance in latent representation between these two versions, denoted as Δd_t, and the corresponding difference in bitrate, denoted as $\Delta b_{i,j,t}$. The derivative is then estimated as:

$$\frac{\partial d_t}{\partial b_{i,j,t}} \approx \frac{\Delta d_t}{\Delta b_{i,j,t}}.$$

Compression Algorithm Selection. In our design, the choice of the compression algorithm is performed for each modality only during the first control round and then fixed throughout the operation. Specifically, all the nodes send both the original data and a version compressed with the lowest available compression level for each candidate algorithm. ZipFM selects the algorithm with the lowest ratio of latent representation shift per bit saved. This design helps ensure that the chosen algorithm is suited to the characteristics of the data modality, while introducing minimal overhead.

Hybrid Scheduling for Fast and Efficient Adaptation. To limit overhead while maintaining adaptability, ZipFM uses a

hybrid scheduling strategy. Each stream begins in a *warm-up phase*, where compression level updates are frequent, enabling rapid convergence to an appropriate compression level. Once the compression setting stabilizes over multiple rounds, the node transitions to a *steady-state phase*, with updates happening less frequently to save bandwidth. This approach ensures that ZipFM can rapidly adapt to new environments at the beginning of the deployment, while minimizing long-term communication cost.

In summary, ZipFM combines a principled understanding of representation robustness, rate-distortion theory, KKT condition, and feedback control to deliver a practical, scalable solution for bandwidth-efficient operation in foundation model-powered multi-node multi-modal IoT sensing systems.

IV. Experiments

To evaluate the effectiveness of ZipFM in real-world IoT sensing scenarios, we conducted extensive experiments across multiple datasets, tasks, and modalities. Our goals were to validate the key insights underlying ZipFM, and quantify its ability to reduce communication cost while preserving model performance. Below, we detail the datasets, experimental protocols, and results supporting our claims.

Datasets. We conducted experiments on three real-world IoT datasets. (1) The M3N-VC dataset [7] includes synchronized audio and seismic signals collected from six spatially distributed nodes across six outdoor scenes. Audio signals are transmitted as spectrograms, while seismic signals are transmitted as time-series segments. (2) The Ridgecrest Seismicity Dataset [9] comprises three-component seismic waveforms from 16 nodes, recorded with high-gain broadband seismometers and high-gain accelerometers, and sent to the server in spectrogram form. (3) The RealWorld-HAR dataset [8] provides time-series data from six body-mounted nodes, including acceleration, gyroscope, and magnetic field modalities.

Tasks and Metrics. Consistent with [1], we evaluate single-vehicle localization on M3N-VC using the average distance error, and assess multi-vehicle joint classification and localization using mAP@r, which computes the mean average precision across all vehicle classes, considering a prediction correct only if both the class label is correct and the predicted location lies within a distance r of the ground truth. For the Ridgecrest Seismicity Dataset, we measure earthquake event localization performance using distance error. On the RealWorld-HAR dataset, we evaluate multi-class human activity recognition via classification accuracy. These metrics reflect realistic IoT sensing objectives while supporting rigorous performance comparisons.

Foundation Model. For all experiments, we adopt the multi-modal, multi-node foundation model proposed in [1].[2] This model is pre-trained with placement awareness, enabling strong generalization in diverse scenarios. Throughout our evaluation, the foundation model remains fixed.

[2]Note that, preprint [1] is currently undergoing blind review for another publication. If accepted, we shall cite the accepted paper. The focus in that paper (see [1]) is on the model training architecture, which is orthogonal to the contribution of the current work (that lies in the data compression policy).

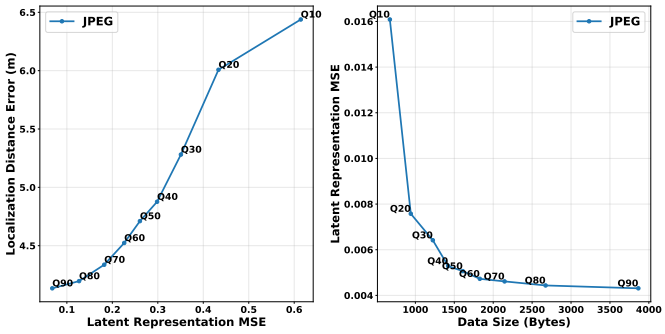

Fig. 1. Analysis of latent representation shift on the M3N-VC dataset. Left: Relationship between latent representation shift and actual drops in localization performance. Right: The latent representation shift exhibits convex behavior with respect to communication bitrate.

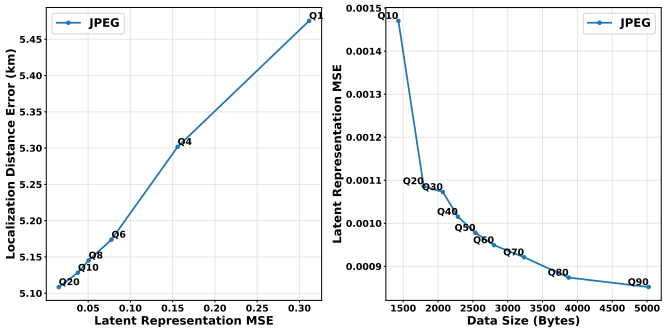

Fig. 2. Analysis of latent representation shift on the Ridgecrest Seismicity Dataset. Left: Relationship between latent representation shift and actual localization error. Right: The latent representation shift exhibits convex behavior with respect to communication bitrate.

Compression Methods. For all our experiments, we employ three widely used lossy compression algorithms for sensory data: JPEG, WebP, and ZFP. For baseline comparisons, we consider uniform compression policies, where all nodes, modalities, and time steps use the same compression method and compression level.

A. Empirical Validation of Our Insights

To empirically validate the use of latent representation shift as a proxy for task performance degradation, and to confirm its convex behavior with respect to communication bitrate, we present an analysis on all three evaluation datasets. Fig.1, Fig.2, and Fig. 3 illustrate these patterns for the M3N-VC, Ridgecrest Seismicity, and RealWorld-HAR datasets, respectively.

For each dataset, the left panel shows the relationship between the latent representation shift and the actual drops in task performance (localization error or classification accuracy). The consistently strong correlation across diverse modalities and tasks supports the viability of latent shift as a task-agnostic, label-free proxy for inference performance.

The right panel in each figure demonstrates the convex trend of latent representation shift as a function of communication

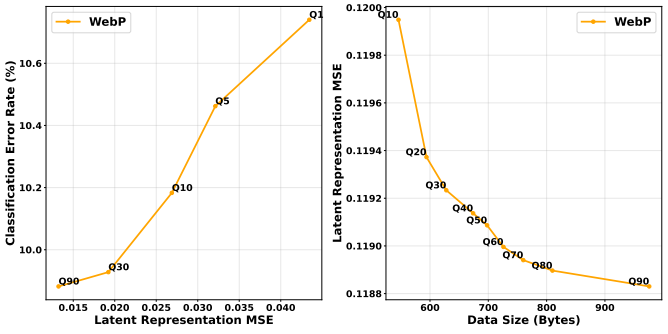

Fig. 3. Analysis of latent representation shift on the RealWorld-HAR dataset. Left: Relationship between latent representation shift and classification accuracy. Right: Convexity of the latent representation shift with respect to communication bitrate.

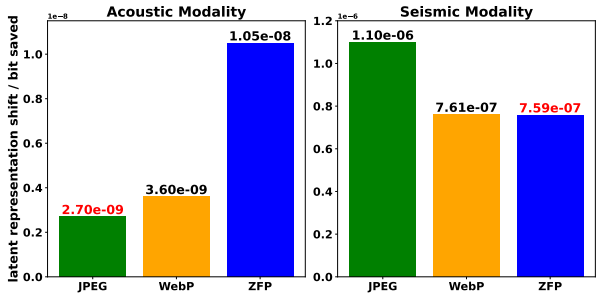

Fig. 4. The ratio of latent representation shift per bit saved compared between different compression algorithms. The selected algorithm for each modality is highlighted in red.

bitrate. As compression is reduced (i.e., bitrate increases), the marginal benefit in reducing latent shift diminishes. This empirical convexity underpins the optimal bitrate allocation mechanism in ZipFM, enabling principled, feedback-based control.

B. Evaluation on M3N-VC Dataset

We begin our evaluation on the M3N-VC dataset, focusing on the single-vehicle localization task. As shown in Fig. 4, ZFP achieves the lowest ratio of latent representation shift per bit saved for the seismic signals, while JPEG performs best for the audio modality. Accordingly, ZipFM selects these algorithms for their respective modalities. As illustrated in Fig. 5, ZipFM achieves a localization error of 4.19 meters while transmitting only about 4% of the original total data traffic. This represents a better trade-off between compression and accuracy compared to baselines that apply a single compression algorithm and compression level uniformly across all nodes, modalities, and time steps. These results highlight the advantage of adaptive algorithm and level selection in achieving a more favorable balance between model performance and communication cost.

Next, we evaluate ZipFM on a more complex setting: the multi-vehicle joint classification and localization task. As shown in Fig. 6, ZipFM maintains an almost unchanged mAP@6m while using only about 4% of the original data traffic. It outperforms all baselines that apply uniform compression algorithms and levels, which either suffer degraded performance or require

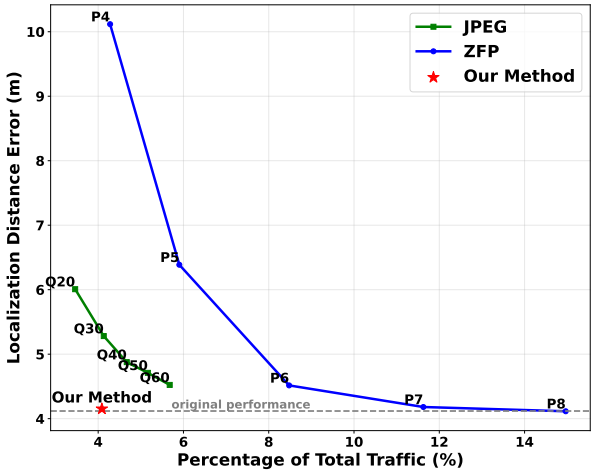

Fig. 5. Localization performance versus communication cost on the M3N-VC dataset. The plot shows the trade-off between localization distance error (y-axis) and the percentage of total traffic (x-axis). Baselines include JPEG (green curve) and ZFP (blue curve), where all nodes and modalities apply a uniform compression setting. The gray dashed line indicates the model performance with original data. ZipFM, marked by a red star, achieves a better balance between localization accuracy and transmission efficiency.

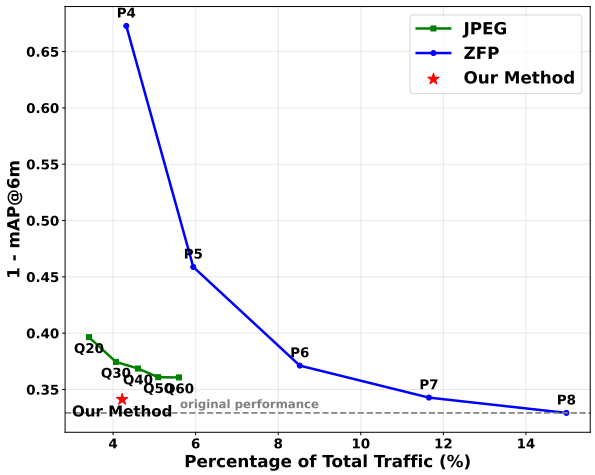

Fig. 6. Localization performance versus communication cost on the M3N-VC dataset. The plot shows the trade-off between 1-mAP@6m (y-axis) and the percentage of total traffic (x-axis). Baselines include JPEG (green curve) and ZFP (blue curve), where all nodes and modalities apply a uniform compression setting. The gray dashed line indicates the model performance with the original data. ZipFM, marked by a red star, achieves a better balance between localization accuracy and transmission efficiency.

significantly more bandwidth. These results further confirm that adaptive, modality-specific compression generalizes effectively to more complex, multi-target scenarios, consistently achieving superior trade-offs between performance and communication efficiency.

To better understand the design choices of ZipFM, we conduct ablation studies focusing on two key aspects: the impact of the *warm-up phase* and the effect of node availability. First, we evaluate a variant that eliminates the warm-up phase, allowing the system to enter the steady-state phase directly. As

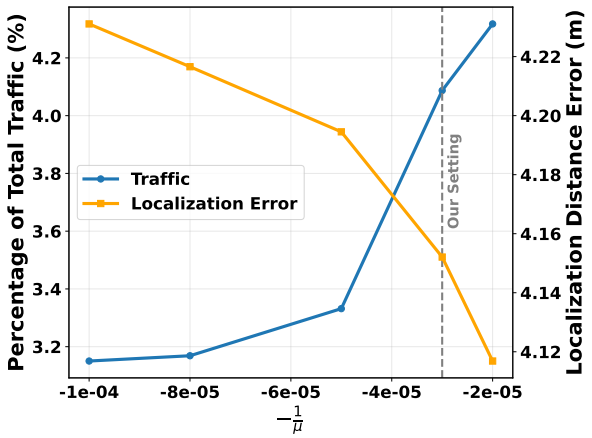

Fig. 7. Effect of different $-\frac{1}{\mu}$ on the trade-off between localization error and communication traffic.

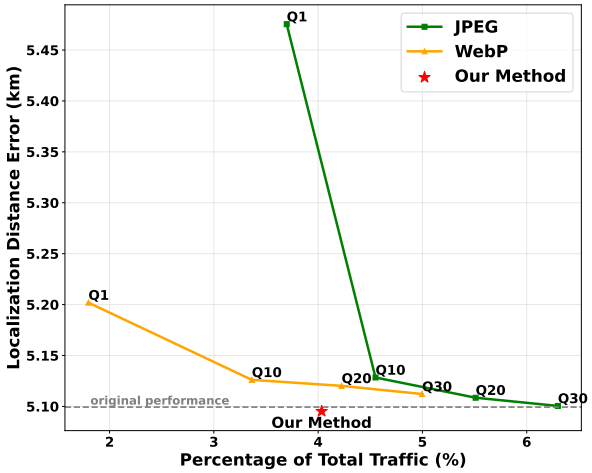

Fig. 8. Localization performance versus communication cost on the Ridgecrest Seismic Dataset. The plot shows the trade-off between localization distance error (y-axis) and the percentage of total traffic (x-axis). Baselines include JPEG (green curve) and WebP (orange curve), where all nodes and modalities apply a uniform compression setting. The gray dashed line indicates the model performance with original data. ZipFM, marked by a red star, achieves a better balance between localization accuracy and transmission efficiency.

shown in Table I, this slows convergence and results in more high-quality data being transmitted during the early stages, ultimately increasing total communication traffic. Second, we examine the system's performance when one of the six nodes is completely removed, resembling a scenario similar to sparse attention in a transformer-based foundation model backbone. This modification leads to degraded localization performance, underscoring that even the small fraction of compressed data retained is critical for the foundation model to accurately interpret events.

We further analyze how the hyperparameter $-\frac{1}{\mu}$ influences the trade-off between model performance and transmission cost in ZipFM. As shown in Fig. 7, decreasing $-\frac{1}{\mu}$ promotes stronger compression, reducing transmission traffic but increasing localization error. This transition is monotonic and smooth, making hyperparameter tuning easier in practice and allowing users to adjust the setting according to their specific requirements.

C. Evaluation on Ridgecrest Seismicity Dataset

In the next experiment, we evaluate ZipFM on the Ridgecrest Seismicity Dataset, focusing on earthquake event localization. This scenario involves relatively high node variability, since different events may not be uniformly detected across all nodes. In this setting, ZipFM selects JPEG for seismometer signals and WebP for accelerometer signals. As shown in Fig. 8, ZipFM achieves low localization error while maintaining a low transmission rate, outperforming the baselines by achieving a more favorable balance between efficiency and accuracy.

To facilitate understanding, we further illustrate the evolution of compression levels over time for several nodes in Fig. 9. As shown, different nodes dynamically adjust their compression levels in response to local signal conditions, enhancing the overall efficiency of the system.

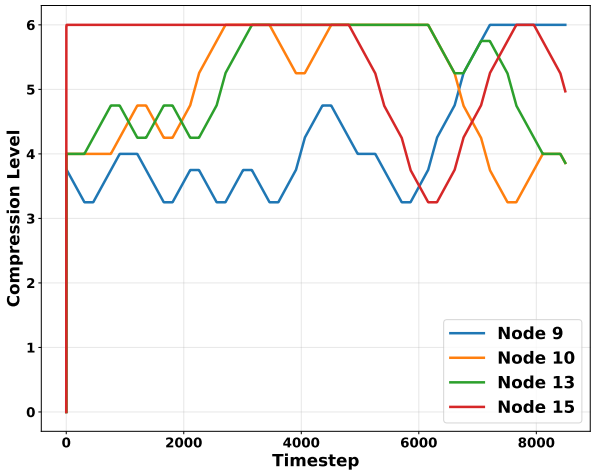

Fig. 9. Evolution of compression levels over time for selected nodes in the Ridgecrest Seismicity Dataset.

D. Evaluation on Realworld HAR Dataset

Finally, we assess ZipFM on the RealWorld-HAR dataset to evaluate its effectiveness for human activity recognition. In this setting, ZipFM selects WebP for accelerometer signals and ZFP for both gyroscope and magnetic field data. As shown in Fig. 10, ZipFM achieves low classification error while maintaining a relatively low transmission cost. Baseline configurations either suffer from higher errors at comparable traffic levels or require greater bandwidth to achieve similar performance. These results highlight the adaptability and effectiveness of ZipFM for IMU data and human activity recognition tasks.

V. Discussion

While ZipFM demonstrates promising results in improving communication efficiency for multi-node, multi-modal IoT foundation model systems, there are several limitations and future exploration directions to consider.

First, ZipFM relies on periodic probing of the latent representation shift to estimate derivatives, which incurs some

TABLE I
ABLATIONS OF ZIPFM.

Method	M3N-VC (H24)				
	Total Traffic (Bytes) (↓)	Percentage Save (%) (↑)	Overhead Traffic (Bytes) (↓)	Localization MSE (m^2) (↓)	Localization Dist. Err. (m) (↓)
ZipFM	16,783,155	95.9	544,739	12.49	4.15
ZipFM w/o warm-up phase	17,417,683	95.8	470,796	12.50	4.16
ZipFM w/ one node dropped	14,183,109	96.5	498,398	13.95	4.37

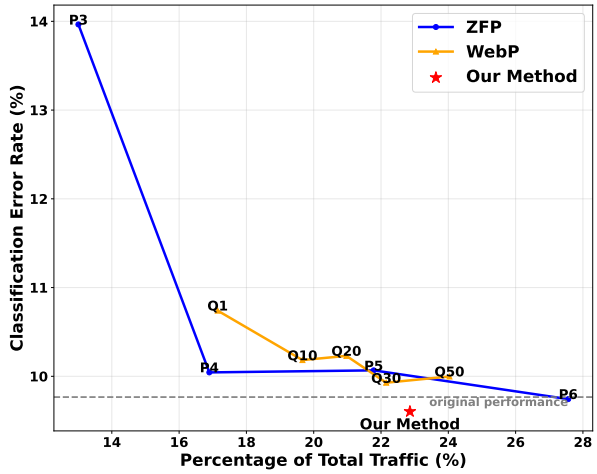

Fig. 10. Activity classification performance versus communication cost on the RealWorld-HAR dataset. The plot shows the trade-off between classification error rate (y-axis) and the percentage of total traffic (x-axis). Baselines include ZFP (blue curve) and WebP (orange curve), where all nodes and modalities apply a uniform compression setting. The gray dashed line indicates the model performance with original data. ZipFM, marked by a red star, achieves a better balance between classification accuracy and transmission efficiency.

additional bandwidth and latency overhead. Although our hybrid scheduling strategy mitigates this cost, highly dynamic environments with rapidly changing data patterns may still experience performance degradation if frequent probing becomes necessary. Developing more lightweight probing techniques remains an important future direction.

Second, ZipFM employs a fixed compression algorithm per modality after an initial selection round. While this design enhances the robustness of the system and simplifies the operations, there may be scenarios where switching compression algorithms over time would be beneficial (e.g., when data characteristics evolve substantially after deployment).

Third, the current formulation of ZipFM assumes a shared backbone running entirely on the central server. In more distributed or hierarchical computing paradigms, such as edge computing or fog computing, the coordination of latent representation monitoring and bitrate allocation could become more complex. Exploring how to develop middleware for these scenarios is an important area for future research.

Fourth, although we have provided both consistent empirical evidence and a rate–distortion–based theoretical rationale for the correlation between latent shift and downstream performance, as well as for the convex behavior of the shift with respect to bitrate, these properties cannot be universally guaranteed for modern, highly non-linear neural networks. Systematically identifying potential failure modes and developing corresponding mitigation strategies would be a valuable direction for future work.

Finally, our lightweight feedback control mechanism, while simple and effective in practice, has not been formally proven to be stable or evaluated under extreme, rapidly varying data distributions. Extending the control framework to offer theoretical convergence guarantees and enhanced adaptivity in highly dynamic environments also represents a promising research opportunity.

VI. CONCLUSIONS

This paper introduces ZipFM, a lightweight, adaptive middleware that enables foundation-model-based multi-modal, multi-node IoT sensing systems to operate more efficiently. By dynamically configuring compression strategies per node, per modality, and per time step, ZipFM achieves significant reductions in communication traffic while maintaining high inference performance. The design of ZipFM is grounded in a novel use of latent representation shift as a proxy for task-agnostic accuracy degradation, together with convex optimization principles and feedback control to balance compression rates across distributed data streams. Extensive experiments on three real-world IoT datasets validate the effectiveness of ZipFM, demonstrating its ability to achieve superior trade-offs between communication efficiency and model fidelity. As intelligent IoT sensing systems incorporate ever more nodes and modalities while increasingly depending on large-scale foundation models, we believe ZipFM can inspire future research on scalable, adaptive, and communication-efficient sensing infrastructures.

ACKNOWLEDGMENT

Research reported in this paper was sponsored in part by DEVCOM ARL under Cooperative Agreement W911NF-172-0196, NSF CNS 20-38817, and the Boeing Company. It was also supported in part by ACE, one of the seven centers in JUMP 2.0, a Semiconductor Research Corporation (SRC) program sponsored by DARPA. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory, DARPA, or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation herein.

REFERENCES

[1] Y. Chen, T. Wang, Y. Lyu, Y. Hu, J. Li, T. Kimura, H. Zhao, Y. Hu, D. Kara, and T. Abdelzaher, "Spar: Self-supervised placement-aware representation learning for multi-node iot systems," *arXiv preprint arXiv:2505.16936*, 2025.

[2] O. Baris, Y. Chen, G. Dong, L. Han, T. Kimura, P. Quan, R. Wang, T. Wang, T. Abdelzaher, M. Bergés *et al.*, "Foundation models for cps-iot: Opportunities and challenges," *arXiv preprint arXiv:2501.16368*, 2025.

[3] J. Bian, A. Al Arafat, H. Xiong, J. Li, L. Li, H. Chen, J. Wang, D. Dou, and Z. Guo, "Machine learning in real-time internet of things (iot) systems: A survey," *IEEE Internet of Things Journal*, vol. 9, no. 11, pp. 8364–8386, 2022.

[4] D. Kara, T. Kimura, S. Liu, J. Li, D. Liu, T. Wang, R. Wang, Y. Chen, Y. Hu, and T. Abdelzaher, "Freqmae: Frequency-aware masked autoencoder for multi-modal iot sensing," in *Proceedings of the ACM Web Conference 2024*, 2024, pp. 2795–2806.

[5] D. Kreković, P. Krivić, I. P. Žarko, M. Kušek, and D. Le-Phuoc, "Reducing communication overhead in the iot-edge-cloud continuum: A survey on protocols and data reduction strategies," *Internet of things*, p. 101553, 2025.

[6] J. D. A. Correa, A. S. R. Pinto, and C. Montez, "Lossy data compression for iot sensors: A review," *Internet of Things*, vol. 19, p. 100516, 2022.

[7] J. Li, Y. Chen, R. Wang, T. Kimura, T. Wang, Y. Lyu, H. Zhao, B. Sun, S. Wu, Y. Hu, D. Kara, B. Tian, K. Nahrstedt, S. Diggavi, J. H. Kim, G. Kimberly, G. Wang, M. Wigness, and T. Abdelzaher, "RestoreML: Practical unsupervised tuning of deployed intelligent iot systems," in *2025 The 21st International Conference on Distributed Computing in Smart Systems and the Internet of Things (DCOSS-IoT)*. IEEE, 2025.

[8] T. Sztyler and H. Stuckenschmidt, "On-body localization of wearable devices: An investigation of position-aware activity recognition," in *2016 IEEE international conference on pervasive computing and communications (PerCom)*. IEEE, 2016, pp. 1–9.

[9] California Institute of Technology (Caltech), "Southern california seismic network," Other/Seismic Network, 1926.

[10] X. Ouyang, X. Shuai, J. Zhou, I. W. Shi, Z. Xie, G. Xing, and J. Huang, "Cosmo: contrastive fusion learning with small data for multimodal human activity recognition," in *Proceedings of the 28th Annual International Conference on Mobile Computing And Networking*, 2022, pp. 324–337.

[11] S. Deldari, H. Xue, A. Saeed, D. V. Smith, and F. D. Salim, "Cocoa: Cross modality contrastive learning for sensor data," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies*, vol. 6, no. 3, pp. 1–28, 2022.

[12] S. Liu, T. Kimura, D. Liu, R. Wang, J. Li, S. Diggavi, M. Srivastava, and T. Abdelzaher, "Focal: Contrastive learning for multimodal time-series sensing signals in factorized orthogonal latent space," *Advances in Neural Information Processing Systems*, vol. 36, pp. 47 309–47 338, 2023.

[13] R. Girdhar, A. El-Nouby, Z. Liu, M. Singh, K. V. Alwala, A. Joulin, and I. Misra, "Imagebind: One embedding space to bind them all," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 2023, pp. 15 180–15 190.

[14] X. Ouyang, J. Wu, T. Kimura, Y. Lin, G. Verma, T. Abdelzaher, and M. Srivastava, "Mmbind: Unleashing the potential of distributed and heterogeneous data for multimodal learning in iot," *arXiv preprint arXiv:2411.12126*, 2024.

[15] T. Kimura, X. Li, O. Hanna, Y. Chen, Y. Chen, D. Kara, T. Wang, J. Li, X. Ouyang, S. Liu *et al.*, "Infomae: Pair-efficient cross-modal alignment for multimodal time-series sensing signals," in *Proceedings of the ACM on Web Conference 2025*, 2025, pp. 3084–3095.

[16] Z. Li, Z. Rao, L. Pan, P. Wang, and Z. Xu, "Ti-mae: Self-supervised masked time series autoencoders," *arXiv preprint arXiv:2301.08871*, 2023.

[17] M. Goswami, K. Szafer, A. Choudhry, Y. Cai, S. Li, and A. Dubrawski, "Moment: A family of open time-series foundation models," *arXiv preprint arXiv:2402.03885*, 2024.

[18] Q. Liu, J. Ye, H. Liang, L. Sun, and B. Du, "Ts-mae: A masked autoencoder for time series representation learning," *Information Sciences*, vol. 690, p. 121576, 2025.

[19] D. Kara, T. Kimura, Y. Chen, J. Li, R. Wang, Y. Chen, T. Wang, S. Liu, and T. Abdelzaher, "Phymask: An adaptive masking paradigm for efficient self-supervised learning in iot," in *Proceedings of the 22nd ACM Conference on Embedded Networked Sensor Systems*, 2024, pp. 97–111.

[20] H. Xu, P. Zhou, R. Tan, M. Li, and G. Shen, "Limu-bert: Unleashing the potential of unlabeled data for imu sensing applications," in *Proceedings of the 19th ACM Conference on Embedded Networked Sensor Systems*, 2021, pp. 220–233.

[21] S. Mo, R. Salakhutdinov, L.-P. Morency, and P. P. Liang, "Iot-lm: Large multisensory language models for the internet of things," *arXiv preprint arXiv:2407.09801*, 2024.

[22] X. Ouyang and M. Srivastava, "Llmsense: Harnessing llms for high-level reasoning over spatiotemporal sensor traces," in *2024 IEEE 3rd Workshop on Machine Learning on Edge in Sensor Systems (SenSys-ML)*. IEEE, 2024, pp. 9–14.

[23] H. Xu, L. Han, Q. Yang, M. Li, and M. Srivastava, "Penetrative ai: Making llms comprehend the physical world," in *Proceedings of the 25th International Workshop on Mobile Computing Systems and Applications*, 2024, pp. 1–7.

[24] D. A. Huffman, "A method for the construction of minimum-redundancy codes," *Proceedings of the IRE*, vol. 40, no. 9, pp. 1098–1101, 1952.

[25] T. A. Welch, "A technique for high-performance data compression," *Computer*, vol. 17, no. 06, pp. 8–19, 1984.

[26] M. A. P. Putra, A. P. Hermawan, D.-S. Kim, and J.-M. Lee, "Data prediction-based energy-efficient architecture for industrial iot," *IEEE Sensors Journal*, vol. 23, no. 14, pp. 15 856–15 866, 2023.

[27] H. Wang, Z. Yemeni, W. M. Ismael, A. Hawbani, and S. H. Alsamhi, "A reliable and energy efficient dual prediction data reduction approach for wsns based on kalman filter," *IET Communications*, vol. 15, no. 18, pp. 2285–2299, 2021.

[28] G. M. Dias, B. Bellalta, and S. Oechsner, "A survey about prediction-based data reduction in wireless sensor networks," *ACM Computing Surveys (CSUR)*, vol. 49, no. 3, pp. 1–35, 2016.

[29] B. Pourghebleh and N. J. Navimipour, "Data aggregation mechanisms in the internet of things: A systematic review of the literature and recommendations for future research," *Journal of Network and Computer Applications*, vol. 97, pp. 23–34, 2017.

[30] M. Amarlingam, P. K. Mishra, P. Rajalakshmi, S. S. Channappayya, and C. S. Sastry, "Novel light weight compressed data aggregation using sparse measurements for iot networks," *Journal of Network and Computer Applications*, vol. 121, pp. 119–134, 2018.

[31] B. A. Begum and S. V. Nandury, "A survey of data aggregation protocols for energy conservation in wsn and iot," *Wireless Communications and Mobile Computing*, vol. 2022, no. 1, p. 8765335, 2022.

[32] Y.-H. Chen, N.-Y. Huang, Y.-H. Chu, M.-H. Li, R.-I. Chang, and C.-H. Wang, "Dynamic bounded-error data compression and aggregation in wireless sensor network," in *SENSORS, 2012 IEEE*. IEEE, 2012, pp. 1–4.

[33] D. Ramijak, A. Pal, and K. Kant, "Pattern mining based compression of iot data," in *Proceedings of the Workshop Program of the 19th International Conference on Distributed Computing and Networking*, 2018, pp. 1–6.

[34] M. I. Mohamed, W. Wu, and M. Moniri, "Adaptive data compression for energy harvesting wireless sensor nodes," in *2013 10th IEEE INTERNATIONAL CONFERENCE ON NETWORKING, SENSING AND CONTROL (ICNSC)*. IEEE, 2013, pp. 633–638.

[35] J. Liu, F. Chen, and D. Wang, "Data compression based on stacked rbm-ae model for wireless sensor networks," *Sensors*, vol. 18, no. 12, p. 4273, 2018.

[36] S. Yao, J. Li, D. Liu, T. Wang, S. Liu, H. Shao, and T. Abdelzaher, "Deep compressive offloading: Speeding up neural network inference by trading edge computation for network latency," in *Proceedings of the 18th conference on embedded networked sensor systems*, 2020, pp. 476–488.

[37] L. Klus, R. Klus, E. S. Lohan, C. Granell, J. Talvitie, M. Valkama, and J. Nurmi, "Direct lightweight temporal compression for wearable sensor data," *IEEE Sensors Letters*, vol. 5, no. 2, pp. 1–4, 2021.

[38] J. Azar, A. Makhoul, M. Barhamgi, and R. Couturier, "An energy efficient iot data compression approach for edge machine learning," *Future Generation Computer Systems*, vol. 96, pp. 168–175, 2019.

[39] S. Chen, J. Liu, K. Wang, and M. Wu, "A hierarchical adaptive spatio-temporal data compression scheme for wireless sensor networks," *Wireless Networks*, vol. 25, pp. 429–438, 2019.

[40] T. M. Cover, *Elements of information theory*. John Wiley & Sons, 1999.

[41] D. Bertsekas, A. Nedic, and A. Ozdaglar, *Convex analysis and optimization*. Athena Scientific, 2003, vol. 1.