CloudSense: Effective CSI Data Compression for Cloud-based WiFi Sensing in Human Pose Estimation Tasks

Anonymous CVPR submission

Paper ID 9265

Abstract

With the rising demand for device-free and privacy-preserving solutions, WiFi-based sensing has become an attractive approach for human pose estimation (HPE). However, high volumes of channel state information (CSI) data can quickly overwhelm edge resources, especially in large-scale applications. This paper addresses this need by introducing `CloudSense`*, an effective compression framework that leverages a vector quantization-based generative adversarial network (VQGAN) to advance the scalability of WiFi-based human sensing.* `CloudSense` *employs a VQGAN-learned codebook for efficient CSI compression, significantly minimizing data transmission costs while preserving the accuracy essential for reliable HPE tasks. We utilize the K-means algorithm to dynamically adjust compression bitrates to cluster a large-scale pre-trained codebook into smaller subsets. Furthermore, a Transformer model is incorporated to mitigate bitrate loss, enhancing robustness in unreliable network conditions. Extensive numerical results demonstrate that* `CloudSense` *significantly outperforms state-of-the-art codecs, reducing the compression of CSI from* 1.368 *Mb/s to* 0.768 *Kb/s with minimal reconstruction error and achieving* 84% *PCK*$_{50}$.

1. Introduction

WiFi-based sensing has recently attracted significant interest within the research community due to its cost-effectiveness, extensive infrastructure, and non-intrusive nature [21, 31, 51, 53]. A critical element of this technology is channel state information (CSI) [5], which captures detailed signal propagation characteristics between transmitting and receiving antennas over multiple subcarriers. This fine-grained data enables the detection of patterns associated with various human activities, including localization [53], human activity recognition (HAR) [31], gesture recognition [51], and even respiration monitoring [21].

Recent studies have explored the integration of deep learning (DL) into WiFi sensing, leveraging its potential to improve WiFi signal processing and human monitoring tasks. For instance, Zhuravchak *et al.* [56] introduced a long short-term memory (LSTM)-based approach for HAR, while WiGRUNT [16] employed ResNet with dual attention for gesture recognition. AFEE-MatNet [31] and Widar3.0 [49] tackled environmental dependencies through extensive pre-processing CSI data and advanced neural networks. However, these methods often require substantial computational resources, which presents challenges for resource-constrained edge devices. To address this, CSI data can be offloaded to cloud servers with greater computational resources, as shown in Fig. 1, but its high dimensionality and sampling rate create a heavy communication burden, potentially disrupting core WiFi functions [32]. Despite recent advancements, no research has fully addressed the combined challenges of CSI compression and sensing [1, 44]. The key limitations include high communication overhead due to decoding demands [44] and difficulties in achieving scalability and efficiency [1]. The existing approaches face significant limitations, including high communication overhead due to decoding demands [44] and difficulties in achieving scalability and efficiency [1]. These challenges prompt a critical question: "Can we develop an effective solution that effectively compresses CSI data to low bitrates at the edge while enabling accurate HPE and ensuring min-

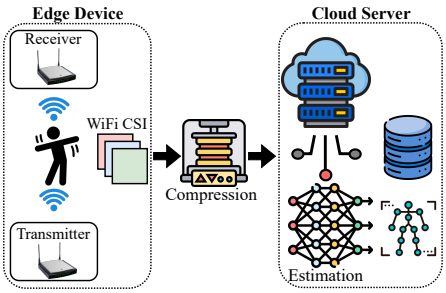


Figure 1. Illustration of the cloud-based WiFi sensing in HPE tasks.

imal reconstruction loss in the cloud?"

We demonstrate in this paper that the answer is surprisingly "Yes" with the proposed `CloudSense`, which achieves high compression rates while maintaining high-quality CSI reconstruction and reliable performance in HPE tasks. Inspired by the architecture of vector quantization-based generative adversarial network (VQGAN) [13], `CloudSense` comprises an edge encoder, a decoder in the cloud and a shared codebook. To improve the system efficiency, we apply K-means clustering to enable a large pre-trained codebook, creating a set of smaller codebooks that represent CSI data through different VQ index maps, allowing for flexible bitrates and adjustable reconstruction quality. Additionally, to mitigate bitstream loss in unstable transmission conditions, we leverage VQGAN's second-stage transformer [37] to predict missing indices from contextual ones, based on the underlying discrete distribution, effectively preventing reconstruction failures.

In summary, the main contributions of the paper are three-fold:

- We propose a novel `CloudSense` framework that achieves high-quality CSI data reconstruction and high accuracy in HPE tasks, compressing CSI data to very low bitrates through the shared codebook's learning capability within the VQGAN architecture.
- We develop a K-means clustering method to reduce the large codebook into smaller versions, enabling variable bitrates and adjustable reconstruction quality.
- For the first time, we propose a second-stage transformer to predict missing indices that help enhance model robustness by mitigating bit loss in challenging network conditions.

2. Related Work

WiFi-based HPE tasks: WiFi infrastructure combined with open-source tools has sparked interest in using CSI for human sensing. Numerous studies (*e.g.* [19, 23, 38]) aim to improve WiFi-based HPE accuracy using off-the-shelf devices, balancing cost-effectiveness and the limitation of low subcarrier counts. Recently, MetaFi [46] and its variants [55] addressed this by enhancing subcarrier resolution to 114 subcarriers, though the resulting high-dimensional CSI data increases transmission overhead. This is challenging for edge IoT devices with limited power and processing capacity, underscoring the need for efficient CSI compression prior to cloud transmission. Our approach introduces a novel method for compressing and transmitting CSI data to improve cloud-based HPE processing.

CSI data compression: The high dimensionality and sampling rate of CSI data impose significant transmission overhead, making compression essential for cloud transmission. For instance, with a 40 MHz WiFi setup (*e.g.* 114 subcarriers), three antenna pairs, and a sampling rate of 500 Hz, the transmission cost reaches 1.368 Mb/s. Traditional methods like compressive sensing [12] and LASSO [9] reduce feedback but struggle with full recovery due to channel matrix sparsity, while advanced algorithms with complex priors [41] also have recovery limitations. Although CSINet [27] employed autoencoders for improved CSI recovery, these methods fall short for CSI-based sensing, which requires both compression and discriminative feature representation. EfficientFi [44] explored VQVAE-based encoding but faced accuracy issues due to a fixed codebook, while RCSNet [1] aimed at real-time but lacked efficiency and scalability. Moreover, previous works assume ideal and complete bitstream reception, which may not be suitable for real-world conditions. `CloudSense`, by contrast, introduces a novel VQGAN-based coding framework for effective CSI compression, integrating K-means clustering and transformers [37] to optimize codebook efficiency, reduce index loss in transmission, and support precise CSI reconstruction for cloud-based HPE tasks.

VQ-based generative model: The development of VQ-VAE [35], which utilizes discrete image representations, has spurred interest in VQ-based generative models, typically implemented in two stages. First, an encoder-decoder quantizer is trained to capture and discretize image latents via a learned codebook. Then, a prior network models the distribution in discrete space. VQGAN [13] extended VQ-VAE by enhancing image quality through adversarial learning [25] and perceptual loss [24], using a transformer [37] as the prior model instead of pixel-CNN [34] to support high-resolution generation. Several works has further improved VQ model fidelity [26, 47, 52] and introduced bidirectional non-autoregressive transformers [4, 50] in the second stage. These models have found broad applications, from image and video generation [4, 15, 17, 20, 42, 47] to text-to-image synthesis [11, 29] and face restoration [18, 40]. Yet, despite these advancements, VQ-based models are largely unexplored for CSI compression. This work addresses this gap, showing how VQGAN can efficiently compress CSI data at extremely low rates.

3. Methodology

3.1. System Overview

Our goal is to optimize CSI data transmission for accurate reconstruction on the server, facilitating precise HPE from processed CSI. The model consists of two main components: an edge model at WiFi access points (APs) and a cloud model (see Fig. 2(a). By dual selective kernel [8], an Encoder E at APs extracts latent features $\mathbf{Z}$ from input CSI data $\mathbf{X}$. These features are then compressed to reduce data size using a specified compression ratio η, with $\mathbf{Z}$ mapped to VQ-indices $\mathbf{I}$ through nearest-neighbor matching in a CSI codebook $\mathbf{e}_k$ (Sec. 3.2) and inverted it back to quantized

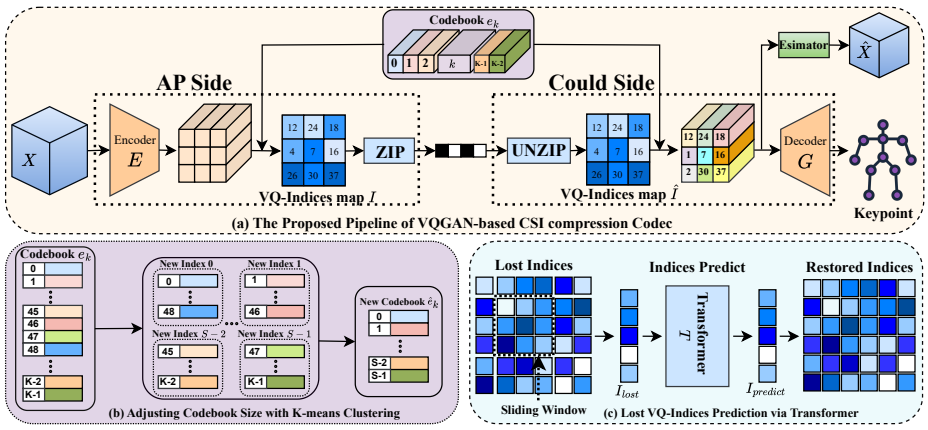


Figure 2. Illustration of the proposed `CloudSense` framework: (a) the overall compression pipeline, (b) the K-means clustering algorithm to adjust the codebook size for variable bitrates, and (c) a Transformer-based approach for predicting lost VQ indices.

vector $\mathbf{Z}_q$. The indices $\mathbf{I}$ are further compressed into a bitstream using lossless techniques (*e.g.* ZIP [10, 30]) before transmitting to the cloud. On the server, `CloudSense` decompresses and reconstructs $\hat{\mathbf{Z}}_q$ by referencing the shared codebook $\mathbf{e}_k$ to retrieve the corresponding codewords in matrix $\hat{\mathbf{I}}$, enabling two primary tasks: i) HPE via an Estimator E_s and ii) CSI reconstruction using Decoder G. For CSI data $\mathbf{X} \in \mathbb{R}^{\frac{F}{M} \times \frac{T}{M} \times C}$, where F, T, and C denote frequency, time, and channel, respectively, the compression pipeline includes five key components:

- **Encoder** $E(\cdot;\theta_E)$ transforms the input signal $\mathbf{X} \in \mathbb{R}^{\frac{F}{M} \times \frac{T}{M} \times C}$ into a latent representation $\mathbf{Z} \in \mathbb{R}^{\frac{F}{M} \times \frac{T}{M} \times D}$, where D is the dimensionality of the latent codes.
- **Codebook** $\mathbf{e}_k \in \mathbb{R}^D, k \in \{1, 2, \cdots, K\}$ maps the latent representation $\mathbf{Z}$ to a sequence of VQ indices $\mathbf{I}$ and reconstructs it into a quantized version $\mathbf{Z}_q \in \mathbb{R}^{\frac{F}{M} \times \frac{T}{M} \times D}$ using the nearest neighbor lookup. A K-means clustering algorithm compresses the codebook to smaller subsets (see Fig. 2(b)).
- **Decoder** $G(\cdot;\theta_G)$ acts as a generator within the GAN framework, converting the quantized latent representation $\hat{\mathbf{Z}}_q$ back into reconstructed CSI data $\hat{\mathbf{X}} \in \mathbb{R}^{F \times T \times C}$.
- **Estimator** $E_s(\cdot;\theta_R)$ produces HPE based on the quantized latent representations $\hat{\mathbf{Z}}_q$, linking the compression pipeline with HPR tasks.
- **Transformer** T predicts missing indices using the context of existing indices, as shown in Fig. 2(c).

3.2. VQ-indices Compression

Unlike existing compression methods [6, 7, 14, 33, 43] that adopt scalar quantization, our approach integrates CNN inductive biases and principles from neural discrete representation learning [13].

VQ-indices process: Our model includes an encoder E and a decoder G that jointly represent CSI using codes from a learned discrete codebook $\mathbf{e}_k \in \mathbb{R}^D, \forall k \in \{1, 2, \cdots, K\}$. The codebook $\mathbf{e}_k$ encodes the latent representation $\mathbf{Z}$ by replacing each vector position with the index of the nearest vector in the codebook, based on Euclidean distance. This process produces a compressed version of the latent representation with minimal loss of quality:

$$\mathbf{I}_{ij} = \underset{k \in 1,2,\ldots,K}{\texttt{argmin}} \parallel \mathbf{Z}_{ij} - \mathbf{e}_k \parallel \tag{1}$$

where i and j denote the positions of vector $\mathbf{Z}_{ij}$, with $\mathbf{I}_{ij}$ representing the corresponding index. We obtain $\mathbf{Z}_q$ through the quantization function $\mathbf{q}(\cdot)$, which assigns each element $\mathbf{Z}_{ij}$ its index $\mathbf{I}_{ij}$:

$$\mathbf{Z}_q = \mathbf{q}(\mathbf{Z}) = \{\mathbf{e}_k \mid \mathbf{I}_{ij}\} \in \mathbb{R}^{\frac{F}{M} \times \frac{T}{M} \times D}. \tag{2}$$

We then use the widely adopted ZIP lossless compression method [10, 30] to compress VQ indices $\mathbf{I}$ into bitstreams, further reducing data size.

VQ loss function: After decoding, the reconstructed latent vectors $\hat{\mathbf{Z}}_q$ are generated by retrieving their corresponding code words based on their indices. The decoder G synthesizes the reconstructed CSI data $\hat{\mathbf{X}} \approx \mathbf{X}$ as

$$\hat{\mathbf{X}} = G(\mathbf{Z}_q) = G(\mathbf{q}(E(\mathbf{X}))). \tag{3}$$

Backpropagation through the non-differentiable quantization operation in (3) is managed with a straight-through gradient estimator. This allows gradients to flow from the decoder to the encoder [2] and enables end-to-end training of both the model and codebook via the following loss function:

$$\begin{aligned}\mathcal{L}_{\texttt{VQ}}(E, D, \hat{e}_s) = ||\mathbf{X} - \hat{\mathbf{X}}||_2^2 + ||\texttt{sg}[E(\mathbf{X})] - \mathbf{Z}_q||^2 \\ + \beta||\texttt{sg}[\mathbf{Z}_q] - E(\mathbf{X})||_2^2\end{aligned} \tag{4}$$

where $sg[\cdot]$ indicates the stop-gradient operation, and $||\texttt{sg}[\mathbf{Z}_q] - E(\mathbf{X})||_2^2$ is the "commitment loss" with weighting factor β [36].

Adjusting variable bitrates based on K-means: The effectiveness of the codebook is crucial for the overall compression efficiency of the system. To enhance this, we have implemented a rate control strategy that allows for the dynamic adjustment of compression bitrates by modifying the codebook size. Specifically, we apply the K-means algorithm to cluster large-scale codebooks, identifying centroids for each cluster. These centroids serve as initial values for the vectors in the resized codebook, allowing us to generate new codebooks of varying sizes, each associated with a distinct set of VQ indices for the compressed data representation.

This approach enables flexible codebook resizing while preserving quality. Specifically, the K-means algorithm begins by randomly selecting centroids from the pre-trained large-scale codebook $\mathbf{e}_k$. It then calculates the Euclidean distance between each codebook vector and the centroids, assigning each vector to the nearest centroid as follows:

$$\mathbf{C}_s = \underset{s\in 1,2,\ldots,S}{\texttt{argmin}} \parallel \mathbf{e}_k - \hat{\mathbf{e}}_s \parallel^2 \tag{5}$$

where $\mathbf{C}_s$ represents the s-th cluster, $\hat{\mathbf{e}}_s$ denotes the centroid of that cluster, and S (with $S < K$) denotes the new codebook size. Following this, the mean of the codebook vectors within each cluster C_j is recalculated, updating the cluster centroids accordingly as $\hat{\mathbf{e}}_s = \frac{1}{|C_s|}\sum_{\mathbf{e}\in C_s}\mathbf{e}$.

These steps are repeated iteratively until the clustering minimizes the associated cost function:

$$\min J(\mathbf{e}_k;\hat{\mathbf{e}}_s) = \min\left(\frac{1}{K}\sum_{k=1}^{K}\|\mathbf{e}_k - \hat{\mathbf{e}}_s\|^2\right). \tag{6}$$

As a result, the newly created codebook $\hat{\mathbf{e}}_s$ from the K-means clustering algorithm can act as an initial point for further fine-tuning, accelerating convergence in the subsequent optimization process.

3.3. Learning Objective

Training strategy: In addition to $\mathcal{L}_{\texttt{VQ}}$, we integrate VQ-GAN, an enhanced version of the original VQVAE, to incorporate adversarial loss and perceptual loss into the proposed approach. This integration is crucial for maintaining high perceptual quality, even at higher compression rates [13]. Specifically, we replace the L_2 loss in Eq. (4) with perceptual loss [24] and introduce a patch-based discriminator D [22], which uses adversarial training to differentiate between real and reconstructed CSI data, leading to

$$\mathcal{L}_{\texttt{GAN}}(\{E,G,\hat{e}_s\},D) = [\log D(\mathbf{X}) + \log(1 - D(\hat{\mathbf{X}}))]. \tag{7}$$

Furthermore, `CloudSense` aims to improve the accuracy of HPE tasks by employing the loss function that measures the distance between the predicted $\hat{\mathbf{Y}}$ and true label $\mathbf{Y}$:

$$\mathcal{L}_{\texttt{keypoint}} = \parallel E_s(\hat{\mathbf{Z}}_q) - \mathbf{Y} \parallel_2^2 = \parallel \hat{\mathbf{Y}} - \mathbf{Y} \parallel_2^2. \tag{8}$$

Algorithm 1 Training and Deployment of `CloudSense`

Input: Labelled CSI data $(\mathbf{X}, \mathbf{Y})$, model weights θ_E, θ_G, θ_{E_s}, CSI codebook $\mathbf{e}_k$, and number of iterations N_{iter}.
Initialize: Encoder E, decoder G, and estimator E_s with parameters θ_E, θ_G, and θ_{E_s}, respectively.
While $i \leq N_{\text{iter}}$ **do**
1. Input the labelled data $(\mathbf{X}, \mathbf{Y})$;
2. Extract features $\mathbf{Z}$ from $\mathbf{X}$ using the encoder $E(\mathbf{X})$. Map $\mathbf{Z}$ to VQ-indices map $\mathbf{I}$ using codebook $\mathbf{e}_k$, and then retrieve quantized features $\mathbf{Z}_q$;
3. Compress $\mathbf{I}$ into a bitstream with ZIP and transmit to the cloud;
4. Decompress the bitstream on the cloud side to retrieve the VQ-indices map $\hat{\mathbf{I}}$ by UNZIP;
5. Restore the quantized feature $\mathbf{Z}_q$ from $\hat{\mathbf{I}}$;
6. Reconstruct CSI data $\hat{\mathbf{X}}$ using decoder G and predict results with estimator E_s;
7. Update encoder θ_E and codebook $\mathbf{e}_s$ by solving (4) with K-means clustering;
8. Update θ_{E_s} and θ_G by minimizing (7) and (8).

▷ Deploy E at edge side, G and E_s at cloud server;
▷ Store the codebook $\mathbf{e}_k$ on both sides.

Overall, our approach employs three primary objectives: $\mathcal{L}_{\texttt{VQ}}$, $\mathcal{L}_{\texttt{GAN}}$, and $\mathcal{L}_{\texttt{Keypoint}}$. These objectives are strategically crafted to achieve the intended performance, as verified through extensive simulations.

Algorithm summary: We are now in position to summarize the `CloudSense` algorithm, providing a clear and accessible outline of the model optimization process. The objective is to find the optimal compression model $\mathcal{O}^* = \{E^*, G^*, E_s^*, \hat{\mathbf{e}}_s^*\}$, detailed as:

$$\begin{aligned}\mathcal{O}^* = \arg\min_{E,G,\hat{\mathbf{e}}_s}\max_{D}\mathbb{E}_{x\sim p(x)}\Big[&\mathcal{L}_{\texttt{VQ}}(E,G,\hat{\mathbf{e}}_s \\ &+ \lambda\mathcal{L}_{\texttt{GAN}}(\{E,G,\hat{\mathbf{e}}_s\},D) + \mathcal{L}_{\texttt{keypoint}}(E_s)\Big]\end{aligned} \tag{9}$$

where λ is calculated as

$$\lambda = \frac{\nabla_{G_L}\mathcal{L}_{\texttt{rec}}}{\nabla_{G_L}[\mathcal{L}_{\texttt{GAN}}] + \delta} \tag{10}$$

with $\mathcal{L}_{\texttt{rec}}$ and $\nabla_{G_L}[\cdot]$ being the perceptual loss [48] and the gradient of the last layer L-th of the decoder, respectively. $\delta = 10^{-6}$ is used to maintain numerical stability. Specifically, we illustrate the `CloudSense` training and deployment in Algorithm 1. Using existing CSI data, we first train the `CloudSense` model offline, then deploy the feature extractor E at WiFi AP, while the cloud server hosts the decoder G and Estimator E_s. The trained CSI codebook is stored on both the WiFi APs and the cloud server for efficient compression and reconstruction.

Discussion: `CloudSense` demonstrates superior performance, surpassing current CSI compression methods due to key advantages: First, `CloudSense`'s multitask learning framework integrates recognition and reconstruction objectives, enabling richer semantic feature extraction and a

more robust representation. Secondly, the learnable codebook functions as both an encoder and a decoder, capturing CSI patterns more effectively for accurate reconstruction, unlike traditional methods that rely on separate quantization optimized with the K-means algorithm. Furthermore, `CloudSense` uses the Transformer approach with the loss function $\mathcal{L}_T$ to minimize discrepancies between the index map $\mathbf{I}$ and its predicted counterpart $\hat{\mathbf{I}}$ in unstable transmission environments, as detailed in the next section.

3.4. Lost VQ-indices Prediction

Bitstream loss can lead to missing index information $\hat{\mathbf{I}}$, causing decoding errors. To mitigate this, we propose a Transformer-based approach in the second stage of the proposed model which aims to predict missing indices during decoding to improve the fidelity of the reconstructed bitstream.

Transformer approach: As discussed in Section 3.2, the input CSI data $\mathbf{X}$ can be represented by a map of the VQ indices $\mathbf{I}$, derived from its encodings. Specifically, the quantized representation of the CSI data $\mathbf{X}$ is defined as $\mathbf{Z}_q = \mathbf{q}(E(\mathbf{X})) \in \mathbb{R}^{\frac{F}{M}\times\frac{T}{M}\times D}$, which corresponds to a sequence of indices $\mathbf{I}\in (0,...,S)^{\frac{F}{M}\times\frac{T}{M}}$ from the codebook. Each index is obtained by substituting the code with its corresponding index in the codebook $\mathbf{e}_s$:

$$\mathbf{I}_{ij} = k, \text{ such that } (\mathbf{Z}_q)_{ij} = \mathbf{Z}_k. \tag{11}$$

By mapping indices back to their respective entries in the codebook, the quantized features $\hat{\mathbf{Z}}_q = \hat{\mathbf{Z}}_{e_{ij}}$ can be easily reconstructed, and the CSI data is decoded as $\hat{\mathbf{X}} = G(\hat{\mathbf{Z}}_q)$.

Once an order is established for the indices in $\mathbf{I}$, the generation of CSI can be framed as an autoregressive prediction task of the next index. A second-stage transformer is trained to predict the probability distribution of each subsequent index $p(\mathbf{I}_i|\mathbf{I}_j)$ with $j < i$. The goal is to maximize the log-likelihood of the data representation, such as

$$\mathcal{L}_{\mathrm{T}} = \mathbb{E}_{x\sim p(x)}\left[-\log p(\mathbf{I})\right] \tag{12}$$

where $p(\mathbf{I}) = \prod_i p(\mathbf{I}_i|\mathbf{I}_j)$. To simulate the potential loss of indices during transmission, we apply a binary mask $M = [m_i]_{i=1}^{N}$ as follows: If $m_i = 1$, the index $\mathbf{I}_i$ is replaced by a special `[mask]` token to indicate its loss; if $m_i = 0$, $\mathbf{I}_i$ remains unchanged. The mask ratio $\alpha \in [0, 1]$ controls the fraction of masked indices, denoted $\mathbf{I}_{\text{lost}}$ as $\alpha \cdot N$. During the storage phase, as shown in Fig. 2(c), the Transformer predicts the probabilities of all potential codebook indices for each masked position i to allow reconstruction of lost indices.

Sliding window: To address the transformer's input sequence length limit N, we apply a sliding window approach, where a window of size $K \times K$ is centered on $\mathbf{I}_i$ for each prediction, as illustrated in Fig. 3. Only indices j within this K-sized window are utilized as input, supporting the transformer's autoregressive function and aligning with the decoding process. This strategy improves the effective use of the spatial structure present in the CSI data. The predicted index $\hat{\mathbf{I}}_i$ is sampled and the reconstructed CSI data $\hat{\mathbf{X}}$ is generated by passing the complete predicted index sequence $\mathbf{I}_{\text{predict}}$ through the decoder G.

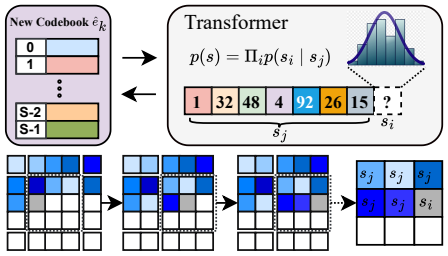


Figure 3. The Transformer process deployed in the second stage.

4. Experiments

4.1. Experimental Settings

Datasets: We consider two datasets: MM-Fi [45] and Wi-Pose[45]. The MM-Fi dataset includes pose annotations with 17 skeleton points collected from a camera sensor and WiFi CSI data from 40 participants, covering 27 action categories across 14 daily activities and 13 rehabilitation exercises. In contrast, Wi-Pose provides pose annotations with 18 skeleton points and WiFi CSI data for 12 distinct actions performed by 12 volunteers, with data randomly split into training and testing sets.

Implementation details: The proposed approach and baselines are implemented in PyTorch and trained on a GeForce RTX 4070 for 50 epochs using the Adam optimizer, with a batch size of 128, a learning rate of 0.001, and momentum of 0.9. To achieve the desired bitrate range, we apply K-Means clustering to reduce the codebook size, selecting from $2, 4, 8, 16, 32, 64, 128, 256, 512, 1024$. The embedding dimension D is set to 256, and the weight λ is 0.5.

Criterion: For evaluation, we assess both CSI restoration quality and HPE task accuracy relative to the compression rate η. Restoration quality is measured using Normalized Mean Squared Error (NMSE) in decibels (dB) [1, 44]. For HPE task accuracy, we apply two common metrics: i) Percent Correct Keypoint (PCK) [39] and ii) MPJPE [39]. The original communication cost is 4 bytes per amplitude data point, totaling (total original data size) $\times$ 4 bytes. In contrast, `CloudSense` transmits only $\mathrm{K}\log_2\mathrm{K}$ bytes, with K as the number of embeddings, so the compression rate η is the ratio of the original cost to that of `CloudSense`.

Baselines: We compare `CloudSense` with five SOTA approaches: three basic CSI feedback models—vanilla

LASSO l_1-solver [9], BM3D-AMP [28], and CSINet [27]—and two advanced compressive systems, EfficientFi [44] and RSCNet [1]. For HPE tasks, we benchmark against three baselines: MetaFi++ [55], PerUnet [54], and WiSPPN [38].

4.2. Results

4.2.1 Performance Comparison with Baselines

Compression Task. We evaluate `CloudSense`'s compression performance and compare it to existing CSI compression models using NMSE, PCK_{20}, and MPJPE metrics. As shown in Tab. 1, `CloudSense` outperforms existing CSI feedback models, including LASSO, DeepCMC, and CSINet, at the same compression rate of $\eta = 4$, and even that achieves better results at $\eta = 570$ while its counterparts operating at $\eta = 4$ while requiring only a negligible additional inference time, on both the MM-Fi and WiPose datasets. When compared to currently CSI-based sensing models, such as RCSNet and EfficientFi, our approach consistently outperforms these methods across all metrics. Notably, `CloudSense` outperforms EfficientFi, achieving a PCK_{20} of 36.93% compared to 16.75% at a compression rate of $\eta = 1710$, while require less inference time. This indicates that our model comprehensively outperforms existing SOTA models across both datasets, showcasing `CloudSense`'s adaptability to diverse conditions.

Human Pose Estimation Task. Tab. 2 provides a comparison between `CloudSense` and existing HPE approaches. Our method primarily focuses on relative pose accuracy, and it significantly outperforms the SOTA approaches for all evaluation criteria, particularly excelling in the low PCK_a as well as in MPJPE metrics. These superior results are achieved with remarkably low computational costs. Meanwhile, PerUnet also achieves good performance on both datasets, but with the cost of high complexity, (*e.g.* approximately 34M parameters). Notably, when compressing data at $\eta = 16$, `CloudSense` remains the impressive results, 46.59% at PCK_{20}, equivalent to the performance of existing HPE methods. This indicates that our encoder and decoder are effectively aligned to compress CSI data for HPE tasks, outperforming current DL models in CSI sensing.

4.2.2 Visualization Results

T-SNE Visualization. As illustrated in Fig. 4a (left), it is a challenge to distinguish different actions due to the similar trends in human poses, making it difficult to classify them as actions in HAR tasks. In contrast, the quantized features of the proposed `CloudSense`, shown in Fig. 4a (right), are distributed into multiple distinct regions (*i.e.*, clusters) within the compressed space. However, because of the inherent similarity in human poses, these clusters lack the sharp boundaries typical in standard classification tasks. Meanwhile, the quantized features for labeled subjects, shown in Fig. 4b (right), are more clearly separated into several discriminative regions, representing a substantial improvement compared to the raw CSI data in Fig. 4b (left).

Pose Visualization. In Fig. 5, we display qualitative results illustrating human skeleton visualizations at different compression rates from the MM-Fi dataset. Our focus is on generating 2D poses, which are then extended to 3D by incorporating a constant vector as the third dimension. Our method reliably generates accurate human poses at low compression rates and maintains pose integrity even under strict compression conditions.

4.2.3 Ablation Study

Hyperparameter Sensitivity. We examine the embedding dimension D and the weight λ with a compression rate of $\eta = 86$, assessing the PCK_{20} accuracy. As shown in Fig. 6, `CloudSense` achieves optimal performance with $D = 128$ and $\lambda = 0.5$, yielding an accuracy of about 45%, and with $D = 256$ and $\lambda = 0.1$, yielding around 46.5%. The lowest accuracy is observed with $D = 64$, particularly at $\lambda = 0.1$. Based on these results, we select $D = 256$ and $\lambda = 0.1$ for all further evaluations in this paper.

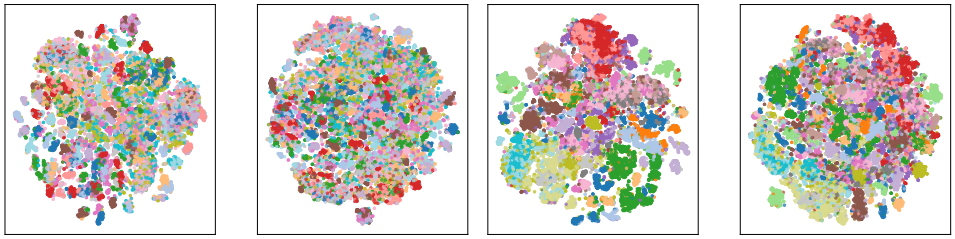
Learning Objective. Tab. 3 evaluates `CloudSense`'s performance across various objective strategies, considering the loss functions from [1, 3, 13, 44]. Obtained results indicate that both our objective strategy and that in [3] achieve the highest performance, though [3] requires a substantially higher inference time. Meanwhile, our approach has the second-fastest inference time, just behind [13], but the performance of [13] is significantly lower compared to ours.

4.3. Analysis and Discussion

Lost VQ-indices Prediction. To assess the Transformer's efficacy in recovering lost indices, we conducted additional simulations by introducing varying error rates, denoted as ϵ, into the indices map, $\hat{\mathbf{I}}$ and the compression rate is fixed at 86. We analyzed performance trends with changing error rates and compared the results with `CloudSense`'s performance without the Transformer, highlighting the Transformer's impact on index recovery. Tab. 4 shows that model performance declines as error rates increase, but this reduction remains minimal, suggesting that the Transformer effectively compensates for lost indices even at a strict error rate of $\epsilon = 0.9$. In contrast, `CloudSense` without the Transformer shows poor performance even at low error rates, with a marked decline as ϵ rises to 0.9, achieving only approximately 14.23% at PCK_{20}. We next conduct simulations of our approach, which features a self-adaptive codebook using K-means clustering, with codebook sizes ranging from

Table 1. Compression performance comparison between different schemes on both MM-Fi and WiPose datasets("N/A" in the HPE accuracy column denotes methods not applicable to this task and **IF** indicates inference time).

		MM-Fi					WiPose				
Method	η	**NMSE(dB)**	**PCK$_{30}$**	**PCK$_{20}$**	**MPJPE**	**IF**	**NMSE(dB)**	**PCK$_{30}$**	**PCK$_{20}$**	**MPJPE**	**IF**
LASSO [9]	4	-12.42	N/A	N/A	N/A	179.32		N/A	N/A	N/A	
DeepCMC [28]	4	-12.58	N/A	N/A	N/A	112.97		N/A	N/A	N/A	
CSINet[27]	4	-4.75	N/A	N/A	N/A	165.12		N/A	N/A	N/A	
	16	-4.25									
	64	-4.05									
	128	-3.72									
RCSNet[1]	4	-7.02	15.21	7.64	346.57	189.13					
	16	-6.53	14.92	7.35	377.19						
	32	-6.36	12.58	6.06	404.52						
	64	-6.17	9.62	5.96	426.68						
EfficientFi[44]	4	-11.94	48.01	24.03	206.62	255.79					
	16	-11.12	46.15	23.58	217.14						
	86	-10.26	44.68	22.18	221.18						
	570	-6.45	40.16	19.34	238.28						
	1710	-3.28	35.62	16.75	257.78						
Ours	4	-15.14	66.16	48.93	156.72	229.38					
	16	-14.82	64.24	46.59	163.21						
	86	-14.42	63.15	44.12	166.87						
	215	-13.74	62.89	42.82	170.98						
	570	-13.17	62.04	39.75	177.26						
	1710	-10.72	56.85	36.93	183.67						



(a) The T-SNE embedding of the raw CSI data and the quantized feature for action label.

(b) The T-SNE embedding of the raw CSI data and the quantized feature for subject label.

Figure 4. Visualization via T-SNE for initial raw CSI representation and Compressed CSI embedding.

Table 2. HPE tasks performance comparison between different schemes on both MM-Fi and WiPose datasets.

MM-Fi					
Method	η	**PCK$_{30}$**	**PCK$_{20}$**	**MPJPE**	**Params**
WiSPPN [38]	N/A	63.21	45.41	166.59	26.78M
PerUnet [54]	N/A	50.12	67.34	154.66	34.51M
MetaFi++ [55]	N/A	45.46	64.44	164.45	26.42M
Ours	2	68.22	54.12	152.16	1.66M
	4	64.24	48.93	156.72	
	16	60.87	46.59	163.21	
WiPose					
Method	η	**PCK$_{5}$**	**PCK$_{10}$**	**MPJPE**	**Params**
WiSPPN [38]	N/A	52.95	64.16	30.37	26.33M
PerUnet [54]	N/A	63.07	71.77	17.12	33.85M
MetaFi++ [55]	N/A	53.64	66.72	18.62	25.58M
Ours	150				
	750				
	1710				

Table 3. Performance Comparison of Different Loss Functions: Best in **bold** and second best in underlined.

Loss Function	**NMSE**	**PCK$_{20}$**	**MPJPE**	**IF**
[13]	-11.54	32.27	221.25	**204.62**
[1]	-11.67	33.14	218.98	205.47
[44]	-13.18	38.62	177.95	229.46
[3]	**-14.26**	**41.23**	**166.74**	267.51
Ours	-14.24	41.21	166.87	229.38

16 to 128, while maintaining a threshold performance of PCK$_{20} > 35\%$. As shown in Tab. 5, the compression rate η increases as the error rate ϵ rises. However, the decrease in η is negligible when ϵ increases from 0.1 to 0.5, due to the effectiveness of the Transformer. On the other hand, when ϵ reaches 0.9, the Transformer becomes less effective. This indicates that the Transformer is most effective when the error rate operates at lower values.

Sliding Window Size. We investigate the effect of vary-

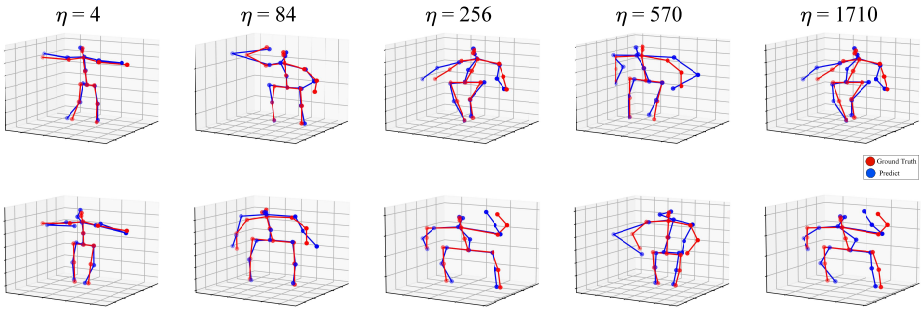


Figure 5. Visualization of the human pose landmarks generated by the vision model (red) and WiFi model (blue) on the MM-Fi at different compression rates.

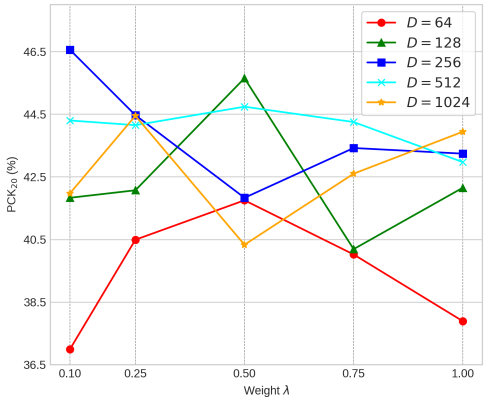


Figure 6. Sensitivity Analysis of Hyper-parameters: Embedding Dimension D and Weight Parameter λ in Eq. 10.

Table 4. Comparison of CloudSense performance with and without Transformer in the presence of lost indices.

	CloudSense			CloudSense Without Transformer		
ϵ	NMSE	PCK_{20}	MPJPE	NMSE	PCK_{20}	MPJPE
0.1	-8.02	39.16	169.45	-6.14	31.64	190.26
0.3	–7.24	36.12	178.63	-5.26	26.63	198.74
0.5	-6.68	32.69	186.54	-4.56	21.16	218.65
0.7	-6.04	30.14	192.68	-3.94	18.63	227.98
0.9	-5.65	26.89	198.86	3.01	14.23	242.12
IF		229.38			207.16	

Table 5. Assessment of CloudSense performance with integrated Transformer and adaptive codebook size using K-means.

ϵ	η	NMSE	PCK_{20}	MPJPE
0.1	215	-6.29	36.51	148.54
0.3	176	**-6.44**	**36.78**	**148.52**
0.5	96	-6.21	35.51	149.58
0.7	64	-5.51	35.78	149.73
0.9	16	-5.29	35.01	157.03

ing sliding window sizes on model performance. Given the sequence length constraints in the transformer's attention mechanism, we test window sizes of 2×2, 3×3, 4×4, and 5×5, with a fixed compression rate $\eta = 86$. As shown in Tab. 6, performance initially improves as window size increases, then declines. The 2×2 window yields the lowest performance, while the optimal result of 164.18 mm is achieved with a 3×3 window. The second-best performance, 172.14 mm, is associated with 5×5, with a minimal gap between 3×3 and 5×5. Thus, 3×3 is recommended to balance accuracy and efficiency.

Table 6. Analysis of the impact of the Transformer's sliding window on CloudSense performance.

$K \times K$	NMSE	PCK_{30}	PCK_{20}	MPJPE
2×2	-7.782	59.152	35.086	168.45
3×3	**-8,282**	**63.158**	**42.961**	**164.18**
4×4	-7,582	60.238	36.562	170.69
5×5	-8,082	62.881	38.163	166.14

Table 7. Ablation Studies on the Effectiveness of K-means Clustering on the MM-Fi Dataset.

Methods	η	NMSE	PCK_{20}	MPJPE
Random initialization	86	-7.716	35.184	181.26
Ours		-8.282	42.961	164.18

Effectiveness of K-means Clustering. To assess the impact of using K-means clustering as the initial step in our fine-tuning process, we perform an ablation study by initializing the codebook randomly rather than with K-means. The results, shown in Tab. 7, reveal that while the randomly initialized codebook still achieves substantial bitrate reduction, the quality of reconstruction and performance on PCK tasks are noticeably lower. This suggests that the expressive capability of the pre-trained codebook acts as a strong prior, playing a critical role in maintaining reconstruction quality.

5. Conclusion

This paper introduced CloudSense, a compression framework that leverages a VQGAN to enhance the scalability of WiFi-based human sensing. Our method achieves state-of-the-art accuracy and reconstruction quality, efficiently com-

pressing original CSI data into lower-bit representations while requiring significantly less inference time compared to existing approaches. This success is due to the flexible integration of adaptive vector quantization using K-means and robust information recovery via a Transformer model. Through validation on the MM-Fi and WiPose datasets, we have demonstrated CloudSense's robustness and generalizability across diverse and challenging scenarios. These achievements pave the way for large-scale applications, particularly on resource-constrained edge devices.

References

[1] Borna Barahimi, Hakam Singh, Hina Tabassum, Omer Waqar, and Mohammad Omer. Rscnet: Dynamic csi compression for cloud-based wifi sensing. In *ICC 2024 - IEEE International Conference on Communications*, pages 4179–4184, 2024. 1, 2, 5, 6, 7

[2] Yoshua Bengio. Estimating or propagating gradients through stochastic neurons. *ArXiv*, abs/1305.2982, 2013. 3

[3] Shiyue Cao, Yueqin Yin, Lianghua Huang, Yu Liu, Xin Zhao, Deli Zhao, and Kaiqi Huang. Efficient-vqgan: Towards high-resolution image generation with efficient vision transformers. *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 7334–7343, 2023. 6, 7

[4] Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T. Freeman. Maskgit: Masked generative image transformer. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11305–11315, 2022. 2

[5] Rongjie Che and Honglong Chen. Channel state information based indoor fingerprinting localization. *Sensors (Basel, Switzerland)*, 23, 2023. 1

[6] Zhengxue Cheng, Heming Sun, Masaru Takeuchi, and J. Katto. Learned image compression with discretized gaussian mixture likelihoods and attention modules. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7936–7945, 2020. 3

[7] Yoojin Choi, Mostafa El-Khamy, and Jungwon Lee. Variable rate deep image compression with a conditional autoencoder. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3146–3154, 2019. 3

[8] Toan D. Gian, Tien Dac Lai, Thien Van Luong, Kok-Seng Wong, and Van-Dinh Nguyen. Hpe-li: Wifi-enabled lightweight dual selective kernel convolution for human pose estimation. In Aleš Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol, editors, *Computer Vision – ECCV 2024*, pages 93–111, Cham, 2025. Springer Nature Switzerland. 2

[9] Ingrid Daubechies, Michel Defrise, and Christine De Mol. An iterative thresholding algorithm for linear inverse problems with a sparsity constraint. *Communications on Pure and Applied Mathematics*, 57, 2003. 2, 6, 7

[10] P. Deutsch. Rfc1951: Deflate compressed data format specification version 1.3. 1996. 3

[11] Ming Ding, Zhuoyi Yang, Wenyi Hong, Wendi Zheng, Chang Zhou, Da Yin, Junyang Lin, Xu Zou, Zhou Shao, Hongxia Yang, and Jie Tang. Cogview: Mastering text-to-image generation via transformers. In *Neural Information Processing Systems*, 2021. 2

[12] David L. Donoho, Arian Maleki, and Andrea Montanari. Message passing algorithms for compressed sensing: Ii. analysis and validation. In *2010 IEEE Information Theory Workshop on Information Theory (ITW 2010, Cairo)*, pages 1–5, 2010. 2

[13] Patrick Esser, Robin Rombach, and Björn Ommer. Taming transformers for high-resolution image synthesis. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12868–12878, 2020. 2, 3, 4, 6, 7

[14] Chenjian Gao, Tongda Xu, Dailan He, Hongwei Qin, and Yan Wang. Flexible neural image compression via code editing. *ArXiv*, abs/2209.09244, 2022. 3

[15] Songwei Ge, Thomas Hayes, Harry Yang, Xi Yin, Guan Pang, David Jacobs, Jia-Bin Huang, and Devi Parikh. Long video generation with time-agnostic vqgan and time-sensitive transformer. In Shai Avidan, Gabriel Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner, editors, *Computer Vision – ECCV 2022*, pages 102–118, Cham, 2022. Springer Nature Switzerland. 2

[16] Yu Gu, Xiang Zhang, Yantong Wang, Meng Wang, Huan Yan, Yusheng Ji, Zhi Liu, Jianhua Li, and Mianxiong Dong. Wigrunt: Wifi-enabled gesture recognition using dual-attention network. *IEEE Transactions on Human-Machine Systems*, 52(4):736–746, 2022. 1

[17] Yuchao Gu, Xintao Wang, Yixiao Ge, Ying Shan, Xiaohu Qie, and Mike Zheng Shou. Rethinking the objectives of vector-quantized tokenizers for image synthesis. *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7631–7640, 2022. 2

[18] Yuchao Gu, Xintao Wang, Liangbin Xie, Chao Dong, Gen Li, Ying Shan, and Ming-Ming Cheng. Vqfr: Blind face restoration with vector-quantized dictionary and parallel decoder. In Shai Avidan, Gabriel Brostow, Moustapha Cissé, Giovanni Maria Farinella, and Tal Hassner, editors, *Computer Vision – ECCV 2022*, pages 126–143, Cham, 2022. Springer Nature Switzerland. 2

[19] Lingchao Guo, Zhaoming Lu, Xiangming Wen, Shuang Zhou, and Zijun Han. From signal to image: Capturing fine-grained human poses with commodity WiFi. *IEEE Commun. Lett.*, 24(4):802–806, 2020. 2

[20] Wenyi Hong, Ming Ding, Wendi Zheng, Xinghan Liu, and Jie Tang. Cogvideo: Large-scale pretraining for text-to-video generation via transformers. *ArXiv*, abs/2205.15868, 2022. 2

[21] Jiaxing Hu, Jianfei Yang, Jenn-Bing Ong, Dazhuo Wang, and Lihua Xie. Resfi: Wifi-enabled device-free respiration detection based on deep learning. In *2022 IEEE 17th International Conference on Control & Automation (ICCA)*, pages 510–515, 2022. 1

[22] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5967–5976, 2017. 4

[23] Wenjun Jiang, Hongfei Xue, Chenglin Miao, Shiyang Wang, Sen Lin, Chong Tian, Srinivasan Murali, Haochen Hu, Zhi Sun, and Lu Su. Towards 3D human pose construction using wifi. *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking*, 2020. 2

[24] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *Computer Vision – ECCV 2016*, pages 694–711, Cham, 2016. Springer International Publishing. 2, 4

[25] Moez Krichen. Generative adversarial networks. In *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, pages 1–7, 2023. 2

[26] Doyup Lee, Chiheon Kim, Saehoon Kim, Minsu Cho, and Wook-Shin Han. Autoregressive image generation using residual quantization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11523–11532, June 2022. 2

[27] Mahdi Boloursaz Mashhadi, Qianqian Yang, and Deniz Gündüz. Distributed deep convolutional compression for massive mimo csi feedback. *IEEE Transactions on Wireless Communications*, 20(4):2621–2633, 2021. 2, 6, 7

[28] Mahdi Boloursaz Mashhadi, Qianqian Yang, and Deniz Gündüz. Distributed deep convolutional compression for massive mimo csi feedback. *IEEE Transactions on Wireless Communications*, 20(4):2621–2633, 2021. 6, 7

[29] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8821–8831. PMLR, 18–24 Jul 2021. 2

[30] David Salomon. Data compression: The complete reference. 2006. 3

[31] Zhenguo Shi, Qingqing Cheng, J. Andrew Zhang, and Richard Yi Da Xu. Environment-robust wifi-based human activity recognition using enhanced csi and deep learning. *IEEE Internet of Things Journal*, 9(24):24643–24654, 2022. 1

[32] Shunpu Tang, Junjuan Xia, Lisheng Fan, Xianfu Lei, Wei Xu, and Arumugam Nallanathan. Dilated convolution based csi feedback compression for massive mimo systems. *IEEE Transactions on Vehicular Technology*, 71(10):11216–11221, 2022. 1

[33] Lucas Theis, Wenzhe Shi, Andrew Cunningham, and Ferenc Huszár. Lossy image compression with compressive autoencoders. *ArXiv*, abs/1703.00395, 2017. 3

[34] Aaron van den Oord, Nal Kalchbrenner, Lasse Espeholt, koray kavukcuoglu, Oriol Vinyals, and Alex Graves. Conditional image generation with pixelcnn decoders. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016. 2

[35] Aäron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning. *ArXiv*, abs/1711.00937, 2017. 2

[36] Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learning. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6309–6318, Red Hook, NY, USA, 2017. Curran Associates Inc. 3

[37] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6000–6010, Red Hook, NY, USA, 2017. Curran Associates Inc. 2

[38] Fei Wang, Stanislav Panev, Ziyi Dai, Jinsong Han, and Dong Huang. Can WiFi estimate person pose? *Clinical Orthopaedics and Related Research(CORR)*, abs/1904.00277, 2019. 2, 6, 7

[39] Jinbao Wang, Shujie Tan, Xiantong Zhen, Shuo Xu, Feng Zheng, Zhenyu He, and Ling Shao. Deep 3D human pose estimation: A review. *Computer Vision and Image Understanding*, 210:103225, 2021. 5

[40] Zhouxia Wang, Jiawei Zhang, Runjian Chen, Wenping Wang, and Ping Luo. Restoreformer: High-quality blind face restoration from undegraded key-value pairs. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 17491–17500, 2022. 2

[41] Chao-Kai Wen, Wan-Ting Shih, and Shi Jin. Deep learning for massive mimo csi feedback. *IEEE Wireless Communications Letters*, 7(5):748–751, 2018. 2

[42] Wilson Yan, Yunzhi Zhang, P. Abbeel, and A. Srinivas. Videogpt: Video generation using vq-vae and transformers. *ArXiv*, abs/2104.10157, 2021. 2

[43] Fei Yang, Luis Herranz, Joost van de Weijer, José A. Iglesias Guitián, Antonio M. López, and Mikhail G. Mozerov. Variable rate deep image compression with modulated autoencoder. *IEEE Signal Processing Letters*, 27:331–335, 2020. 3

[44] Jianfei Yang, Xinyan Chen, Han Zou, Dazhuo Wang, Qianwen Xu, and Lihua Xie. Efficientfi: Toward large-scale lightweight wifi sensing via csi compression. *IEEE Internet of Things Journal*, 9(15):13086–13095, 2022. 1, 2, 5, 6, 7

[45] Jianfei Yang, He Huang, Yunjiao Zhou, Xinyan Chen, Yuecong Xu, Shenghai Yuan, Han Zou, Chris Xiaoxuan Lu, and Lihua Xie. MM-Fi: Multi-modal non-intrusive 4D human dataset for versatile wireless sensing. In *Thirty-seventh Conf. Neural Infor. Process. Sys. Data. and Bench. Track*, 2023. 5

[46] Jianfei Yang, Yunjiao Zhou, He Huang, Han Zou, and Lihua Xie. MetaFi: Device-free pose estimation via commodity WiFi for metaverse avatar simulation. In *IEEE 8th World Int. of Things (WF-IoT)*, pages 1–6, 2022. 2

[47] Jiahui Yu, Xin Li, Jing Yu Koh, Han Zhang, Ruoming Pang, James Qin, Alexander Ku, Yuanzhong Xu, Jason Baldridge, and Yonghui Wu. Vector-quantized image modeling with improved vqgan. *ArXiv*, abs/2110.04627, 2021. 2

[48] Richard Zhang, Phillip Isola, Alexei A. Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. *2018 IEEE/CVF Con-*

ference on Computer Vision and Pattern Recognition, pages 586–595, 2018. 4

[49] Yi Zhang, Yue Zheng, Kun Qian, Guidong Zhang, Yunhao Liu, Chenshu Wu, and Zheng Yang. Widar3.0: Zero-effort cross-domain gesture recognition with wi-fi. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(11):8671–8688, 2022. 1

[50] Zhu Zhang, Jianxin Ma, Chang Zhou, Rui Men, Zhikang Li, Ming Ding, Jie Tang, Jingren Zhou, and Hongxia Yang. M6-ufc: Unifying multi-modal controls for conditional image synthesis via non-autoregressive generative transformers. 2021. 2

[51] Yanchao Zhao, Ran Gao, Shangqing Liu, Lei Xie, Jie Wu, Huawei Tu, and Bing Chen. Device-free secure interaction with hand gestures in wifi-enabled iot environment. *IEEE Internet of Things Journal*, 8(7):5619–5631, 2021. 1

[52] Chuanxia Zheng, Tung-Long Vuong, Jianfei Cai, and Dinh Phung. Movq: Modulating quantized vectors for high-fidelity image generation. In S. Koyejo, S. Mohamed, A. Agarwal, D. Belgrave, K. Cho, and A. Oh, editors, *Advances in Neural Information Processing Systems*, volume 35, pages 23412–23425. Curran Associates, Inc., 2022. 2

[53] Rui Zhou, Meng Hao, Xiang Lu, Mingjie Tang, and Yang Fu. Device-free localization based on csi fingerprints and deep neural networks. In *2018 15th Annual IEEE International Conference on Sensing, Communication, and Networking (SECON)*, pages 1–9, 2018. 1

[54] Yue Zhou, Aichun Zhu, Caojie Xu, Fangqiang Hu, and Yifeng Li. Perunet: Deep signal channel attention in unet for WiFi-based human pose estimation. *IEEE Sensors J.*, 22(20):19750–19760, 2022. 6, 7

[55] Yunjiao Zhou, He Huang, Shenghai Yuan, Han Zou, Lihua Xie, and Jianfei Yang. MetaFi++: WiFi-enabled transformer-based human pose estimation for metaverse avatar simulation. *IEEE Internet of Things J.*, 10(16):14128–14136, 2023. 2, 6, 7

[56] Andrii Zhuravchak, Oleg Kapshii, and Evangelos Pournaras. Human activity recognition based on wi-fi csi data -a deep neural network approach. *Procedia Computer Science*, 198:59–66, 2022. 12th International Conference on Emerging Ubiquitous Systems and Pervasive Networks / 11th International Conference on Current and Future Trends of Information and Communication Technologies in Healthcare. 1