Eliminating CSI Feedback Overhead via Deep Learning-Based Data Hiding

Jiajia Guo, *Graduate Student Member, IEEE*, Chao-Kai Wen, *Senior Member, IEEE*, and Shi Jin, *Senior Member, IEEE*

***Abstract*— Channel state information (CSI) plays a crucial role in the capacity of multiple-input and multiple-output systems, but CSI feedback occupies substantial precious transmission resources in frequency-division duplexing (FDD) systems. In this work, we propose a data hiding-based CSI feedback framework, namely, EliCsiNet, to eliminate the CSI feedback overhead in FDD systems through deep learning. The key idea is to hide/superimpose CSI in transmitted messages (e.g., images) with no transmission resource occupation and few effects on message semantics. Concretely, we introduce a novel neural network framework in which the user extracts and hides CSI features in images, and the base station recovers the CSI from the transmitted images. However, the essential source coding (e.g., JPEG compression) before data transmission causes two problems in the proposed EliCsiNet framework when applied in practical systems. First, the compression inevitably disturbs the information of the hidden CSI in images and affects the CSI reconstruction accuracy. Therefore, a two-stage separable training strategy, which includes coding-free end-to-end and coding-aware decoder-only training, is adopted to reduce these effects. Second, the bit length of the images coded via JPEG is unpredictable and uncontrollable, and CSI superimposition may lead to an increase in the bit length of the coded images. To avoid this issue, we divide a full image into several sub-blocks and select the one with the smallest length increment. Image entropy is also introduced to accelerate block selection. Simulation results demonstrate that the proposed EliCsiNet framework can eliminate the CSI feedback overhead with few effects on the features properties of transmitted images, including image quality and bit length.**

***Index Terms*— CSI feedback, deep learning, data hiding, JPEG compression.**

I. Introduction

ADVANCED multiple-input and multiple-output (MIMO) techniques, such as massive MIMO [2], multi-user MIMO [3], distributed MIMO [4], and coordinated multipoint [5], have been proposed to improve the spectral efficiency of cellular systems. However, all advanced MIMO techniques require the base station (BS) to know the accurate downlink channel state information (CSI), which helps BS beamform its signals to users. In time-division duplexing (TDD) systems, downlink CSI can be inferred from uplink CSI owing to channel reciprocity, which is not the case in frequency-division duplexing (FDD) systems. This reciprocity no longer holds in FDD systems, and the only means to obtain CSI in existing FDD systems is to have the user perform channel estimation and then feed back the estimated CSI after quantization to BS [6]. The feedback overhead is excessive in today's cellular systems and will inevitably increase in the future because of numerous antennas at BS [7]. Developing a high-accuracy feedback strategy, which reduces or even eliminates the feedback overhead, is critical and has huge implications for massive MIMO systems.

Researchers have attempted to eliminate the CSI feedback overhead. For example, [8] introduced superimposed coding (SC) to CSI feedback to avoid uplink resource occupation. In SC-based CSI feedback, the estimated downlink CSI is spread to form a block signal, which is then superimposed onto the uplink user data sequences. However, the interference between the user data and the downlink CSI is difficult to cancel, thereby increasing the symbol error rate. That is, the elimination of the CSI feedback overhead in SC occurs at the expense of user data transmission quality. Moreover, [9] introduced deep learning (DL) to SC-based CSI feedback and achieved performance comparable with that of conventional SC-based feedback [8].

This work aims to enable BS to obtain the downlink CSI *without* additional uplink resource occupation. Although the feedback overhead is eliminated from the SC-based CSI feedback, the superimposition of extra information onto data sequences, leads to inevitable performance degradation of the original transmission in the form of the symbol error rate [8], making it undesirable in practical systems. Thus, superimposing downlink CSI onto the transmitted messages requires answering the following essential question: *How do we superimpose CSI onto the transmitted messages without affecting the message quality?*

Manuscript received 16 December 2021; revised 9 March 2022; accepted 22 April 2022. Date of publication 8 June 2022; date of current version 18 July 2022. This work was supported in part by the National Natural Science Foundation of China (NSFC) under Grant 61941104 and Grant 61921004, in part by the Key Research and Development Program of Shandong Province under Grant 2020CXGC010108, and in part by the OPPO Research Fund. The work of Chao-Kai Wen was supported in part by the Ministry of Science and Technology of Taiwan under Grant MOST 110-2224-E-110-001. The work of Jiajia Guo was supported in part by the Scientific Research Foundation of Graduate School of Southeast University under Grant YBPY2118. An earlier version of this paper was presented in part at the IEEE ICC 2022. *(Corresponding author: Shi Jin.)*

Jiajia Guo and Shi Jin are with the National Mobile Communications Research Laboratory, Southeast University, Nanjing 210096, China (e-mail: jiajiaguo@seu.edu.cn; jinshi@seu.edu.cn).

Chao-Kai Wen is with the Institute of Communications Engineering, National Sun Yat-sen University, Kaohsiung 80424, Taiwan, and also with the Southern Taiwan Industry Promotion Center, National Sun Yat-sen University, Kaohsiung 80424, Taiwan (e-mail: chaokai.wen@mail.nsysu.edu.tw).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/JSAC.2022.3180806.

Digital Object Identifier 10.1109/JSAC.2022.3180806

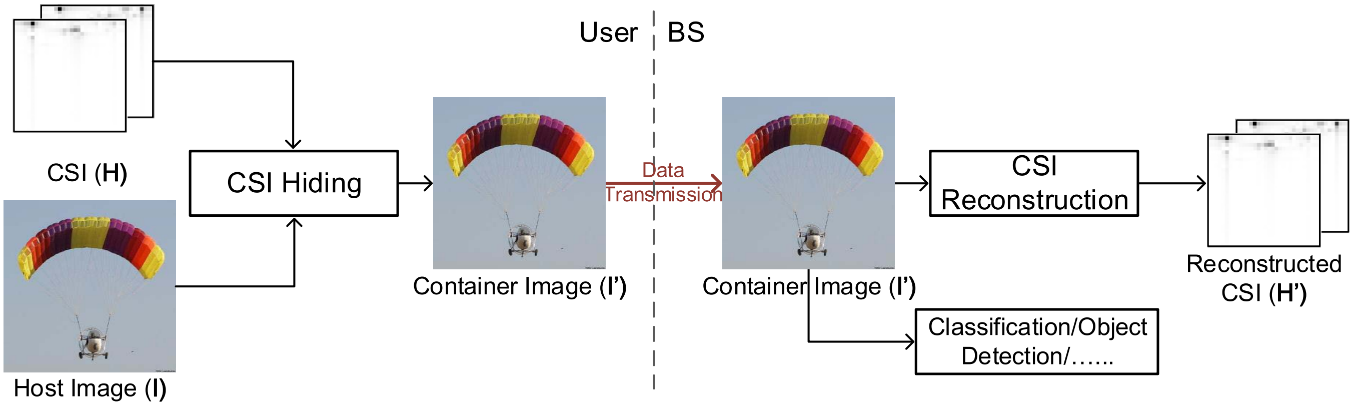


Fig. 1. Diagram of the proposed EliCsiNet, which superimposes downlink CSI onto the transmitted messages (e.g., images) with no occupation of uplink transmission and few effects on message semantics. The user hides the estimated CSI ($\mathbf{H}$) within host image $\mathbf{I}$ then transmits the container image ($\mathbf{I}'$) to BS. BS recovers the CSI from the received container image. The hiding and reconstruction operations are realized by NN-based modules.

Inspired by semantic communication, which focuses on transmitting the correct meaning of messages rather than guaranteeing the correct reception of every transmitted bit [10]–[12], we propose EliCsiNet (shown in Fig. 1), a novel system that superimposes downlink CSI onto the transmitted messages (e.g., images) with no occupation of uplink transmission and few effects on message semantics.

Before discussing the framework of EliCsiNet, we explain the main motivation of EliCsiNet. Given that extra information superimposition inevitably affects the originally transmitted bits and message semantics, we cannot directly superimpose downlink CSI onto bitstreams. However, a few bit or symbol errors are tolerable for a high system capacity in semantic communication. For example, real-time videos and images are continuously transmitted to the cloud computing data center for processing (e.g., object detection) in autonomous driving [13] through uplink transmission, in which low transmission latency should be guaranteed for driving safety. Hence, the information that is useful for the subsequent image tasks should be transmitted with high quality, and redundant data can be removed to reduce bandwidth occupation [14]. Thus, in semantic communication, semantic errors, rather than bit or symbol errors, are minimized [15], and the system focuses on transmitting the information relevant to the transmission goal, namely, the message meaning. However, SC-based feedback superimposes CSI onto the bits/symbols without any consideration of message semantics. Sometimes bit or symbol errors seriously affect message semantics, and sometimes they do not affect message semantics. That is, the effects are uncontrollable and unpredictable, thereby making SC-based CSI feedback infeasible for practical systems. Therefore, during CSI superimposition, the semantics of the transmitted messages should be considered and retained, and the received message should not be completely similar to the original one.

Motivated by this situation, we adopt a DL-based data hiding approach that embeds information into other messages with few effects on the original semantics [16].

(1) Data hiding enables a secret message to be hidden in several transport media, such as digital images, and its key property is its imperceptibility for achieving the fundamental goal of being hidden [16], [17]. In other words, the images containing secret messages cannot be discovered by the human eye, and data hiding has few effects on the subsequent tasks, such as classification and object detection. Meanwhile, the receiver can recover the hidden secret message with high quality. In this work, as shown in Fig. 1, we use images as an example and hide the downlink CSI in natural images. However, the key idea of this work can be extended to a scenario, in which other transport media (such as texts [18], [19]) rather than images are considered. Specifically, fueled by the neural network (NN) framework proposed in [20], the proposed EliCsiNet framework is composed of three subnets, namely, preparation network (PreNet), hiding network (HidNet), and reconstruction network (RecNet). These subnets are trained together, and the optimization goals of HidNet and RecNet are to maximize the quality of the container image and the reconstructed CSI, respectively. Through end-to-end learning, the EliCsiNet framework can superimpose downlink CSI onto the natural images with few effects on image quality.

(2) In practical systems, images are pre-processed with lossy compression (i.e., source coding) to reduce the occupation of the transmission bandwidth and guarantee low latency [16]. Joint Photographic Experts Group (JPEG) [21], [22] is the most widely used compression technique in the source coding of images. Robustness to lossy compression is a major challenge in data hiding [23]. In this work, source coding causes two critical problems.

First, the compression operation can be regarded as adding noise to the container image. The information on the CSI contained in the container image may be corrupted, and the BS cannot recover the CSI from the received compressed image. Similar to the quantization operation, JPEG compression is non-differentiable and cannot be directly embedded into end-to-end training. Thus, the key problem is how to embed JPEG compression into NN training. Inspired by the quantization strategy proposed in [24], we propose a two-stage separable strategy to reduce the effects of JPEG compression on reconstruction accuracy at the

BS. Concretely, coding-free end-to-end training and coding-aware decoder-only training are adopted in the two training stages.

Second, the image is transmitted in the form of bitstreams, which are generated by the JPEG encoder. The superimposition of downlink CSI inevitably affects JPEG compression, including the bit number after compression. The bit number of the coded images may increase while the downlink CSI is being superimposed. When the number of increased bits is large, the data hiding for CSI feedback is not worthwhile. In this work, we divide images into several blocks, hide CSI in the different blocks, and select the one whose bit number does not increase after JPEG coding. This process is time- and computation-consuming and therefore is infeasible for practical systems. Image entropy is introduced to accelerate this selection.

A. Related Work

Several possible methods have been proposed to reduce the feedback overhead. Codebook-based CSI feedback, such as Type I and II codebooks in 5G new radio (NR), is adopted by the existing systems [25]. However, the CSI codebook size usually increases exponentially with the antenna number, thereby requiring considerable resources of control channel and computational effort [26]. Another common method to reduce downlink CSI feedback is compressed sensing (CS) [27]. Under the CSI sparsity assumption, compressed CSI can be reconstructed with acceptable accuracy by several CS-based iterative algorithms, such as orthogonal matching pursuit [28]. However, these iterative algorithms are complex and infeasible for use in practical systems. Moreover, the two methods mentioned above ignore the fact that user channels are highly related to the environment, and the environment around BS remains relatively stable.

The feedback overhead can be considerably reduced if environmental knowledge is introduced to the feedback process. The main challenge is how to extract and exploit environmental knowledge. Recently, DL, which automatically extracts knowledge from data through NNs, has shown great potential in CSI feedback [29], [30]. Specifically, the CsiNet proposed in [31] realized channel compression and reconstruction through an NN-based autoencoder. Simulation results have shown that DL can considerably improve feedback performance while lowering complexity and latency. Thus, DL-based CSI feedback has been a popular topic in the communications field. DL-based CSI feedback also attracts significant research efforts from the industry. It has been listed as the first of three selected initial use cases in the Third Generation Partnership Project (3GPP) study item [32] of Release 18 for 5G-Advanced. Moreover, [33] applied DL to CSI feedback in wireless local area networks (WLANs) and implemented the proposed LB-SciFi on a wireless testbed. The experiment results in [33] showed that LB-SciFi offers an average of 73% feedback overhead reduction and increases system throughput by approximately 69% in practical systems. However, CSI feedback still occupies a huge uplink resource, which is an inevitable stumbling block in several new communication scenarios, such as uplink centric broadband communication (UCBC) [34]–[36].

Overall, the existing feedback works reduce the feedback overhead at the expense of feedback accuracy or eliminate the feedback overhead at the expense of data transmission quality.

B. Contributions

The goal of this work is not to develop novel NN architectures or introduce novel DL tricks. This work aims to introduce a new solution to reduce/eliminate the feedback overhead of downlink CSI, thereby serving a guide for future research. The main contributions of this work are summarized as follows.

- *Elimination of feedback overhead with few effects on message semantics.* To eliminate the feedback overhead, we propose a DL-based hiding framework for downlink CSI acquisition. We use digital images as an example and hide downlink CSI in the images. The NNs at the user side hide the estimated CSI in the images, and the NNs at the BS recover the CSI from the received container images. The superimposition of downlink CSI has few effects on image semantics.
- *High robustness to the noise introduced by source coding.* The existence of source coding, such as JPEG compression, considerably disturbs the CSI reconstruction at the BS. Given that JPEG compression is non-differentiable and cannot be embedded into end-to-end learning, a two-stage separate training strategy is proposed to reduce the effects of JPEG compression on CSI accuracy. JPEG compression is not considered at the first training stage. At the second stage, only the NNs at the BS are fine-tuned with the compressed container images.
- *No bit increase in data transmission after source coding.* The digital images are transmitted in the form of bitstreams generated by the JPEG encoder. The hidden information may increase the bit number. We propose dividing a full image into several blocks and selecting the one whose bit number does not increase after CSI hiding. Moreover, image entropy is introduced to accelerate the selection.

C. Paper Organization

The remainder of this paper is organized as follows. Section II introduces the transmission model and the formulation of the DL-based CSI feedback methods. Section III presents a preliminary introduction to data hiding and how to combine DL-based data hiding with CSI feedback (i.e., the EliCsiNet framework). Section IV introduces the effects of JPEG compression and two potential solutions. Section V provides the numerical simulation results of the proposed methods. Section VI concludes this work.

II. System Model

After introducing the adopted transmission model, we present the formulation of DL-based CSI feedback.

A. Transmission Model

This work considers a single-cell downlink FDD massive MIMO system, where the BS equipped with $N_{\mathrm{t}} \gg 1$ antennas serves several single-antenna users. The system adopts orthogonal frequency division multiplexing (OFDM) transmission over $\tilde{N}_{\mathrm{c}}$ subcarriers. Assuming that the BS in this system employs linear precoding, the received signal at the user side over the k-th subcarrier can be written as:

$$y_n = \tilde{\mathbf{h}}_k^H \mathbf{v}_k x_k + n_k, \tag{1}$$

where $\tilde{\mathbf{h}}_k \in \mathbb{C}^{N_{\mathrm{t}} \times 1}$ denotes the downlink channel vector of the k-th subcarrier, $\mathbf{v}_k \in \mathbb{C}^{N_{\mathrm{t}} \times 1}$ represents the linear downlink precoding vector, x_k is the data-bearing symbol, and n_k is the complex additive noise of the k-th subcarrier. The CSI matrix in the spatial-frequency domain can be written as $\tilde{\mathbf{H}} = [\tilde{\mathbf{h}}_1, \tilde{\mathbf{h}}_2, \cdots, \tilde{\mathbf{h}}_{N_{\mathrm{c}}}] \in \mathbb{C}^{\tilde{N}_{\mathrm{c}} \times N_{\mathrm{t}}}$.

The precoding vector is designed by the BS based on the knowledge of downlink CSI $\tilde{\mathbf{H}}$, and the accuracy of the obtained downlink CSI significantly affects the precoding performance. Thus, the user has to feed back channel $\tilde{\mathbf{H}}$[1] as accurately as possible. For example, the number of total feedback CSI parameters is $2N_{\mathrm{t}}\tilde{N}_{\mathrm{c}}$ when the downlink CSI is perfectly fed back to the BS. This feedback overhead is extremely huge and infeasible for practical systems.

CSI is sparse in the angular-delay domain [27], [37]. Therefore, 2D discrete Fourier transform (DFT) is adopted as follows:

$$\mathbf{H} = \mathbf{F}_{\mathrm{c}} \tilde{\mathbf{H}} \mathbf{F}_{\mathrm{t}}, \tag{2}$$

where $\mathbf{F}_{\mathrm{c}} \in \mathbb{C}^{\tilde{N}_{\mathrm{c}} \times \tilde{N}_{\mathrm{c}}}$ and $\mathbf{F}_{\mathrm{t}} \in \mathbb{C}^{N_{\mathrm{t}} \times N_{\mathrm{t}}}$ are DFT matrices. Given that the time delay among multi-path arrivals often lies within a limited period in practical systems, only the first N_c rows of $\mathbf{H}$ contain values [31]. Similar to [31], the rows in $\mathbf{H}$, except for the first N_{c} rows, are removed. In the rest, the $N_{\mathrm{c}} \times N_{\mathrm{t}}$ truncated channel matrix is still denoted by $\mathbf{H}$ by abuse of notation.

B. DL-Based CSI Feedback Formulation

In DL-based CSI feedback, the autoencoder architecture is composed of an encoder and a decoder. At the user side, the NN-based encoder compresses the estimated downlink CSI $\mathbf{H}$ by reducing the neuron number of the fully connected (FC) layer [31] or using pooling/downsampling layers [38]. Then, lossy quantization [24] and lossless entropy coding [38] are adopted to generate bitstreams for transmission. Upon receiving the feedback codewords,[2] the BS decompresses the CSI from the codewords by using FC or upsampling layers then refines the coarse reconstructed CSI with stacked convolutional layers, such as the RefineNet block [31]. The entire DL-based CSI feedback process can be formulated as

$$\hat{\mathbf{H}} = \mathrm{f}_{\mathrm{de}}\left(\Big(\mathcal{Q}\big(\mathrm{f}_{\mathrm{en}}(\mathbf{H}; \Theta_{\mathrm{en}})\big)\Big); \Theta_{\mathrm{de}}\right), \tag{3}$$

where $\mathrm{f}_{\mathrm{en}}(\cdot)$ and $\mathrm{f}_{\mathrm{de}}(\cdot)$ denote the encoder at the user side and the decoder at the BS, respectively; Θ_{en} and Θ_{de} represent the NN weights in the encoder and decoder, respectively; and $\mathcal{Q}(\cdot)$ is the quantization operation.[3] The NNs are trained with a large number of CSI samples, and the most widely used cost function is MSE.

III. CSI Feedback Using DL-Based Data Hiding

In this section, we present a preliminary introduction to data hiding. Then, we introduce the EliCsiNet framework and NN architecture and show how to combine DL-based data hiding with CSI feedback.

A. Introduction to Data Hiding

In a scenario of secret communication, Alice is the sender, and Bob is the recipient. Alice attempts to hide secret information S within the transmitted medium (cover/host) C, which can be images or videos. The encoded medium is defined as container C', which contains secret information S. Then, container C' is transmitted to Bob. Upon obtaining the container, Bob recovers the secret, and the revealed secret is denoted as S'. This process can be formulated as

$$C' = \mathcal{H}(S, C), \tag{4a}$$

$$S' = \mathcal{R}(C'), \tag{4b}$$

where $\mathcal{H}(\cdot)$ and $\mathcal{R}(\cdot)$ represent the hiding and revealing operations, respectively.

To ensure imperceptibility, the difference between cover C and container C' should be minimized as

$$\min_{\mathcal{H}} \quad dist_C(C, C') = dist_C\Big(C, \mathcal{H}(S, C)\Big), \tag{5}$$

where $dist_C(\cdot)$ is the distance metric between the cover and container. Meanwhile, to ensure successful hidden data transmission, the difference between secret S and revealed information S' should be minimized as

$$\min_{\mathcal{R}} \quad dist_S(S, S') = dist_S\Big(S, \mathcal{R}(C')\Big), \tag{6}$$

where $dist_S(\cdot)$ is the distance metric between the original and revealed secrets. MSE is the most widely used distance metric. The loss function of the entire process can be formulated as

$$L = dist_S(S, S') + \beta \cdot dist_C(C, C'), \tag{7}$$

where $\beta \in [0, 1]$ is a scaling factor used to achieve a balance between the secret and the cover reconstructions.

B. Framework and NN Architecture of EliCsiNet

In DL-based CSI feedback, the NN framework is based on the autoencoder architecture, where the encoder at the user side and the decoder at the BS compress and recover the downlink CSI $\mathbf{H}$, respectively. The autoencoder architecture is naturally suitable for the CSI feedback problem, which is similar to image compression [40]. Therefore, the proposed EliCsiNet framework is similar to the autoencoder architecture, and the NN modules are deployed at the user side and the BS.

[1] In this paper, we assume that the CSI has been accurately estimated by the user.

[2] Unlike [39], the uplink transmission in this work is assumed to be perfect.

[3] Given that the entropy coding is lossless, the entropy coding is not involved in this formulation.

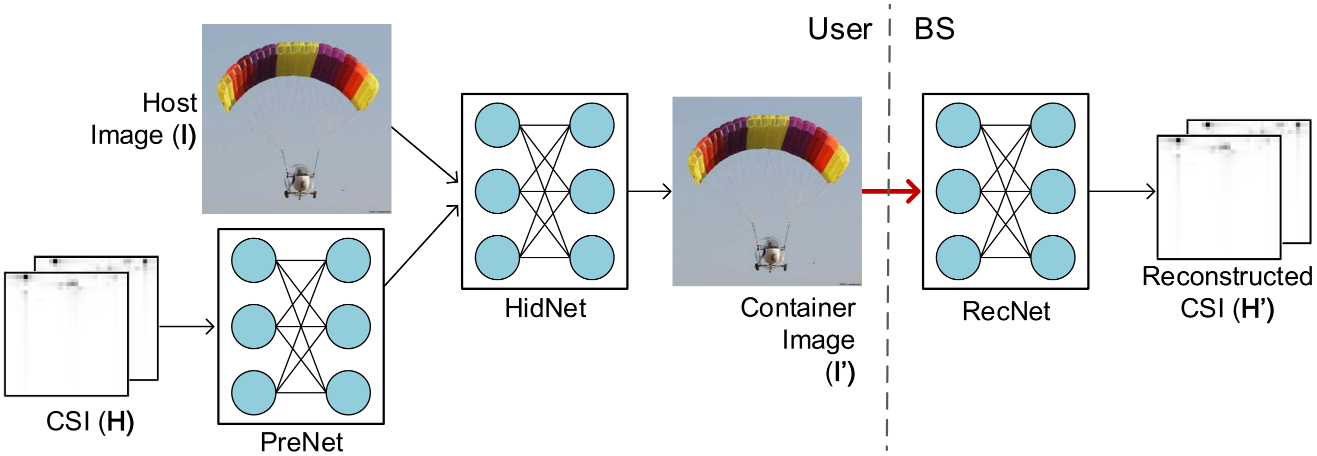


Fig. 2. Three main components of the proposed EliCsiNet framework. PreNet (left) transforms CSI $\mathbf{H}$ into features to be hidden. HidNet (center) superimposes the CSI features onto host image $\mathbf{I}$. RecNet (right) reveals the CSI from container image $\mathbf{I}'$.

1) Framework: As shown in Fig. 2, the EliCsiNet framework has three components: PreNet, HidNet, and RecNet. The first two modules are at the user, and the last one is deployed at the BS. In the following text, we describe the main functions of the three modules.

- *PreNet* is used to prepare the downlink CSI $\mathbf{H}$ to be hidden, that is, to transform the CSI into features that can be easily superimposed onto the host image. Concatenation is the most widely used operation to combine two inputs or features [41]. The dimension of CSI $\mathbf{H}$ is $N_{\rm t} \times N_{\rm c} \times 2$, and that of host image $\mathbf{I}$ is $M \times M \times 3$.[4] Thus, the original CSI cannot be directly concatenated with the image. In this work, an increase in dimension is realized by the upsampling operation, and the NNs of PreNet are composed of upsampling and convolutional layers. The output of PreNet has the same size as that of the host image.
- *HidNet* is used to hide the CSI features extracted by PreNet in host image $\mathbf{I}$. The inputs of this subnet are host image $\mathbf{I}$ and the output of the previous PreNet. The host image and the CSI features are concatenated, and the output shape is $M \times M \times (3+K)$, where K is the number of CSI features. Then, the $M \times M \times (3+K)$ matrix is processed by convolutional layers. The output of this subnet is container image $\mathbf{I}'$, which has the same dimension as the host image, that is, $M \times M \times 3$.
- *RecNet* is deployed at the BS and is used to extract/reveal the hidden CSI from container image $\mathbf{I}'$. In this subnet, the dimension of the input container image is reduced until the feature has the same size as the CSI.

The entire process can be formulated as

$$\mathbf{H}' = \mathcal{R}(\mathbf{I}'; \Theta_{\rm R}) = \mathcal{R}\Bigg(\mathcal{H}\Big(\big(\mathcal{P}(\mathbf{H}; \Theta_{\rm P}), \mathbf{I}\big), \Theta_{\rm H}\Big); \Theta_{\rm R}\Bigg), \quad (8)$$

where $\mathcal{R}$, $\mathcal{H}$, and $\mathcal{P}$ represent the operations of RecNet, HidNet, and PreNet, respectively, and $\Theta_{\rm R}$, $\Theta_{\rm H}$, and $\Theta_{\rm P}$ denote the corresponding NN weights, respectively. Although the three modules are deployed at the user side and the BS, they are trained as a single NN via an end-to-end approach.

2) NN Architecture: The detailed NN architecture is shown in Fig. 3. For simplicity, we assume that the dimensions of the CSI and the host image in the figure are $32 \times 32 \times 2$ [31] and $256 \times 256 \times 3$, respectively. If the dimensions differ, then the number of dimension reduction/dimension increase blocks will change accordingly. The CSI and the host image are normalized to $(-1, 1)$. Given that FC layers are not adopted, the number of NN weights is much smaller than that in [31]. Moreover, the computational complexity can be reduced by the NN compression mentioned in [43].

As mentioned in Section III-A, the cost function adopted in this work is

$$L = \|\mathbf{H} - \mathbf{H}'\| + \beta \cdot \|\mathbf{I} - \mathbf{I}'\|, \quad (9)$$

where $\|\cdot\|$ represents the MSE distance.

IV. CSI Feedback Using DL-Based Data Hiding Against JPEG Compression

In practical transmission systems, an image should be encoded via source coding techniques, such as JPEG, before being sent to the transmission modules. Several challenges arise with source coding. In this section, we provide a preliminary introduction to JPEG compression and the challenge introduced by this source coding. Then, we introduce two methods to handle this challenge.

A. Introduction to JPEG Compression

The goal of JPEG compression is to reduce the bitstreams of images with controllable and tolerable errors. We briefly introduce the encoder of JPEG compression. The decoder performs the opposite operation and is thus not repeated here. As shown in Fig. 4, the encoder of JPEG compression is composed of four main components, namely, pre-processing, discrete cosine transform (DCT), quantization, and entropy coding modules [21], [22].

- *Pre-processing:* The shape of the input image is $M \times M$. However, the compression is performed on each

[4] For example, the dimension of the truncated CSI in [31] is $32 \times 32 \times 2$, and that of the natural images in [42] is $224 \times 224 \times 3$.

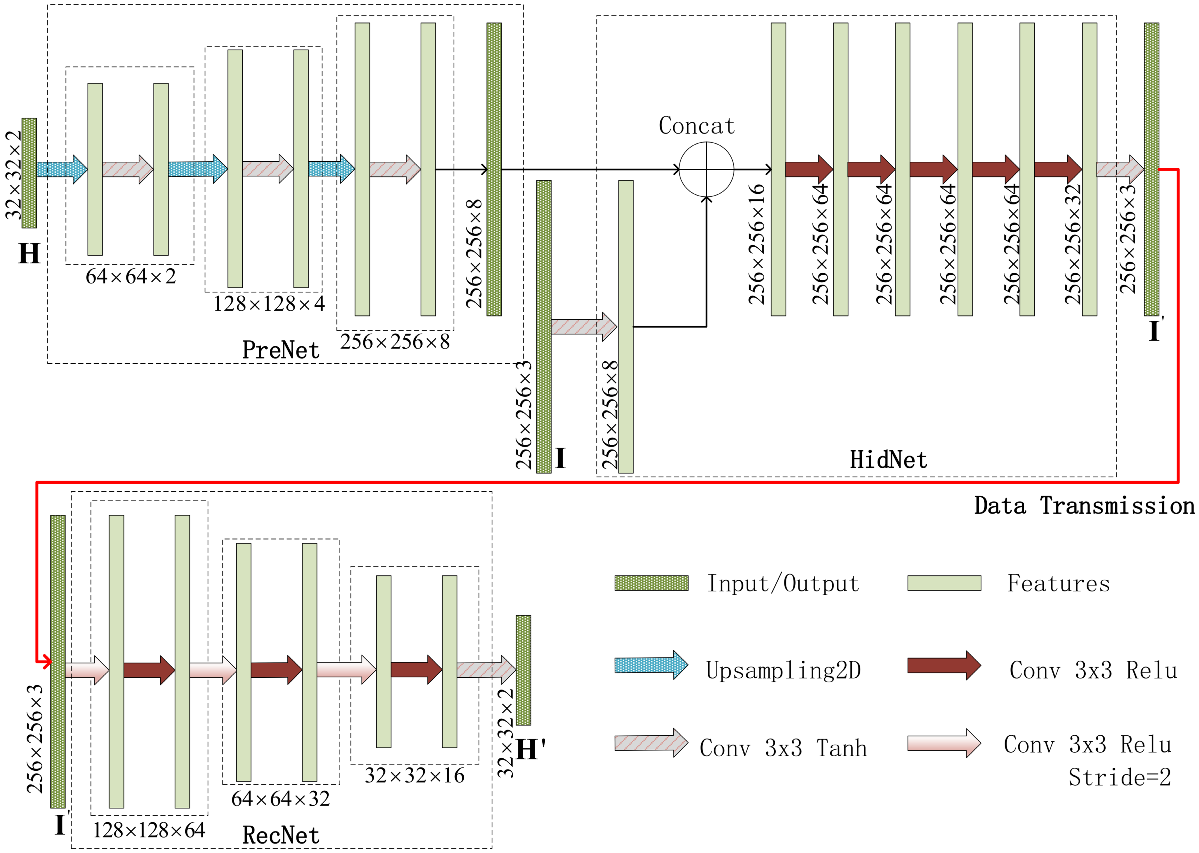


Fig. 3. Detailed NN architecture of the proposed EliCsiNet framework, which includes PreNet, HidNet, and RecNet. We assume that the dimensions of the CSI and host image are $32 \times 32 \times 2$ and $256 \times 256 \times 3$, respectively.

8×8 block rather than on the full image. Therefore, the image is grouped into 8×8 non-overlapping blocks.

- *DCT:* The 8×8 blocks are inputted to the DCT operation, which can be regarded as a harmonic analyzer. The obtained DCT coefficient values are the relative amount of the 2D spatial frequencies contained in the previous 64-point input block signal [21]. The coefficients can be grouped into two kinds: direct current (DC) and alternating current (AC) coefficients. The changes in the sample value in such a small block (i.e., 8×8) are often small. Therefore, the low frequencies, especially DC, occupy the main energy, and the high frequencies have zero or near-zero amplitudes.
- *Quantization:* The DCT frequency coefficients are uniformly quantized with a quantization table, which is determined by the quality of the JPEG compression. The quantization step of each frequency coefficient is different and decreases with frequency. The highest quantization precision is used to quantize the DC coefficient.
- *Entropy coding:* To further reduce the length of the bitstream, lossless entropy coding encodes the quantized DCT coefficients compactly. Huffman coding is adopted in JPEG compression. However, before entropy coding, some complicated pre-processing, such as reordering, is necessary. This article does not introduce too many details due to space limitations, but interested readers may refer to [21].

B. Problems Caused by JPEG Compression

In Section III, we introduce the EliCsiNet framework, which superimposes the downlink CSI onto the host image. In practical transmission systems, the image should be encoded through source coding techniques, such as JPEG, before being sent to the transmission modules. Therefore, (8) should be written as

$$\begin{aligned}\mathbf{H}' &= \mathcal{R}(\mathcal{C}(\mathbf{I}'); \Theta_{\mathrm{R}}) \\ &= \mathcal{R}\Bigg(\mathcal{C}\Big(\mathcal{H}\Big(\big(\mathcal{P}(\mathbf{H}; \Theta_{\mathrm{P}}), \mathbf{I}\big); \Theta_{\mathrm{H}}\Big)\Big); \Theta_{\mathrm{R}}\Bigg), \qquad (10)\end{aligned}$$

where $\mathcal{C}$ represents the source coding in container image $\mathbf{I}'$. JPEG compression reduces the bitstream length of container image $\mathbf{I}'$ at the expense of image quality. This can be regarded as noise $\mathbf{n}_{JPEG}$, which disturbs the image quality as $\mathbf{I}' \leftarrow \mathbf{I}' + \mathbf{n}_{JPEG}$. The introduction of JPEG compression to data hiding raises the following concerns.

- The information contained in container image $\mathbf{I}'$ can be divided into two parts: information of the host image and downlink CSI. Given that JPEG is designed for image compression, the effects of JPEG on host image $\mathbf{I}$ are tolerable. JPEG compression inevitably destroys the information of the downlink CSI $\mathbf{H}$ contained in container image $\mathbf{I}'$. The accuracy of extracting CSI from the container image considerably decreases, making the proposed EliCsiNet infeasible for practical systems.

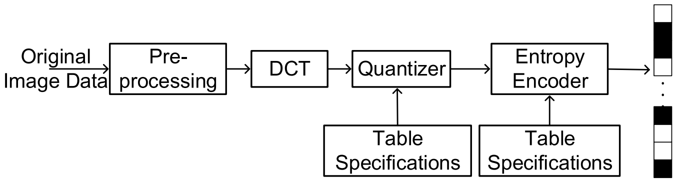


Fig. 4. Fundamental building blocks of the JPEG encoder, including pre-processing, DCT, quantizer, and entropy coding. This process is complex and non-differentiable, making it difficult to be embedded into the end-to-end training of the proposed EliCsiNet framework.

- The goal of the proposed EliCsiNet framework is to eliminate CSI feedback overhead, that is, feedback CSI without uplink resource occupation. In Section III, we assume that the image is transmitted with full precision without source coding. The dimensions of the images before and after data hiding does not change. Therefore, no extra overhead is occupied when the downlink CSI is hidden within the image. However, if JPEG compression is employed, the bitstream length is determined by the image context, and the problem becomes increasingly complicated.

  As mentioned in Section IV-A, the encoder of JPEG compression has four main components. Accordingly, an image with abundant details, that is, high-frequency components, will occupy much space after JPEG coding. If the superimposition of CSI adds extra details to host images, the bit number of the container image coded by JPEG may increase. If the bit number of the coded container image is much larger than that of the host image, extra transmission resources will be occupied, and the proposed EliCsiNet will become meaningless.

C. Two-Stage Separate Training Strategy

A natural idea to reduce the effects of JPEG compression on CSI reconstruction is to embed JPEG compression onto the proposed EliCsiNet framework. However, we observe two main problems. First, the quantization at the JPEG encoder is complex.[5] The quantization step is determined by a pre-defined table, and that of each coefficient value is different. Moreover, the quantization operation is non-differentiable, which hinders the propagation of gradients during NN training. Therefore, this operation is difficult to embed onto the NNs.

Although one-stage end-to-end learning is straightforward, its major limitation is the need for a continuous gradient. Motivated by [24], [44], this study introduces a two-stage separate training strategy to address the limitation of one-stage end-to-end learning, thereby reducing the effects of JPEG compression on reconstruction accuracy.

1) Stage I: Coding-Free End-to-End Training: The goal of this stage is to train the NNs (i.e., PreNet and HidNet) before JPEG compression. However, RecNet is still jointly training with PreNet and HidNet at this stage for gradient backpropagation. The loss function of this stage is similar to that in Section III, i.e., (9). This training stage is essential. If the NNs used to generate container images are not powerful enough, the performance of the reconstruction at the BS will be inevitably poor.

2) Stage II: Coding-Aware Decoder-Only Training: Through the first training stage, the NN weights of PreNet and HidNet are obtained to hide downlink CSI within the host image. Then, a JPEG compression module is introduced between the proposed HidNet and RecNet. That is, the output of the trained HidNet is encoded by the JPEG encoder (as shown in Fig. 4), and immediately afterward, the JPEG decoder at the BS recovers the container image from the transmitted bitstreams and sends it to the proposed RecNet. Therefore, only RecNet is paid attention to after the first training stage, and the NN weights of the trained PreNet and HidNet are fixed and unchanged during the second stage.

During the second stage, we generate training sample pairs for training the coding-aware RecNet. The CSI and image datasets are inputted to the NNs at the user side, and the outputs are the container images. We use the encoder and decoder of JPEG compression to encode and reconstruct the container images. The reconstructed images are collected as the input training dataset of RecNet, and the original downlink CSI samples are collected as the label. The loss function utilized during the second stage is different from that used during the first stage; it focuses on minimizing the error between the original and reconstructed CSI as follows:

$$L = \|\mathbf{H} - \mathcal{R}\big(\mathcal{C}(\mathbf{I}'); \mathbf{\Theta}_{\mathrm{R}}\big)\|. \tag{11}$$

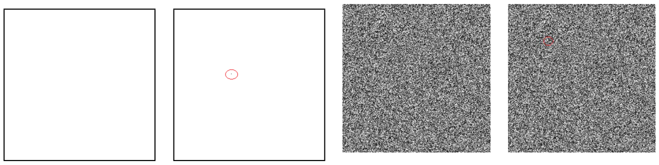
To accelerate the training, the NN weights of RecNet are initialized using the RecNet trained during the first stage rather than one that is trained from scratch.

D. Block-Based CSI Hiding With Image Entropy

The goal of this work is to eliminate the overhead of CSI feedback. Although specific feedback is no longer needed, an extra data transmission overhead may exist when downlink CSI is hidden within host images. On the one hand, in the special example shown in Fig. 5(a), the host image is pure white. The file size of the coded image is 2,278 bytes via JPEG. As shown in Fig. 5(b), we add just a black dot to this pure white image, and the file size increases from 2,278 bytes to 2,288 bytes. In this case, the little information superimposed on the host image leads to an overhead increase in data transmission, which is lost rather than gained. On the other hand, in the host image composed of random noise (shown in Fig. 5(c)), the file size is 66,519 bytes. When a black dot is added to this image, the file size becomes 66, 509 bytes. In this case, no extra transmission overheads are needed.

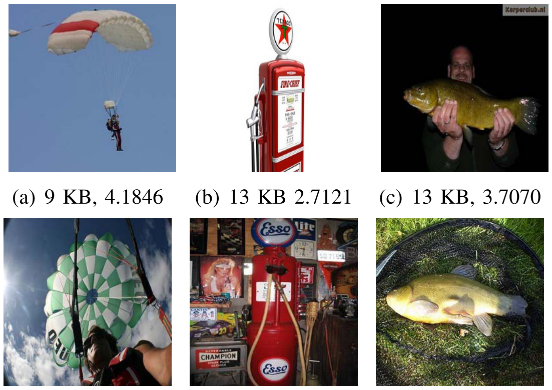
The principle of JPEG compression produces this observation. A slight change in the quantized DC coefficient leads to visual distortion of the smooth area in the image [45]. In other words, the smooth areas (i.e., low-frequency coefficients) are unsuitable for data hiding. The authors in [46] claimed that the secret message should be hidden in high-frequency coefficients, thereby controlling the file size of the container images. However, in the proposed EliCsiNet framework, the optimization goal of HidNet is to minimize

[5] Given that the entropy coding in JPEG compression is lossless, it is ignored in this part for simplicity.



(a) 2,278 bytes (b) 2,288 bytes (c) 66,519 bytes (d) 66,509 bytes

Fig. 5. Examples of two special cases, where the first host image (a) is pure white, and the second one (c) is composed of random noise. A black dot enclosed by a red circle is randomly added to the two host images. (b) and (d) show the corresponding images with a random black dot. The caption of each sub-figure represents the file size after JPEG encoding. The file size of the pure white image increases when the dot is added.



(a) 9 KB, 4.1846 (b) 13 KB 2.7121 (c) 13 KB, 3.7070

(d) 30 KB, 7.8315 (e) 36 KB 7.4704 (f) 49 KB, 7.6638

Fig. 6. Examples of the host images used in this work. The dimension of all images is $256 \times 256 \times 3$, and the classification labels of the images in the same column are the same. The caption of each sub-figure represents the file size of each encoded image and the corresponding entropy. The file sizes of the images on the first row are much smaller than those on the second row. The file size of the images with low entropy is relatively small.

the difference between the host and the container images, and it does not consider where the difference is located. The effects on file size will be few if the difference is located in the high-frequency part. Therefore, the key problem in hiding CSI without a considerable bit length increase is to find the high-frequency part and then hide CSI in it.

1) Block-Based CSI Hiding: Fig. 6 shows several image samples used in this work. The samples on the second row have much more details than those in the first row. The details are in the form of high-frequency coefficients. According to the analysis above, the images with abundant details, rather than smooth images, should be selected to hide downlink CSI. However, whether the transmitted image has abundant or minimal details is uncontrollable and unpredictable.

In this work, we propose to divide the full image into several blocks and select one of them to hide downlink CSI. Block-based image processing is widely adopted in DL-based computer vision algorithms, such as image compression [47], [48]. One of the advantages of block-based processing is the reduction in the computation requirement. The most widely adopted NN layer in computer vision is the convolutional layer. Assuming that the feature shape of the NN input is $W \times W$ and the feature number is C_{in}, the number of floating-point operations (FLOPs) is defined as follows [49]:

$$\text{FLOPs} = 2W^2(C_{in}K^2+1)C_{out}, \tag{12}$$

where K is the filter width and C_{out} is the number of output features. The computation complexity will decrease by 75% if the width and height are half of the original ones.

The goal of dividing images into several blocks is to select the one that is not smooth. We use the first sub-figure in Fig. 6 as an example. As shown in Fig. 7, the original image is divided into four non-overlapping sub-blocks. The first two blocks contain the most information in the original image, and the last two, especially the third one (c), contain little information and details. Therefore, the file sizes of the last two blocks are considerably smaller than those of the first two blocks. If the downlink CSI is hidden in the last two blocks, which have few high-frequency coefficients, then the file size will increase unavoidably. Therefore, the first two blocks, rather than the last two blocks, should be selected to hide downlink CSI. This selection causes a new problem, that is, how to choose the suitable sub-block from the full image.

Exhaustive search is a simple method. The downlink CSI is hidden in the sub-blocks parallelly by using the proposed PreNet and HidNet. Then, each container block is encoded by the JPEG encoder. The block whose file size does not increase or increases just a little is selected to hide the downlink CSI. However, the complexity of the exhaustive search is huge. Therefore, the block selection process should be carefully designed and optimized.

2) Entropy-Aided Block Selection: Entropy provides a lower bound on the average number of bits per pixel required to encode an image without introducing errors for the uncorrelated images [50]. When an image is smooth and contains few details, its entropy is low. In this work, we convert the colorful images to grayscale images and calculate their entropy by using the MATLAB function “entropy” [51]. In Fig. 6, the entropy values of the images on the first row are much lower than those on the second row. The four sub-blocks in Fig. 7 also differ in terms of entropy values. The third block, which only shows the blue sky, has the lowest entropy value. Therefore, we introduce the entropy value of the image block to block selection.

Fig. 8 shows the workflow of the proposed entropy-aided block selection, which entails six steps, as follows: (1) dividing the full image into several non-overlapping blocks, (2) calculating the entropy value of each block parallelly, (3) selecting the block with the largest entropy value, (4) hiding the downlink CSI in the selected image block, (5) calculating the bit length of the container block, and (6) outputting the selected container block if the bit length change is acceptable; otherwise, the third step and subsequent steps are repeated, and the block with the second-largest entropy value is selected. If all blocks do not meet the requirement, then the block with the smallest increase in bit length will be selected.

In the sixth step, the bit length change should be compared with a threshold. When the increase in bit length is much smaller than the CSI feedback without data hiding, the increase is tolerable. For example, according to [24], the number of feedback bits must be at least 320 to ensure that the normalized mean squared error (NMSE) is below -10 dB for the indoor

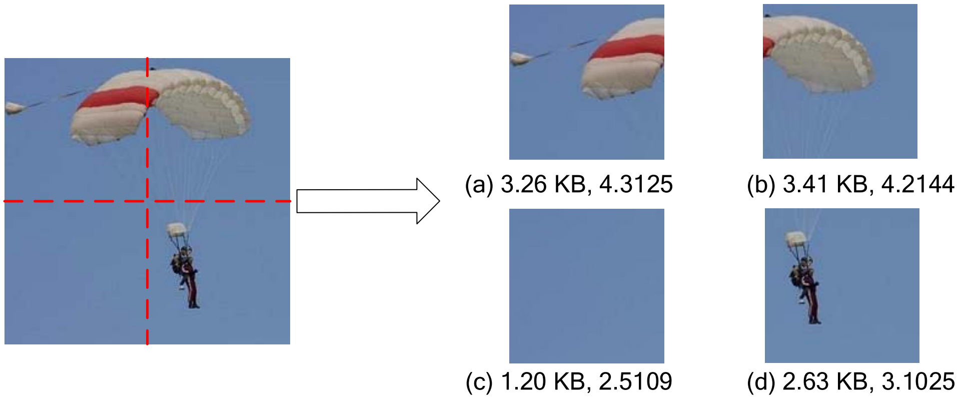


Fig. 7. Example of dividing an image into several blocks. The original $256 \times 256 \times 3$ image is divided into four non-overlapping sub-blocks, and the dimension of each block is $64 \times 64 \times 3$. The caption of each sub-block represents its corresponding file size and entropy. The blocks on the second row, are much smoother than those on the first row.

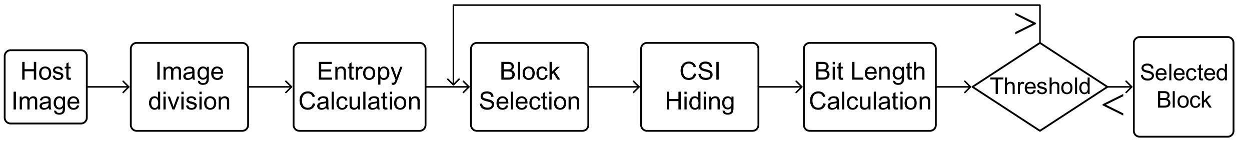


Fig. 8. Workflow of the proposed entropy-aided block selection, which is composed of six main steps.

scenario.[6] If the bit increase in data hiding-aided CSI feedback is 50, then the cost is acceptable. Therefore, we do not strictly require it to not have any increase in bit length.

V. Numerical Simulation Results

This section presents the details of the simulation, including dataset generation and the NN training setting. Then, the proposed EliCsiNet without source coding, including the feedback accuracy and effects on image classification, is evaluated. The effects of JPEG compression on the EliCsiNet framework and the proposed entropy-aided block-based CSI hiding strategy are also determined.

A. Simulation Setting

1) Dataset Generation: Two datasets containing downlink CSI and natural images are used to evaluate the proposed methods.

- The downlink CSI used in this work is the same as that used in [24], [31]. The COST 2100 MIMO channel model [52] is adopted, and an indoor picocellular scenario is considered. The single-antenna users are randomly positioned in a square area with lengths of 20m, and the BS equipped with 32 uniform linear array antennas is positioned at the center. The carrier frequency is 5.3 GHz, and the subcarrier number is 1,024. The CSI dimension after the pre-processing is $32 \times 32 \times 2$. Following the setting in [31], the dataset is randomly divided into three parts, that is, training, validation, and test datasets, each of which contains 100,000, 30,000, and 20,000 CSI samples, respectively.
- We select several images from the ImageNet database to generate the host images. This subset contains about 13,000 images and spans 10 object classes, including fish, dog, cassette player, chainsaw, church, horn, truck, gas pump, golf ball, and parachute. All images are reshaped as $256 \times 256 \times 3$ and randomly divided into training, validation, and test datasets, each of which has 75%, 10%, and 15% samples, respectively.

2) NN Training Setting: All simulations are implemented on TensorFlow v2.3.0 with an NVIDIA DGX-2 workstation. The batch size, learning rate, and training epoch are set as 128, 0.001, and 50, respectively. Adaptive moment estimation (Adam) is used as the NN optimizer. The other settings of the NNs follow the default setting in TensorFlow. Moreover, NMSE is utilized to evaluate the obtained CSI's accuracy.

B. Performance of the Proposed EliCsiNet Without Source Coding

In this subsection, we assume that source coding does not exist after CSI hiding, and the container image is transmitted losslessly. We evaluate the feedback accuracy of the proposed EliCsiNet and then evaluate the effects of CSI hiding on the perceptual quality and the subsequent image task.

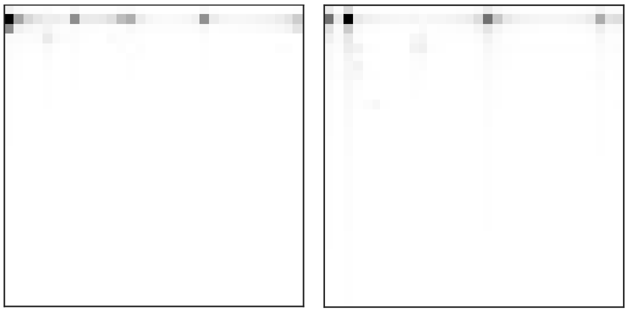
1) CSI Accuracy: Table I shows the NMSE performance of the CSI obtained using the proposed EliCsiNet framework without source coding. The scaling factor β used for the balance between the image and CSI reconstructions is set as 0.001, 0.005, 0.025, 0.05, and 0.1. When β is set as 0.1, the NNs in EliCsiNet mainly focus on the quality of the container images, and the CSI reconstruction error has a negligible influence on the optimization goal, namely, loss function. Therefore, in this case, RecNet at the BS cannot recover the downlink CSI. When the scaling factor β is below 0.1, the proposed methods can accurately reconstruct the downlink CSI. The reconstruction performance increases with the decrease in β. All NMSE values are below -18 dB.

[6] This observation can be found in Table IV of [24].

TABLE I

NMSE (dB) PERFORMANCE OF THE OBTAINED CSI AND IMAGES FRAMEWORK AND CLASSIFICATION ACCURACY OF THE CONTAINER IMAGES USING THE PROPOSED EliCsiNet

β	0.001	0.005	0.025	0.05	0.1
CSI NMSE (dB)	−19.71	−18.57	−18.59	−18.23	0
Image NMSE (dB)	−18.68	−22.73	−29.80	−32.34	−35.36
Image VIF	0.878	0.936	0.964	0.971	0.985
Image UQI	0.969	0.982	0.990	0.993	0.997
Classification accuracy	80.42% (3.14%↓)	82.23% (1.33%↓)	83.35% (0.21%↓)	83.64% (0.08%↑)	-



(a) (b)

Fig. 9. Examples of two indoor channel gray "images." Their numbers in the public test dataset are 1500 and 2400.

In [24], the NMSE of CsiNet+ is -17.23 dB when the feedback bit number is 1,280. In other words, the proposed method saves an overhead of 1,280 bits.

2) Effects on Host Images: The elimination of the CSI feedback overhead comes at the expense of the transmitted image's quality. The accuracy of the container images is shown in Table I. In the table, three objective and subjective image quality assessment metrics, namely, NMSE, visual information fidelity (VIF) [53], and universal image quality index (UQI) [54], are adopted to evaluate the effects of the proposed methods on the original image. The performance of the container image under the three metrics is high. The two CSI gray "images" in Fig 9 are hidden in Figs. 10(a) and 10(k). The errors decrease with scaling factor β. When β is 0.05, the host and container images are almost indistinguishable by the naked eye. The second host image is much smoother than the first one. Therefore, the difference between the second host image and its corresponding container image can be easily determined relative to the first one.

In certain scenarios, such as autonomous driving, the images transmitted to the server would be adopted in several image tasks, such as object detection and image classification. Although the simulation results presented above indicate that CSI hiding has few effects on image quality, whether it affects the subsequent image tasks needs to be determined. Therefore, in this part, we evaluate the effect of CSI hiding on image classification. As mentioned in Section V-A.1, the image dataset is selected from ImageNet and spans 10 object classes. Therefore, we fine-tune VGG16,[7] which is pre-trained using the ImageNet dataset. Specifically, we fix the NN weights of the convolutional layers and set the neuron number of the last FC layer as 10. Then, we use the image dataset adopted in this work to re-train the FC layers in VGG16.

TABLE II

EFFECTS OF JPEG COMPRESSION ON THE PROPOSED EliCsiNet, INCLUDING THE NMSEs OF CSI AND IMAGES, ΔB, AND CLASSIFICATION ACCURACY

β	0.001	0.005	0.025	0.05
CSI NMSE (dB)	−12.63	−14.52	−8.61	−2.93
Image NMSE (dB)	−18.99	−23.17	−30.61	−32.61
ΔB	−564	+2008	+744	+240
Classification accuracy	80.81%	82.42%	83.84%	83.95%

The classification accuracy of the original host images is 83.56%. As indicated in Table I, when scaling factor β is relatively low, such as 0.001, CSI hiding considerably affects classification accuracy, which decreases by 3.14%, because the NNs focus on reconstructing CSI accurately, and the quality of the container images is ignored. Accuracy increases with scaling factor β. When β is set as 0.05, CSI hiding has no effects on image classification. In this case, the NMSE of the obtained CSI is -18.23 dB and still acceptable for practical systems.

Overall, when the source coding is not considered, the proposed EliCsiNet framework can feed back the downlink CSI accurately with no extra resource occupation and few effects on the original images.

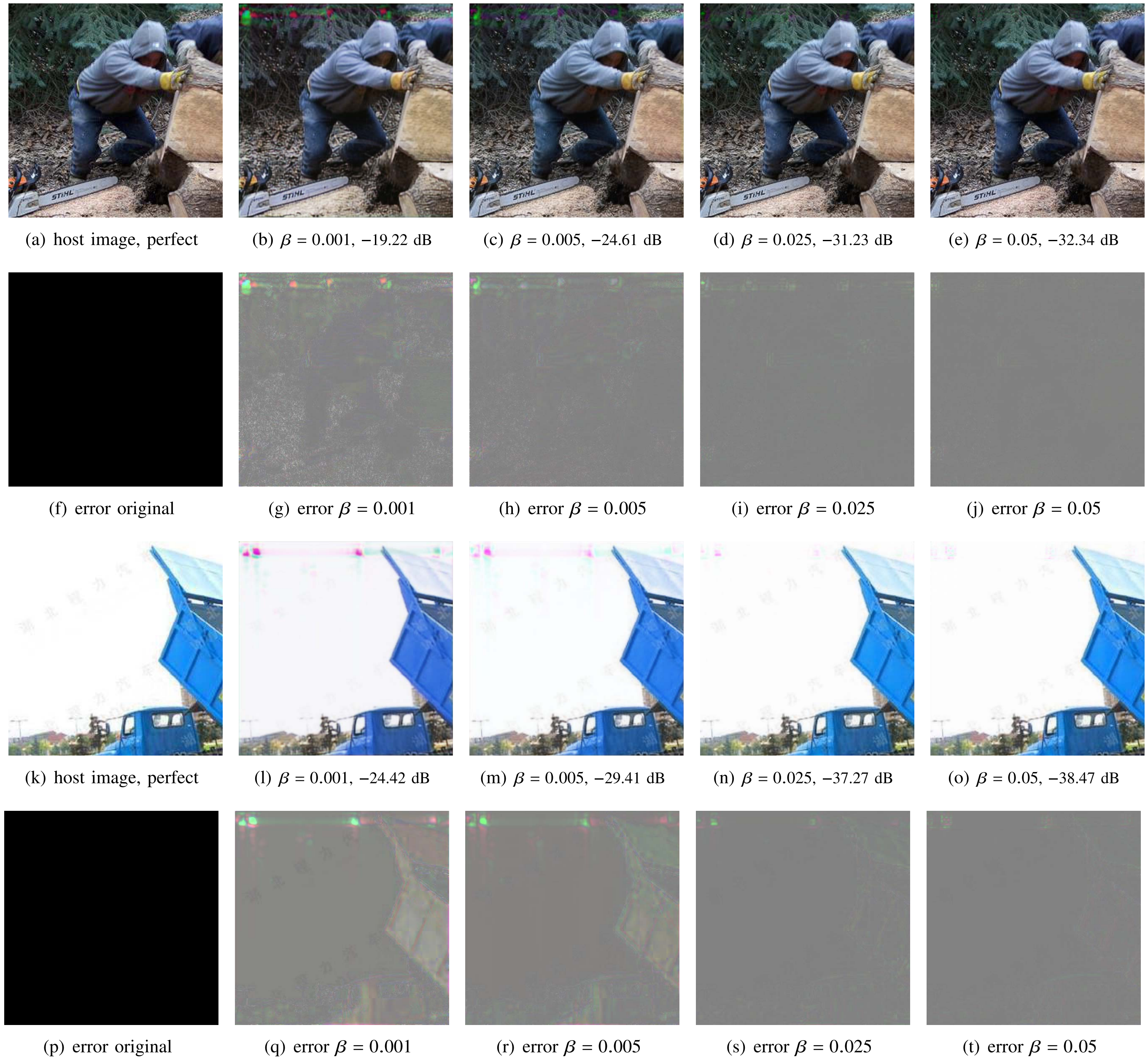
C. Performance of the Proposed EliCsiNet With JPEG Compression

This subsection shows the effects of JPEG compression on CSI accuracy and the bit length of the encoded container images. The performance of the proposed two-stage separable training strategy and entropy-aided block-based EliCsiNet is evaluated.

1) Effects of JPEG Compression: Table II summarizes the effects of JPEG compression on the proposed EliCsiNet, including the NMSEs of CSI and images, bit length increase ΔB, and classification accuracy. The quality of JPEG compression is set as 90.

- First, the quality of the CSI reconstructed by RecNet considerably decreases. For example, as shown in Table I, the CSI NMSE is -18.23 dB when β is set as 0.05 and JPEG compression is not considered. In this case, the performance gap is up to 15.3 dB, which is large. The introduction of source coding makes the proposed EliCsiNet framework infeasible for practical systems.
- Second, as presented in Tables I and II, the quality of the container images is improved a little in compression with the one without JPEG compression. As shown in

[7] https://tensorflow.google.cn/api_docs/python/tf/keras/applications/vgg16/VGG16?hl=en


(a) host image, perfect (b) β = 0.001, −19.22 dB (c) β = 0.005, −24.61 dB (d) β = 0.025, −31.23 dB (e) β = 0.05, −32.34 dB

(f) error original (g) error β = 0.001 (h) error β = 0.005 (i) error β = 0.025 (j) error β = 0.05

(k) host image, perfect (l) β = 0.001, −24.42 dB (m) β = 0.005, −29.41 dB (n) β = 0.025, −37.27 dB (o) β = 0.05, −38.47 dB

(p) error original (q) error β = 0.001 (r) error β = 0.005 (s) error β = 0.025 (t) error β = 0.05

Fig. 10. Examples of two host images and the corresponding container ((b)-(e), (l)-(o)) and error ((g)-(j), (q)-(t)) images. Two different CSI "images" are hidden in (a) and (k). The errors decrease with scaling factor β. When β is 0.05, the host and container images are almost indistinguishable by the naked eye. The second host image is much smoother than the first one. Therefore, the difference between the second host image and its corresponding container image can be easily determined relative to the first one.

Fig. 10, the superimposition of downlink CSI onto the images introduces extra details (high-frequency coefficients in the DCT domain) to the images. The JPEG encoder reduces the high-frequency information during quantization. Therefore, the hidden CSI information is dropped during coding, and the image quality is improved slightly. Therefore, the introduction of JPEG compression has a few positive effects on image classification.

- Third, the bit length of the images encoded by the JPEG algorithm is changed. When scaling factor β is set as 0.005, 0.025, and 0.05, the increase in bit length is enhanced by the decrease in β. When β is large, such as 0.05, the superimposition of CSI has few effects on the images, and the container image is nearly the same as the host image. Therefore, the bit length increase is small. However, when scaling factor β is small, such as 0.001, the bit length does not increase but decreases instead. As shown in Figs. 10(g) and 10(q), the quality of the container images is low, and many details are eliminated. For example, the shallow text watermark in Fig. 10(k) cannot be found in Fig. 10(l). Therefore, the bit length is reduced in this case.

Fig. 11 shows the cumulative distribution function of the increase in bit length when the scaling factor is set as 0.025 and 0.05. When β is 0.025, 40% of the increments exceed 1,000 bits. When β is 0.05, 20% of the increments

TABLE III

NMSE (dB) PERFORMANCE OF THE OBTAINED CSI WITH A TWO-STAGE SEPARABLE STRATEGY

β	0.001	0.005	0.025	0.05
No coding	−19.71	−18.57	**−18.59**	**−18.23**
With coding	−12.63	−14.52	−8.61	−2.93
With coding and two-stage training	**−22.08**	**−22.45**	−15.94	−12.46

exceed 1,000. This situation leads to a substantial extra overhead, thereby rendering the proposed method unreasonable. Moreover, the bit length changes are distributed over a large area mainly from $-2{,}000$ to 4,000. The performance is extremely unstable, making it infeasible for practical systems.

The simulation presented above shows the effects of JPEG compression on the proposed EliCsiNet and reveals the necessity of the two-stage separable training strategy and the entropy-aided block-based CSI hiding.

2) Performance of the Two-Stage Separable Training Strategy: Table III shows the NMSE performance of the CSI obtained with the two-stage separable strategy. In this table, "No coding" and "With coding" mean that the container image is transmitted without and with JPEG compression, respectively. "With coding and two-stage training" represents that the container image is transmitted after JPEG compression, and the RecNet at the BS is trained by the proposed two-stage training approach. Relative to the one without two-stage separable training, the proposed strategy considerably improves the CSI reconstruction performance. For example, when scaling factor β is 0.05, the NMSE improvement is up to 9.53 dB. Moreover, with the help of the proposed strategy, the system with JPEG compression exhibits a comparable performance relative to the one without any source coding.

When scaling factor β is low, namely, 0.001 and 0.005, the reconstruction accuracy of the NNs with JPEG compression is even higher than that of NNs without any source coding. This observation was also reported in [24], which showed that the quantization of feedback codewords sometimes brings extra performance gains relative to analog CSI feedback. JPEG compression and quantization introduce noise to codewords, and this introduction generally reduces reconstruction performance. In this case, noise has a positive rather than negative impact on CSI reconstruction. During training, this kind of noise decreases the gap between training and validation losses. Specifically, it improves NN generalization, thereby leading to a highly accurate reconstruction.

As shown in Tables II and III, NMSE is -12.46 dB when the increase in bit length is 240. In this case, the feedback overhead can be regarded as 240 bits. According to [24], 512 bits need to be fed back when NMSE is -12.58 dB. Therefore, even when the entropy-aided block-based CSI hiding strategy is not adopted, the proposed method can reduce the feedback overhead by more than 50%.

3) Performance of the Entropy-Aided Block-Based EliCsiNet: We divide each $256 \times 256 \times 3$ image into four non-overlapping sub-blocks, as shown in Fig. 7. As shown by Table II, image quality and classification accuracy considerably decrease when scaling factor β is set as 0.001. Therefore, β is set as 0.005, 0.025, and 0.05 during the simulation. Table IV shows the performance of the proposed methods.

TABLE IV

PERFORMANCE OF THE PROPOSED ENTROPY-AIDED BLOCK-BASED EliCsiNet FRAMEWORK

β		0.005	0.025	0.05
ΔB	Original	+2008	+744	+240
	Random selection	+452	+158	+62
	Exhaustive search	-37	−43	−53
	Entropy 1*	+128	+35	+9
	Entropy 2**	+27	−12	−26
CSI NMSE (dB)		−12.60	−6.20	−4.91
Image NMSE (dB)		−29.57	−29.55	−29.63
Classification accuracy		83.61%	83.58%	83.51%

[*] The "Entropy 1" represents the method that selects the sub-block with the largest entropy.

[**] The "Entropy 2" represents the method that selects the sub-block with a smaller bit length increase in the first two sub-blocks with the largest entropy.

First, we evaluate the effects of the block selection strategy on bit length change ΔB. The increase in bit length is huge, making the proposed method unreasonable. Exhaustive search is then adopted, and the bit length of each block is calculated. The smallest ΔB is regarded as the expense of hiding CSI in images. In this case, the bit number of the container images is smaller than that of the host images. That is, no extra overheads are incurred, and the elimination of the feedback overhead is not at the expense of bit length increase in the transmitted images.

However, the complexity of the exhaustive search is high. Thus, we evaluate the performance of the introduction of image entropy. When we directly select the sub-block with the largest entropy, the bit length increase is much smaller than that of random selection. For example, when the scaling factor is set as 0.005, the extra bit numbers of random selection and "Entropy 1" are 452 and 128, respectively. However, the increase in "Entropy 1" is still much larger than that in the exhaustive search. This result shows that the mapping between entropy and bit length change does not strictly hold. If the length increase of the selected sub-block is larger than a predefined threshold, we can select the sub-block with the second-largest entropy and compare the length changes of the two selected blocks to search for the suitable block. As shown in this table, the bit length increase can be reduced when the first two blocks are selected and compared.

Second, we evaluate the effects of the proposed block-based EliCsiNet with two-stage separable training. As indicated in Tables III and IV, the accuracy of the CSI obtained by the block-based EliCsiNet is much lower than that obtained by the full image-based EliCsiNet. This comparison is unfair because the larger the image is, the more the information can be hidden. For example, we divide the full image into many $7 \times 7 \times 3$ sub-blocks. These sub-blocks are smooth and cannot be hidden with downlink CSI. The images in practical systems are much larger than the images in ImageNet. Therefore, the sub-blocks in practical systems are larger and can contain more information with high accuracy. Although the block is relatively small, it still can eliminate the CSI feedback overhead. For example, in [24], 512 bits needed to be fed back when NMSE was -12.58 dB. In this work, if the

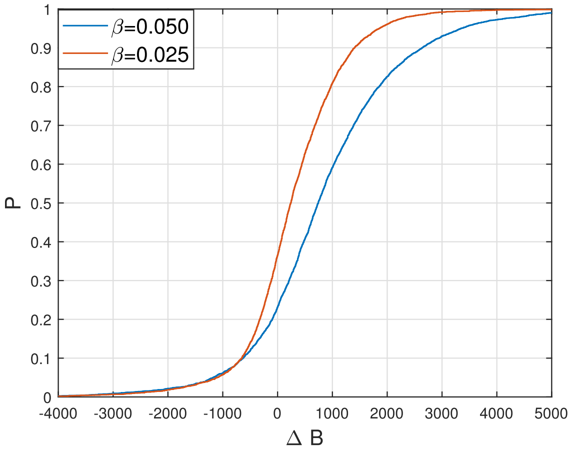


Fig. 11. Cumulative distribution function of the increase in bit length. Scaling factor β is set as 0.025 and 0.05. When β is 0.025, 40% of the increments exceed 1,000 bits. When β is 0.05, 20% of the increments exceed 1,000. This situation leads to a substantial extra overhead, thereby making the proposed method unreasonable.

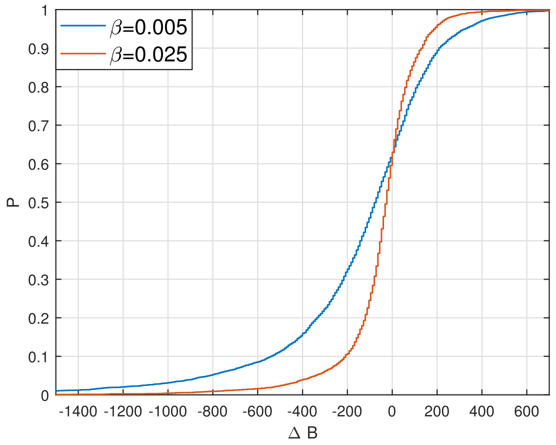


Fig. 12. Cumulative distribution function of the increase in bit length when scaling factor β is set as 0.005 and 0.025. Only 10% of the increments exceed 200 bits. The performance of the proposed block-based method is stable.

exhaustive search was adopted, then no extra bits would be needed when β is 0.005 and the CSI NMSE is -12.60 dB.

Fig. 12 shows the cumulative distribution function of the increase in bit length when scaling factor β is set as 0.005 and 0.025. According to this figure, 90% of the length increments are below 200. Therefore, the performance of the proposed block-based EliCsiNet is stable, and the proposed method is feasible for practical systems.

4) Complexity of the Proposed NNs: Wireless systems have a high requirement of algorithm complexity. The complexity of the CSI feedback and the delay introduced to the original message need to be evaluated. During the NN inference, the average running times of the hiding and the reconstruction operations in this work are approximately 0.0025 and 0.0008 s, respectively. Therefore, the delay introduced in the original message transmission is 0.0025 s. The network's running speed can be further accelerated due to the following reasons:

- During the NN design, we did not consider the NN complexity; thus, the NNs adopted in this work became redundant. If an efficient NN architecture, such as shuffle NN unit in [55], was adopted, then the NN's complexity can be significantly reduced.
- NN compression techniques can further reduce the NN complexity [43]. For example, weight quantization/binarization [43], [56], knowledge distillation [57], and weight pruning [43] were introduced to DL-based CSI feedback to reduce NN's complexity.
- We ran the EliCsiNet on a GPU platform. General purpose platforms cannot be directly deployed in communications. Specific platforms, such as the application-specific integrated circuit (ASIC), might be delicately designed to accelerate inference speed and reduce energy consumption.

VI. Conclusion and Future Work

We developed a DL-based CSI hiding framework, namely, EliCsiNet, to eliminate the CSI feedback overhead in FDD MIMO systems. Unlike most existing DL-based feedback frameworks that focus on reducing the overhead through an autoencoder architecture, we introduced DL-based data hiding to the CSI feedback problem. The key idea was to hide/superimpose downlink CSI in the transmitted messages (e.g., images) with no transmission resource occupation and few effects on message semantics.

First, we designed a novel NN framework, in which PreNet extracts the CSI features, HidNet superimposes the CSI features onto the original images, and RecNet at the user side recovers the CSI from the transmitted images. Second, we examined the effects of source coding on the proposed EliCsiNet and used JPEG compression as an example. To ensure CSI reconstruction accuracy after JPEG compression, we developed a two-stage separable training strategy. Third, to avoid a large increase in the bit length of the coded images, we divided the full image into several sub-blocks and selected the one with the smallest length increase. Moreover, image entropy was introduced to accelerate the block selection. Our simulation revealed that the proposed EliCsiNet framework can eliminate the CSI feedback overhead with few effects on the transmitted images, including image quality and bit length.

Some challenges are worth exploring further in the future despite the promising results of the DL-based data hiding-enabled CSI feedback.

- First, the performance of the data hiding-aided CSI feedback scheme needs to be evaluated when different transmitted media, such as texts, are considered.
- Second, the complexity of the NNs adopted in this work needs to be reduced by introducing novel NN architectures and NN compression techniques.
- Third, the imperfect uplink transmission needs to be considered. JPEG assumes that the transmission link is perfect, making it sensitive to uplink noise. Therefore, much more advanced JPEG compression techniques, such as the wireless extension of JPEG 2000 (dubbed JPWL) [58], need to be introduced to this work, and the CSI feedback performance needs to be evaluated when imperfect uplink transmissions are considered.
- Fourth, the reconstructed CSI undoubtedly contains the image information and can be exploited to improve the

quality of container images if extremely high image quality and extremely low transmission latency are required.

- Lastly, the mapping between entropy and bit length does not strictly hold. More advanced selection strategies need to be explored to reduce the bit length increase.

References

[1] J. Guo, C.-K. Wen, and S. Jin, "Deep data hiding-based CSI feedback overhead elimination: An initial investigation," in *Proc. IEEE ICC*, May 2022, pp. 1–6.

[2] L. Lu, G. Y. Li, A. L. Swindlehurst, A. Ashikhmin, and R. Zhang, "An overview of massive MIMO: Benefits and challenges," *IEEE J. Sel. Topics Signal Process.*, vol. 8, no. 5, pp. 742–758, Oct. 2014.

[3] Q. H. Spencer, C. B. Peel, A. L. Swindlehurst, and M. Haardt, "An introduction to the multi-user MIMO downlink," *IEEE Commun. Mag.*, vol. 42, no. 10, pp. 60–67, Oct. 2004.

[4] M. Matthaiou, C. Zhong, M. R. McKay, and T. Ratnarajah, "Sum rate analysis of ZF receivers in distributed MIMO systems," *IEEE J. Sel. Areas Commun.*, vol. 31, no. 2, pp. 180–191, Feb. 2013.

[5] D. Lee *et al.*, "Coordinated multipoint transmission and reception in LTE-advanced: Deployment scenarios and operational challenges," *IEEE Commun. Mag.*, vol. 50, no. 2, pp. 148–155, Feb. 2012.

[6] D. J. Love, R. W. Heath, Jr., V. K. Lau, D. Gesbert, B. D. Rao, and M. Andrews, "An overview of limited feedback in wireless communication systems," *IEEE J. Sel. Areas Commun.*, vol. 26, no. 8, pp. 1341–1365, Oct. 2008.

[7] C. Luo, J. Ji, Q. Wang, X. Chen, and P. Li, "Channel state information prediction for 5G wireless communications: A deep learning approach," *IEEE Trans. Netw. Sci. Eng.*, vol. 7, no. 1, pp. 227–236, Jan. 2020.

[8] D. Xu, Y. Huang, and L. Yang, "Feedback of downlink channel state information based on superimposed coding," *IEEE Commun. Lett.*, vol. 11, no. 3, pp. 240–242, Mar. 2007.

[9] C. Qing, B. Cai, Q. Yang, J. Wang, and C. Huang, "Deep learning for CSI feedback based on superimposed coding," *IEEE Access*, vol. 7, pp. 93723–93733, 2019.

[10] E. C. Strinati and S. Barbarossa, "6G networks: Beyond Shannon towards semantic and goal-oriented communications," *Comput. Netw.*, vol. 190, May 2021, Art. no. 107930.

[11] G. Shi, Y. Xiao, Y. Li, and X. Xie, "From semantic communication to semantic-aware networking: Model, architecture, and open problems," *IEEE Commun. Mag.*, vol. 59, no. 8, pp. 44–50, Aug. 2021.

[12] Z. Zhang, Q. Yang, S. He, M. Sun, and J. Chen, "Wireless transmission of images with the assistance of multi-level semantic information," 2022, *arXiv:2202.04754*.

[13] J. Wang, J. Liu, and N. Kato, "Networking and communications in autonomous driving: A survey," *IEEE Commun. Surveys Tuts.*, vol. 21, no. 2, pp. 1243–1274, 2nd Quart., 2019.

[14] S.-W. Kim, K. Ko, H. Ko, and V. C. M. Leung, "Edge-network-assisted real-time object detection framework for autonomous driving," *IEEE Netw.*, vol. 35, no. 1, pp. 177–183, Jan. 2021.

[15] H. Xie, Z. Qin, G. Y. Li, and B.-H. Juang, "Deep learning enabled semantic communication systems," *IEEE Trans. Signal Process.*, vol. 69, pp. 2663–2675, 2021.

[16] C. Zhang, C. Lin, P. Benz, K. Chen, W. Zhang, and I. So Kweon, "A brief survey on deep learning based data hiding," 2021, *arXiv:2103.01607*.

[17] A. Rasmi, B. Arunkumar, and V. M. Anees, "A comprehensive review of digital data hiding techniques," *Pattern Recognit. Image Anal.*, vol. 29, no. 4, pp. 639–646, Oct. 2019.

[18] Z. Yang, X. Guo, Z. Chen, Y. Huang, and Y. Zhang, "RNN-stega: Linguistic steganography based on recurrent neural networks," *IEEE Trans. Inf. Forensics Security*, vol. 14, no. 5, pp. 1280–1295, May 2019.

[19] S. Abdelnabi and M. Fritz, "Adversarial watermarking transformer: Towards tracing text provenance with data hiding," in *Proc. IEEE Symp. Secur. Privacy (SP)*, May 2021, pp. 121–140.

[20] S. Baluja, "Hiding images within images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 42, no. 7, pp. 1685–1697, Jul. 2020.

[21] G. Wallace, "The JPEG still picture compression standard," *IEEE Trans. Consum. Electron.*, vol. 38, no. 1, pp. 18–34, Feb. 1992.

[22] M. Rabbani and R. Joshi, "An overview of the JPEG 2000 still image compression standard," *Signal Process., Image Commun.*, vol. 17, no. 1, pp. 3–48, Jan. 2002.

[23] C. Zhang, A. Karjauv, P. Benz, and I. S. Kweon, "Towards robust data hiding against (JPEG) compression: A pseudo-differentiable deep learning approach," 2020, *arXiv:2101.00973*.

[24] J. Guo, C.-K. Wen, S. Jin, and G. Y. Li, "Convolutional neural network-based multiple-rate compressive sensing for massive MIMO CSI feedback: Design, simulation, and analysis," *IEEE Trans. Wireless Commun.*, vol. 19, no. 4, pp. 2827–2840, Apr. 2020.

[25] A. Ghosh, "5G new radio (NR): Physical layer overview and performance," in *Proc. IEEE Commun. Theory Workshop*, May 2018, pp. 1–38.

[26] Y. Zhang, C. Wei, R. Zheng, N. Wang, and J. Hou, "Enabling 3D MIMO in LTE-advanced—From the CSI acquisition perspective," in *Proc. IEEE ICNC*, Feb. 2015, pp. 42–46.

[27] Z. Qin, J. Fan, Y. Liu, Y. Gao, and G. Y. Li, "Sparse representation for wireless communications: A compressive sensing approach," *IEEE Signal Process. Mag.*, vol. 35, no. 3, pp. 40–58, May 2018.

[28] P.-H. Kuo, H. T. Kung, and P.-A. Ting, "Compressive sensing based channel feedback protocols for spatially-correlated massive antenna arrays," in *Proc. IEEE Wireless Commun. Netw. Conf. (WCNC)*, Apr. 2012, pp. 492–497.

[29] Z. Qin, H. Ye, G. Y. Li, and B. H. F. Juang, "Deep learning in physical layer communications," *IEEE Wireless Commun.*, vol. 26, no. 2, pp. 93–99, Mar. 2019.

[30] D. C. Nguyen *et al.*, "Enabling AI in future wireless networks: A data life cycle perspective," *IEEE Commun. Surveys Tuts.*, vol. 23, no. 1, pp. 553–595, 1st Quart., 2021.

[31] C.-K. Wen, W.-T. Shih, and S. Jin, "Deep learning for massive MIMO CSI feedback," *IEEE Wireless Commun. Lett.*, vol. 7, no. 5, pp. 748–751, Oct. 2018.

[32] *New SI: Study on Artificial Intelligence (AI)/Machine Learning (ML) for NR Air Interface*, document RP-213599, 3GPP, Moderator (Qualcomm), Dec. 2021. Accessed: Mar. 1, 2022. [Online]. Available: https://www.3gpp.org/ftp/tsg_ran/TSG_RAN/TSGR_94e/Docs/RP-213599.zip

[33] P. K. Sangdeh, H. Pirayesh, A. Mobiny, and H. Zeng, "LB-SciFi: Online learning-based channel feedback for MU-MIMO in wireless LANs," in *Proc. IEEE 28th Int. Conf. Netw. Protocols (ICNP)*, Oct. 2020, pp. 1–11.

[34] K. B. Letaief, Y. Shi, J. Lu, and J. Lu, "Edge artificial intelligence for 6G: Vision, enabling technologies, and applications," *IEEE J. Sel. Areas Commun.*, vol. 40, no. 1, pp. 5–36, Jan. 2022.

[35] P. Sun, "Evolution to 5.5G and 6G and key applications," in *Unleashing the Power of 5GtoB in Industries*. Singapore: Springer, 2021, pp. 281–287.

[36] Huawei. (2021). *NetX2025 Target Network Technical White Paper*. [Online]. Available: https://carrier.huawei.com/~/media/CNBGV2/download/program/netx2025/netx-2025-target-network-technical-white-paper-en.pdf

[37] P. Liang, J. Fan, W. Shen, Z. Qin, and G. Y. Li, "Deep learning and compressive sensing-based CSI feedback in FDD massive MIMO systems," *IEEE Trans. Veh. Technol.*, vol. 69, no. 8, pp. 9217–9222, Aug. 2020.

[38] Q. Yang, M. B. Mashhadi, and D. Gündüz, "Deep convolutional compression for massive MIMO CSI feedback," in *Proc. IEEE 29th Int. Workshop Mach. Learn. Signal Process. (MLSP)*, Oct. 2019, pp. 1–6.

[39] H. Ye, F. Gao, J. Qian, H. Wang, and G. Y. Li, "Deep learning-based denoise network for CSI feedback in FDD massive MIMO systems," *IEEE Commun. Lett.*, vol. 24, no. 8, pp. 1742–1746, Aug. 2020.

[40] Y. Hu, W. Yang, Z. Ma, and J. Liu, "Learning end-to-end lossy image compression: A benchmark," *IEEE Trans. Pattern Anal. Mach. Intell.*, early access, Mar. 11, 2021, doi: 10.1109/TPAMI.2021.3065339.

[41] Y. Yang, F. Gao, C. Xing, J. An, and A. Alkhateeb, "Deep multimodal learning: Merging sensory data for massive MIMO channel prediction," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 7, pp. 1885–1898, Jul. 2021.

[42] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. 26th NIPS*, 2012, pp. 1097–1105.

[43] J. Guo, J. Wang, C.-K. Wen, S. Jin, and G. Y. Li, "Compression and acceleration of neural networks for communications," *IEEE Wireless Commun.*, vol. 27, no. 4, pp. 110–117, Aug. 2020.

[44] Y. Liu, M. Guo, J. Zhang, Y. Zhu, and X. Xie, "A novel two-stage separable deep learning framework for practical blind watermarking," in *Proc. 27th ACM Int. Conf. Multimedia*, Oct. 2019, pp. 1509–1517.

[45] K.-H. Jung and K.-Y. Yoo, "Data hiding method using image interpolation," *Comput. Standards Interfaces*, vol. 31, no. 2, pp. 465–470, Feb. 2009.

[46] Y. Li, S. Yao, K. Yang, Y.-A. Tan, and Q. Zhang, "A high-imperceptibility and histogram-shifting data hiding scheme for JPEG images," *IEEE Access*, vol. 7, pp. 73573–73582, 2019.

[47] W. Shi, F. Jiang, S. Liu, and D. Zhao, "Image compressed sensing using convolutional neural network," *IEEE Trans. Image Process.*, vol. 29, pp. 375–388, 2020.

[48] Y. Wu, X. Li, Z. Zhang, X. Jin, and Z. Chen, "Learned block-based hybrid image compression," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 32, no. 6, pp. 3978–3990, Jun. 2022.

[49] P. Molchanov, S. Tyree, T. Karras, T. Aila, and J. Kautz, "Pruning convolutional neural networks for resource efficient inference," in *Proc. 5th ICLR*, 2017, pp. 1–17.

[50] R. Dony *et al.*, "Karhunen–Loeve transform," in *The Transform and Data Compression Handbook*, vol. 1, nos. 1–34. Boca Raton, FL, USA: CRC Press, 2001, p. 29.

[51] MathWorks. *Entropy of Grayscale Image*. Accessed: Nov. 10, 2021. [Online]. Available: https://ww2.mathworks.cn/help/images/ref/entropy.html?lang=en

[52] L. Liu *et al.*, "The COST 2100 MIMO channel model," *IEEE Wireless Commun.*, vol. 19, no. 6, pp. 92–99, Dec. 2012.

[53] H. R. Sheikh and A. C. Bovik, "Image information and visual quality," *IEEE Trans. Image Process.*, vol. 15, no. 2, pp. 430–444, Feb. 2006.

[54] Z. Wang and A. C. Bovik, "A universal image quality index," *IEEE Signal Process. Lett.*, vol. 9, no. 3, pp. 81–84, Aug. 2002.

[55] Z. Cao, W.-T. Shih, J. Guo, C.-K. Wen, and S. Jin, "Lightweight convolutional neural networks for CSI feedback in massive MIMO," *IEEE Commun. Lett.*, vol. 25, no. 8, pp. 2624–2628, Aug. 2021.

[56] Z. Lu, J. Wang, and J. Song, "Binary neural network aided CSI feedback in massive MIMO system," *IEEE Wireless Commun. Lett.*, vol. 10, no. 6, pp. 1305–1308, Jun. 2021.

[57] H. Tang, J. Guo, M. Matthaiou, C.-K. Wen, and S. Jin, "Knowledge-distillation-aided lightweight neural network for massive MIMO CSI feedback," in *Proc. IEEE 94th Veh. Technol. Conf. (VTC-Fall)*, Sep. 2021, pp. 1–5.

[58] F. Dufaux, G. Baruffa, F. Frescura, and D. Nicholson, "JPWL—An extension of JPEG 2000 for wireless imaging," in *Proc. IEEE Int. Symp. Circuits Syst.*, May 2006, pp. 3873–4870.

Jiajia Guo (Graduate Student Member, IEEE) received the B.S. degree from the Nanjing University of Science and Technology, Nanjing, China, in 2016, and the M.S. degree from the University of Science and Technology of China, Hefei, China, in 2019. He is currently pursuing the Ph.D. degree in information and communications engineering with Southeast University, Nanjing. His current research interests include deep learning for physical layer, distributed learning, and neural network compression. He was a recipient of the Exemplary Reviewer of IEEE WIRELESS COMMUNICATIONS LETTERS in 2021.

Chao-Kai Wen (Senior Member, IEEE) received the Ph.D. degree from the Institute of Communications Engineering, National Tsing Hua University, Taiwan, in 2004. He was with the Industrial Technology Research Institute, Hsinchu, Taiwan, and MediaTek Inc., Hsinchu, from 2004 to 2009, where he was engaged in broadband digital transceiver design. Since 2009, he has been with the Institute of Communications Engineering, National Sun Yat-sen University, Kaohsiung, Taiwan, where he is currently a Professor. His research interest includes optimization in wireless multimedia networks.

Shi Jin (Senior Member, IEEE) received the B.S. degree in communications engineering from the Guilin University of Electronic Technology, Guilin, China, in 1996, the M.S. degree from the Nanjing University of Posts and Telecommunications, Nanjing, China, in 2003, and the Ph.D. degree in information and communications engineering from Southeast University, Nanjing, in 2007. From June 2007 to October 2009, he was a Research Fellow with the Adastral Park Research Campus, University College London, London, U.K. He is currently with the Faculty of the National Mobile Communications Research Laboratory, Southeast University. His research interests include space time wireless communications, random matrix theory, and information theory. He and his coauthors have been awarded the 2011 IEEE Communications Society Stephen O. Rice Prize Paper Award in the field of communication theory and the 2010 Young Author Best Paper Award by the IEEE Signal Processing Society. He served as an Associate Editor for the IEEE TRANSACTIONS ON WIRELESS COMMUNICATIONS, IEEE COMMUNICATIONS LETTERS, and *IET Communications*.