Lossless Compression Framework Using Lossy Prior for High-Resolution Remote Sensing Images

Enjia Gu, Yongshan Zhang , *Member, IEEE*, Xinxin Wang , and Xinwei Jiang

Abstract—**Lossless compression of remote sensing images is critically important for minimizing storage requirements while preserving the complete integrity of the data. The main challenge in lossless compression lies in striking a good balance between reasonable compression durations and high compression ratios. In this article, we introduce an innovative lossless compression framework that uniquely utilizes lossy compression data as prior knowledge to enhance the compression process. Our framework employs a checkerboard segmentation technique to divides the original remote sensing image into various subimages. The main diagonal subimages are compressed using a traditional lossy method to obtain prior knowledge for facilitating the compression of all subimages. These subimages are then subjected to lossless compression using our newly developed lossy prior probability prediction network (LP3Net) and arithmetic coding in a specific order. The proposed LP3Net is an advanced network architecture, consisting of an image preprocessing module, a channel enhancement module, and a pixel probability transformer module, to learn the discrete probability distribution of each pixel within every subimage, enhancing the accuracy and efficiency of the compression process. Experiments on high-resolution remote sensing image datasets demonstrate the effectiveness and efficiency of the proposed LP3Net and lossless compression framework, achieving a minimum of 4.57% improvement over traditional compression methods and 1.86% improvement over deep learning-based compression methods.**

Index Terms—**Arithmetic coding, checkerboard segmentation, discrete probability prediction, JPEG XL resampling, lossless remote sensing image compression.**

I. INTRODUCTION

REMOTE sensing images are obtained from spacecraft, satellites or high-altitude platforms using high-precision sensors to capture information about the Earth's surface from a distance [1], [2]. From military reconnaissance to geographic information systems, and environmental monitoring to disaster management, remote sensing images have become a crucial means for obtaining and analyzing information about the Earth's surface [3], [4], [5]. With the rapid development of remote sensing technology, these devices capture remote sensing images that contain increasingly large amounts of information, along with improved image quality. However, the high quality of these remote sensing images have also led to a dramatic increase in data volume, posing significant technical challenges for data storage and transmission.

Image compression methods are originally proposed for conventional RGB images that can be categorized into lossy and lossless image compression [6], [7], [8]. Evidently, lossy compression results in the loss of original data, which can significantly reduce storage space while sacrificing a certain degree of visual quality. In terms of traditional methods, lossy compression algorithms include JPEG, BPG, and AVIF, among others. For instance, JPEG reduces storage size by converting images to a format that separates brightness from color data, then compresses them using techniques like downsampling and Huffman coding. In recent years, a substantial amount of research based on deep learning has been conducted in the field of lossy compression. Yang et al. [7] proposed a lossy image compression framework based on diffusion generative models, employing a conditional diffusion model as the decoder to enhance compression and reconstruction by introducing a "content" latent variable. Kumari et al. [9] developed a lossy image compression scheme called quantization-aware ResNet variational autoencoder, enhancing variational autoencoders with hierarchical structures and quantization-aware training for improved rate-distortion performance and efficient entropy coding. In contrast, lossless compression algorithms preserve the complete original data of images. Classical lossless image compression algorithms include PNG, JPEG 2000, and JPEG XL. JPEG XL employs a combination of modern coding techniques including advanced prediction, transformation, and entropy coding methods to achieve efficient lossless compression with excellent performance across a wide range of image types. Recently, deep learning-based lossless compression methods have also been progressively explored. Wang et al. [10] proposed an end-to-end lossless image compression method that combines an autoregressive model with a multivariant mixture distribution channel-conditioning model to enhance compression performance. Kamisli et al. [11] introduced a scale-based autoregressive deep learning method for lossless image compression that reduces computational complexity while maintaining compression performance. Although the aforementioned algorithms demonstrate superior compression performance on traditional RGB images, their effectiveness on remote sensing images tends to be more limited.

Compared to traditional RGB images, remote sensing images exhibit a higher spatial resolution. In addition, remote sensing images captured by the same high-altitude equipment display a high degree of structural complexity and significant redundancy in color blocks. Consequently, developing specialized algorithms for compressing remote sensing images also presents considerable challenges. Therefore, research on remote sensing image compression is not as extensive as that on traditional RGB image compression. Similar to RGB images, remote sensing image compression can be also categorized into lossy and lossless methods. Currently, most scholars focus on lossy compression techniques for remote sensing images, Fu and Du [12] introduced a learned lossy image compression framework for high-resolution remote sensing images, employing a mixed hyperprior network that integrates transformer and convolutional neural network-based approaches for accurate entropy estimation and a three-stage training strategy to enhance network refinement. Verdú et al. [13] presented a new fixed-quality image compression method for remote sensing images, utilizing a neural architecture that leverages advanced statistical modeling to efficiently estimate reconstruction quality without introducing artifacts. Abramova et al. [14] proposed a fast and accurate procedure for setting the quantization step in lossy remote sensing image compression using discrete cosine transform coders, based on predicting mean square errors through a simple analysis of image complexity, effectively controlling distortion in compressed images while maintaining desired image quality. Ye et al. [15] introduced the global feature-assisted segmentation compression network, which employs a dual-branch segmentation strategy enhanced by global features and an attention mechanism, alongside code rate allocation and a joint hyperpriority decoding and entropy coding network, to achieve high compression ratios while preserving small target details in remote sensing images. Although these lossy compression algorithms significantly reduce the data volume of remote sensing images, they may lose some critical original information and adversely affect further applications of remote sensing images, such as classification and semantic segmentation. Given the high acquisition costs of remote sensing images and their significant value in military, agriculture, and natural disaster research, the study of lossless compression holds greater potential value.

Lossless remote sensing image compression technology ensures that no information is lost during the compression and decompression of remote sensing images. This technology is especially important for applications that demand the integrity and originality of data. Wang and Zhang [16] proposed a remote sensing image compression technique using ANS entropy coding, which outperforms Golomb coding in JPEG-LS in terms of simplicity, efficiency, and reduced redundancy, effectively enhancing compression ratios, reducing data downlink volume, and increasing transmission speeds for large-format satellite images. Kumari and Sriramulu [9] introduced a novel lossless image compression methodology using K-means clustering, which effectively reduces image file sizes without compromising quality by grouping similar data points to minimize redundancy. Furthermore, research on lossless remote sensing

image compression technology utilizing deep learning is currently a hot topic in the field. Feng et al. [17] proposed a novel deep learning-based lossless compression framework for three-channel remote sensing images. This framework employs a coarse-to-fine approach with checkerboard segmentation and utilizes a probability prediction network (P2Net) combined with arithmetic coding. It is designed to effectively preserve structural information and image details within the prior knowledge, thereby enhancing the overall efficiency of lossless compression. Kahali et al. [18] proposed a novel method for lossless image compression using arithmetic encoding combined with the low-power capabilities of spiking neural networks on neuromorphic platforms, specifically designed to address the challenges of on-board data processing and transmission in small satellites with limited power and computational capacity. Anuradha et al. [19] introduced an advanced lossless compression technique for hyperspectral imaging, combining deep recurrent neural networks with discrete wavelet transform to effectively capture and compress high-dimensional spectral and spatial information, enhancing storage and transmission efficiency for large-scale multispectral data collection. However, the compression effectiveness and efficiency of these techniques on high-resolution remote sensing images remains generally moderate.

Inspired by the abovementioned observations, this article proposes a lossless compression framework based on lossy prior, as shown in Fig. 1. Specifically, considering the high similarity in the color blocks of remote sensing images, we adopt a checkerboard segmentation technique to divide the original image into subimages. Subsequently, we employ a lossy compression method to compress subimages from the main diagonal to serve as prior knowledge. Then, using the proposed lossy prior probability prediction network (LP3Net) and arithmetic coding, the lossy-compressed subimages from the main diagonal and the previously encoded subimages are used as prior knowledge to compress the remaining subimages one by one. The proposed LP3Net consists of an image preprocessing module, a channel enhancement module, and a pixel probability transformer module, designed to learn the discrete probability distribution of each pixel in every subimage. For the subimages requiring compression, the discrete probability distribution is combined to enable lossless compression using arithmetic coding. Experiments on multiple high-resolution remote sensing image datasets demonstrate the effectiveness and superiority of the proposed framework compared to the state-of-the-art methods.

In summary, our contributions are shown as follows.

1) We propose a lossless compression framework with lossy prior knowledge based on checkerboard segmentation and JPEG XL resampling. This framework compresses remote sensing images in a coarse-to-fine manner, ensuring that no original image information is lost.

2) Based on prior knowledge, we design LP3Net, consisting of an image preprocessing module, a channel enhancement module, and a probability prediction module. This network learns the probability distribution of each pixel to enable lossless compression with arithmetic coding.
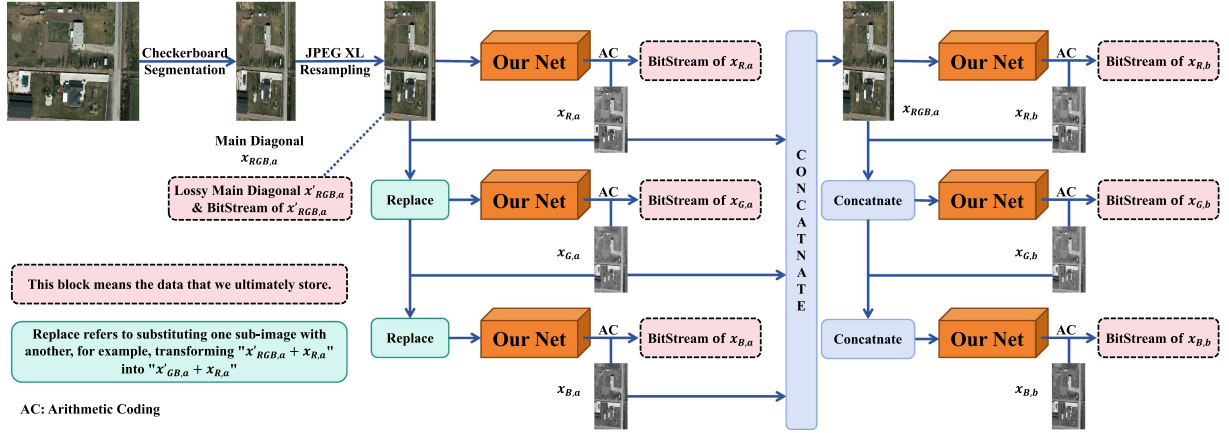
Fig. 1. Overall procedure of the proposed lossless compression framework designed for remote sensing images. Given a remote sensing image as input, checkerboard segmentation is initially applied to divide the image into six subimages. The main diagonal subimages are compressed using a lossy method to produce prior knowledge, which is then used to predict the discrete probability distribution of the entire image by LP3Net. The prediction process is performed in a specific order, with the main diagonal subimages predicted first, followed by the counter-diagonal subimages. The discrete probability distribution of each subimage is then encoded using arithmetic coding to complete the compression process.

3) Experiments on remote sensing datasets verify the effectiveness the proposed method compared to the state-of-the-art methods.

The rest of this article is organized as follows. Section II reviews the related works. Section III presents detailed descriptions of the proposed lossless compression framework and the specifically designed LP3Net. Section V discusses comprehensive experiments along with their corresponding analyses. Finally, Section VI concludes this article.

II. RELATED WORKS

A. Lossy Remote Sensing Image Compression

Lossy remote sensing image compression sacrifices the quality of the original data to achieve a reduction in storage size. Zhang et al. [20], proposed the spatial-temporal context model for remote sensing image compression, utilizing a stacked diagonal masked module and spatial-temporal contextual adaptive coding to effectively harness spatial scope and temporal redundancy in remote sensing images for enhanced compression efficiency and accuracy in downstream processing. Shi et al. [21] introduced the enhanced global feature-guided network for remote sensing image compression, which incorporates a pyramid vision transformer and a novel quadruple-filtered noise reduction attention module, using multilevel feature processing and a global visual feature guidance module to more effectively compress and preserve critical visual features of remote sensing images. Zhang et al. [22] proposed a novel compression framework for Gaofen satellite multispectral images that employs a lightweight bit-depth recovery network to split the image into most significant bits (MSB) and least significant bits (LSB), where the MSB is losslessly compressed and the LSB is efficiently encoded using the network, effectively addressing the challenge of maintaining high reconstruction quality while improving compression ratios. The work proposed by Ma et al. [23] introduced a novel spatiotemporal compression

pipeline for remote sensing images based on lossy compression methods, using adaptive encoding techniques with a multilayer perceptron neural network to effectively reduce the data volume.

Lossy compression of remote sensing images, despite achieving significant reductions in data volume with compression ratios ranging from 1:5 to 1:100, inevitably involves the loss of crucial data. This is particularly significant given the invaluable nature of remote sensing imagery, where every bit of data can be critical for accurate analysis and application.

B. Lossless Remote Sensing Image Compression

Unlike lossy remote sensing image compression, lossless compression reduces data volume without any loss of information or degradation in image quality. This method ensures that the compressed image can be perfectly restored, with each pixel exactly matching the original image. Joshi and Rani [24] proposed a prediction-based lossless compression algorithm for on-board satellite hyperspectral data, which utilizes nonbinary tree traversal and nearest neighbor methods with neighbor-driven decision-making in the preprocessing stage. This approach effectively handles the high data acquisition rates and limited computational resources in satellites, reducing computational complexity while maintaining error resilience. Gao et al. [25] proposed an improved lossless compression method that optimally reorders bands based on their correlation, groups them using the minimum spanning tree algorithm to enhance interspectral correlations, and employs adaptive prediction to minimize redundancy. This method then applies binary arithmetic coding to the residuals, achieving efficient lossless compression. Valsesia et al. [26] proposed a predictive neural network named LineRWKV for hyperspectral image compression onboard spacecrafts, utilizing a hybrid attentive-recursive operation that combines the representational advantages of Transformers with the efficiency of recurrent neural networks to reduce memory usage and manage the computational complexity. Zhu and Hu [27] proposed

a three-level cascaded prediction-based lossless compression method for hyperspectral imagery, which employs a k-means algorithm for preprocessing, a cascaded predictor combining similar neighborhood mean, recursive least squares, and least mean square predictors to respectively address spatial and spectral redundancies, and a Huffman encoder for efficient coding of prediction residuals, thereby reducing computational complexity and enhancing compression efficiency. Feng et al. [17] proposed a deep learning-based lossless compression framework for remote sensing images, employing a coarse-to-fine approach with checkerboard segmentation to preserve detail and structure, and a novel P2Net coupled with arithmetic coding to efficiently encode images, significantly enhancing compression efficiency without information loss.

While existing methods achieve lossless compression and maintain data integrity for remote sensing images, they tend to have lower compression efficiency when applied to high-resolution three-channel images. Consequently, we are developing an efficient deep learning-based framework specifically designed for lossless compression of high-resolution three-channel remote sensing images, aimed at maximizing compression efficiency within reasonable encoding and decoding times.

III. PROPOSED METHOD

A. Overview

The overall proposed lossless compression framework for remote sensing images is depicted in Fig. 1. The compression process begins by applying a checkerboard segmentation to the original 3-channel remote sensing image. This segmentation technique divides the image into six subimages across spatial and channel dimensions. Three subimages from the main diagonal segmentation are initially compressed using a resampling method to produce their lossy-compressed versions that serve as prior knowledge. Since we do not retain any lossless prior knowledge, we develop a specialized prediction network with lossy prior to predict the discrete probability distribution of the entire remote sensing image for lossless compression. It should be noted that the discrete probability distribution for the entire image is predicted in a specific order.

Given that our stored prior knowledge is based on the main diagonal's original images, we first predict the discrete probability distribution of the main diagonal images on a per-channel basis. For instance, when compressing the first channel subimage of the main diagonal, the lossy-compressed subimages from the main diagonal are used as input. When compressing the second-channel subimage of the main diagonal, both the lossy-compressed subimages from the main diagonal and the already compressed first-channel subimage are used as inputs. This process is repeated for the third-channel main diagonal subimage to predict the discrete probability distributions. When predicting the first-channel subimage of the counter-diagonal, the compressed main diagonal subimages are used as inputs. Similarly, for compressing the second-channel subimage of the counter-diagonal, both the compressed main diagonal subimages and the compressed first-channel subimage from the counter-diagonal are used as inputs. This process continues for the third-channel
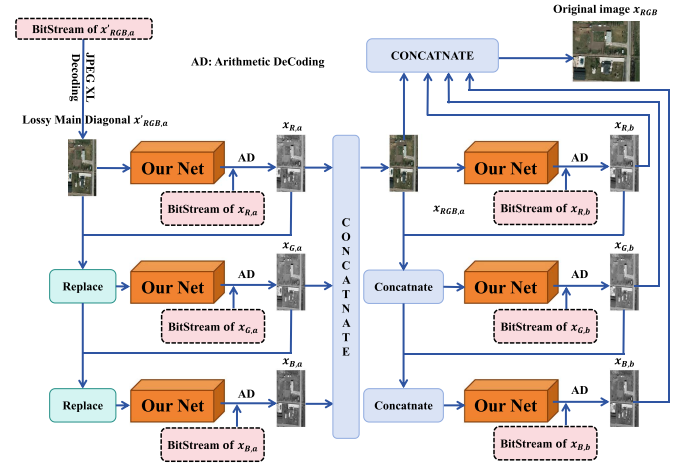

Fig. 2. Graphical illustration of decompression process.

counter-diagonal subimage, until predictions for the discrete probability distributions of all six subimages are completed. Subsequently, each subimage is encoded in sequence using arithmetic coding to finalize the compression of the remote sensing image. Each step of the compression process is performed in a specific order to optimize the compression efficiency of the remote sensing images. As shown in Fig. 1, our framework involves multiple concatenation and replacement operations. However, these operations do not significantly increase the overall computational overhead, as the data remains in GPU memory without undergoing computation or network inference. The primary time cost of this framework is attributed to the network inference itself.

Fig. 2 illustrates the inverse process of Fig. 1, depicting the decompression procedure. In the decompression stage, we first decode the encoded stream of the lossy-compressed image to obtain the lossy version of the main diagonal subimages. When decoding the first-channel original subimage of the main diagonal, the already decoded lossy main diagonal image is used as input. Similarly, when decoding the second-channel subimage of the main diagonal, both the lossy main diagonal subimages and the reconstructed first-channel main diagonal subimage are used as inputs. Subsequently, the counter-diagonal subimages are decoded in a similar recursive manner. This process continues until all subimages are fully recovered. By following this structured decoding order, the original remote sensing image is completely reconstructed, ensuring lossless decompression.

B. Checkerboard Segmentation

It is worth mentioning that, unlike the work in [28], we do not perform an initial color space transformation before compressing the remote sensing images. This is because reversible color space transformation introduce additional bit overhead [29]. It will complicate the network's prediction tasks.

For learning algorithms that perform compression on a subimage basis, it is crucial to divide the original image into suitable subimages for compression. Following the work in [17], we
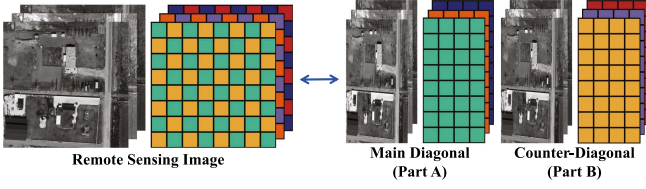
Fig. 3. Graphical illustration of three-channel checkerboard segmentation, with the main diagonal subimages referred to as "Part A" and the counter-diagonal subimages as "Part B."

Fig. 4. Graphical illustration of the effects of resampling parameters 2, 4, and 8 on image quality, with parameters detailed in the image.

apply checkerboard segmentation to the original remote sensing image $x \in \mathbb{R}^{H \times W \times 3}$, where $H \times W$ represents the spatial dimensions and 3 denotes the number of channels. This segmentation produces six subimages $x_{c,s} \in \mathbb{R}^{H \times \frac{W}{2} \times 1}$, with both spatial and channel-wise divisions. Here, c stands for the channel index ($c \in R, G, B$) and s refers to the spatial location index ($s \in a, b$). In particular, $x_{R,a}$, $x_{G,a}$, and $x_{B,a}$ are the subimages derived from the main diagonal cuts across the three channels, whereas $x_{R,b}$, $x_{G,b}$, and $x_{B,b}$ are the subimages obtained from the counter-diagonal cuts on the three channels. This segmentation strategy preserves the details and structural integrity of the remote sensing images to the greatest extent, facilitating the compression of the subsequent subimages. Fig. 3 illustrates the process of using checkerboard segmentation to derive subimages from the original three-channel image. For each channel, the checkerboard segmentation can be expressed by the following formula:

$$S(i,j) = \begin{cases} \text{Part A,} & \text{if}(i+j) \bmod 2 = 0 \\ \text{Part B,} & \text{if}(i+j) \bmod 2 = 1. \end{cases} \quad (1)$$

Specifically, for each pixel position (i,j), if the sum of the indices $i + j$ is even, the pixel is assigned to Part A; if the sum is odd, the pixel is assigned to Part B. This segmentation method is independent of the number of image channels, meaning it can be applied to images with any number of channels.

The primary reason we chose a fixed checkerboard segmentation scheme is its ability to effectively preserve the structural and color block information of the original image. As shown in Fig. 2, it is difficult to visually distinguish between the image contents in part A and part B because they exhibit a high degree of similarity in structure and color blocks. This segmentation approach ensures that the overall structure and color block features of the image are well preserved during the compression process, allowing the subsequently designed network to more accurately learn the structural and color block information in the image, leading to a more precise discrete probability distribution. As a fixed scheme, checkerboard segmentation provides a stable and efficient compression method when dealing with high-resolution remote sensing images, helping to reduce processing complexity while ensuring reliable performance.

It is worth noting that this segmentation method differs from the checkerboard context model [30], where pixels at the checkerboard positions are zeroed out to accommodate masked convolution, without altering the size of the processed image. In contrast, our checkerboard segmentation method modifies the image structure, resulting in a change in image size.
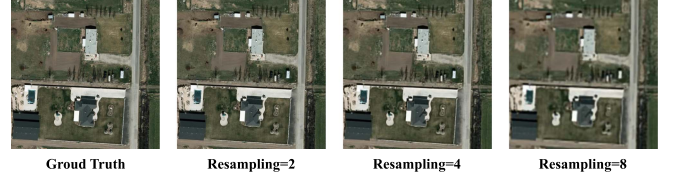
C. JPEG XL Resampling

In the JPEG XL encoder, resampling is a preprocessing technique applied to images during encoding to achieve an optimal balance between compression efficiency and image quality. Specifically, the encoder performs downsampling based on a designated resampling factor during encoding, while the decoder restores the image resolution using the corresponding upsampling algorithm during decoding. This approach effectively reduces data volume while preserving as much visual quality as possible. However, excessive downsampling may result in a loss of image details, especially in images with abundant high-frequency details. Therefore, appropriately setting the resampling parameter is crucial to strike a balance between image quality and compression efficiency. Specific parameter settings are discussed in Section V-D2.

Building on this technique, our method applies resampling to the data in the main diagonal regions, based on a checkerboard segmentation. The lossy-compressed data processed by JPEG XL resampling is saved as prior knowledge, serving as the initial input to the network. Our approach leverages resampling's ability to preserve critical original image information under low-bitrate conditions, providing effective initial features for the network and enhancing the model's ability to capture fine image details. In addtion, Fig. 4 illustrates the image quality under different resampling parameters.

D. Lossy Prior Probability Prediction Network

In the previous work, we obtain six subimages based on the checkerboard segmentation technique, and organize them into six parts. Specifically, the three parts $x_{R,a}$, $x_{G,a}$, and $x_{B,a}$ are subimages corresponding to the main diagonal on the three channels, while the other three parts $x_{R,b}$, $x_{G,b}$, and $x_{B,b}$ represent the subimages along the counter-diagonal for each channel. It is noteworthy that $x_{R,a}$, $x_{G,a}$, and $x_{B,a}$ are first compressed using the JPEG XL resampling encoder to obtain their lossy representations, denoted as $x'_{\text{RGB},a}$. This outcome serves as lossy prior knowledge for subsequent learning. In addition, all six parts require compression using our proposed LP3Net and arithmetic coding. Thus, each of them requires a network to predict their distinct discrete probability, leading to a compression framework consisting of six subnetworks.

Specifically, $x'_{\text{RGB},a}$ serves as prior knowledge and is input into the first LP3Net to compress $x_{R,a}$. Next, $x'_{\text{RGB},a}$ and $x_{R,a}$ are provided as prior knowledge to the second LP3Net to compress $x_{G,a}$. Fig. 5 illustrates the process of replacing part of $x'_{\text{RGB},a}$ with $x_{R,a}$. Similarly, $x'_{\text{RGB},a}$, $x_{R,a}$ and $x_{G,a}$ serve as
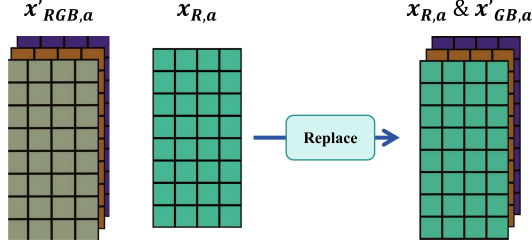
Fig. 5. Graphical illustration of the replacement process where $x_{R,a}$ replaces $x'_{\text{RGB},a}$.

prior knowledge for the third LP3Net to compress $x_{B,a}$. It is essential to note that, starting with the fourth LP3Net, the prior knowledge is replaced by the combination of $x_{R,a}$, $x_{G,a}$, and $x_{B,a}$ (denoted as $x_{\text{RGB},a}$) instead of $x'_{\text{RGB},a}$, as the real data for $x_{R,a}$, $x_{G,a}$, and $x_{B,a}$ is already available, allowing us to discard $x'_{\text{RGB},a}$. Subsequently, $x_{\text{RGB},a}$ serve as prior knowledge for the fourth LP3Net to compress $x_{R,b}$, $x_{\text{RGB},a}$ and $x_{R,b}$ serve as prior knowledge for the fifth LP3Net to compress $x_{G,b}$. Finally, $x_{\text{RGB},a}$, $x_{R,b}$, and $x_{G,b}$ are used to compress $x_{B,b}$ with the sixth LP3Net.

As illustrated in Fig. 6, the proposed LP3Net consists of the image preprocessing module, channel enhancement module, and pixel probability transformer module. Here, we take $x'_{\text{RGB},a}$ and $x_{R,a}$ as input to explain how each module of LP3Net utilizes lossy prior knowledge to perform discrete probability distribution predictions for each pixel.

1) Image Preprocessing Module: In our network architecture, the first module is designated as the image preprocessing module. This module primarily enhances the initial input images while adjusting the number of channel to meet the input requirements of subsequent modules. It incorporates a convolutional layer with a kernel size of 3, a stride of 1, and padding of 1, and is set to output 64 channels. The number of input channels varies depending on the specific networks (i.e., LP3Net). The number of input channels is 3 for the first four networks, 4 for the fifth network, and 5 for the sixth network, while all other parameters remain constant.

This module, by conducting preliminary convolutional processing, not only enhances the feature representation of the images but also lays a foundation for deeper feature extraction and learning processes [31]. Moreover, the design of this module reflects a consideration for the flexibility of network inputs, as it dynamically adjusts the number of input channel. This adjustment allows the model to handle different stages of prior knowledge, thereby optimizing the overall information processing capacity of the network. Such a design not only improves the adaptability of the model but also enhances the efficiency and robustness of the network in utilizing information across various image channels.

2) Channel Enhancement Module: The channel enhancement module, serving as the second core component of our network, is structured around four E-RFDN-L blocks. Each E-RFDN-L block is composed of three specialized submodules: the shallow residual like block (SRLB), the simple convolution

block (SCB), and the enhanced spatial attention block (ESAB). In addition, it includes a convolutional layer with a kernel size of 1, stride of 1, padding of 0, input channels of 128, and output channels of 64. The specific structure is illustrated in Fig. 7. These submodules, initially proposed by Liu [32] for image super-resolution, employ minimal parameters to maximize their efficiency in feature enhancement. After processing through the image preprocessing module, we continuously apply four E-RFDN-L blocks to enhance the features, followed by a concatenation operation to merge the outputs of these four blocks. Subsequently, we design a convolution with an input channel of 256, output channel of 512, kernel size of 1, stride of 1, and no padding, followed by an lrelu activation function to elevate the dimensionality to 512. In addition, we implement another convolution with both input and output channels of 512, kernel size of 3, stride of 1, and padding of 1 to further enhance the up-scaled results. For the final output, a convolution with an input channel of 64, output channel of 512, kernel size of 1, stride of 1, and no padding, is designed to process the results post the image preprocessing module. These results are then added together to further augment the capability of discrete probability distribution prediction for each pixel. Here is a detailed introduction to these submodules.

SRLB: Inspired by the shallow residual block (SRB) [32], we design a similar structure to meet the specific needs of our application. In the design of SRLB, we implement a convolutional layer with a kernel size of 3, a stride of 1, and padding of 1. Both the input and output channels are set to 64, and the layer is complemented by a parametric ReLU (PReLU) activation function [33] to enhance nonlinearity. In addition, this submodule includes a residual connection to further enhance the network's learning capacity. By enabling the direct transmission of input information to the output, the residual connection helps mitigate the issue of gradient vanishing during deep network training, while also improving the stability of the learned feature representations. Similar to SRB, this design enables our model to handle complex feature learning tasks more effectively, significantly improving both its performance and generalization capabilities.

SCB: The SCB is an efficient and compact network component. This submodule consists of a convolutional layer with a kernel size of either 1 or 3, a stride of 1, and padding adjusted according to the kernel size: no padding for a kernel size of 1, and padding of 1 for a kernel size of 3, as illustrated in Fig. 7. The input and output channels are set to 64 and 32, respectively. Following the convolutional layer, a PReLU activation function is integrated to introduce the necessary nonlinear processing capabilities. Unlike SRLB, SCB simplifies its architecture by not introducing a residual connection. The primary function of SCB is to extract key features from the input data, thereby capturing the essential parts of the enhanced features and laying the groundwork for subsequent processing steps.

ESAB: The ESAB is intricately designed, as shown in Fig. 6. This submodule comprises multiple convolutions, a variety of activation functions, pooling operations, and bilinear interpolation. Its primary purpose is to enhance the spatial distribution of residual features through a spatial attention mechanism,
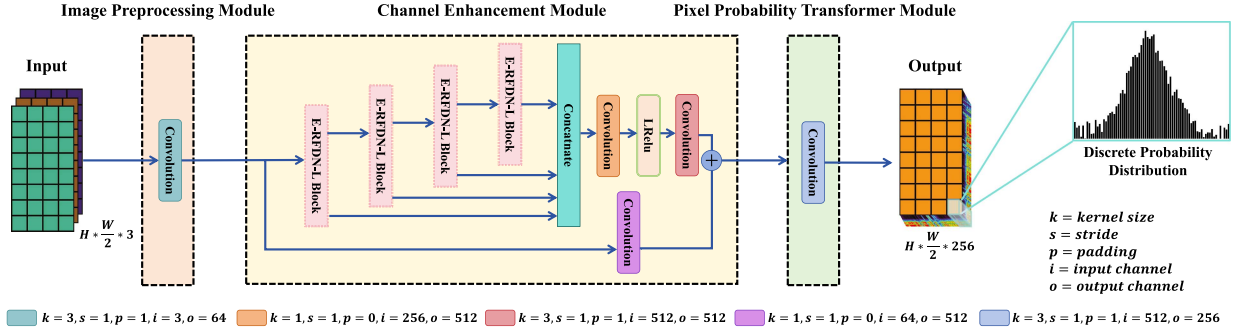
Fig. 6.   Detailed architecture of the proposed LP3Net, which consists of the image preprocessing module, channel enhancement module, and pixel probability transformer module.
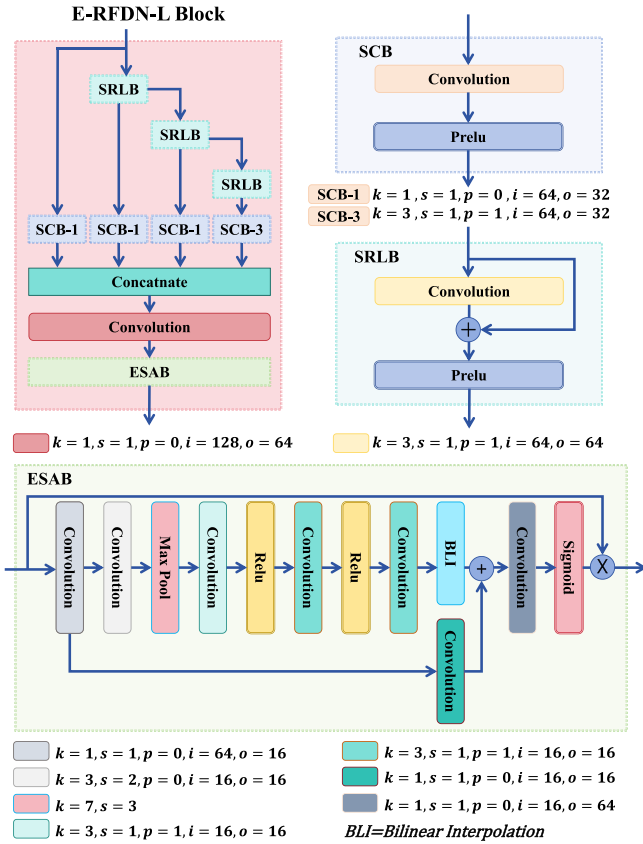

Fig. 7.   Structure of the E-RFDN-L block in the channel enhancement module, which consists of the SRLB, SCB, a convolution layer, and ESAB.

enabling the network to focus more precisely on the key spatial content of the image. In addition, ESAB utilizes strided convolutions and large window max-pooling to expand its receptive field, while employing a convolution layer with a kernel size of 1 at the onset of the module to reduce channel dimensions, thereby maintaining the module's lightweight design. This architecture not only enhances the efficacy of the attention mechanism but also significantly improves the performance of discrete probability distribution predictions.

3) Pixel Probability Transformer Module: The pixel probability transformer module is the final component of our network.

This module employs a simple convolutional layer to predict the discrete probability distribution for each pixel. Utilizing a kernel size of 3, a stride of 1, and padding of 1, the convolutional layer processes input data with 512 channels obtained by the channel enhancement module to get output data with 256 channels. After processing through this module, the dimensions of data change from $H \times \frac{W}{2} \times 512$ to $H \times \frac{W}{2} \times 256$. Subsequently, we apply a softmax function to transform the output of this module. The integration of the softmax function allows for a seamless mapping of the convolutional results to the discrete probability distribution, thereby achieving our ultimate objective.

4) Loss Function: To train our proposed LP3Net, we utilize the bits per subpixel (bpsp) loss [17] to optimize the efficiency of lossless compression for remote sensing images to the greatest extent. The formulation of this loss function is defined as follows:

$$\mathcal{L}_{\text{bpsp}} = -\frac{1}{N} \sum_{i=1}^{N} \log_2 \left(p\left(x_i | y\right) + \epsilon\right) \tag{2}$$

where y represents the input to the network, and it is important to note that each subnetwork's input varies during the compression process. Detailed implementation specifics are described in Section III-D. $p(x_i | y)$ represents the conditional discrete probability distribution of the ith pixel x_i given input y, calculated using a softmax function as discussed in Section III-D3. The constant ϵ is used to ensure numerical stability by preventing the logarithm from approaching zero. N denotes the total number of pixels in the remote sensing image. As shown in (2), the loss function computes the negative average of all pixels' predicted probabilities, using logarithmic transformations with base 2.

E. Arithmetic Coding

After obtaining the discrete probability distribution, it becomes feasible to apply arithmetic coding to the data from a designated channel of the subimages, subsequently storing the encoded bitstream on the device. The specific steps of the arithmetic coding process are outlined as follows:

$$L_0 = 0, \quad H_0 = 1 \tag{3}$$

$$c(x_i) = p(0) + p(1) + \cdots + p(s_{x_i} - 1) \tag{4}$$

$$L_i = L_{i-1} + (H_{i-1} - L_{i-1}) \cdot c(x_i) \tag{5}$$

$$H_i = L_{i-1} + (H_{i-1} - L_{i-1}) \cdot (c(x_i) + p(s_{x_i})). \tag{6}$$

Here, L and H are the lower and upper bounds of the interval, respectively. The initial bounds of the interval are set as $L_0 = 0$ and $H_0 = 1$. $p(1), p(2), \ldots, p(s_{x_i} - 1)$ represent the probabilities associated with pixel values ranging from 0 to $s_{x_i} - 1$. $c(x_i)$ calculates the cumulative probability for the ith encoding pixel x_i, which has the actual pixel value s_{x_i}. For pixel x_i, L_i and H_i are the lower and upper bounds of the interval, respectively. The compressed image is ultimately represented by any number within the final interval $[L_n, H_n)$ converted into a binary bitstream. More detailed explanations and implementations of this encoding process can be found in the referenced work [34].

It should be noted that the encoding process can be parallelized, as all original data is available during this phase. In parallel encoding, only one JPEG XL encoding and one network inference is required, along with six arithmetic encodings. In contrast, decoding requires sequential computation, relying on partial data from previous subimages for predictive steps. In serial decoding, one JPEG XL decoding, six network inferences, and six arithmetic decodings must be performed recursively.

IV. DISCUSSIONS

We have innovatively proposed a lossless compression framework for high-resolution remote sensing images. First, we obtain lossy prior knowledge through checkerboard segmentation and JPEG XL resampling method. Then, we utilize the proposed LP3Net to predict the discrete probability distribution of the entire image for arithmetic encoding, yielding the final compression result. Our framework achieves state-of-the-art performance in lossless compression of high-resolution remote sensing images. However, its current limitation is the inability to handle multispectral remote sensing images, which we plan to address in future work to further extend its applicability.

V. EXPERIMENTS

A. Experimental Setup

1) Datasets: We evaluate the proposed framework on three high-resolution remote sensing image datasets, with their details provided as follows. Fig. 8 illustrates some representative images from these datasets.

1) *AID:* This dataset comprises a diverse collection of aerial scenes, including categories such as church, meadow, farmland, desert, industrial, forest, center, brideg, and commercial. It contains 10 000 images, each with a resolution of 600×600, distributed across 30 distinct categories.
2) *LoveDA:* This dataset was gathered from the cities of Wuhan, Changzhou, and Nanjing, utilizing imagery obtained from Google Earth. It contains a total of 5987 images, all having a consistent resolution of 1024×1024.
3) *VEDAI:* This dataset is used for vehicle detection in aerial imagery, depicting vehicles with various orientations, lighting conditions, and occlusions. It includes 1210 images, each with a resolution of 1024×1024.

Fig. 8. Representative images from three remote sensing image datasets. A: AID. B: LoveDA. C: VEDAI.

2) Compared Methods: To demonstrate the effectiveness of the proposed method, we employ both traditional learning methods and advanced deep learning techniques for comparison. The traditional lossless methods include PNG [35], JPEG 2000 [36], WebP [37], WebP2 [38], AVIF [39], JPEG XL [40], and decomposed soft compression (DSC) [41], while the advanced deep learning methods contain L3C [34] and LC-FDNet [28].

3) Setting Details: To train our network, we randomly extract patches of size 128×128 from the input images. The network is trained using the Adam optimizer over 1500 epochs, with a batch size of 16. The initial learning rate is set to $1e^{-3}$ and decreases by a factor of 0.1 every 500 epochs. Since not all remote sensing images can be evenly divided by 2, padding is applied to images with dimensions $H \times W$ (where $H\%2 = 1$ and $W\%2 = 1$) by adding $(W \times 1, H \times 1)$ pixels during the experiments. To ensure the optimal performance for the compared methods, we obtained their code from online sources or directly from the authors and conducted experiments using the parameter settings specified in their original papers.

To evaluate the compression performance, we use bpsp as the primary metric. For lossless compression, bpp is three times the value of bpsp for such images, making the two metrics equivalent in this context. Thus, we focus solely on bpsp for our evaluation. A lower bpsp value indicates better compression performance. All experiments are conducted using PyTorch on a system with an NVIDIA A100 80 G GPU and 256 GB of RAM.

B. Compression Results

Before conducting comparisons, we calculate the mean squared error (MSE) between the original and decompressed images. The experimental results consistently show that the MSE values are zero, indicating that the decompressed images are identical to the original ones, thereby confirming lossless compression.

Table I presents the bpsp results for various lossless compression methods. The optimal results are highlighted in bold. Our method consistently outperforms both traditional nonlearning

TABLE I
QUANTITATIVE COMPARISON OF VARIOUS LOSSLESS COMPRESSION METHODS BASED ON bpsp ON THREE REMOTE SENSING IMAGE DATASETS

Methods	AID		LoveDA		VEDAI	
PNG	4.44085	+78.03%	4.19277	+41.41%	4.41051	+77.57%
JPEG 2000	3.16092	+26.72%	3.40195	+14.74%	3.41666	+37.55%
WebP	3.03563	+21.70%	3.42915	+15.65%	3.57016	+43.73%
WebP2	2.82425	+13.22%	3.20888	+8.22%	3.28917	+32.42%
AVIF	3.61490	+44.92%	3.87690	+30.75%	3.73431	+50.34%
JEPG XL	2.60831	+4.57%	3.13353	+5.68%	3.19338	+28.57%
DSC	2.99172	+19.94%	3.49048	+17.72%	3.47648	+39.96%
L3C	3.15727	+26.57%	3.46960	+17.02%	3.45751	+39.20%
LC-FDNet	2.54070	+1.86%	3.03223	+2.27%	2.79033	+12.34%
Ours	**2.49440**		**2.96503**		**2.48386**	

Fig. 9. Efficient compression representation of a remote sensing image from the VEDAI dataset, demonstrating significant storage reduction using our proposed method.

TABLE II
ANALYSIS OF PROCESSING SPEEDS FOR VARIOUS COMPRESSION METHODS ACROSS THREE REMOTE SENSING IMAGE DATASETS, QUANTIFIED IN SECONDS

Time	AID	LoveDA	VEDAI
PNG	/	/	/
JPEG 2000	0.3825	0.9416	0.9522
WebP	0.1456	0.3666	0.3754
WebP2	0.2061	0.5149	0.5057
AVIF	0.2371	0.6407	0.6302
JPEG XL	0.1964	0.4301	0.4502
DSC	33.7481	99.5492	97.9812
L3C	0.4649	1.2690	1.2780
LC-FDNet	3.3134	6.8462	6.7322
Ours	1.7576	4.7362	4.7626

and advanced learning approaches across remote sensing image datasets, reducing storage space by at least 1.86% in all scenarios. Specifically, it surpasses the best-performing traditional method, JPEG XL, as well as the leading deep learning method, LC-FDNet. These results demonstrate the effectiveness and adaptability of our method for different types of remote sensing image data. For the AID dataset, our method surpasses JPEG XL by 4.57% and LC-FDNet by 1.86%. On the LoveDA dataset, it outperforms JPEG XL by 5.68% and LC-FDNet by 2.27%. Similarly, for the VEDAI dataset, our method exceeds JPEG XL by 28.57% and LC-FDNet by 12.34%. These results emphasize the robustness and high efficiency of our method across various high-resolution remote sensing image datasets.

To validate the effectiveness of our proposed method, we randomly select an image from the VEDAI dataset as an illustrative example. As shown in Fig. 9, the files outfile.0, outfile.1, and outfile.2 represent the bitstream data for $x_{R,a}$, $x_{G,a}$, and $x_{B,a}$, while the files outfile.3, outfile.4, and outfile.5 represent the bitstream data for $x_{R,b}$, $x_{G,b}$, and $x_{B,b}$. The file named pad contains the number of edge padding pixels, and test_img.jxl represents $x'_{\mathrm{RGB},a}$. Our results demonstrate that in the storage system, our compression scheme requires only 995 KB to store this example image, compared to the original PNG format requiring 1734 KB for storage. This indicates that our method needs only 53.4% of the storage space to effectively preserve all information.

C. Inference Time

Our framework not only excels in compression performance but also demonstrates rapid inference speeds, as shown in Table II. When compared with cutting-edge deep learning-based compression techniques, our method achieves faster inference times than LC-FDNet, although it is slightly outpaced by L3C. In contrast to traditional codecs, our approach is somewhat slower than most methods except DSC. It is evident that the inference speed of our framework remains within a practical range for real-world applications, even when addressing large-scale challenges. Specifically, compressing a 1024×1024 image takes less than 5 s, affirming the scalability and practicality of our framework. Table II illustrates that as the image resolution increases, the inference speed decreases, emphasizing the influence of image size on compression time. Our framework achieves a good balance between high compression efficiency and fast inference speed, making it particularly well-suited for demanding applications, such as real-time remote sensing data analysis and large-scale environmental monitoring.

Overall, our framework demonstrates exceptional efficiency in processing diverse remote sensing datasets across multiple environments, including urban, agricultural, and forested landscapes. It excels in compressing high-resolution satellite imagery and aerial photographs, ensuring that the integrity and quality of the images are meticulously preserved. This capability is particularly advantageous for extensive remote sensing applications, as it significantly enhances data manageability, reduces associated storage and transmission costs, and improves the overall flexibility and effectiveness of remote sensing image analysis. By adeptly handling various image types and complexities, our framework provides robust solutions that cater

TABLE III
BPSP PERFORMANCE WITH DIFFERENT PRIOR KNOWLEDGE ON REMOTE
SENSING IMAGE DATASETS

Settings		AID	LoveDA	VEDAI
Prior	Prediction			
Part A	Part B	2.49440	2.96503	2.48386
Part B	Part A	2.50242	2.97400	2.48495

TABLE IV
IMPACT OF RESAMPLING SETTINGS ON COMPRESSION EFFICIENCY AND SSIM
OF THE PRIOR KNOWLEDGE

Resampling	AID		LoveDA		VEDAI	
	BPSP	SSIM	BPSP	SSIM	BPSP	SSIM
2	0.2434	0.8892	0.1997	0.8656	0.2090	0.8864
4	0.0603	0.6577	0.0464	0.6819	0.0442	0.6651
8	0.0193	0.5113	0.0137	0.5829	0.0128	0.5546

TABLE V
BPSP PERFORMANCE OF OUR LP3NET WITH VARYING RESAMPLING SETTINGS
ON REMOTE SENSING IMAGE DATASETS

Resampling	AID		LoveDA		VEDAI	
8	2.97910	+19.43%	3.20095	+7.96%	2.70628	+8.95%
4	2.64571	+6.07%	3.06140	+3.25%	2.60305	+4.80%
2	2.49440		2.96503		2.48386	

TABLE VI
BPSP PERFORMANCE OF OUR LP3NET WITH VARYING BATCH SIZE ON
REMOTE SENSING IMAGE DATASETS

#Batch	AID	LoveDA	VEDAI
32	2.49440	2.96503	2.48386
16	2.51607	2.96579	2.50880
8	2.50767	2.96528	2.50636

to the dynamic demands of remote sensing technologies and applications.

D. Ablation Studies

1) Selection of Prior and Prediction Area: In our experimental setup, we explore the use of lossy subimages from the main diagonal as prior knowledge to assist in predicting the discrete probability distributions of all subimages. This strategy capitalizes on the information from the main diagonal, which provides critical insights into the overall structure and content of the image, thereby aiding the proposed LP3Net in more accurately estimating the discrete probability distributions of each subimage. Similarly, we investigate the use of lossy subimages from the counter-diagonal as prior information for predicting the discrete probability distributions of all subimages. This complementary strategy of data utilization aims to fully exploit the spatial correlations within the image to enhance the overall performance of the compression framework.

To thoroughly evaluate the influence of these two lossy diagonal-based prior information strategies on compression performance, we carry out experiments using the optimal parameter settings established earlier. Table III presents a summary of the compression results, comparing the use of lossy main diagonal and lossy counter-diagonal data as prior information across various datasets. As indicated in Table III, using lossy main diagonal data shows a slight performance advantage over using counter-diagonal data. Given the symmetry of our approach, it is apparent that either type of data as prior knowledge yields excellent results. Thus, we can infer that our framework effectively utilizes the inherent spatial relationships within the image to optimize compression outcomes, demonstrating its potential for efficient compression in remote sensing imagery scenarios. This indicates that our image compression framework intelligently leverages the symmetric properties of image data, achieving efficient compression of remote sensing images.

2) Effect of JPEG XL Resampling Setting: In Section III-C, we mention that suitable JPEG XL resampling parameters are crucial for compression rates. In JPEG XL resampling, there are three parameter options: 2×2, 4×4, and 8×8. To thoroughly understand the impact of different resampling settings on image compression effectiveness, we design and conduct a series of experiments to evaluate the performance of these three resampling configurations across various datasets. In these experiments, all models maintained consistency in architecture and other training parameters, with only the resampling size varying to ensure comparability of results. Each model undergoes a standardized fine-tuning process to control variables and ensures the validity of the experimental outcomes.

As shown in Table IV, with an increase in the resampling level, SSIM significantly decreases, reflecting a more pronounced loss of image structure and detail. However, the reduction in prior knowledge's BPSP is not as substantial. The sharp decline in SSIM, indicating the loss of structure and detail, negatively impacts the performance of LP3Net. As indicated in Table V, it is evident that using 2×2 resampling setting achieves optimal performance. Compared to 4×4 resampling setting, it increases the compression rate by at least 3.25%. Fo the AID dataset, it improves the compression rate by 6.07%. Relative to 8×8 resampling setting, the minimum increase in compression rate is 7.96%, with a maximum improvement of 19.43%. Overall, 2×2 resampling setting demonstrates exceptionally superior compression efficiency in JPEG XL. Despite consuming more storage space for prior knowledge compared to other subsampling options, it significantly enhances the performance of network parts, achieving notable compression performance advantages across various datasets. This superior performance makes 2×2 resampling as an ideal choice for training LP3Net.

3) Effect of Batch Size: Due to the varying computational resources available on different training devices, it is essential to evaluate the impact of different batch sizes on network performance. To comprehensively assess the influence of batch size on network training and its performance, we design a series of experiments. In these experiments, all models are consistent in architecture and other training parameters, with only the batch size varying to ensure comparability of the results. Each model undergoes a standardized fine-tuning process to control variables and ensure the validity of the experimental outcomes.

We compare network performance across different datasets under various batch size configurations to understand the actual impact. The experimental results are summarized in Table VI. The results indicate that smaller batch sizes do not significantly

impact model performance, and their performance discrepancies are minimal and can be considered as noise. This demonstrates that, although adjustments in batch size do influence training and performance to some extent, our network exhibits considerable robustness to changes in this parameter. In other words, variations in batch size do not significantly affect network performance, showcasing our network's good adaptability and stability under different data conditions. This robustness is crucial for practical applications, ensuring that the network maintains excellent performance under diverse operational conditions.

VI. Conclusion

In this article, we proposed an LP3Net and integrated it into our lossless compression framework for high-resolution remote sensing images. The proposed compression framework initially segments original remote sensing images using a checkerboard segmentation technique. The main diagonal subimages undergo lossy compression using JPEG XL resampling, and these compressed data serves as prior knowledge for progressively compressing each subimage. The LP3Net architecture consists of an image preprocessing module, a channel enhancement module, and a pixel probability transformer module. The discrete probability distributions obtained through LP3Net are used to encode the images using arithmetic coding. This learning scheme allows LP3Net to leverage the prior compressed data effectively, enhancing the accuracy of its predictions and thereby improving the overall lossless compression performance across various high-resolution remote sensing datasets. Our experimental results confirm that LP3Net achieves excellent compression ratios.

Future research could explore several directions to enhance the current framework. One direction is extending its support to multispectral remote sensing images by leveraging interband correlations. Another is enabling classification and segmentation tasks directly in the compressed domain to improve efficiency. These advancements would further enhance the framework's applicability to real-time analysis in resource-constrained environments.

References

[1] Y. Zhang, S. Yan, L. Zhang, and B. Du, "Fast projected fuzzy clustering with anchor guidance for multimodal remote sensing imagery," *IEEE Trans. Image Process.*, vol. 33, pp. 4640–4653, Aug. 2024.

[2] Y. Zhang, G. Jiang, Z. Cai, and Y. Zhou, "Bipartite graph-based projected clustering with local region guidance for hyperspectral imagery," *IEEE Trans. Multimedia*, vol. 26, pp. 9551–9563, Apr. 2024.

[3] P. Duan, S. Hu, X. Kang, and S. Li, "Shadow removal of hyperspectral remote sensing images with multiexposure fusion," *IEEE Trans. Geosci. Remote Sens.*, vol. 60, Sep. 2022, Art. no. 5537211.

[4] P. Duan, P. Ghamisi, X. Kang, B. Rasti, S. Li, and R. Gloaguen, "Fusion of dual spatial information for hyperspectral image classification," *IEEE Trans. Geosci. Remote Sens.*, vol. 59, no. 9, pp. 7726–7738, Sep. 2021.

[5] Y. Zhang, S. Yan, X. Jiang, L. Zhang, Z. Cai, and J. Li, "Dual graph learning affinity propagation for multimodal remote sensing image clustering," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, Jun. 2024, Art. no. 5521713.

[6] Z. Duan, M. Lu, J. Ma, Y. Huang, Z. Ma, and F. Zhu, "QARV: Quantization-aware resnet VAE for lossy image compression," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 1, pp. 436–450, Jan. 2024.

[7] R. Yang and S. Mandt, "Lossy image compression with conditional diffusion models," *Adv. Neural Inf. Process. Syst.*, vol. 36, pp. 1–25, 2024.

[8] Y. Bai, X. Liu, K. Wang, X. Ji, X. Wu, and W. Gao, "Deep lossy plus residual coding for lossless and near-lossless image compression," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 46, no. 5, pp. 3577–3594, May 2024.

[9] R. Kumari and S. Sriramulu, "Lossless image compression using k-means clustering in color pixel domain," in *Proc. IEEE Int. Conf. Comput., Power Commun. Technol.*, 2024, vol. 5, pp. 1925–1933.

[10] R. Wang, J. Liu, H. Sun, and J. Katto, "Learned lossless image compression with combined channel-conditioning models and autoregressive modules," *IEEE Access*, vol. 11, pp. 73462–73469, 2023.

[11] F. Kamisli, "Learned lossless image compression through interpolation with low complexity," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 33, no. 12, pp. 7832–7841, Dec. 2023.

[12] C. Fu and B. Du, "Remote sensing image compression based on the multiple prior information," *Remote Sens.*, vol. 15, no. 8, 2023, Art. no. 2211.

[13] S. M. i. Verdú, M. Chabert, T. Oberlin, and J. Serra-Sagristà, "Fixed-quality compression of remote sensing images with neural networks," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 17, pp. 12169–12180, Jul. 2024.

[14] V. Abramova, V. Lukin, S. Abramov, S. Kryvenko, P. Lech, and K. Okarma, "A fast and accurate prediction of distortions in DCT-based lossy image compression," *Electronics*, vol. 12, no. 11, 2023, Art. no. 2347.

[15] W. Ye, W. Lei, W. Zhang, T. Yu, and X. Feng, "GFScompnet: Remote sensing image compression network based on global feature-assisted segmentation," *Multimedia Tools Appl.*, vol. 83, pp. 67103–67127, 2024.

[16] D. Wang and Y. Zhang, "A lossless compression of remote sensing images based on ANS entropy coding algorithm," *Proc. SPIE*, vol. 13088, pp. 17–24, 2024.

[17] X. Feng, E. Gu, Y. Zhang, and A. Li, "Probability prediction network with checkerboard prior for lossless remote sensing image compression," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 17, pp. 17971–17982, Sep. 2024.

[18] S. Kahali, S. Dey, C. Kadway, A. Mukherjee, A. Pal, and M. Suri, "Low-power lossless image compression on small satellite edge using spiking neural network," in *Proc. Int. Joint Conf. Neural Netw.*, 2023, pp. 1–8.

[19] D. Anuradha, G. C. Sekhar, A. Mishra, P. Thapar, Y. A. Baker El-Ebiary, and M. Syamala, "Efficient compression for remote sensing: Multispectral transform and deep recurrent neural networks for lossless hyper-spectral imagine," *Int. J. Adv. Comput. Sci. Appl.*, vol. 15, no. 2, 2024, pp. 531–541.

[20] J. Zhang et al., "Spatial-temporal context model for remote sensing imagery compression," in *Proc. 32nd ACM Int. Conf. Multimedia*, 2024, pp. 6063–6072.

[21] C. Shi, K. Shi, F. Zhu, Z. Zeng, M. Ding, and Z. Jin, "An enhanced global feature-guided network based on multiple filtering noise reduction for remote sensing image compression," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, Nov. 2024, Art. no. 5646220.

[22] W. Zhang, D. Li, H. Zhang, P. Yu, and W. Gao, "Lightweight bit-depth recovery network for gaofen satellite multispectral image compression," *IEEE J. Sel. Topics Appl. Earth Observ. Remote Sens.*, vol. 17, pp. 18635–18651, Oct. 2024.

[23] X. Ma, "High-resolution image compression algorithms in remote sensing imaging," *Displays*, vol. 79, 2023, Art. no. 102462.

[24] V. Joshi and J. S. Rani, "A simple lossless algorithm for on-board satellite hyperspectral data compression," *IEEE Geosci. Remote Sens. Lett.*, vol. 20, May 2023, Art. no. 5504305.

[25] X. Gao, L. Wang, T. Li, and J. Xie, "A method of reordering lossless compression of hyperspectral images," *ISPRS Ann. Photogrammetry, Remote Sens. Spatial Inf. Sci.*, vol. 10, pp. 821–826, 2023.

[26] D. Valsesia, T. Bianchi, and E. Magli, "Onboard deep lossless and near-lossless predictive coding of hyperspectral images with line-based attention," *IEEE Trans. Geosci. Remote Sens.*, vol. 62, pp. 1–14, 2024, *arXiv:2403.17677*.

[27] F. Zhu and H. Hu, "Lossless compression for hyperspectral images using cascaded prediction," in *Proc. 8th Int. Conf. Commun., Image Signal Process.*, 2023, pp. 265–269.

[28] H. Rhee, Y. I. Jang, S. Kim, and N. I. Cho, "Lc-fdnet: Learned lossless image compression with frequency decomposition network," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 6033–6042.

[29] S.-C. Pei and J.-J. Ding, "Improved reversible integer-to-integer color transforms," in *Proc. IEEE Int. Conf. Image Process.*, 2009, pp. 473–476.

[30] D. He, Y. Zheng, B. Sun, Y. Wang, and H. Qin, "Checkerboard context model for efficient learned image compression," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 14771–14780.

[31] Q. Wang, J. Cheng, Q. Gao, G. Zhao, and L. Jiao, "Deep multi-view subspace clustering with unified and discriminative learning," *IEEE Trans. Multimedia*, vol. 23, pp. 3483–3493, Sep. 2020.

[32] J. Liu, J. Tang, and G. Wu, "Residual feature distillation network for lightweight image super-resolution," in *Proc. Computer Vis.–ECCV 2020 Workshops*, 2020, pp. 41–55.

[33] K. He, X. Zhang, S. Ren, and J. Sun, "Delving deep into rectifiers: Surpassing human-level performance on imagenet classification," in *Proc. IEEE Int. Conf. Computer Vis.*, 2015, pp. 1026–1034.

[34] F. Mentzer, E. Agustsson, M. Tschannen, R. Timofte, and L. V. Gool, "Practical full resolution learned lossless image compression," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2019, pp. 10629–10638.

[35] T. Boutell, "PNG (portable network graphics) specification version 1.0," Tech. Rep., 1997.

[36] Z. Si and K. Shen, "Research on the webp image format," in *Proc. Adv. Graph. Commun. Packag. Technol. Mater.*, 2016, pp. 271–277.

[37] G. Ginesu, M. Pintus, and D. D. Giusto, "Objective assessment of the webp image coding algorithm," *Signal Process. Image Commun.*, vol. 27, no. 8, pp. 867–874, 2012.

[38] D. Barina, "Comparison of lossless image formats," *Comput. Sci. Res. Notes - CSRN*, pp. 339–342, May 2021. [Online]. Available: http://wscg. zcu.cz/WSCG2021/Poster/J02.pdf

[39] N. Barman and M. G. Martini, "An evaluation of the next-generation image coding standard AVIF," in *Proc. Int. Conf. Qual. Multimedia Exp.*, 2020, pp. 1–4.

[40] J. Alakuijala et al., "JPEG XL next-generation image compression architecture and coding tools," in *Proc. Adv. Digit. Image Process. LII*, 2019, vol. 11137, pp. 112–124.

[41] G. Liu, W. Li, and F. Duan, "Decomposed Soft Compression for Remote Sensing Image," in *Proc. IEEE Int. Conf. Robot. Biomimetics*, 2022, pp. 650–655.

Yongshan Zhang (Member, IEEE) received the B.Eng. and Ph.D. degrees in computer science from the China University of Geosciences, Wuhan, China, in 2014 and 2019, respectively.

She has been a member of the BDSC Laboratory, University of Illinois at Chicago, Chicago, IL, USA, and the VIP Lab, University of Macau, Macau, China. She is currently an Associate Professor with the School of Computer Science, China University of Geosciences. Her current research interests include machine learning and remote sensing image analysis.

Xinxin Wang is currently working toward the Ph.D. degree in computer science with the Department of Computer and Information Science, University of Macau, Macau, China.

His research interests include multiview learning and hyperspectral image processing.

Enjia Gu is currently working toward the M.S. degree in computer science with the School of Computer Science, China University of Geosciences, Wuhan, China.

His research interests include deep learning and lossless remote sensing image compression.

Xinwei Jiang received the Ph.D. degree in computer science from the Huazhong University of Science and Technology, Wuhan, China, in 2012.

He is currently an Associate Professor with the China University of Geosciences, Wuhan. His research interests include nonparametric statistical models, dimensionality reduction, and hyperspectral image processing.