frontiers in Plant Science

ORIGINAL RESEARCH
published: 28 February 2022
doi: 10.3389/fpls.2022.849606

Green Visual Sensor of Plant: An Energy-Efficient Compressive Video Sensing in the Internet of Things

Ran Li[1]*, *Yihao Yang*[1] *and Fengyuan Sun*[2]

[1] School of Computer and Information Technology, Xinyang Normal University, Xinyang, China, [2] Guangxi Key Laboratory of Wireless Wideband Communication and Signal Processing, Guilin University of Electronic Technology, Guilin, China

Internet of Things (IoT) realizes the real-time video monitoring of plant propagation or growth in the wild. However, the monitoring time is seriously limited by the battery capacity of the visual sensor, which poses a challenge to the long-working plant monitoring. Video coding is the most consuming component in a visual sensor, it is important to design an energy-efficient video codec in order to extend the time of monitoring plants. This article presents an energy-efficient Compressive Video Sensing (CVS) system to make the visual sensor green. We fuse a context-based allocation into CVS to improve the reconstruction quality with fewer computations. Especially, considering the practicality of CVS, we extract the contexts of video frames from compressive measurements but not from original pixels. Adapting to these contexts, more measurements are allocated to capture the complex structures but fewer to the simple structures. This adaptive allocation enables the low-complexity recovery algorithm to produce high-quality reconstructed video sequences. Experimental results show that by deploying the proposed context-based CVS system on the visual sensor, the rate-distortion performance is significantly improved when comparing it with some state-of-the-art methods, and the computational complexity is also reduced, resulting in a low energy consumption.

Keywords: Internet of Things, visual sensor, Compressive Video Sensing, context extraction, linear recovery, plant monitoring

OPEN ACCESS

Edited by:
Yu Xue,
Nanjing University of Information Science and Technology, China

Reviewed by:
Romany Mansour,
The New Valley University, Egypt
Zijian Qiao,
Ningbo University, China
Khan Muhammad,
Sejong University, South Korea

***Correspondence:**
Ran Li
liran@xynu.edu.cn

Specialty section:
This article was submitted to Sustainable and Intelligent Phytoprotection, a section of the journal Frontiers in Plant Science

Received: *06 January 2022*
Accepted: *24 January 2022*
Published: *28 February 2022*

Citation:
Li R, Yang Y and Sun F (2022) Green Visual Sensor of Plant: An Energy-Efficient Compressive Video Sensing in the Internet of Things. Front. Plant Sci. 13:849606. doi: 10.3389/fpls.2022.849606

1. INTRODUCTION

In the Internet of Things (IoT), the plant propagation process or plant growth can be monitored by visual sensors. One benefit from the framework of IoT, a large amount of data on the plant can be gathered in a central server, and the valuable information can be achieved by analyzing the data in real-time. However, with the limited processing capabilities and power/energy budget of visual sensors, it is a challenge for video monitoring of plant to compress large-scale video sequences by using the traditional codec, e.g., H.264/AVC and HEVC (Sullivan et al., 2012), so the existing works have developed low-complexity and energy-efficient video codecs, in which Distributed Video Coding (DVC) (Girod et al., 2005) and Compressive Video Sensing (CVS) (Baraniuk et al., 2017) have attracted more attention in industry and academia. Different from DVC, CVS dispenses with the feedback and virtual channels (Unde and Pattathil, 2020), which makes the codec framework simpler. Meanwhile, CVS provides a low-complexity encoder because of its theoretic foundation, Compressive Sensing (CS) (Baraniuk, 2007), realizes the capture of

video frames at a rate significantly below the Nyquist rate. Currently, many researchers recognize that CVS is a potential scheme to compress the video sequences in the IoT framework, and especially for wireless video monitoring of plants, the CVS scheme can assist visual sensors to efficiently reduce the energy consumptions, however, its rate-distortion performances are still far from satisfactory.

The objective of this article is to improve the rate-distortion performance of CVS, providing high-quality video monitoring of plants with low energy consumption. To achieve this objective, the existing works focus on how to design excellent recovery algorithms, and they are keen on mixing various advanced tools into the CVS framework, e.g., the latest popular Deep Neural Network (DNN) (Palangi et al., 2016; Zhao et al., 2020; Tran et al., 2021). Though effective, they bear a heavy computational burden. Different from these works, we try to exploit the capability of CS to capture important structures, improving the reconstruction quality only armed with some simple recovery algorithms. It is well known that the context feature (Shechtman and Irani, 2007; Romano and Elad, 2016) is a good structure for visual quality, and, therefore, in this article, we focus on how to fuse contexts into CVS for an obvious improvement of reconstruction quality.

Compressive Video Sensing consists of three essential steps including CS measurement, measurements quantization, and reconstruction. CS measurement is a process of randomly sampling each video frame, in which the block-based (Gan, 2007; Bigot et al., 2016) or structurally (Do et al., 2012; Zhang et al., 2015) random matrix is often used to ensure the small memory requirement. Output by CS measurement, all measurements are required to be quantized as bits, then transmitted to the decoder. The straightforward solution to incorporating quantization into CVS is simply to apply Scalar Quantization (SQ), but it brings a big error. For block-based sampling, Differential Pulse Code Modulation (DPCM) (Mun and Fowler, 2012) can be used, and it exploits the correlations between blocks to improve the rate-distortion performance. Based on DPCM, many works also proposed some efficient predictive schemes (Zhang et al., 2013; Gao et al., 2015) to quantize CS measurements. Reconstruction is deployed at the decoder, and it uses quantized measurements to reconstruct the video sequence by the CS recovery algorithm. At present, the reconstruction can be implemented by one of the three types: frame-by-frame (Chen Y. et al., 2020; Trevisi et al., 2020), three-dimensional (3D) (Qiu et al., 2015; Tachella et al., 2020), and distributed strategies (Zhang et al., 2020; Zhen et al., 2020). The frame-by-frame reconstruction performs a CS recovery algorithm to reconstruct each video frame independently, and it has a poor rate-distortion performance due to neglecting the correlations between frames. The 3D reconstruction designs some complex representation models to once reconstruct a whole video sequence or a Group Of Pictures (GOP), e.g., Li et al. (2020) proposed the Scalable Structured CVS (SS-CVS) framework, which learns the union of data-driven subspaces model to reconstruct GOPs. However, it has a defect in 3D reconstruction that the huge memory and high computational complexity are required to be invested at decoder. Derived from the decoding strategy of DVC, the distributed reconstruction divides the input video sequence into non-key frames and key frames and reconstructs each non-key frame by the CS recovery algorithm with the aid of its neighboring key frames. With a small memory and a low computational complexity, the distributed reconstruction improves the rate-distortion performance by exploiting the motions between frames, so many existing works focus on it to design the CVS systems, e.g., Ma et al. proposed the DIStributed video Coding Using Compressed Sampling (DISCUCS) (Prades-Nebot et al., 2009), Gan et al. proposed the DIStributeCOmpressed video Sensing (DISCOS) (Do et al., 2009), Fowler et al. proposed the Multi-Hypothesis Block CS (MH-BCS) system (Chen et al., 2011; Tramel and Fowler, 2011; Azghani et al., 2016), etc. The core of distributed reconstruction is the Multi-Hypothesis (MH) predictive technique, which uses a linear combination of blocks in key frames to interpolate the blocks in non-key frames. As one of the state-of-the-art techniques, the MH prediction is widely applied to distributed reconstruction. Recently, some works try to modify the implementation of MH prediction, e.g., Chen C. et al. (2020) added the iterative Reweighted TIKhonov-regularized scheme into MH prediction (MH-RTIK), causing a significant improvement of CVS performance. CS theory indicates that the precise recovery requires enough CS measurements. With insufficient CS measurements, the excellent CS recovery algorithm still cannot prevent the degradation of reconstruction quality, however, by adaptively allocating CS measurements based on local structures of the image, a simple recovery algorithm can also provide a good reconstruction quality (Yu et al., 2010; Taimori and Marvasti, 2018; Zammit and Wassell, 2020). Judging from the above facts, the adaptive allocation is a potential way to improve the rate-distortion performance of the CVS system with a light codec.

This article presents a context-based CVS system, of which the core is the allocation of CS measurements adapted by context structures at the encoder. Based on these adaptive measurements, by combining linear estimation and MH prediction into distributed reconstruction, the decoder provides a satisfying reconstruction quality with low memory and computational cost. The contributions of the proposed context-based CVS is to solve the following issues:

(1) How to extract the context structures from CS measurements? Traditional methods use pixels to compute the context features, but it costs lots of computations at the encoder, resulting in impracticality for CVS. Especially when the encoder is realized by Compressive Imaging (CI) devices (Liu et al., 2019; Deng et al., 2021), due to the unavailability of original pixels, it is impossible to perform the traditional methods. Considering the low dimensionality and availability of CS measurements, it is practical in CVS to extract context structures from CS measurements.
(2) How to adaptively allocate CS measurements by context structures? Contexts measure the correlations between pixels, and their distribution reveals some meaningful structures, e.g., smoothness, edges, textures, etc. With the same recovery quality, fewer necessary measurements are required for simple structures and more for complex structures. According to the distribution of contexts, an efficient

allocation is designed to avoid insufficiency or redundancy of measurements.

(3) How to quantize the adaptive measurements? Adaptive allocation makes blocks have different numbers of CS measurements, as a result, the traditional prediction cannot be applied to quantization. Due to the insufficient capability of SQ, an appropriate prediction scheme is required to reduce the quantization error.

Experimental results show that the proposed context-based CVS system outputs the high-quality reconstructed video sequences when monitoring plant growth or propagation and improves the rate-distortion performances when compared with the state-of-the-art CVS systems, which demonstrates the effectiveness of context-based allocation for CVS.

The rest of this article is organized as follows. Section 2 briefly overviews Plant Monitoring System, CVS, and describes the traditional method to extract context features. Section 3 presents the proposed context based CVS system. Experimental results are provided in Section 4, and we conclude this article in Section 5.

2. RELATED WORKS

2.1. Plant Monitoring System

In modern agriculture, it is essential to monitor plant propagation or growth for guaranteeing productivity. The labor costs can be efficiently reduced by automatically capturing the architectural parameters of the plant, so more and more attention has been paid to the design of the plant monitoring system (Somov et al., 2018; Grimblatt et al., 2021; Rayhana et al., 2021). Early, lots of systems are designed to monitor the various environmental parameters on plant growth, such as humidity, temperature, solar illuminance, etc., e.g., James and Maheshwar (2016) used multiple sensors to measure the soil data of plants and transmitted these data to the mobile phone by Raspberry Pi; Okayasu et al. (2017) developed a self-powered wireless monitoring device that is equipped with some environmental sensors; Guo et al. (2018) added big-data services to analyze the environmental data on plant growth. These environmental parameters indirectly indicate the process of plant growth, and they cannot record the visual scenes on plant growth, resulting in the unavailability of the physical structure parameters on plants. To realize the visual monitoring of plants, some works have started to integrate the visual sensors into the plant monitoring system, e.g., Peng et al. (2022) used the binocular camera to capture video sequences on a plant and used the structure from motion method (Piermattei et al., 2019) to extract the 3-D information of a plant; Sajith et al. (2019) designed a complex network to derive the plant growth parameters from the monitoring images; Akila et al. (2017) extracted the plant color and texture by the visual monitoring system. From the above, it can be seen that the visual sensor or camera is used to capture the video sequences on plant growth, and these video sequences are compressed as bitstream which is transmitted to the IoT cloud for further analyzing. As the core of visual sensors, the video compression is a major energy consumer, so a challenge that we face for the visual monitoring system of the plant is to design an energy-efficient video coding scheme to extend the working time of the visual sensor. In the framework of IoT, CVS is a potential coding scheme to reduce the energy consumption of visual sensors. The following briefly overviews the CVS systems.

2.2. CVS System

Compressive Video Sensing is the marriage of CS theory and DVC, which reduces the encoding costs and enhances the robustness to noise, thus becoming a potential video codec for wireless visual sensors. At the encoder, to satisfy low complexity and fast computation, the block-based CS sampling is performed on each video frame independently, i.e., the ith video frame $\boldsymbol{f}_i$ of size $N_1 \times N_2$ is partitioned into non-overlapping blocks of size $B \times B$, each block is vectorized as $\boldsymbol{x}_{i,j}$ of length N_b, and the CS measurements $\boldsymbol{y}_{i,j}$ of $\boldsymbol{x}_{i,j}$ are output by

$$\boldsymbol{y}_{i,j} = \boldsymbol{\Phi}_{i,j} \cdot \boldsymbol{x}_{i,j} \tag{1}$$

where $\boldsymbol{\Phi}_{i,j}$ is called as the measurement matrix and can be constructed by some random matrices, e.g., Gaussian, Bernoulli, structural random matrix, etc. By setting the length of $\boldsymbol{y}_{i,j}$ to be $M_{i,j}$, the size of $\boldsymbol{\Phi}_{i,j}$ is fixed to be $M_{i,j} \times N_b$, and the subrate S_i of $\boldsymbol{f}_i$ is defined as

$$S_i = \frac{M_i}{N} = \frac{\sum_{j=1}^{J} M_{i,j}}{N_1 \times N_2} \tag{2}$$

where N is the number of total pixels in $\boldsymbol{f}_i$, M_i is the number of CS measurements for $\boldsymbol{f}_i$, and J is the number of blocks in $\boldsymbol{f}_i$. In CI application, an optical device is designed to perform Equation (1), and directly output the CS measurements. To ensure a stable recovery, L video frames are gathered to form a GOP, in which the first frame, called the key frame, is set to be a high subrate, and others, called the non-key frame, are set to be a low subrate. After quantization, all CS measurements of GOP are packaged and transmitted to decoder.

At the decoder, by using the received CS measurements, the frame-by-frame, 3D, or distributed strategy is performed to reconstruct the GOP. For frame-by-frame, the reconstruction model can be represented by

$$\left\{\hat{\boldsymbol{x}}_{i,j}\right\}_{j=1}^{J} = arg \min_{\left\{\boldsymbol{x}_{i,j}\right\}_{j=1}^{J}} \left\{ \sum_{j=1}^{J} \left\| \boldsymbol{y}_{i,j} - \boldsymbol{\Phi}_{i,j} \cdot \boldsymbol{x}_{i,j} \right\|_2^2 + \alpha \cdot \sum_{j=1}^{J} \left\| \boldsymbol{\Psi} \cdot \boldsymbol{x}_{i,j} \right\|_1 \right\} \tag{3}$$

where $\boldsymbol{\Psi}$ denotes the 2D sparse representation basis, α is a regularization factor, $\|\cdot\|_2$ denotes ℓ_2 norm, and $\|\cdot\|_1$ denotes ℓ_1 norm. The model (3) can be solved by some non-linear optimization algorithms, e.g., Alternating Direction Method of Multipliers (ADMM) (Yang et al., 2020), and all reconstructed blocks are spliced into the estimated frame $\hat{\boldsymbol{f}}_i$. The frame-by-frame model uses only the spatial correlations, so its rate-distortion performance is unsatisfactory. The 3D reconstruction

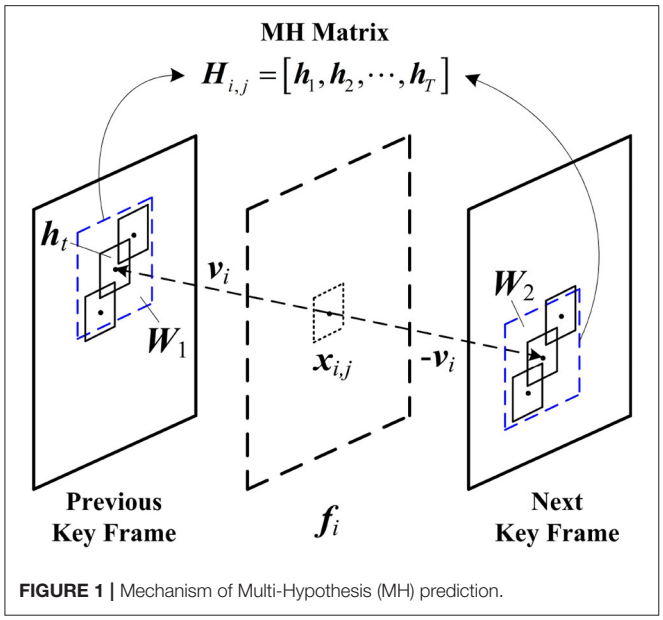

FIGURE 1 | Mechanism of Multi-Hypothesis (MH) prediction.

model fully considers the spatial-temporal correlations and it can be represented by

$$\left\{\hat{\boldsymbol{x}}_{i,j}\,|_{i=1}^{L}\right\}_{j=1}^{J} = arg \min_{\left\{\boldsymbol{x}_{i,j}|_{i=1}^{L}\right\}_{j=1}^{J}} \left\{ \sum_{i=1}^{L}\sum_{j=1}^{J} \|\boldsymbol{y}_{i,j} - \boldsymbol{\Phi}_{i,j}\cdot\boldsymbol{x}_{i,j}\|_2^2 + \alpha \cdot \sum_{i=1}^{L}\sum_{j=1}^{J} \left|\boldsymbol{\Gamma}\cdot\left[\boldsymbol{x}_{1,j}, \boldsymbol{x}_{2,j}, \cdots, \boldsymbol{x}_{L,j}\right]\right| \right\} \tag{4}$$

where $\boldsymbol{\Gamma}$ denotes the 3D sparse representation basis, and it is used to remove the spatial-temporal redundancies between blocks. Though effective, model (4) results in a heavy computational burden. Different from the 3D reconstruction, the distributed reconstruction uses the motion-compensation based prediction technique to expose the spatial-temporal redundancies between blocks. **Figure 1** shows the mechanism of MH prediction, which is commonly used in distributed reconstruction. MH prediction collects the spatial-temporal neighboring blocks in key frames to construct an MH matrix $\boldsymbol{H}_{i,j}$. According to the motion vector $\boldsymbol{v}_i$ of $\boldsymbol{x}_{i,j}$, the motion-aligned windows $\boldsymbol{W}_1$ and $\boldsymbol{W}_2$ of sizes $W \times W$ are, respectively, located on the previous and the next key frames, and all candidate blocks in $\boldsymbol{W}_1$ and $\boldsymbol{W}_2$ are extracted as the hypotheses $\{\boldsymbol{h}_t\}_{t=1}^{T}$ of $\boldsymbol{x}_{i,j}$, producing $\boldsymbol{H}_{i,j} = [\boldsymbol{h}_1, \boldsymbol{h}_2, \cdots, \boldsymbol{h}_T]$, in which $T = W^2$. By using MH prediction, the distributed reconstruction is modeled as a Least-Squares (LS) problem as follows:

$$\hat{\boldsymbol{w}}_{i,j} = arg\min_{\boldsymbol{w}} \left\{ \|\boldsymbol{y}_{i,j} - \boldsymbol{\Phi}_{i,j}\cdot\boldsymbol{H}_{i,j}\cdot\boldsymbol{w}\|_2^2 + \beta\cdot\|\boldsymbol{\Theta}\cdot\boldsymbol{w}\|_2^2 \right\} \tag{5}$$

$$\hat{\boldsymbol{x}}_{i,j} = \boldsymbol{H}_{i,j}\cdot\hat{\boldsymbol{w}}_{i,j} \tag{6}$$

where $\boldsymbol{\Theta}$ is the Tikhonov matrix, and β is a regularization factor. $\boldsymbol{\Theta}$ is a diagonal matrix and constructed by

$$\boldsymbol{\Theta} = \begin{bmatrix} \|\boldsymbol{y}_{i,j} - \boldsymbol{\Phi}_{i,j}\cdot\boldsymbol{h}_1\|_2 & & 0 \\ & \ddots & \\ 0 & & \|\boldsymbol{y}_{i,j} - \boldsymbol{\Phi}_{i,j}\cdot\boldsymbol{h}_T\|_2 \end{bmatrix} \tag{7}$$

With this structure, $\boldsymbol{\Theta}$ assigns weights of small magnitude to hypotheses mostly dissimilar from $\boldsymbol{x}_{i,j}$. The LS problem can be fast solved by the Conjugate Gradient algorithm (Zhang et al., 2018), which significantly reduces the computational complexity of distributed reconstruction. Due to the full exploitation of spatial-temporal correlations between blocks, the MH prediction enables the distributed reconstruction to provide superior recovery. From the above, in order to realize a light decoder and ensure a good recovery at the same time, distributed reconstruction is a wise way.

2.3. Contexts

Compressive Sensing theory indicates that the sparsity K of the signal determines its required number M of CS measurements by precise recovery. An empirical rule (Becker and Bobin, 2011) is that the precise recovery can be achieved if

$$M \geq 4\cdot K \tag{8}$$

In the block-based CS sampling, this rule can be used to avoid the redundancy or insufficiency of CS measurements for blocks, i.e., adapted by the sparsity, each block is allocated to the appropriate number of CS measurements. The sparsity is defined as the number of coefficients with significant magnitude in a representation, and its calculation has not a strict mathematical formula. For images, the sparsity can be revealed by some features, e.g., edge, variance, gradient, etc., and these features are applied into adaptive allocation, leading to the improvement of recovery quality. The simple features only describe the correlations between pixels, but the structures of blocks are not taken into consideration, thus we require some complex features to improve the efficiency of adaptive allocation. In Ref. Romano and Elad (2016), the self-similarity descriptor (Shechtman and Irani, 2007) is used to extract the contexts of blocks, which represents how similar a central block is to its large surrounding windows. Contexts contain the internal structures and external relations among blocks, and it is a potential feature to better reveal the sparsity variation. The following briefly describes how to extract the contexts in an image.

The context feature expresses the similarities between a central block and those of its large surrounding windows. As illustrated in **Figure 2**, for a central block $\boldsymbol{x}_p$ in an image, its similarity weights are computed by

$$s_{p,q} = \exp\left\{-\frac{\|\boldsymbol{x}_p - \boldsymbol{x}_q\|_2^2}{2\sigma^2}\right\}, \forall q \in \Omega_d\left(p\right) \tag{9}$$

where $\boldsymbol{x}_q$ denotes the qth surrounding block in a neighborhood $\Omega_d\left(p\right)$ of size $d \times d$, and σ is a normalization factor. The range

of $s_{p,q}$ is [0, 1], in which a large value indicates that the blocks $\boldsymbol{x}_p$ and $\boldsymbol{x}_q$ are highly similar, and a small value indicates that the two are substantially different. All weights constitute a correlation surface $\boldsymbol{U}_p = \left[s_{p,q} | \forall q \in \Omega_d(p)\right]$, of which the statistics reveal the self-similarity of $\boldsymbol{x}_p$. To measure the statistics, the correlation surface of $\boldsymbol{x}_p$ is rearranged into a histogram of b bins, of which the normalization is regarded as the context feature $\boldsymbol{g}_p$ of $\boldsymbol{x}_p$.

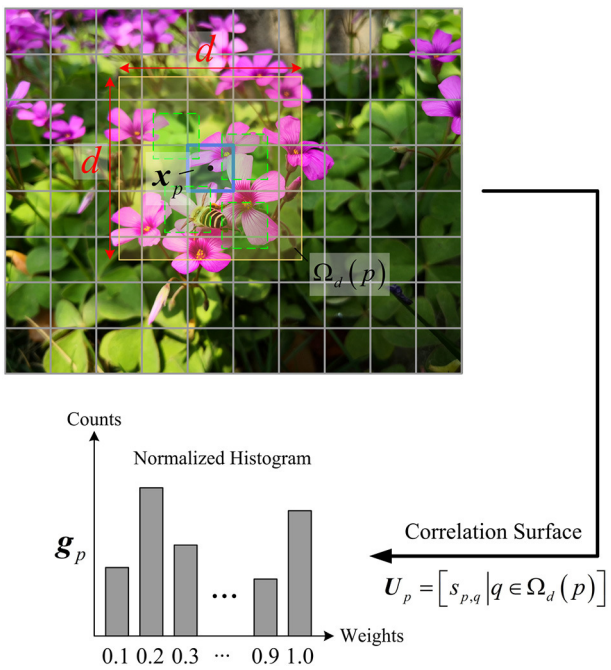

FIGURE 2 | Illustration on the traditional extraction of contexts.

The context feature $\boldsymbol{g}_p$ is an empirical distribution of the co-occurrences of $\boldsymbol{x}_p$ in its large surroundings, which measures the correlations between $\boldsymbol{x}_p$ to its surroundings. When $\boldsymbol{g}_p$ is biased toward the left bins, it can be concluded that the majority of $s_{p,q}$ are small, indicating the block $\boldsymbol{x}_p$ is unique, i.e., it originates from a highly textured and non-repetitive area, so its sparsity is relatively high. When $\boldsymbol{g}_p$ is biased toward the right bins, it means that most of $s_{p,q}$ are high, indicating that the block $\boldsymbol{x}_p$ has many co-occurrences in its surroundings, i.e., it originates from a large flat area, so its sparsity is low. From the above, we can see that the context feature accurately describes the geometric structure of a block with respect to its surrounding blocks, thus it is naturally sensitive to the sparsity variation. However, in CVS, the traditional method is impractical due to the unavailability of original pixels or high computational complexity. Therefore, it is challenging to extract the context feature by using CS measurements of blocks.

3. PROPOSED CONTEXT BASED CVS SYSTEM

3.1. System Architecture

As shown in **Figure 3**, we describe the architecture of the proposed context-based CVS system in detail. The input video sequence is divided into several GOPs of length L, and each GOP_k is successively encoded as Packet_k. After receiving this packet, the decoder reconstructs the corresponding $\widehat{\text{GOP}}_k$, and all reconstructed GOPs are regrouped as the entire video sequence.

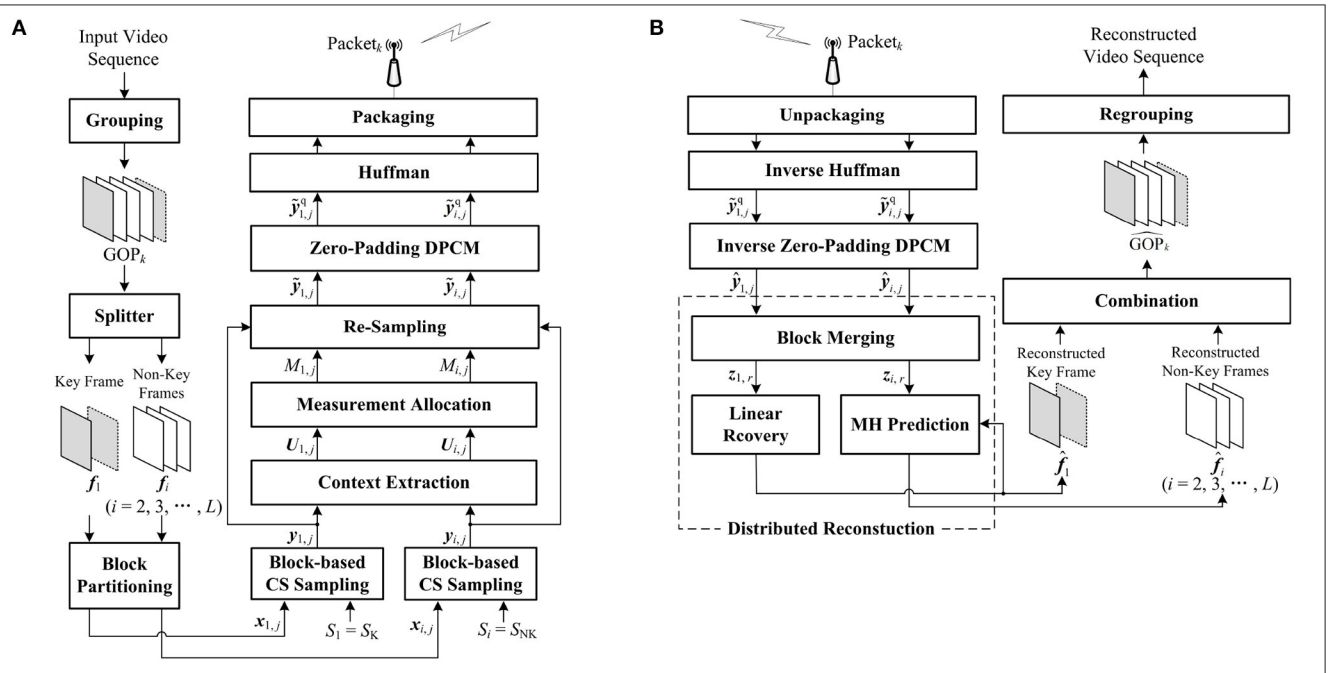

FIGURE 3 | Architecture of the proposed context-based Compressive Video Sensing (CVS) system: **(A)** encoder framework, **(B)** decoder framework.

Figure 3A presents the process of encoding GOP_k. The key frame $\boldsymbol{f}_1$ is split from GOP_k, and others $\{\boldsymbol{f}_i\}_{i=2}^{L}$ are regarded as the non-key frames. The key frame $\boldsymbol{f}_1$ and the ith non-key frame $\boldsymbol{f}_i$ are partitioned into J non-overlapping blocks $\{\boldsymbol{x}_{1,j}\}_{j=1}^{J}$ and $\{\boldsymbol{x}_{i,j}\}_{j=1}^{J}$ of size $B \times B$, respectively. For the key frame $\boldsymbol{f}_1$, we set a high subrate $S_1 = S_K$ to sample the blocks $\{\boldsymbol{x}_{1,j}\}_{j=1}^{J}$ and generate the CS measurements $\{\boldsymbol{y}_{1,j}\}_{j=1}^{J}$ according to Equation (1). The blocks $\{\boldsymbol{x}_{i,j}\}_{j=1}^{J}$ in the non-key frame $\boldsymbol{f}_i$ are sampled at a low subrate $S_i = S_{NK}$, producing the corresponding CS measurements $\{\boldsymbol{y}_{i,j}\}_{j=1}^{J}$ by Equation (1). For $\boldsymbol{f}_1$ and $\boldsymbol{f}_i$, based on the preset subrates, CS measurements are uniformly allocated to each block, however, without considering the structures of blocks, the uniform allocation results in either redundancy or insufficiency of CS measurements for some blocks. To improve the efficiency of block-based CS sampling, the core of the encoder is to perform the adaptive allocation by contexts of blocks. Different from traditional methods, the contexts $\boldsymbol{U}_{1,j}$ and $\boldsymbol{U}_{i,j}$ of $\boldsymbol{x}_{1,j}$ and $\boldsymbol{x}_{i,j}$ are, respectively, extracted by using the CS measurements $\boldsymbol{y}_{1,j}$ and $\boldsymbol{y}_{i,j}$, which makes CVS system more practical. After context extraction, according to the contexts $\boldsymbol{U}_{1,j}$ and $\boldsymbol{U}_{i,j}$, the numbers of CS measurements of $\boldsymbol{x}_{1,j}$ and $\boldsymbol{x}_{i,j}$ are modified as $M_{1,j}$ and $M_{i,j}$ by adaptive allocation. According to $M_{1,j}$ and $M_{i,j}$, by removing the redundancy or supplementing the insufficiency in $\boldsymbol{y}_{1,j}$ and $\boldsymbol{y}_{i,j}$, $\boldsymbol{x}_{1,j}$ and $\boldsymbol{x}_{i,j}$ are re-sampled as $\tilde{\boldsymbol{y}}_{1,j}$ and $\tilde{\boldsymbol{y}}_{i,j}$, respectively. DPCM cannot be used to quantize the adaptive measurements with different numbers. To overcome this defect of DPCM, we fuse zero padding into DPCM and predictively quantize $\tilde{\boldsymbol{y}}_{1,j}$ and $\tilde{\boldsymbol{y}}_{i,j}$ as $\tilde{\boldsymbol{y}}^{q}_{1,j}$ and $\tilde{\boldsymbol{y}}^{q}_{i,j}$. Finally, all quantized CS measurements are encoded as bits by Huffman and packaged as $Packet_k$.

Figure 3B presents the process of decoding $Packet_k$. After unpackaging $Packet_k$, the inversions of Huffman and zero-padding DPCM are implemented, and the CS measurements of $\boldsymbol{x}_{1,j}$ and $\boldsymbol{x}_{i,j}$ are recovered as $\hat{\boldsymbol{y}}_{1,j}$ and $\hat{\boldsymbol{y}}_{i,j}$ which have some quantization errors with their originals $\tilde{\boldsymbol{y}}_{1,j}$ and $\tilde{\boldsymbol{y}}_{i,j}$. The distributed reconstruction is performed to reconstruct the key frame $\boldsymbol{f}_1$ and the non-key frames $\{\boldsymbol{f}_i\}_{i=2}^{L}$. To suppress the blocking artifacts in the reconstructed frames, we realize the recovery of large blocks by merging the CS measurements

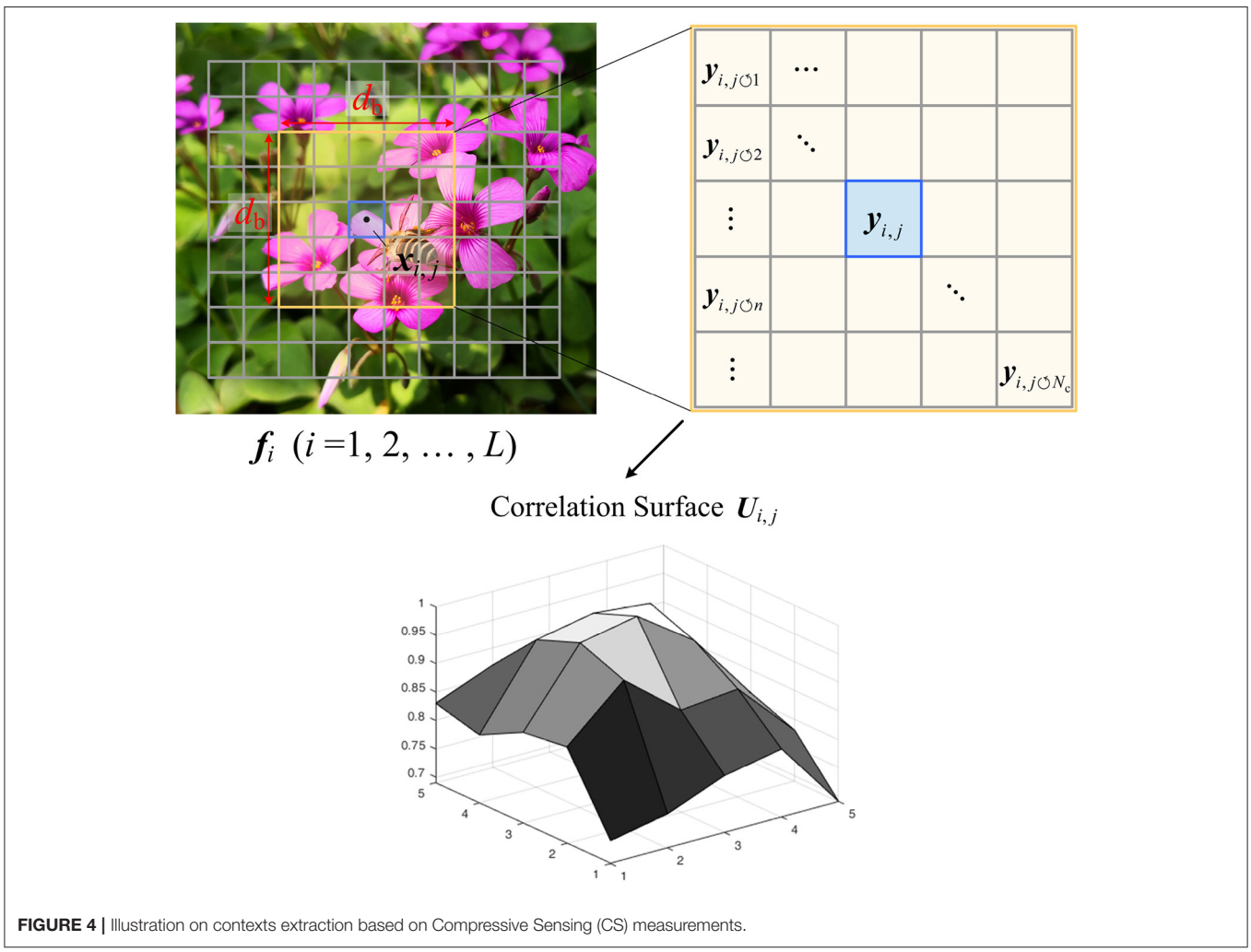

FIGURE 4 | Illustration on contexts extraction based on Compressive Sensing (CS) measurements.

of the spatially neighboring blocks, so the CS measurements of $\boldsymbol{f}_1$ and $\boldsymbol{f}_i$ are updated as $\boldsymbol{z}_{1,r}$ and $\boldsymbol{z}_{i,r}$ for large blocks. Based on $\boldsymbol{z}_{1,r}$, the reconstructed key frame $\hat{\boldsymbol{f}}_1$ is produced by using a linear recovery model, which rapidly recovers each block by a matrix-vector product. Regarding the previous and the next reconstructed key frames as references, the MH prediction outputs the reconstructed non-key frame $\hat{\boldsymbol{f}}_i$ by using $\boldsymbol{z}_{i,r}$. Finally, all reconstructed frames are combined into $\widehat{\mathrm{GOP}}_k$. Details of the core parts, including contexts extraction, measurements allocation, zero-padding DPCM, and distribution reconstruction, are described in the following subsections.

Algorithm 1 | Allocating the appropriate numbers of Compressive Sensing (CS) measurements to blocks.

Require: S_i - Subrate of $\boldsymbol{f}_i$, $P_{i,j}$ - Distribution on the sparsity of blocks $\boldsymbol{x}_{i,j}$, $j = 1, 2, \cdots, J$, N - Total number of pixels in $\boldsymbol{f}_i$, N_b - Block length;

Initial measurement number $m_{i,j}^{(0)} = \mathrm{Round}\left(P_{i,j} S_{i,j} N\right)$, and Round $(\cdot)$ is a rounding operator;
Restrict $m_{i,j}^{(0)}$ to not be larger than $0.9N_\mathrm{b}$, i.e., $m_{i,j}^{(0)} = \mathrm{Min}\left(m_{i,j}^{(0)}, 0.9N_\mathrm{b}\right)$, in which Min $(\cdot)$ is a minimization operator;
Set $M_\mathrm{sup} = S_i \cdot N - \sum_{j=1}^{J} m_{i,j}^{(0)}$, and $iter = 0$;
while $M_\mathrm{sup} > 0$, increment $iter$ by 1 **do**
  if $M_\mathrm{sup} < J$ **then**
    Randomly select M_sup blocks, and their measurement numbers are incremented by 1;
    Update $m_{i,j}^{(iter)}$, and set $M_{i,j} = m_{i,j}^{(iter)}$;
    Break;
  else
    $m_{i,j}^{(iter+1)} = m_{i,j}^{(iter)} + 1$
    $M_\mathrm{sup} = M_\mathrm{sup} - J$
  end if
end while
return $M_{i,j}, j = 1, 2, \cdots, J$.

3.2. Context Extraction

In the proposed CVS system, the context features are extracted by using the CS measurements of blocks. As illustrated in **Figure 4**, we compute the correlation surface $\boldsymbol{U}_{i,j}$ of $\boldsymbol{x}_{i,j}$ in $\boldsymbol{f}_i$ as its contexts, in which $i = 1, 2, \cdots, L$. In the surrounding window of size $d_\mathrm{b} \times d_\mathrm{b}$ centered on $\boldsymbol{x}_{i,j}$, we cannot extract the original blocks pixel-by-pixel due to the unavailability of original pixels, but can only use the CS measurements $\left\{\boldsymbol{y}_{i,j\bigcirc n}\right\}_{n=1}^{N_\mathrm{c}}$ of non-overlapping blocks $\left\{\boldsymbol{x}_{i,j\bigcirc n}\right\}_{n=1}^{N_\mathrm{c}}$, in which $N_\mathrm{c} = d_\mathrm{b}^2$. According to CS theory, the measurement matrix $\boldsymbol{\Phi}_{i,j}$ holds the Restricted Isometry Property (RIP) (Candès and Wakin, 2008) for blocks $\left\{\boldsymbol{x}_{i,j}\right\}_{j=1}^{J}$, which implies that all pairwise distances between original blocks can be well preserved in the measurement space, i.e.,

$$\begin{aligned} \|\boldsymbol{x}_{i,j} - \boldsymbol{x}_{i,j\bigcirc n}\|_2 &\approx \|\boldsymbol{\Phi}_{i,j} \cdot \boldsymbol{x}_{i,j} - \boldsymbol{\Phi}_{i,j} \cdot \boldsymbol{x}_{i,j\bigcirc n}\|_2 \\ &= \|\boldsymbol{y}_{i,j} - \boldsymbol{y}_{i,j\bigcirc n}\|_2, \forall n \in \{1, 2, \cdots, N_\mathrm{c}\} \end{aligned} \tag{10}$$

where it is noted that all blocks share the same measurement matrix $\boldsymbol{\Phi}_{i,j}$ due to the uniform allocation. Based on Equation

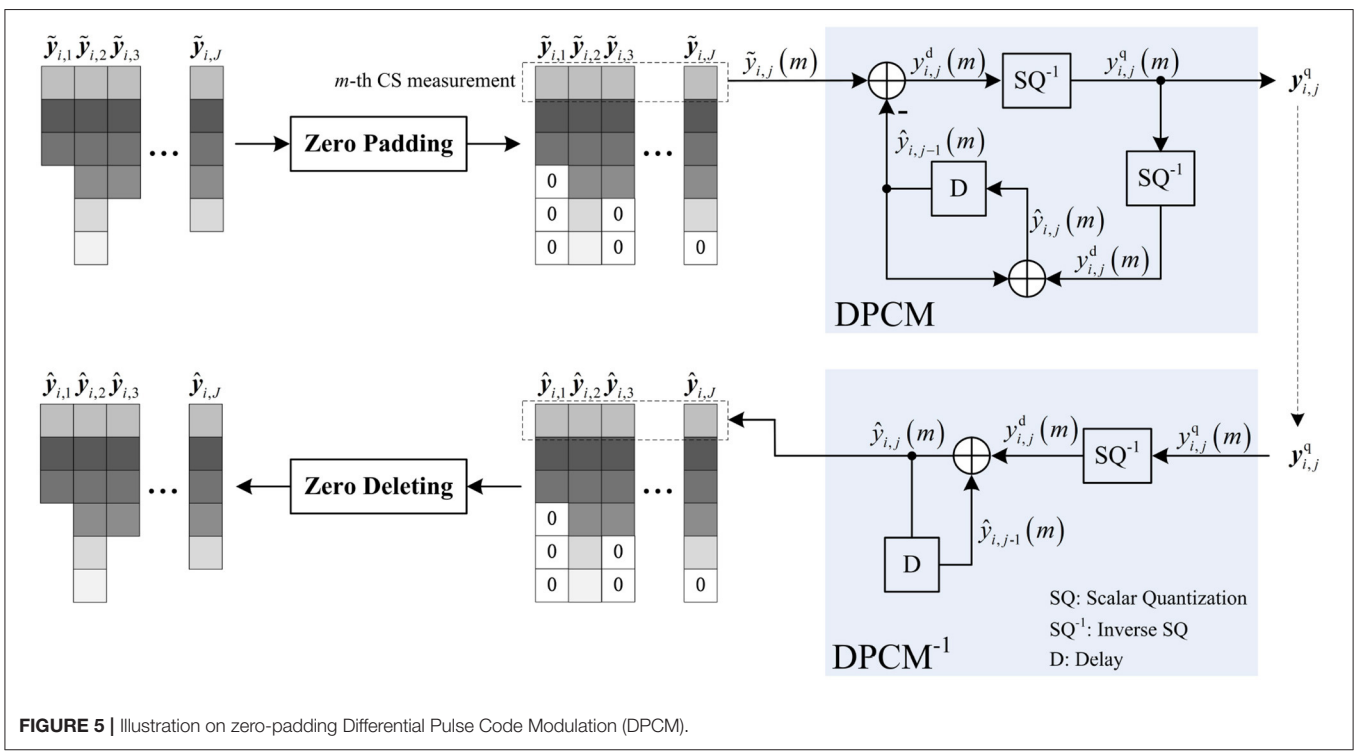

FIGURE 5 | Illustration on zero-padding Differential Pulse Code Modulation (DPCM).

(10), the similarity weights between $\boldsymbol{x}_{i,j}$ and $\boldsymbol{x}_{i,j\circlearrowleft n}$ can be estimated by

$$s_{i,j\circlearrowleft n} = \exp\left\{-\frac{\|\boldsymbol{y}_{i,j} - \boldsymbol{y}_{i,j\circlearrowleft n}\|_2^2}{2\sigma^2}\right\}, \forall n \in \{1, 2, \cdots, N_c\} \quad (11)$$

All weights constitute the correlation surface $\boldsymbol{U}_{i,j}$ as follows:

$$\boldsymbol{U}_{i,j} = \left[s_{i,j\circlearrowleft n} \,|\forall n \in \{1, 2, \cdots, N_c\}\right] \quad (12)$$

To compactly represent the contexts of $\boldsymbol{x}_{i,j}$, we compute the mean $u_{i,j}$ of $\boldsymbol{U}_{i,j}$ as the context feature, i.e.,

$$u_{i,j} = \frac{1}{N_c}\sum_{n=1}^{N_c} s_{i,j\circlearrowleft n} \quad (13)$$

3.3. Measurement Allocation

By exploiting the context feature $u_{i,j}$ of $\boldsymbol{x}_{i,j}$, we set the appropriate number of CS measurements for $\boldsymbol{x}_{i,j}$, and remove the redundancy or supplement the insufficiency in $\boldsymbol{y}_{i,j}$. The magnitudes of context features are high in smooth regions, and the magnitudes are low in the edge and texture regions, so it is found that the experience that the context feature is inversely proportional to the sparsity. Based on this experience, we can describe the distribution on the sparsity degrees of blocks by

$$P_{i,j} = \frac{u_{i,j}^{-1}}{\sum_{j=1}^{J} u_{i,j}^{-1}} \quad (14)$$

According to the present subrate S_i of $\boldsymbol{f}_i$, we construct the allocation model of CS measurements for blocks as follows:

$$\begin{aligned} M_{i,j} &= arg\min_{m_{i,j}} \sum_{j=1}^{J} \left(m_{i,j} - P_{i,j}\cdot S_i \cdot N\right) \\ \text{s.t.} &\sum_{j=1}^{j} m_{i,j} = S_i \cdot N, m_{i,j} \le 0.9 \cdot N_b, m_{i,j} \in \mathbb{N}^+ \end{aligned} \quad (15)$$

where N is the total number of pixels in $\boldsymbol{f}_i$, N_b is the block length, $m_{i,j}$ is a positive integer, and its upper bound is set to be $0.9 \cdot N_b$. The model (15) is solved according to **Algorithm 1** and outputs the final number $M_{i,j}$ of CS measurements for $\boldsymbol{x}_{i,j}$.

3.4. Zero-Padding DPCM

Due to the adaptive allocation, the lengths of the re-sampled CS measurements $\left\{\tilde{y}_{i,j}\right\}_{j=1}^{L}$ vary. Compared with SQ, DPCM provides better rate-distortion performance by adding the

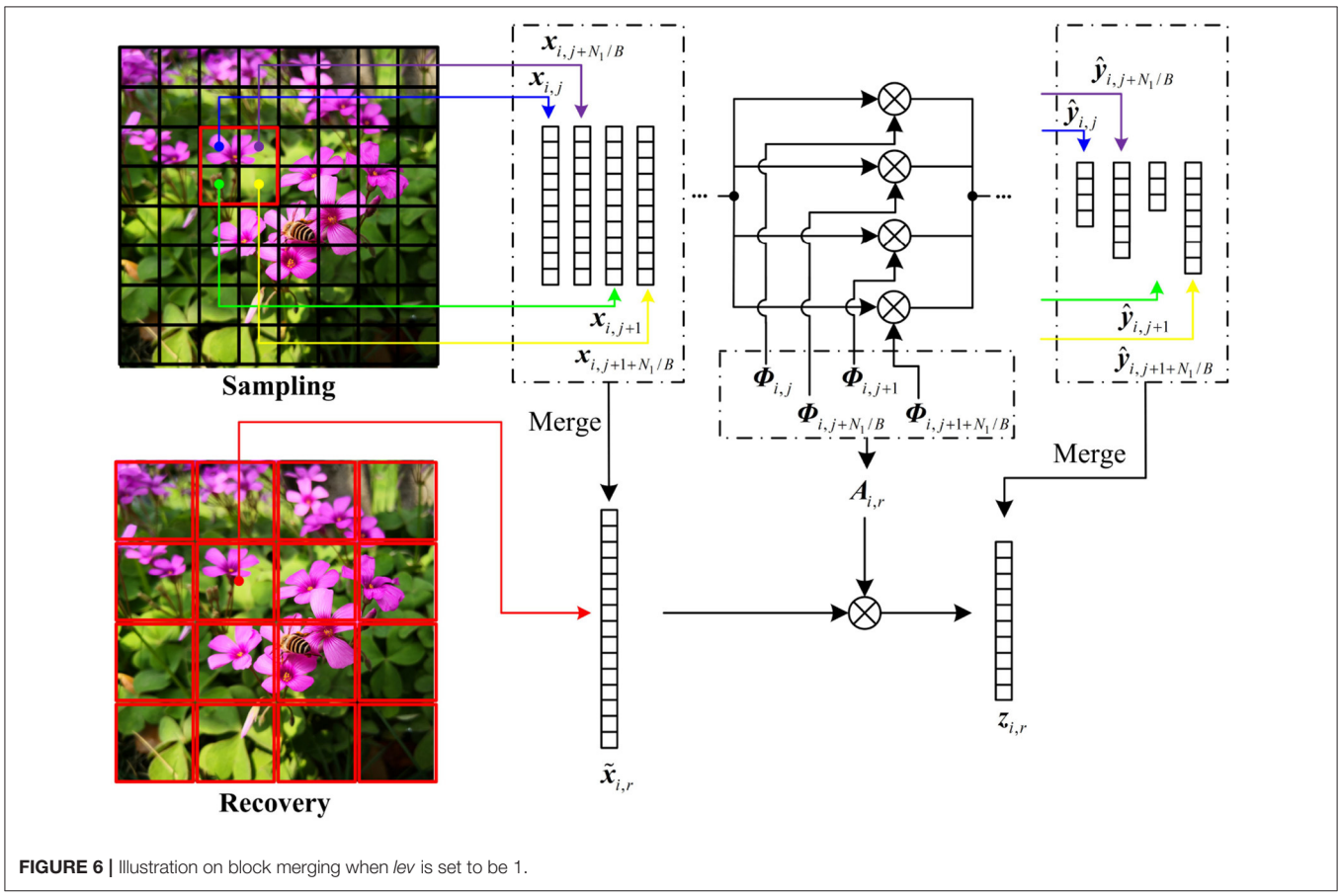

FIGURE 6 | Illustration on block merging when *lev* is set to be 1.

predictive scheme into the quantization of block-based CS measurements. However, DPCM requires that all blocks have the same number of CS measurements, as a result, DPCM cannot be used to quantize $\left\{\tilde{y}_{i,j}\right\}_{j=1}^{L}$. To make DPCM adapt to the adaptive allocation, we propose zero-padding DPCM, whose implementation is shown in **Figure 5**. Before inputting $\tilde{y}_{i,j}$ to DCPM, we fill zeros in the last of $\tilde{y}_{i,j}$ to make its length the same as others. After obtaining the de-quantized CS measurements $\hat{y}_{i,j}$, we delete the zeros in the last of $\hat{y}_{i,j}$ to recover its original length $M_{i,j}$. By zero padding, each measurement in $\hat{y}_{i,j-1}$ can be used to predict the corresponding measurement in $\hat{y}_{i,j}$, and especially when there is predictive measurement $\hat{y}_{i,j-1}(m)$ of the m-th measurement $\tilde{y}_{i,j}(m)$, the residual $y_{i,j}^{\mathrm{d}}(m)$ can be significantly reduced due to the intrinsic spatial correlation between $\tilde{y}_{i,j}$ and $\tilde{y}_{i,j-1}$. The rate-distortion curves of the reconstructed *Foreman*, *Mobile*, and *Football* sequences are presented when zero-padding DPCM and SQ are, respectively, used to quantize the adaptive CS measurements (shown in **Supplementary Figure 1**), in which the rate-distortion curve is measured in terms of the Peak Signal-to-Noise Ratio (PSNR) in dB and bitrate in bits per pixel (bpp), and the linear recovery algorithm presented in subsection 3.5 is used to recover each video frame. It can be seen that zero-padding DPCM presents competitive performance with SQ at low bitrates but as the bitrate increases, its improvement of performance over SQ is increasingly significant. From these results, we find that the efficiency of zero-padding DPCM relies on the correlation between block-based CS measurements. With insufficient measurements, the correlation is weakened by the filling of excessive zeros, causing the performance degradation, but when measurements are sufficient, a high correlation is maintained, so the performance improvement stands out. From the above, zero-padding DPCM is more suitable for adaptive measurements compared with SQ.

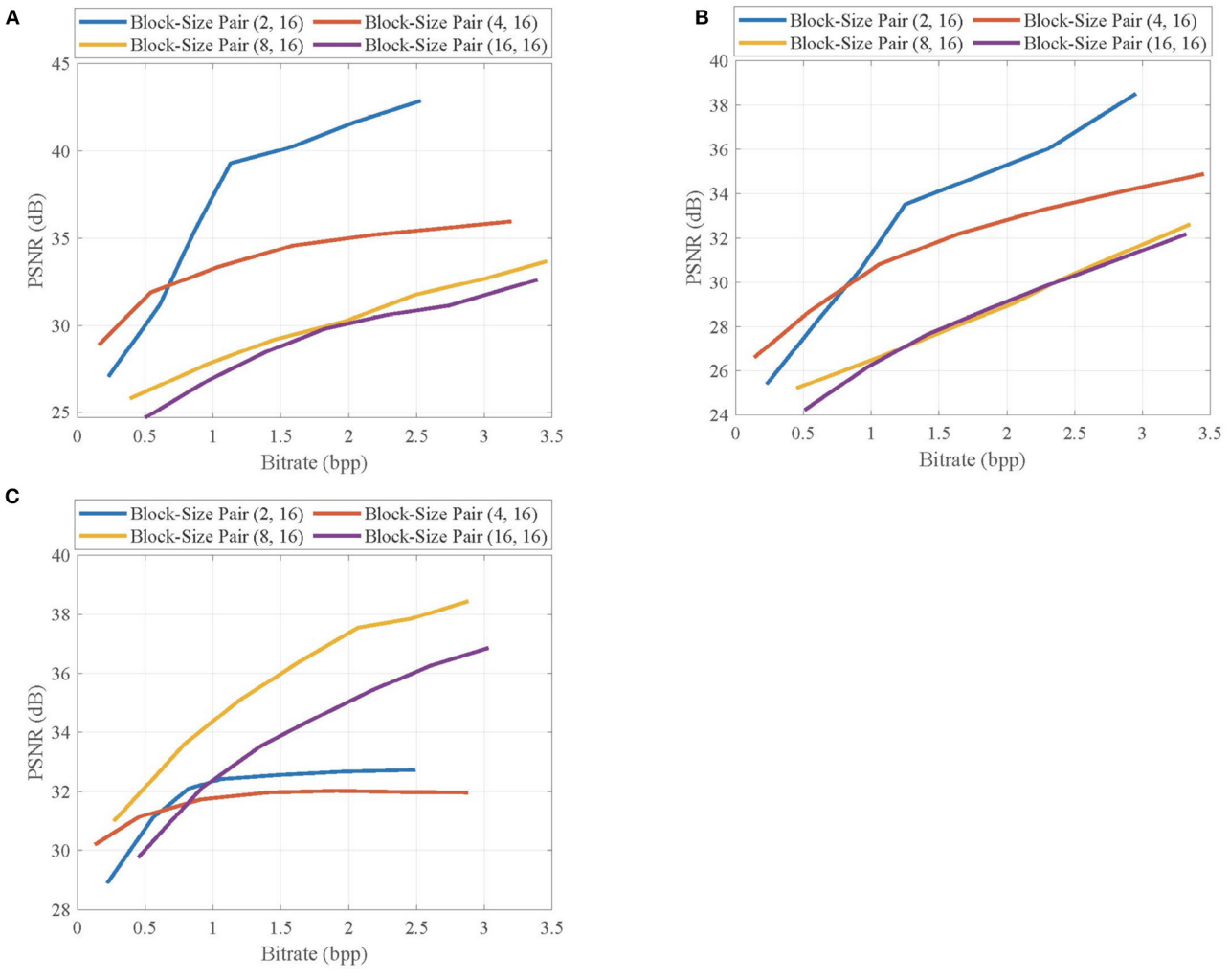

FIGURE 7 | Rate-distortion curves of the reconstructed key frames in **(A)** *Foreman*, **(B)** *BlowingBubbles*, and **(C)** *ParkScene* sequences at different block-size pairs.

3.5. Distributed Reconstruction

At decoder, the distributed strategy is performed to reconstruct the key frame $\boldsymbol{f}_1$ and the non-key frames $\{\boldsymbol{f}_i\}_{i=2}^{L}$, in which $\boldsymbol{f}_1$ is estimated by a linear recovery model, and $\boldsymbol{f}_i$ is produced by MH prediction. To highlight the complex structures by contexts, a small block size is more desired at the encoder. However, the small block size causes serious blocking artifacts due to the differences of neighboring blocks in recovery quality. To suppress the blocking artifacts, we merge the CS measurements $\{\hat{\boldsymbol{y}}_{i,j}\}_{j=1}^{J}$ of the small blocks $\{\boldsymbol{x}_{i,j}\}_{j=1}^{J}$ into those $\{\boldsymbol{z}_{i,r}\}_{r=1}^{R}$ of the large blocks $\{\tilde{\boldsymbol{x}}_{i,r}\}_{r=1}^{R}$ and realize the sampling of small blocks and the recovery of large blocks. The size $B_{lev} \times B_{lev}$ of large block is set to be

$$B_{lev} = 2^{lev} \cdot B, lev = 1, 2, \cdots \tag{16}$$

in which lev is a positive integer. The number R of large blocks is N/B_{lev}^2, and it is smaller than the number J of small blocks. **Figure 6** illustrates the block merging when lev is set to be 1. The four neighboring blocks $\boldsymbol{x}_{i,j}$, $\boldsymbol{x}_{i,j+1}$, $\boldsymbol{x}_{i,j+N_1/B}$, $\boldsymbol{x}_{i,j+1+N_1/B}$ are merged into a large block $\tilde{\boldsymbol{x}}_{i,r}$, and their CS measurements $\hat{\boldsymbol{y}}_{i,j}$, $\hat{\boldsymbol{y}}_{i,j+1}$, $\hat{\boldsymbol{y}}_{i,j+N_1/B}$, and $\hat{\boldsymbol{y}}_{i,j+1+N_1/B}$ are spliced into $\boldsymbol{z}_{i,r}$ in rows, i.e.,

$$\boldsymbol{z}_{i,r} = \begin{bmatrix} \hat{\boldsymbol{y}}_{i,j} \\ \hat{\boldsymbol{y}}_{i,j+1} \\ \hat{\boldsymbol{y}}_{i,j+N_1/B} \\ \hat{\boldsymbol{y}}_{i,j+1+N_1/B} \end{bmatrix} \approx \boldsymbol{\Lambda}_{i,r} \cdot \begin{bmatrix} \boldsymbol{x}_{i,j} \\ \boldsymbol{x}_{i,j+1} \\ \boldsymbol{x}_{i,j+N_1/B} \\ \boldsymbol{x}_{i,j+1+N_1/B} \end{bmatrix} \tag{17}$$

$$\boldsymbol{\Lambda}_{i,r} = \begin{bmatrix} \boldsymbol{\Phi}_{i,j} & & & 0 \\ & \boldsymbol{\Phi}_{i,j+1} & & \\ & & \boldsymbol{\Phi}_{i,j+N_1/B} & \\ 0 & & & \boldsymbol{\Phi}_{i,j+1+N_1/B} \end{bmatrix} \tag{18}$$

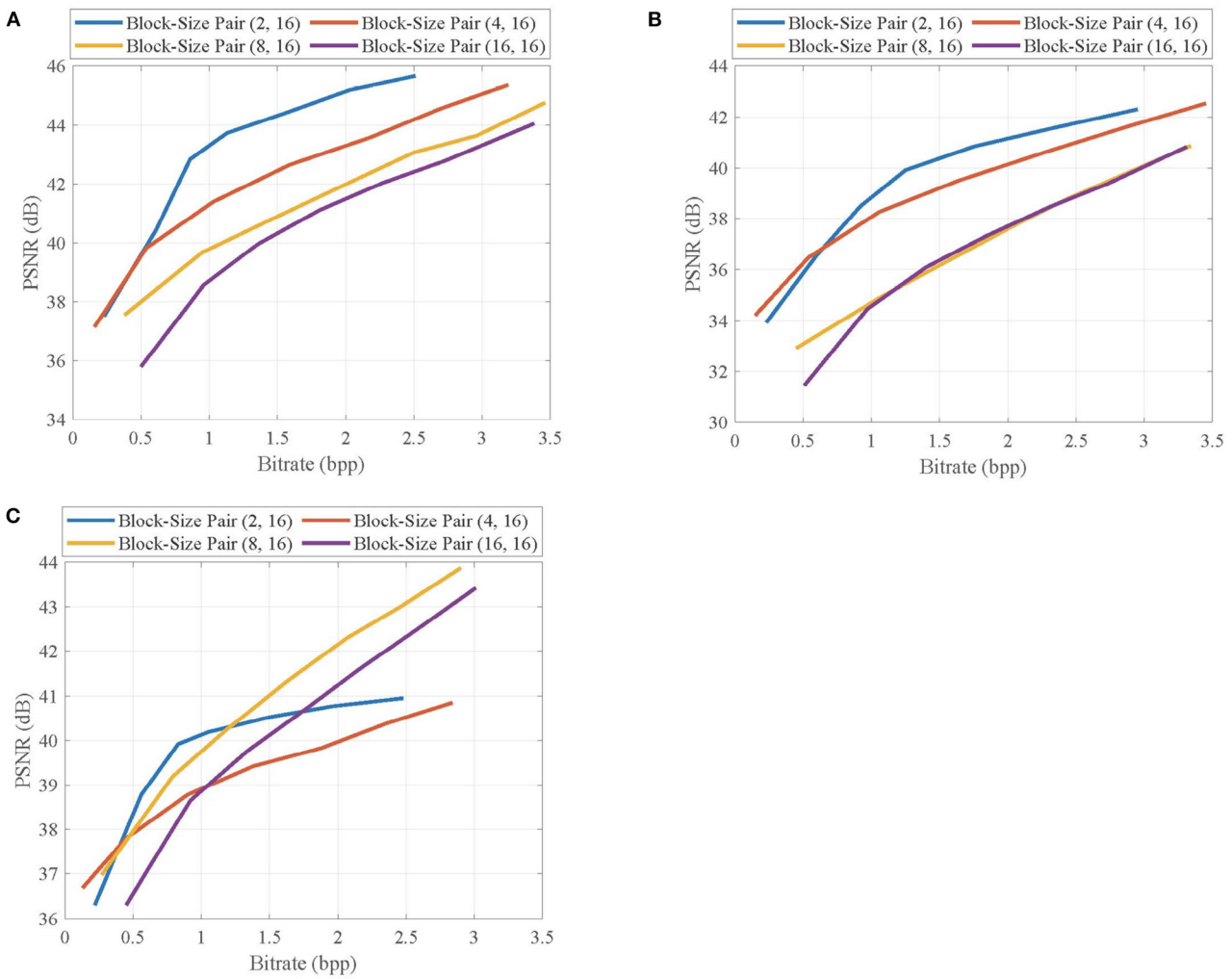

FIGURE 8 | Rate-distortion curves of the reconstructed non-key frames in **(A)** *Foreman*, **(B)** *BlowingBubbles*, and **(C)** *ParkScene* sequences at different block-size pairs.

in which $\boldsymbol{\Lambda}_{i,r}$ is the diagonal matrix composed of the block measurement matrices $\boldsymbol{\Phi}_{i,j}$, $\boldsymbol{\Phi}_{i,j+1}$, $\boldsymbol{\Phi}_{i,j+N_1/B}$, and $\boldsymbol{\Phi}_{i,j+1+N_1/B}$, N_1 is the total number of rows in $\boldsymbol{f}_i$, and B is the block size of the small block. To make $\boldsymbol{z}_{i,r}$, the CS measurements of $\tilde{\boldsymbol{x}}_{i,r}$, we transform $\tilde{\boldsymbol{x}}_{i,r}$ as

$$\begin{bmatrix} \boldsymbol{x}_{i,j} \\ \boldsymbol{x}_{i,j+1} \\ \boldsymbol{x}_{i,j+N_1/B} \\ \boldsymbol{x}_{i,j+1+N_1/B} \end{bmatrix} = \boldsymbol{I} \cdot \tilde{\boldsymbol{x}}_{i,r} \tag{19}$$

in which $\boldsymbol{I}$ is an elementary column transformation matrix. Plugging Equation (19) into Equation (17), we build the bridge between $\tilde{\boldsymbol{x}}_{i,r}$ and $\boldsymbol{z}_{i,r}$ by

$$\boldsymbol{z}_{i,r} \approx \boldsymbol{\Lambda}_{i,r} \cdot \boldsymbol{I} \cdot \tilde{\boldsymbol{x}}_{i,r} = \boldsymbol{A}_{i,r} \cdot \tilde{\boldsymbol{x}}_{i,r} \tag{20}$$

in which $\boldsymbol{A}_{i,r} = \boldsymbol{\Lambda}_{i,r} \cdot \boldsymbol{I}$. According to Equation (20), the large block $\tilde{\boldsymbol{x}}_{i,r}$ can be recovered by using $\boldsymbol{z}_{i,r}$. When *lev* is set to be larger than 1, the block merging can be done in manner similar to the above.

After the block merging, we use $\{\boldsymbol{z}_{1,r}\}_{r=1}^{R}$ to recover the key frame $\boldsymbol{f}_1$. The block $\tilde{\boldsymbol{x}}_{1,r}$ of $\boldsymbol{f}_1$ is linearly estimated by

$$\hat{\boldsymbol{x}}_{1,r} = \boldsymbol{P}_{1,r} \cdot \boldsymbol{z}_{1,r} \tag{21}$$

in which $\boldsymbol{P}_{1,r}$ is the transformation matrix produced by the following model:

$$\boldsymbol{P}_{1,r} = arg \min_{\boldsymbol{P}} \left\{ E\left[\| \tilde{\boldsymbol{x}}_{1,r} - \boldsymbol{P} \cdot \boldsymbol{z}_{1,r} \|_2^2 \right] \right\} \tag{22}$$

in which $E[\cdot]$ denotes the expectation function. The model (22) outputs the optimal transformation matrix to minimize the mean square error between $\tilde{\boldsymbol{x}}_{1,r}$ and its estimator $\hat{\boldsymbol{x}}_{1,r}$, and it can be

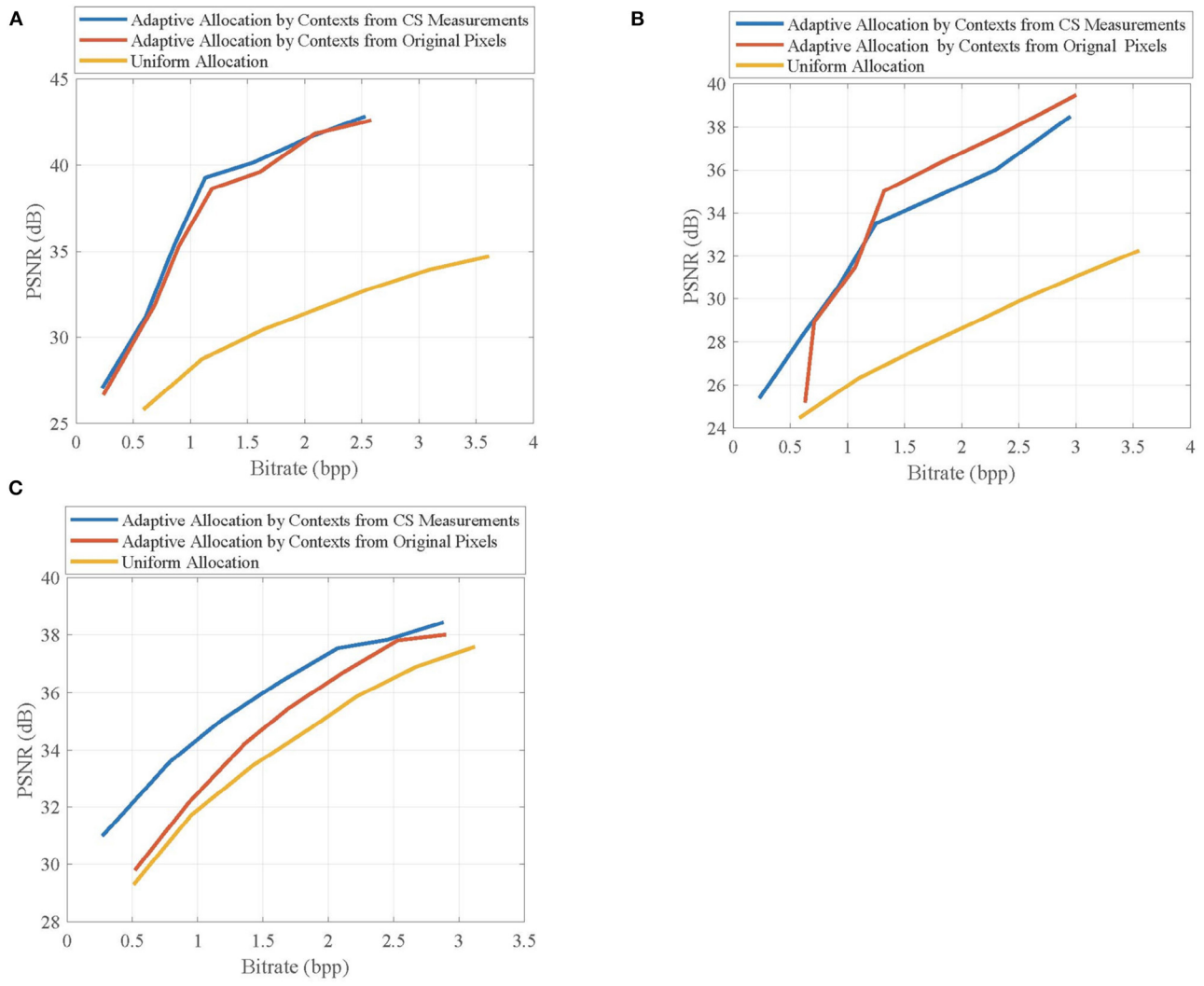

FIGURE 9 | Rate-distortion curves of the reconstructed key frames in **(A)** *Foreman*, **(B)** *BlowingBubbles*, and **(C)** *ParkScene* sequences when using different allocation schemes. For *Foreman* and *BlowingBubbles*, the block-size pair is set to be (2, 16), and for *ParkScene*, the block-size is set to be (8, 16).

solved by making the gradient of objective function equal to 0, producing

$$\boldsymbol{P}_{1,r} = E\left[\tilde{\boldsymbol{x}}_{1,r}\boldsymbol{z}_{1,r}^{\mathrm{T}}\right] E^{-1}\left[\boldsymbol{z}_{1,r}\boldsymbol{z}_{1,r}^{\mathrm{T}}\right] \tag{23}$$

Plugging Equation (20) into Equation (23), we get

$$\boldsymbol{P}_{1,r} = \boldsymbol{Cor}_{\mathrm{xx}} \cdot \boldsymbol{A}_{1,r}^{\mathrm{T}} \left(\boldsymbol{A}_{1,r} \cdot \boldsymbol{Cor}_{\mathrm{xx}} \cdot \boldsymbol{A}_{1,r}^{\mathrm{T}}\right)^{-1} \tag{24}$$

$$\boldsymbol{Cor}_{\mathrm{xx}} = E\left[\tilde{\boldsymbol{x}}_{1,r}\tilde{\boldsymbol{x}}_{1,r}^{\mathrm{T}}\right] \tag{25}$$

in which $\boldsymbol{Cor}_{\mathrm{xx}}$ is the auto-correlation matrix of $\tilde{\boldsymbol{x}}_{1,r}$, and its element $Cor_{\mathrm{xx}}[m,n]$ is estimated as follows:

$$Cor_{\mathrm{xx}}[m,n] = 0.95^{\delta_{m,n}} \tag{26}$$

in which $\delta_{m,n}$ is the Euclidean distance between two pixels $\tilde{x}_{1,r}(m)$ and $\tilde{x}_{1,r}(n)$ in $\tilde{\boldsymbol{x}}_{1,r}$. When the subrate is set to be large, the linear recovery model can provide excellent visual quality while costing fewer computations.

4. EXPERIMENTAL RESULTS

We evaluate the proposed CVS system on video sequences with various resolutions, including seven CIF (352 × 288) sequences *Akiyo*, *Bus*, *Container*, *Coastguard*, *Football*, *Foreman*, *Hall*, one WQVGA (416 × 240) sequence *BlowingBubbles*, and one 1080p (1920 × 1080) sequence *ParkScene*. In the proposed CVS system, the window size $d_{\mathrm{b}} \times d_{\mathrm{b}}$ and the normalization factor σ are, respectively, set to be 11 × 11 and 10 for the context extraction, the window size $W \times W$ and the regularization factor β are, respectively, set to be 21×21 and 0.25 for the MH prediction, and

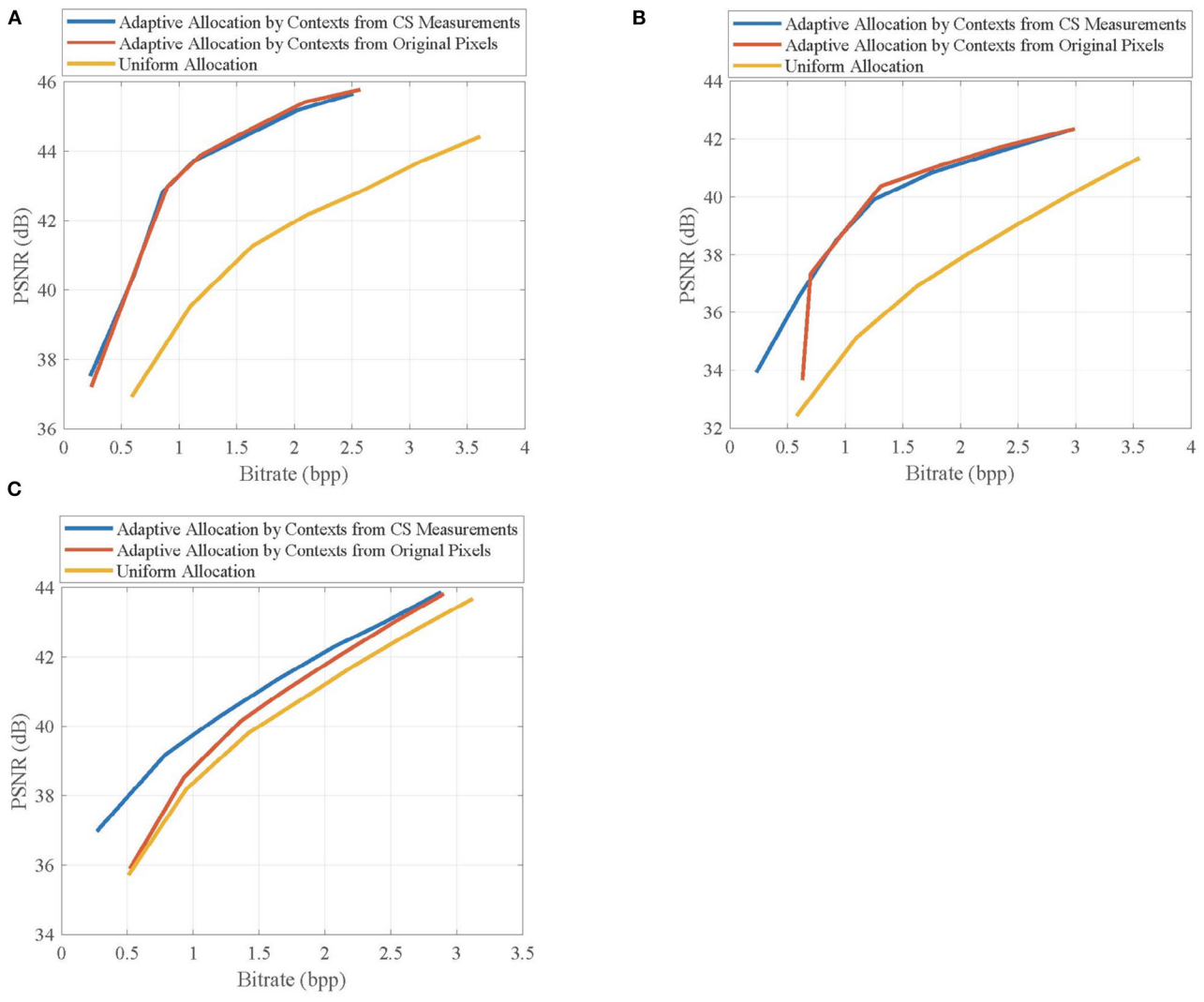

FIGURE 10 | Rate-distortion curves of the reconstructed non-key frames in **(A)** *Foreman*, **(B)** *BlowingBubbles*, and **(C)** *ParkScene* sequences when using different allocation schemes. For *Foreman* and *BlowingBubbles*, the block-size pair is set to be (2, 16), and for *ParkScene*, the block-size is set to be (8, 16).

the measurement matrix is produced by Gaussian distribution. First, we discuss the effects of different block sizes on the proposed CVS system. Second, we evaluate the performance improvement resulting from the used context extraction. Finally, we compare the proposed CVS system with two state-of-the-art CVS systems: SS-CVS (Li et al., 2020) and MH-RTIK (Chen C. et al., 2020) in terms of the rate-distortion performance. PSNR is used to evaluate the qualities of reconstructed video sequences, and the bitrate denotes the average amount of bits per pixel to encode a video sequence. The variation of PSNR with bitrate is called the rate-distortion performance. The computational complexity is measured by the execution time. Experiments are implemented with MATLAB on a workstation with 3.30-GHz CPU and 8 GB RAM.

4.1. Effects of Block Sizes

In the proposed CVS system, in order to highlight the complex structures by contexts, we desire a small block size at encoder, but at decoder, a large block size is desired to suppress the blocking artifacts in the reconstructed video frames. We set a block-size pair (B, B_{lev}), in which B and B_{lev} are the block sizes for sampling and recovery, respectively, and evaluate the effects of different block-size pairs on the reconstruction qualities of key frames and non-key frames.

First, we select the first frames of *Foreman*, *BlowingBubbles*, and *ParkScene* sequences as the key frames, which are linearly recovered, and show their rate-distortion curves at different block-size pairs in **Figure 7**. For *Foreman* and *BlowingBubbles* with the low resolution, the block-size pair (4, 16) achieves higher PSNR values than others with low bitrates, but the rate-distortion curve for the block-size pair (2, 16) rapidly increases as the bitrate increases and significant PSNR gains are achieved when compared with other block-size pairs. These results indicate that the small blocks used in adaptive allocation and large blocks for linear recovery fit together well. For *ParkScene* with high resolution, when the block size B for sampling is set to be too small, e.g., $B = 2$, no block can contain sufficient structures, causing the rate-distortion performance to degenerate as the bitrate increases, but a suitable block size for sampling is set, e.g., $B = 8$, PSNR gains can be significantly improved.

Then, we select the second frames of *Foreman*, *BlowingBubbles*, and *ParkScene* sequences as the non-key frames, which are recovered by MH prediction based on the reconstructed previous and next key frames at the subrate 0.7, and show their rate-distortion curves at different block-size pairs in **Figure 8**. Similar to the results from key frames, for *Foreman* and *BlowingBubbles*, the better rate-distortion performance is achieved when the block-size pair is set to be (2, 16), and for *ParkScene*, in order to prevent the loss of structures, the block size for sampling is appropriately set to be 8.

Given the above, we can see that the bad effects resulting from the extraction of contexts can be suppressed by the block

TABLE 1 | Average Peak Signal-to-Noise Ratio (PSNR) (dB) for reconstructed video sequences by the proposed Compressive Video Sensing (CVS) system, Scalable Structured CVS (SS-CVS) (Trevisi et al., 2020), and Multi-Hypothesis Reweighted TIKhonov (MH-RTIK) (Chen C. et al., 2020) at subrates 0.1 to 0.5.

Sequence	Resolution	Algorithm	Subrate S_{NK}				
			0.1	0.2	0.3	0.4	0.5
GOP Length $L = 2$							
Container	CIF	MH-RTIK	33.67	34.76	35.08	35.28	35.47
		Proposed	**38.74**	**39.92**	**40.38**	**40.47**	**40.61**
Coastguard		MH-RTIK	33.12	34.26	34.69	35.08	35.43
		Proposed	**35.80**	**37.22**	**38.30**	**38.89**	**39.45**
Hall		MH-RTIK	37.10	38.01	38.39	38.65	38.91
		Proposed	**38.26**	**39.69**	**40.82**	**41.23**	**41.50**
Foreman		MH-RTIK	36.52	37.09	37.56	37.96	38.60
		Proposed	**38.13**	**39.66**	**40.87**	**41.41**	**41.78**
GOP Length $L = 10$							
Akiyo	CIF	SS-CVS	17.70	24.80	33.06	36.55	39.23
		Proposed	**40.75**	**43.50**	**45.28**	**45.09**	**45.56**
Bus		SS-CVS	18.65	23.57	25.71	27.67	30.10
		Proposed	**25.65**	**38.31**	**30.97**	**32.97**	**34.00**
Football		SS-CVS	15.52	23.95	27.87	30.33	32.93
		Proposed	**28.98**	**32.67**	**35.78**	**36.55**	**37.28**
Foreman		SS-CVS	13.40	20.51	28.07	32.90	35.25
		Proposed	**33.18**	**36.00**	**38.55**	**39.54**	**40.22**
BlowingBubble	QWVGA	SS-CVS	16.93	23.50	28.47	30.70	32.84
		Proposed	**30.13**	**32.17**	**33.58**	**35.01**	**35.68**
ParkScene	1080P	SS-CVS	23.19	30.04	33.14	35.53	36.62
		Proposed	**33.01**	**35.18**	**36.79**	**37.97**	**38.67**

merging, therefore, the quality improvement from contexts-based allocation is further enhanced.

4.2. Effects of Contexts

In the proposed CVS system, the contexts are extracted from CS measurements and used to adaptively allocate the CS measurements for blocks, leading to the improvement of reconstruction quality. To verify the validity of contexts from CS measurements on the quality improvement, we evaluate the effects of different allocation schemes on the rate-distortion performance of the proposed CVS system. The uniform allocation is used as a benchmark, and the adaptive allocation uses the contexts extracted from CS measurements and original pixels, respectively.

Figure 9 shows the rate-distortion curves of the reconstructed key frames when using different allocation schemes, in which the key frames are, respectively, taken from the first frames of *Foreman*, *BlowingBubbles*, and *ParkScene* sequences. It can be seen that adaptive allocation outperforms uniform allocation in PSNR values at any bitrate, indicating that contexts contribute to quality improvement. Importantly, the contexts from CS measurements are competitive with those from original pixels, and their performance gaps are very small, which means that CS measurements can better represent the contexts of blocks.

Figure 10 shows the rate-distortion curves of the reconstructed non-key frames when using different allocation schemes, in which the non-key frames are, respectively, taken from the second frames of *Foreman*, *BlowingBubbles*, and *ParkScene* sequences. It can be seen that the adaptive allocation is still effective for MH prediction, and it can significantly improve the rate-distortion performances when compared with uniform allocation. The contexts from CS measurements have similar efficiency of allocation to that of contexts from original pixels, which proves that the merits of adaptive allocation can still be maintained in the measurement domain.

The above results indicate that the contexts extracted by CS measurements prompt the adaptive allocation to improve the reconstruction quality of CVS system, which makes the

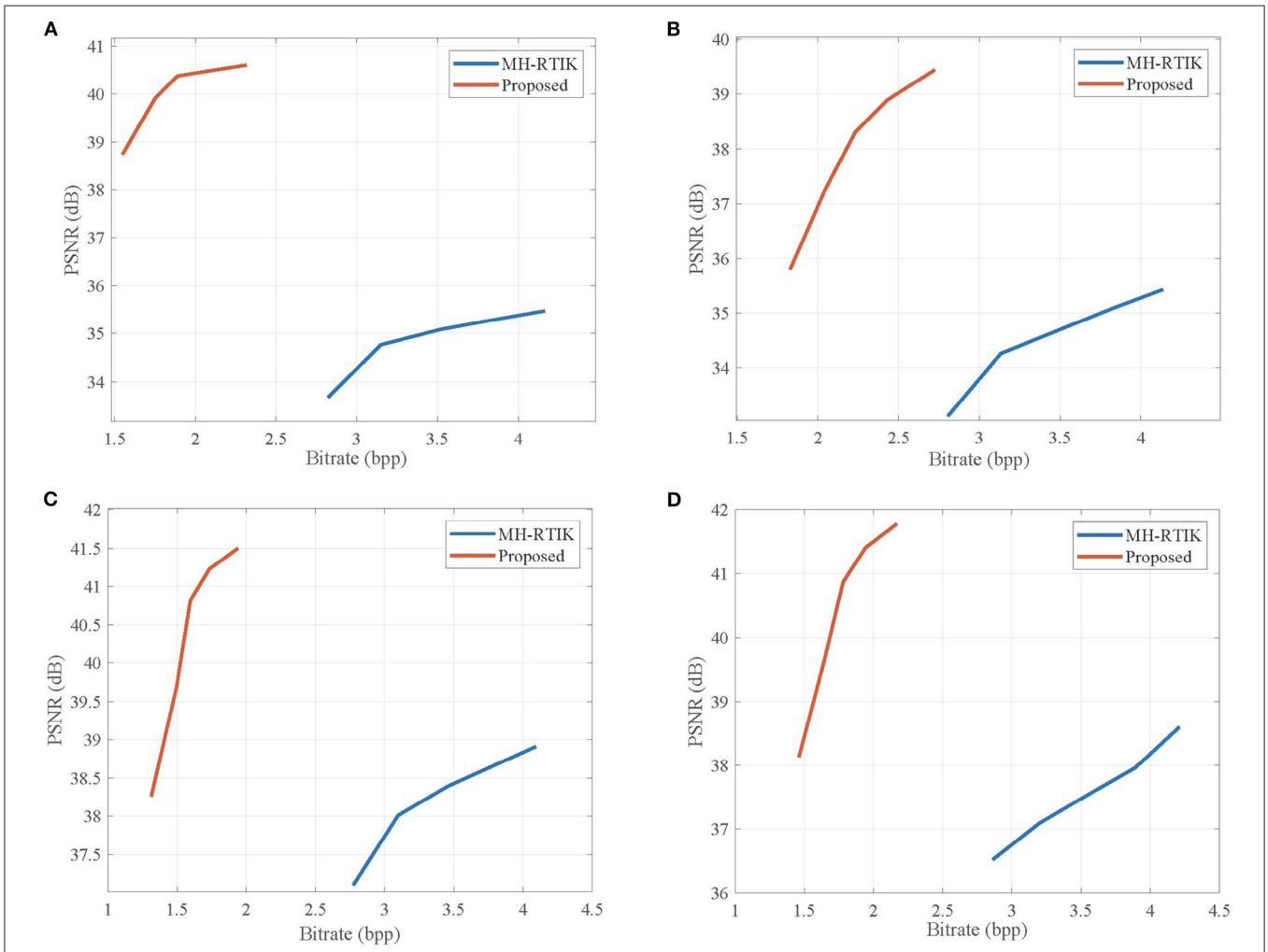

FIGURE 11 | Rate-distortion curves obtained by the proposed CVS system and Multi-Hypothesis TIKhonov (MH-TIK) (Chen C. et al., 2020) for **(A)** *Container*, **(B)** *Coastguard*, **(C)** *Hall*, and **(D)** *Foreman* sequences. Note that the length *L* of GOP is set to be 2.

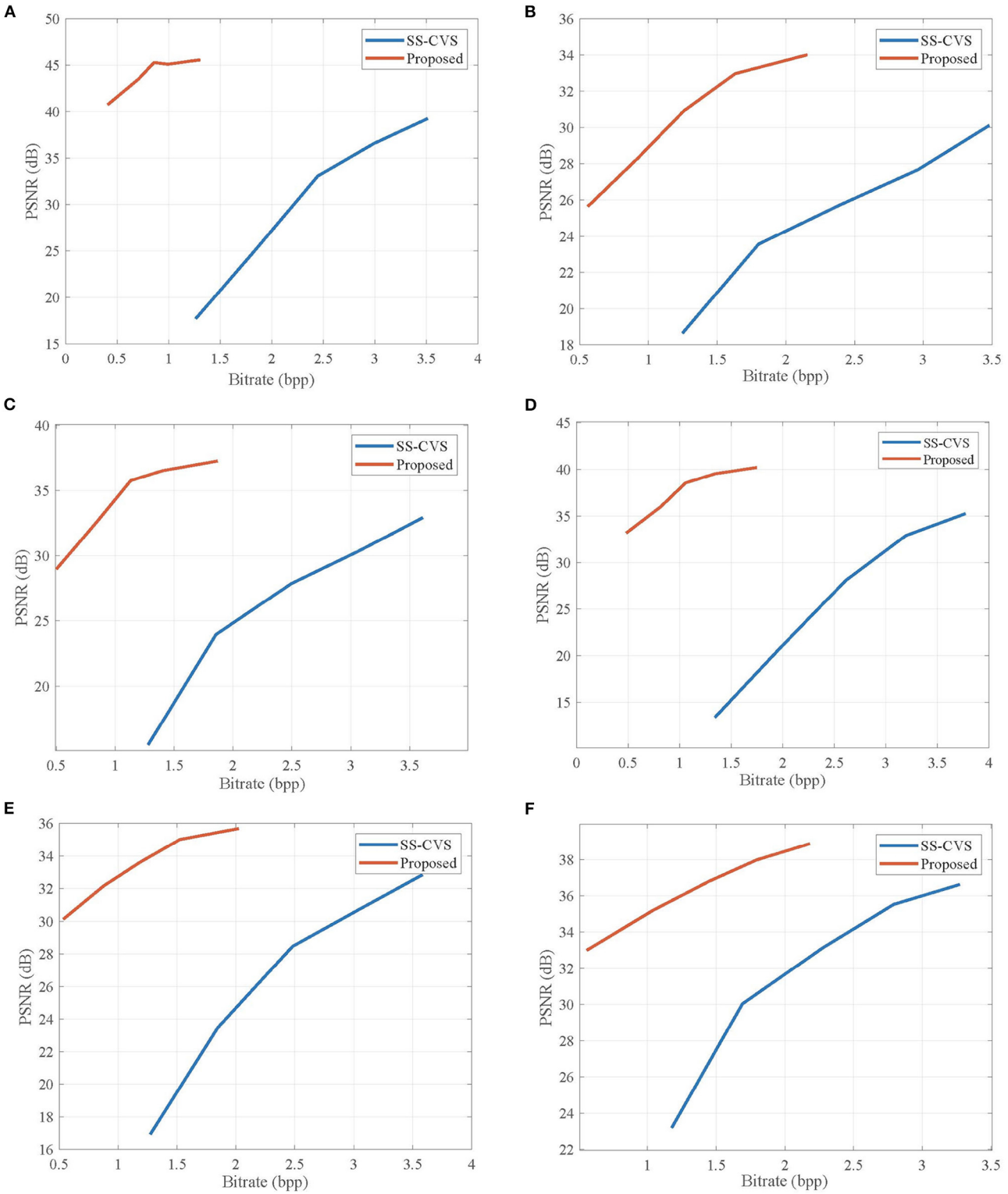

FIGURE 12 | Rate-distortion curves obtained by the proposed CVS system and Scalable Structured CVS (SS-CVS) (Li et al., 2020) for **(A)** *Akiyo*, **(B)** *Bus*, **(C)** *Football*, **(D)** *Foreman*, **(E)** *BlowingBubble*, and **(F)** *ParkScene* sequences. Note that the length *L* of GOP is set to be 10.

TABLE 2 | Average encoding time (s/frame) and decoding time (s/frame) on video sequences with different resolutions for the proposed CVS system, SS-CVS (Li et al., 2020), and MH-RTIK (Chen C. et al., 2020).

Resolution	Algorithm	Encoding Time (s/frame)	Decoding Time (s/frame)
Average on Subrates S_{NK} 0.1 to 0.5			
CIF	MH-RTIK	0.17	19.34
	Proposed	0.63	4.48
Average on Subrates $S_{NK} = 0.6$			
CIF	SS-CVS	5.40	21.22
	Proposed	0.64	7.79
QWVGA	SS-CVS	4.90	17.23
	Proposed	0.64	7.69
1080P	SS-CVS	108.10	401.8
	Proposed	1.83	162.47

proposed CVS system more suitable to the applications with limited resources.

4.3. Performance Comparisons

We evaluate the performance of the proposed CVS system by comparing it with the two state-of-the-art CVS systems: SS-CVS (Li et al., 2020) and MH-RTIK (Chen C. et al., 2020). To make a fair comparison, we keep the parameter settings of SS-CVS and MH-RTIK in their original reports, some important details are repeated as follows:

1) SS-CVS: the system consists of one base layer and one enhancement layer; the block size is set to be 16; the length of GOP is 10; the subrate of key frame is set to be 0.9; the dimension of the subspace is 10; the number of subspaces is 50.
2) MH-RTIK: the sub-block extraction is used; the number of hypotheses is 40; the block size is set to be 16; the length of GOP is 2; the subrate of key frame is set to be 0.7.

In addition, we employ SQ and Huffman in SS-CVS and MH-RTIK to compress the CS measurements. For the proposed CVS system, the block-size pair is set to be $(2, 16)$ for CIF and QWVGA sequences and $(8, 16)$ for 1080P sequences, the subrate S_K of key frame is set to be 0.7, the results under the GOP length $L = 2$ are compared with those of MH-RTIK, and the results under the GOP length $L = 10$ are compared with those of SS-CVS.

Table 1 lists the average PSNR values for the reconstructed video sequences by the proposed CVS system, SS-CVS, and MH-RTIK when the subrate S_{NK} of non-key frame varies from 0.1 to 0.5. Compared with MH-RTIK, the proposed CVS system achieves obvious PSNR gains at any subrate, e.g., the average PSNR gain is 2.824 dB for the *Foreman* sequence. Compared with SS-CVS, the proposed CVS system also presents higher PSNR values at any subrate, and especially for low subrates, PSNR gains are significant, e.g., when the subrate is 0.1, PSNR gains are 9.82, 13.20, and 19.78 dB for *ParkScene*, *BlowingBubble*, *Foreman* sequences, respectively. **Figures 11**, **12** show the rate-distortion curves for the proposed CVS system, MH-RTIK, and SS-CVS. Due to the implementation of zero-padding DPCM, the performance improvement of the proposed CVS system is further enhanced when compared with MH-RTIK and SS-CVS. By the objective evaluation of the reconstruction quality, it can be indicated that the proposed CVS system can significantly improve the qualities of the reconstructed video sequences.

Table 2 lists the average encoding time (s/frame) and decoding time (s/frame) on video sequences with different resolutions for the proposed CVS system, SS-CVS, and MH-RTIK. We compute the average execution time on the range $[0.1, 0.5]$ of subrate S_{NK} for the proposed CVS system and compare it with that of MH-RTIK for CIF sequences. The encoding speed of the proposed CVS system is slowed down due to the contexts-based adaptive allocation, and its encoding time is 0.63 s per frame, larger than that of MH-RTIK. Assisted by the simple linear recovery, the proposed CVS system reduces the decoding complexity, and only costs 4.48 s to reconstruct a video frame, however, MH-RTIK requires 19.34 s per frame. Under the subrate $S_{NK} = 0.6$, the execution time of the proposed CVS algorithm is compared with that of SS-CVS for the CIF, QWVGA, and 1080P video sequences, respectively. Compared with SS-CVS, the proposed CVS system costs less encoding time, and the encoding time does not dramatically increase as the resolution increases, e.g., for 1080P sequence, the proposed CVS system only costs 1.83 s per frame, but SS-CVS costs 108.10 s. In SS-CVS, the subspace clustering and the basis derivation are implemented at the encoder, and they lead to more encoding costs than the adaptive allocation in the proposed CVS system. The proposed CVS system costs less decoding time than SS-CVS, and its decoding costs also grow more slowly when compared with SS-CVS, e.g., for 1080P sequence, the proposed CVS system costs 162.47 s per frame, and the SS-CVS costs 401.8 s. The heavy computational burdens for SS-CVS derive from the non-linear subspace learning, but the decoding complexity of the proposed CVS system is limited benefiting from the linear recovery and prediction. From the above, we can see that the proposed CVS system still keeps a low computational complexity while providing better rate-distortion performance.

5. CONCLUSION

In this article, a context-based CVS system is proposed to improve the visual quality of the reconstructed video sequences. At the encoder, the CS measurements are adaptively allocated for blocks according to the contexts of video frames. Innovatively, the contexts are extracted by CS measurements. Although the extraction of contexts is independent of original pixels, these contexts can still better reveal the structural complexity of each block. To guarantee better rate-distortion performance, the zero-padding DPCM is proposed to quantize these adaptive measurements. At the decoder, the key frames are reconstructed by linear recovery, and these non-key frames are reconstructed by MH prediction. Thanks to the effectiveness of context-based adaptive allocation, the simple recovery schemes also provide the comfortable visual quality. Experimental results show that the proposed CVS system improves the rate-distortion performances

when compared with two state-of-the-art CVS systems, including MH-RTIK and SS-CVS, and guarantees a low computational complexity.

As the research in this article is exploratory, there are many intriguing questions that future work should consider. First, the estimation of block sparsity should be analyzed in mathematics. Second, we will investigate how to fuse the quantization into adaptive allocation. More importantly, we will deploy the adaptive CVS system on an actual hardware platform.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/**Supplementary Material**, further inquiries can be directed to the corresponding author/s.

AUTHOR CONTRIBUTIONS

RL: designed the study and drafted the manuscript. YY: conducted experiments and analyzed the data. FS: critically reviewed and improved the manuscript. All authors have read and approved the final version of the manuscript.

FUNDING

This work was supported in part by the Project of Science and Technology, Department of Henan Province in China (212102210106), National Natural Science Foundation of China (31872704), Innovation Team Support Plan of University Science and Technology of Henan Province in China (19IRTSTHN014), and Guangxi Key Laboratory of Wireless Wideband Communication and Signal Processing of China.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fpls.2022.849606/full#supplementary-material

REFERENCES

Akila, I., Sivakumar, A., and Swaminathan, S. (2017). "Automation in plant growth monitoring using high-precision image classification and virtual height measurement techniques," in *2017 International Conference on Innovations in Information, Embedded and Communication Systems (ICIIECS)* (Coimbatore), 1–4.

Azghani, M., Karimi, M., and Marvasti, F. (2016). Multihypothesis compressed video sensing technique. *IEEE Trans. Circuits Syst. Video Technol.* 26, 627–635. doi: 10.1109/TCSVT.2015.2418586

Baraniuk, R. G. (2007). Compressive sensing. *IEEE Signal Process. Mag.* 24, 118–121. doi: 10.1109/MSP.2007.4286571

Baraniuk, R. G., Goldstein, T., Sankaranarayanan, A. C., Studer, C., Veeraraghavan, A., and Wakin, M. B. (2017). Compressive video sensing: algorithms, architectures, and applications. *IEEE Signal Process. Mag.* 34, 52–66. doi: 10.1109/MSP.2016.2602099

Becker S., and Bobin J., C. E. (2011). Nesta: a fast and accurate first-order method for sparse recovery. *SIAM J. Imag. Sci.* 4, 1–39. doi: 10.1137/090756855

Bigot, J., Boyer, C., and Weiss, P. (2016). An analysis of block sampling strategies in compressed sensing. *IEEE Trans. Inf. Theory* 62, 2125–2139. doi: 10.1109/TIT.2016.2524628

Candès, E. J., and Wakin, M. B. (2008). An introduction to compressive sampling. *IEEE Signal Process. Mag.* 25, 21–30. doi: 10.1109/MSP.2007.914731

Chen, C., Tramel, E. W., and Fowler, J. E. (2011). "Compressed-sensing recovery of images and video using multihypothesis predictions," in *2011 Conference Record of the Forty Fifth Asilomar Conference on Signals, Systems and Computers (ASILOMAR)* (Pacific Grove, CA), 1193–1198.

Chen, C., Zhou, C., Liu, P., and Zhang, D. (2020). Iterative reweighted tikhonov-regularized multihypothesis prediction scheme for distributed compressive video sensing. *IEEE Trans. Circuits Syst. Video Technol.* 30, 1–10. doi: 10.1109/TCSVT.2018.2886310

Chen, Y., Huang, T.-Z., He, W., Yokoya, N., and Zhao, X.-L. (2020). Hyperspectral image compressive sensing reconstruction using subspace-based nonlocal tensor ring decomposition. *IEEE Trans. Image Process.* 29, 6813–6828. doi: 10.1109/TIP.2020.2994411

Deng, C., Zhang, Y., Mao, Y., Fan, J., Suo, J., Zhang, Z., et al. (2021). Sinusoidal sampling enhanced compressive camera for high speed imaging. *IEEE Trans. Pattern Anal. Mach. Intell.* 43, 1380–1393. doi: 10.1109/TPAMI.2019.2946567

Do, T. T., Chen, Y., Nguyen, D. T., Nguyen, N., Gan, L., and Tran, T. D. (2009). "Distributed compressed video sensing," in *2009 16th IEEE International Conference on Image Processing (ICIP)* (Baltimore, MD), 1393–1396.

Do, T. T., Gan, L., Nguyen, N. H., and Tran, T. D. (2012). Fast and efficient compressive sensing using structurally random matrices. *IEEE Trans. Signal Process.* 60, 139–154. doi: 10.1109/TSP.2011.2170977

Gan, L. (2007). "Block compressed sensing of natural images," in *2007 15th International Conference on Digital Signal Processing* (Cardiff), 403–406.

Gao, X., Zhang, J., Che, W., Fan, X., and Zhao, D. (2015). "Block-based compressive sensing coding of natural images by local structural measurement matrix," in *2015 Data Compression Conference* (Snowbird, UT), 133–142.

Girod, B., Aaron, A., Rane, S., and Rebollo-Monedero, D. (2005). Distributed video coding. *Proc. IEEE* 93, 71–83. doi: 10.1109/JPROC.2004.839619

Grimblatt, V., Jégo, C., Ferré, G., and Rivet, F. (2021). How to feed a growing population—an iot approach to crop health and growth. *IEEE J. Emerg. Sel. Top. Circuits Syst.* 11, 435–448. doi: 10.1109/JETCAS.2021.3099778

Guo, L., Liu, Y., Hao, H., Han, J., and Liao, T. (2018). "Growth monitoring and planting decision supporting for pear during the whole growth stage based on pie-landscape system," in *2018 7th International Conference on Agro-geoinformatics (Agro-geoinformatics)* (Hangzhou), 1–4.

James, J., and Maheshwar P, M. (2016). "Plant growth monitoring system, with dynamic user-interface," in *2016 IEEE Region 10 Humanitarian Technology Conference (R10-HTC)* (Agra), 1–5.

Li, Y., Dai, W., Zou, J., Xiong, H., and Zheng, Y. F. (2020). Scalable structured compressive video sampling with hierarchical subspace learning. *IEEE Trans. Circuits Syst. Video Technol.* 30, 3528–3543. doi: 10.1109/TCSVT.2019.2939370

Liu, Y., Yuan, X., Suo, J., Brady, D. J., and Dai, Q. (2019). Rank minimization for snapshot compressive imaging. *IEEE Trans. Pattern Anal. Mach. Intell.* 41, 2990–3006. doi: 10.1109/TPAMI.2018.2873587

Mun, S., and Fowler, J. E. (2012). "Dpcm for quantized block-based compressed sensing of images," in *2012 Proceedings of the 20th European Signal Processing Conference (EUSIPCO)* (Bucharest), 1424–1428.

Okayasu, T., Nugroho, A. P., Sakai, A., Arita, D., Yoshinaga, T., Taniguchi, R.-I., et al. (2017). "Affordable field environmental monitoring and plant growth measurement system for smart agriculture," in *2017 Eleventh International Conference on Sensing Technology (ICST)* (Sydney, NSW), 1–4.

Palangi, H., Ward, R., and Deng, L. (2016). Distributed compressive sensing: a deep learning approach. *IEEE Trans. Signal Process.* 64, 4504–4518. doi: 10.1109/TSP.2016.2557301

Peng, Y., Yang, M., Zhao, G., and Cao, G. (2022). Binocular-vision-based structure from motion for 3-d reconstruction of plants. *IEEE Geosci. Remote Sens. Lett.* 19, 1–5. doi: 10.1109/LGRS.2021.3105106

Piermattei, L., Karel, W., Wang, D., Wieser, M., Mokroš, M., Surový, P., et al. (2019). Terrestrial structure from motion photogrammetry for deriving forest inventory data. *Remote Sens.* 11, 950. doi: 10.3390/rs11080950

Prades-Nebot, J., Ma, Y., and Huang, T. (2009). "Distributed video coding using compressive sampling," in *2009 Picture Coding Symposium* (Chicago, IL), 1–4.

Qiu, W., Zhou, J., Zhao, H., and Fu, Q. (2015). Three-dimensional sparse turntable microwave imaging based on compressive sensing. *IEEE Geosci. Remote Sens. Lett.* 12, 826–830. doi: 10.1109/LGRS.2014.2363238

Rayhana, R., Xiao, G. G., and Liu, Z. (2021). Printed sensor technologies for monitoring applications in smart farming: a review. *IEEE Trans. Instrum. Meas.* 70, 1–19. doi: 10.1109/TIM.2021.3112234

Romano, Y., and Elad, M. (2016). Con-patch: when a patch meets its context. *IEEE Trans. Image Process.* 25, 3967–3978. doi: 10.1109/TIP.2016.2576402

Sajith V. V. V., Gopalakrishnan E. A., Sowmya V., and Soman K. P. (2019). "A complex network approach for plant growth analysis using images," in *2019 International Conference on Communication and Signal Processing (ICCSP)* (Chennai), 0249–0253.

Shechtman, E., and Irani, M. (2007). "Matching local self-similarities across images and videos," in *2007 IEEE Conference on Computer Vision and Pattern Recognition* (Minneapolis, MN), 1–8.

Somov, A., Shadrin, D., Fastovets, I., Nikitin, A., Matveev, S., Seledets, I., et al. (2018). Pervasive agriculture: iot-enabled greenhouse for plant growth control. *IEEE Pervasive Comput.* 17, 65–75. doi: 10.1109/MPRV.2018.2873849

Sullivan, G. J., Ohm, J.-R., Han, W.-J., and Wiegand, T. (2012). Overview of the high efficiency video coding (hevc) standard. *IEEE Trans. Circuits syst. Video Technol.* 22, 1649–1668. doi: 10.1109/TCSVT.2012.2221191

Tachella, J., Altmann, Y., Márquez, M., Arguello-Fuentes, H., Tourneret, J.-Y., and McLaughlin, S. (2020). Bayesian 3d reconstruction of subsampled multispectral single-photon lidar signals. *IEEE Trans. Comput. Imag.* 6, 208–220. doi: 10.1109/TCI.2019.2945204

Taimori, A., and Marvasti, F. (2018). Adaptive sparse image sampling and recovery. *IEEE Trans. Comput. Imag.* 4, 311–325. doi: 10.1109/TCI.2018.2833625

Tramel, E. W., and Fowler, J. E. (2011). "Video compressed sensing with multihypothesis," in *2011 Data Compression Conference* (Snowbird, UT), 193–202.

Tran, D. T., Yamaç, M., Degerli, A., Gabbouj, M., and Iosifidis, A. (2021). Multilinear compressive learning. *IEEE Trans. Neural Netw. Learn. Syst.* 32, 1512–1524. doi: 10.1109/TNNLS.2020.2984831

Trevisi, M., Akbari, A., Trocan, M., Rodríguez-Vázquez, A., and Carmona-Galán, R. (2020). Compressive imaging using rip-compliant cmos imager architecture and landweber reconstruction. *IEEE Trans. Circuits Syst. Video Technol.* 30, 387–399. doi: 10.1109/TCSVT.2019.2892178

Unde, A. S., and Pattathil, D. P. (2020). Adaptive compressive video coding for embedded camera sensors: compressed domain motion and measurements estimation. *IEEE Trans. Mob. Comput.* 19, 2250–2263. doi: 10.1109/TMC.2019.2926271

Yang, Y., Sun, J., Li, H., and Xu, Z. (2020). Admm-csnet: a deep learning approach for image compressive sensing. *IEEE Trans. Pattern Anal. Mach. Intell.* 42, 521–538. doi: 10.1109/TPAMI.2018.2883941

Yu, Y., Wang, B., and Zhang, L. (2010). Saliency-based compressive sampling for image signals. *IEEE Signal Process. Lett.* 17, 973–976. doi: 10.1109/LSP.2010.2080673

Zammit, J., and Wassell, I. J. (2020). Adaptive block compressive sensing: Toward a real-time and low-complexity implementation. *IEEE Access* 8, 120999–121013. doi: 10.1109/ACCESS.2020.3006861

Zhang, J., Zhao, D., and Jiang, F. (2013). "Spatially directional predictive coding for block-based compressive sensing of natural images," in *2013 IEEE International Conference on Image Processing* (Melbourne, VIC), 1021–1025.

Zhang, M., Wang, X., Chen, X., and Zhang, A. (2018). The kernel conjugate gradient algorithms. *IEEE Trans. Signal Process.* 66, 4377–4387. doi: 10.1109/TSP.2018.2853109

Zhang, P., Gan, L., Sun, S., and Ling, C. (2015). Modulated unit-norm tight frames for compressed sensing. *IEEE Trans. Signal Process.* 63, 3974–3985. doi: 10.1109/TSP.2015.2425809

Zhang, R., Wu, S., Wang, Y., and Jiao, J. (2020). High-performance distributed compressive video sensing: Jointly exploiting the hevc motion estimation and the ℓ_1-ℓ_1 reconstruction. *IEEE Access* 8, 31306–31316. doi: 10.1109/ACCESS.2020.2973392

Zhao, Z., Xie, X., Liu, W., and Pan, Q. (2020). A hybrid-3d convolutional network for video compressive sensing. *IEEE Access* 8, 20503–20513. doi: 10.1109/ACCESS.2020.2969290

Zhen, C., De-rong, C., and Jiu-lu, G. (2020). "A deep learning based distributed compressive video sensing reconstruction algorithm for small reconnaissance uav," in *2020 3rd International Conference on Unmanned Systems (ICUS)* (Harbin), 668–672.

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.