Sensor Applications

Compressive Sensing for Remote Flood Monitoring

Vahid Abolghasemi[1*], and Mohammad Hossein Anisi[1*]

[1] School of Computer Science and Electronic Engineering, University of Essex, Colchester, CO4 3SQ, UK
[*] Senior Member, IEEE

Abstract—Although wireless sensor networks (WSNs) are considered as one of the prominent solutions for flood monitoring; however, the energy constraint nature of the sensors is still a technical challenge. In this paper, we tackle this problem by proposing a novel energy-efficient remote flood monitoring system, enabled by compressive sensing. The proposed approach compressively captures water level data using; *i*) a random block-based sampler, and *ii*) a gradient-based compressive sensing approach, at a very low rate, exploiting water level data variability over time. Through extensive experiments on real water-level dataset, we show that the number of packet transmissions as well as the size of packets are significantly reduced. The results also demonstrate significant energy reduction in sensing and transmission. Moreover, data reconstruction from compressed samples are of high quality with negligible degradation, compared to classic compression techniques, even at high compression rates.

Index Terms—Compressive sensing, Sparse recovery, Energy efficiency, Wireless sensor network, Water level, Remote monitoring.

I. INTRODUCTION

Current flood warning analysis which relies on water-level sensors and precipitation forecasts is not capable of providing near-real-time and automated flood monitoring. In this regard, wireless sensor network (WSN) as one of the key enabling technologies of Internet of Things (IoT), is one of the most efficient technologies that could be used for remote and real-time data collection from rivers. With WSN, several sensors are connected with each other and deployed in the side of the river. This not only lets the managers to understand any sudden changes in the river but also let them to generate a large database of river water levels that could be used to predict the upcoming floods [1]. However, continuous and real-time monitoring of the river requires frequent transmission of data which drains the limited energy of the sensors.

WSN has been used in several flood monitoring applications as an efficient technology for remote data collection [2]–[7]. For instance, authors in [4] proposed an architecture for a flash flood alert system. The system consists of a WSN and a sink node located close to the WSN to collect it. Periodically, or in case of abnormal conditions, the sink sends information via cellular network to a command center. The work in [8] describes the implementation of a river monitoring system using an event-based component model for WSN. It monitors flood, pollution and human tampering and warns potential stakeholders whenever they are at risk. In [9], a real-time flood monitoring and warning system has been presented for a selected area of the southern part of Thailand. In this system, water condition is monitored remotely using a WSN that transmits the required data to an application sever via General Packet Radio Service (GPRS). However, mitigating the number of redundant packets to reduce the energy consumption of the sensor nodes has been remained a challenge [10].

Recently, compressive sensing (CS) [11] has unveiled promising performance in multimedia data transmission [12] and IoT applications by offering low sampling rate [13]. Since CS offers data acquisition from the filed of interest with sub-Nyquist sampling rate, it can significantly mitigate transmission and power consumption costs of IoT systems. CS can also be deployed on an embedded system within an intermediate layer for the purpose of local processing of received data from sensors [14]. Therefore, data aggregation, storing, and reconstruction of original data with the aim of extraction of salient information and important features can be performed on the cloud which hosts the application layer. Moreover, CS has shown to
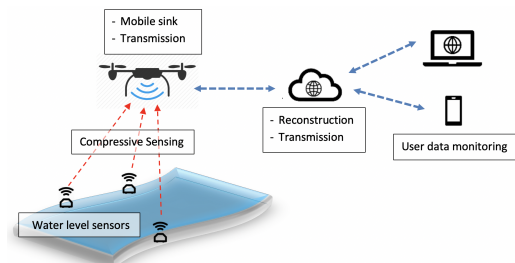

Fig. 1.   Layout of an exemplary CS-based remote flood monitoring.

provide higher compression rates and better reconstruction quality [15], compared to traditional methods e.g. based on Hadamard or Wavelet transform. Despite such great advantages, limited applications of CS-based IoT/WSN systems have been reported in the literature. Most existing works that combine CS with WSN focus on general methodological perspective rather than application-specific criteria. Among those few, a cost-effective compressive sensing approach was proposed for wireless transmission of fetal Electrocardiogram (ECG) and Electroencephalogram (EEG) signals [16], [17]. A distributed adaptive CS is proposed in [18] for complex meteorological data with spatial and temporal correlation. A comprehensive report on use of CS for WSN with detailed quantitative analysis with popular sensors can be found in [19].

In this paper, considering that power management is one of the main issues in WSNs and data transmission is the most energy consuming task in a sensor node, we reduce the number of transmissions based on CS technique. To the best of our knowledge, this is the first work that addresses compressive sensing for wireless monitoring of flood events. Taking into account the slow-variation of water levels, we propose two novel methods by utilising a block structure and Gradient operator to compressively acquire data samples. Figure 1 illustrates the proposed topology where low-cost water level sensors are deployed across the river of interest. In this layout, data collection can be done using a drone that approaches the sensor nodes and acts as a mobile sink to reduce transmission distance and save more energy [20]. Therefore, only limited number of packets, filtered by CS, are transmitted to the drone from a close distance that greatly reduce the power consumption and extend the network lifetime.

II. PROPOSED APPROACH

Let us assume that a stream of water level samples from L sensor nodes, deployed in the rivers, are to be compressively gathered at the gateway (mobile sink in Figure 1). This can be achieved by using a

Corresponding author: V. Abolghasemi (e-mail: v.abolghasemi@essex.ac.uk).

block-based strategy [21]. In this approach, we assume that original blocks (segments) acquired by the sensor nodes, within a certain period of time, is composed of n discrete samples, represented by:

$$\mathbf{x}_l = [x_l(0), x_l(1), \ldots, x_l(n-1)]^\top \tag{1}$$

where $l \in [1, 2, \ldots, L]$ denotes the l-th sensor node, L is the total number of nodes, and $[\cdot]^\top$ denotes the transpose operator. These samples, however, are not to be sent to the sink. Instead, a compressed version of segments with size $m << n$ will be received by the sink. Here, the compressive sensing problem can be mathematically expressed as:

$$\mathbf{y}_l = \mathbf{\Phi}\mathbf{x}_l + \mathbf{v}_l \tag{2}$$

where $\mathbf{y}_l \in \mathbb{R}^m$ is the measurement vector, and $\mathbf{\Phi} \in \mathbb{R}^{m \times n}$ is called the measurement matrix (sensing matrix). Measurement noise is represented by vector $\mathbf{v}_l$ which is incurred during the compression procedure. The measurement matrix $\mathbf{\Phi}$ plays a crucial role in both compression and reconstruction phases. It should be designed to provide high compression ratio (i.e. $m << n$) and at the same time guarantee recovery of $\mathbf{x}_l$ from measurements $\mathbf{y}_l$. On the other hand, successful reconstruction relies on the key assumption of *sparsity* on $\mathbf{x}_l$, meaning that most entries are zero. This assumption does not always hold in the primary domain (e.g. time), thus, one may need to transform the data into a different domain (e.g. frequency). It has been shown that problem (2) still holds if $\mathbf{x}_l$ can be made sparse by transformation into a different domain, e.g. Discrete Cosine Transform (DCT). This condition can turn equation (2) into:

$$\mathbf{y}_l = \mathbf{\Phi}\mathbf{\Psi}\mathbf{s}_l \tag{3}$$

where $\mathbf{\Psi}$ is an $n \times n$ matrix of orthogonal basis, and $\mathbf{s}_l \in \mathbb{R}^n$ is the sparse coefficients vector associated to $\mathbf{x}_l$. After receiving the compressed vectors at the sink, i.e. $\{\mathbf{y}_1, \mathbf{y}_2, \ldots, \mathbf{y}_l\}$, they are transmitted to the cloud computer for further processing whereby the original data, i.e. $\{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_l\}$, is reconstructed (Figure 1). In a typical send-and-receive scenario, as proposed in this paper, the measurements $\mathbf{y}_l$ are frequently transmitted to the receiver, whereas $\mathbf{\Phi}$ and $\mathbf{\Psi}$ are known to the receiver. At the receiver, $\mathbf{x}_l$ (or $\mathbf{s}_l$, equivalently) is to be reconstructed from $\mathbf{y}_l$, $\mathbf{\Phi}$ and $\mathbf{\Psi}$. The reconstruction stage does not consume any energy of the WSN. Numerous techniques have been proposed in the literature for such reconstruction, the most famous ones are Basis Pursuit (BP) [22] and Orthogonal Matching Pursuit (OMP) [23]. However, not all methods are efficient and one needs to design techniques depending on the data of interest [24]. As mentioned above, finding sparse representation of the data of interest is crucial for achieving a successful CS system. In what follows, we propose two solutions for this problem: i) using block sparse Bayesian learning model in WSN (BSBL–WSN); and ii) proposing a novel method, termed gradient compressive sensing WSN (GCS–WSN), for compression and transmission of water level samples.

A. BSBL–WSN

Blocks of compressed measurements via matrix multiplication, i.e. (2), will be transmitted to the cloud server for further processing. This process is called CS encoding which can be performed entirely in the analog domain at the sensor nodes. In other words, vector $\mathbf{y}_l$ can be directly acquired by a Random Demodulator (RD) module operating at a sub-Nyquist sampling rate. As this process is to be implemented for data blocks, we have found it very close to a recent work presented by Zhang et al. [17] for telemonitoring of physiological data through a wireless body-area network. Here, we utilise this method for sparse representation of water level data for efficient transmission through a wireless sensor network. Zhang et al. [17] proposed a method that can be efficiently used for compression and reconstruction of

EEG data. They initially considered ECG signals which naturally have block structure [16], but later showed that such assumption can be valid for arbitrary waveform such as EEG without a distinct block structure [17]. This motivated us to use the same concept for water level data too. In our proposed block sparse Bayesian learning method, BSBL–WSN, the input signal $\mathbf{x}_l$ is partitioned into concatenation of several non-overlapping blocks as shown in (1). By removing the baseline from each segment (i.e. removing the offset), we can observe that only a few blocks are non-sparse among these blocks. This is due to the fact that water level is normally stable in normal conditions and sudden changes occur mainly in case of flooding. This behaviour allows us to model the water level data in form of a block-sparse data. We model the block-sparse signal $\mathbf{x}_l$ as a parameterised multivariate Gaussian distribution:

$$p(\mathbf{x}_l; \gamma_l, \mathbf{C}_l) \sim \mathcal{N}(\mathbf{0}, \gamma_l \mathbf{C}_l), \ l = 1, \cdots, L. \tag{4}$$

where γ_l is a non-negative parameter controlling the block-sparsity of $\mathbf{x}_l$. Also, $\mathbf{C}_l \in \mathbb{R}^{n \times n}$ is a positive-definite matrix to model the correlation structure of the corresponding block. The posterior of $\mathbf{x}$ can then be obtained via [17]:

$$p(\mathbf{x}_l | \mathbf{y}_l; \lambda, \gamma_l, \mathbf{C}_l) = \mathcal{N}(\mu_x, \mathbf{\Sigma}_x), \tag{5}$$

with
$$\mu_x = \mathbf{\Sigma}_0 \mathbf{\Phi}^T \left(\lambda \mathbf{I} + \mathbf{\Phi}\mathbf{\Sigma}_0 \mathbf{\Phi}^T\right)^{-1} \mathbf{y}, \mathbf{\Sigma}_x = \left(\mathbf{\Sigma}_0^{-1} + \frac{1}{\lambda}\mathbf{\Phi}^T \mathbf{\Phi}\right)^{-1}$$

In order to estimate the original vector $\mathbf{x}_l$, we first need to find the parameters λ (non-negative regularisation scalar), γ_l, $\mathbf{C}_l$. It has been shown that these parameters can be estimated by a using a Type II maximum likelihood procedure [17]. Therefore, we follow the Expectation Maximization (EM) method and define these learning rules:

$$\gamma_l \leftarrow \frac{1}{n}\mathrm{Tr}[\mathbf{C}_l^{-1}(\mathbf{\Sigma}_x + \mu_x \mu_x^T)] \tag{6}$$

$$\lambda \leftarrow \frac{||\mathbf{y}_l - \mathbf{\Phi}\mu_x||_2^2 + \mathrm{Tr}(\mathbf{\Sigma}_x \mathbf{\Phi}^T \mathbf{\Phi})}{m} \tag{7}$$

The above learning-based rules can estimate the model parameters, and finally the vector $\mathbf{x}_l$ can be estimated using the Maximum-A-Posterior (MAP):

$$\hat{\mathbf{x}}_l \leftarrow \mathbf{\Sigma}_0 \mathbf{\Phi}^T \left(\lambda \mathbf{I} + \mathbf{\Phi}\mathbf{\Sigma}_0 \mathbf{\Phi}^T\right)^{-1} \mathbf{y}_l. \tag{8}$$

In general, various structures such as random binary or random Gaussian distribution have been used for measurement matrix in the literature. However, there are evidences that using *sparse binary matrix*, which are much easier to implement in embedded systems, can lead to more energy-efficient compression compared to other structures [18]. Therefore, we consider random binary $\mathbf{\Phi}$, and inverse DCT dictionary $\mathbf{\Psi}$.

B. GCS–WSN

We will see later in Section III that the proposed BSBL–WSN performs very accurately and can recover the original samples on the cloud. However, the reconstruction process explained above is not as simple as traditional methods such as BP [22] or OMP [23]. In this section, our aim is to propose a very simple CS techniques for water level data so that traditional reconstruction methods can be applied too. Naturally, water level data has small fluctuations, and sharp rises occur only in case of flooding events. In fact, crucial information exist mainly at few (sparse) transition points. In order to model these changes in a mathematical form, we propose to use Gradient (derivative) function which measures the variations of a function. In other words, it can *sparsify* our data as it is a requirement for successful implementation of CS. The process of converting water level data into its sparse counterpart can be simply implemented by
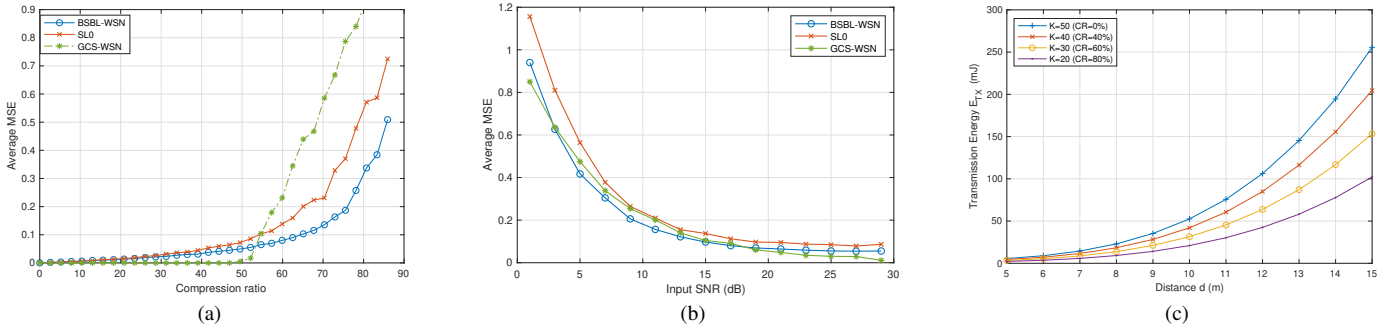
Fig. 2. Performance evaluation of the proposed approach: Average reconstruction error between original data and reconstructed data vs (a) various compression rates and (b) various input SNRs (dB); (c) Transmission energy vs Transmission distance at various compression rates.
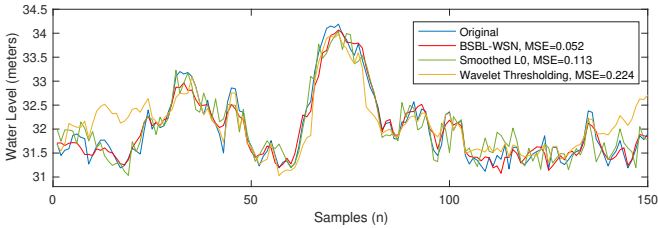

Fig. 3. Example of water level data reconstructed from 50% compressed samples using two CS-based methods (BSBL and SL0), and one classic compression technique, i.e., Wavelet Thresholding.

an analog differentiator at the sensor nodes. Here, we mathematically define the gradient of $\mathbf{x}_l$ as:

$$\mathbf{x}'_l(t) = \nabla \mathbf{x}_l(t) = \frac{\partial \mathbf{x}_l(t)}{\partial t} \tag{9}$$

where t is the time variable which is replaced by n after acquisition process, and $\mathbf{x}'_l$ is the sparse vector. Now, the compressive sensing problem can be re-defined and directly applied to $\mathbf{x}'_l$ via $\mathbf{y}'_l = \mathbf{\Phi}\mathbf{x}'_l$. We call the proposed approach as Gradient Compressive Sensing (GCS). After obtaining the compressed measurements $\mathbf{y}'_l$ at the gateway, they are sent to the cloud server where the original signal, i.e., $\mathbf{x}'_l$, is recovered by a CS algorithm using the shared measurement matrix $\mathbf{\Phi}$. Most common sparse recovery methods such as BP [22], or OMP [23], can be used for this purpose. The recovered data encompasses the information of water level variations. However, if needed, the integral of $\mathbf{x}'_l$ can be calculated to yield $\mathbf{x}_l$.

III. EXPERIMENTAL RESULTS

Extensive experiments were conducted to evaluate the performance of the proposed methods with real water level dataset retrieved from an online flood information system in Malaysia[1]. The dataset includes water level samples captured from 15 sensor nodes deployed on various rivers. There are 150 data points for a 24-hour period.

A. CS performance

An example of 150-point water level data are first compressed using random binary matrix of size 70×150 (equivalent to 50% compression ratio). Then, the the proposed BSBL–WSN method is applied to the compressed sampled to reconstruct the original data points. For comparison, we also applied Smoothed–ℓ_0 (SL0) [25] and classic Wavelet thresholding to reconstruct the original samples. SL0 uses a smooth measure of ℓ_0-norm to recover sparsest vector from the compressed measurements. Figure 3 shows the results of this experiment along with the obtained Mean Square Error (MSE). It is clearly observed that the original signal has been successfully

[1]http://infobanjir.water.gov.my/real_time.cfm

recovered where BSBL–WSN outperforms both SL0 and Wavelet thresholding, with a much smaller reconstruction error.

In order to evaluate the robustness of the proposed approaches, we conducted an experiment to reconstruct all water level data at various compression ratios. Figure 2 (a) illustrates the average reconstruction performance of original flood data points from compressed measurements at various compression rates. In this experiment, 50 trials have been conducted at each compression rate, and random binary sensing matrix (measurement matrix) have been used. As observed from Figure 2 (a), the average MSE is acceptable for up to around 75% compression rate using the proposed BSBL–WSN. It is also observed that BSBL outperforms SL0 as expected. Interestingly, Figure 2 (a) reveals that the proposed GCS–WSN provides greater reconstruction performance among all other methods in the graph. However, such performance significantly drops at 50% onward where average MSE sharply increases. The reason of this significant drop could be due to weakness of BP method (used for sparse reconstruction) to recover highly compressed samples.

Next, the robustness of the proposed methods against noisy inputs is assessed. To do this, White Gaussian Noise (WGN) is added to the original samples with different Signal-to-Noise-ratio (SNR) levels. Then, compressive sensing is applied and the average reconstruction errors over 50 trials are recorded. The compression ratio is kept at 50% in all trials. Figure 2 (b) presents average MSEs versus varying input SNR from 1 to 30 dB. A monotonic reduction in MSE is observed for all the methods where the proposed methods, i.e., BSBL–WSN and GCS–WSN outperform SL0.

B. Energy efficiency

In the proposed method, unlike the conventional WSN, instead of transmitting each captured sample, we transmit a number of samples in a compressed packet. So that, the senor nodes can save considerable amount of energy by transmitting much less number of packets [26]. We discuss the energy model used and show how the compression techniques have reduced the energy consumption of the nodes. Let us denote the required energy for sampling, computation, and communication by E_{sm}, E_{cmp}, E_{cmm}, respectively. In order to calculate the energy saving of sampling as a result of applying CS, we use the following equation [19]:

$$E_{saving} \approx \left(\frac{n-m}{n}\right)(E_{sm} + E_{cmm} + E_{cmp}). \tag{10}$$

Recall that n and m are number of samples before and after compression, respectively (i.e. $m < n$). We have assumed transmission distance between the sensor nodes and the mobile sink to be 10 meters. In addition, normalized sampling and computation energies with respect to communication energy are calculated as $E_{sm} = 4.2 \times 10^{-4}$ and $E_{cmp} = 0.11$, respectively [19]. Note that both BSBL and GCS methods consume the same amount of sensing energy as they follow

similar sampling procedures. Table 1 provides the results of saved energy using Eq. (10) at different compression rates. As observed from this table, when the compression rate is increased, the amount of energy saving becomes greater, correspondingly. However, there should be a compromise between these two factors with respect to reconstruction error. For instance, one can go for up to 70% compression on the sensors if BSBL–WSN is used for reconstruction (with MSE = 0.1440). Using GCS–WSN at this rate and beyond is not recommended as the associated MSE is not acceptable (MSE = 0.5852). In addition to the advantages of CS in reducing the energy consumed for sensing, considerable power to transmit the samples is saved. In fact, CS allows us to combine a number of packets into one packet and reduce the number of transmissions. Although, the size of a compressed packet is larger than a packet with a single sample but because of transmitting much less number of packets, sensor nodes consume less energy in transmitter ($\mathcal{E}_{elec}$) and spend low power in transmitter amplifier (E_{amp}) to achieve an acceptable signal noise ratio. The energy used for transmitting a packet has been calculated as follows [27]:

$$E_{TX}(K, d) = \mathcal{E}_{elec}K + \mathcal{E}_{amp}Kd^4 \qquad (11)$$

where K is the number of bits to be transmitted in each packet, d is the distance between a sensor node and the sink. We assume a two-ray ground model where $\mathcal{E}_{elec}$ = 50 nJ/bit, and $\mathcal{E}_{amp}$ = 100 pJ/bit/m^2. Figure 2 (c) depicts the transmission energy E_{TX} (11) versus distances within $d = [5\ 15]$ meters for various compression rates. As observed from this figure, for longer distances more energy is required as expected. Also, increasing the compression ratio (equivalent to decreasing K) can significantly reduce the transmission costs compared to the no-compression case (i.e. CR = 0).

TABLE 1. Energy saving performance at various compression rates.

m, n	CR (%)	E_{saving}	MSE BSBL	MSE GCS
50, 50	0	0	0.0076	7×10^{-20}
45, 50	10	0.11	0.0074	2×10^{-10}
35, 50	30	0.33	0.0208	1×10^{-9}
25, 50	50	0.55	0.0435	0.0050
15, 50	70	0.77	0.1440	0.5852
10, 50	80	0.88	0.2613	0.9326

IV. CONCLUSIONS

In this paper, the problem of energy-efficient water level data transmission in WSN was addressed. The research focuses on reducing the number of packet transmissions using compressive sensing as a promising resolution to conserve the energy of the sensor nodes. Two different methods, based on block-sparse modeling and Gradient function, were proposed to capture the input samples in a compressed fashion. Our experiments have shown significant reduction of energy consumption during data acquisition. Notably, we observed that the proposed techniques allow accurate and robust reconstruction of original samples at the receiver side (i.e. cloud).

REFERENCES

[1] S. A. Soleymani, S. Goudarzi, M. H. Anisi, W. H. Hassan, M. Y. I. Idris, S. Shamshirband, N. M. Noor, and I. Ahmedy, "A novel method to water level prediction using rbf and ffa," *Water Resources Management*, vol. 30, no. 9, pp. 3265–3283, 2016.

[2] R. Chaudhry, S. Tapaswi, and N. Kumar, "A green multicast routing algorithm for smart sensor networks in disaster management," *IEEE Trans. on Green Comm. and Net.*, vol. 3, no. 1, pp. 215–226, 2019.

[3] M. L. Silva, L. N. S. Júnior, A. L. L. Aquino, and J. d. C. Lima, "Jsensor: A parallel simulator for huge wireless sensor networks applications," *IEEE Trans. on Par. and Dist. Sys.*, vol. 30, no. 10, pp. 2296–2308, 2019.

[4] M. Castillo Effer, D. H. Quintela, W. Moreno, R. Jordan, and W. Westhoff, "Wireless sensor networks for flash-flood alerting," in *Proceedings of the Fifth IEEE Int. Caracas Conf. on Devices, Circuits and Systems, 2004.*, vol. 1, 2004, pp. 142–146.

[5] M. Abdullah, "Simulation of wireless sensor network for flood monitoring system," in *Design, User Experience, and Usability. User Experience Design for Everyday Life Applications and Services*, A. Marcus, Ed. Cham: Springer International Publishing, 2014, pp. 255–264.

[6] T. M. Thekkil and N. Prabakaran, "Real-time wsn based early flood detection and control monitoring system," in *International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT)*, 2017, pp. 1709–1713.

[7] R. Marin-Perez, J. Garcia-Pintado, and A. S. Gomez, "A real-time measurement system for long-life flood monitoring and warning applications," *Sensors*, vol. 12, no. 4, pp. 4213–4236, 2012.

[8] J. Ueyama, D. Hughes, N. Matthys, W. Horre, S. Michiels, C. Huygens, and W. Joosen, "An event-based component model for sensor networks: a case study for river monitoring," in *SBRC2010 Conf.*, 2010.

[9] J. Sunkpho and C. Ootamakorn, "Real-time flood monitoring and warning system," *Journal of Science and Technology*, vol. 33, no. 2, pp. 227–235, 2011.

[10] S. O. Olatinwo and T. Joubert, "Energy efficient solutions in wireless sensor systems for water quality monitoring: A review," *IEEE Sensors Journal*, vol. 19, no. 5, pp. 1596–1625, 2019.

[11] D. L. Donoho, "Compressed sensing," *IEEE Trans. on Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, 2006.

[12] A. Bose and S. P. Maity, "Spread spectrum watermark detection on degraded compressed sensing," *IEEE Sensors Letters*, vol. 1, no. 5, pp. 1–4, 2017.

[13] H. Djelouat, A. Amira, and F. Bensaali, "Compressive sensing-based iot applications: A review," *J. of Sensor and Actuator Networks*, vol. 7, no. 4, p. 45, 2018.

[14] M. Skhiri, S. Bdiri, and F. Derbel, "Power aware wireless sensor networks based on compressive sensing," in *2018 IEEE Int. Inst. and Measurement Technology Conference (I2MTC)*, 2018, pp. 1–5.

[15] X. Yuan and R. Haimi-Cohen, "Image compression based on compressive sensing: End-to-end comparison with jpeg," *IEEE Transactions on Multimedia*, vol. 22, no. 11, pp. 2889–2904, 2020.

[16] Z. Zhang, T. Jung, S. Makeig, and B. D. Rao, "Compressed sensing for energy-efficient wireless telemonitoring of noninvasive fetal ECG via block sparse bayesian learning," *IEEE Trans. on Biomed. Eng.*, vol. 60, no. 2, pp. 300–309, 2013.

[17] Z. Zhang', T. Jung, S. Makeig, and B. D. Rao, "Compressed sensing of EEG for wireless telemonitoring with low energy consumption and inexpensive hardware," *IEEE Trans. on Biomed. Eng.*, vol. 60, no. 1, pp. 221–224, 2013.

[18] Z. Chen, J. Ranieri, R. Zhang, and M. Vetterli, "DASS: Distributed adaptive sparse sensing," *IEEE Trans. on Wireless Comm.*, vol. 14, no. 5, pp. 2571–2583, 2015.

[19] M. Razzaque and S. Dobson, "Energy-efficient sensing in wireless sensor networks using compressed sensing," *Sensors*, vol. 14, no. 2, pp. 2822–2859, Feb. 2014.

[20] S. Goudarzi, N. Kama, M. H. Anisi, S. Zeadally, and S. Mumtaz, "Data collection using unmanned aerial vehicles for internet of things platforms," *Computers & Electrical Engineering*, vol. 75, pp. 1–15, 2019.

[21] V. Abolghasemi, S. Sanei, S. Ferdowsi, F. Ghaderi, and A. Belcher, "Segmented compressive sensing," in *2009 IEEE/SP 15th Workshop on Statistical Signal Processing*, 2009, pp. 630–633.

[22] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM Rev.*, vol. 43, no. 1, p. 129–159, Jan. 2001.

[23] J. A. Tropp, "Greed is good: algorithmic results for sparse approximation," *IEEE Trans. on Inf. Theory*, vol. 50, no. 10, pp. 2231–2242, 2004.

[24] E. Crespo Marques, N. Maciel, L. Naviner, H. Cai, and J. Yang, "A review of sparse recovery algorithms," *IEEE Access*, vol. 7, pp. 1300–1322, 2019.

[25] H. Mohimani, M. Babaie-Zadeh, and C. Jutten, "A fast approach for overcomplete sparse decomposition based on smoothed ℓ_0 norm," *IEEE Trans. on Sig. Proc.*, vol. 57, no. 1, pp. 289–301, January 2009.

[26] S. S. Anjum, R. M. Noor, M. H. Anisi, I. B. Ahmedy, F. Othman, M. Alam, and M. K. Khan, "Energy management in rfid-sensor networks: Taxonomy and challenges," *IEEE Internet of Things Journal*, vol. 6, no. 1, pp. 250–266, 2019.

[27] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Trans. on Wireless Comm.*, vol. 1, no. 4, pp. 660–670, 2002.