

Article

A DH-KSVD Algorithm for Efficient Compression of Shock Wave Data

Jiarong Liu ^{1,2}, Yonghong Ding ^{1,2,*} and Wenbin You ^{1,*}

¹ State Key Laboratory of Extreme Environment Optoelectronic Dynamic Measurement Technology and Instrument, North University of China, Taiyuan 030051, China; s202405070@st.nuc.edu.cn

² School of Information and Communication Engineering, North University of China, Taiyuan 030051, China

* Correspondence: dingyonghong@nuc.edu.cn (Y.D.); youwenbin@nuc.edu.cn (W.Y.)

Abstract

To address low training efficiency and poor reconstruction in traditional K Singular Value Decomposition (KSVD) for compressive sensing of shock wave signals, this study proposes an improved algorithm, DH-KSVD, integrating dynamic pruning and hybrid coding. The dynamic pruning mechanism eliminates redundant atoms according to their contributions and adaptive thresholds, while incorporating residual features to enhance dictionary compactness and training efficiency. The hybrid sparse constraint integrates the sparsity of ℓ_0 -Orthogonal Matching Pursuit (OMP) with the noise robustness of ℓ_1 -Least Absolute Shrinkage and Selection Operator (LASSO), dynamically adjusting their relative weights to enhance both coding quality and reconstruction stability. Experiments on typical shock wave datasets show that, compared with Discrete Cosine Transform (DCT), KSVD, and feature-based segmented dictionary methods (termed CC-KSVD), DH-KSVD reduces average training time by 46.4%, 31%, and 13.7%, respectively. At a Compression Ratio (CR) of 0.7, the Root Mean Square Error (RMSE) decreases by 67.1%, 65.7%, and 36.2%, while the Peak Signal-to-Noise Ratio (PSNR) increases by 35.5%, 39.8%, and 11.8%, respectively. The proposed algorithm markedly improves training efficiency and achieves lower RMSE and higher PSNR under high compression ratios, providing an effective solution for compressing long-duration, transient shock wave signals.

Keywords: shock wave signal; compressive sensing (CS); k singular value decomposition (KSVD); orthogonal matching pursuit (OMP); least absolute shrinkage and selection operator (LASSO)



Academic Editor: Luis Javier García Villalba

Received: 1 September 2025

Revised: 28 September 2025

Accepted: 30 September 2025

Published: 1 October 2025

Citation: Liu, J.; Ding, Y.; You, W. A DH-KSVD Algorithm for Efficient Compression of Shock Wave Data. *Appl. Sci.* **2025**, *15*, 10640. <https://doi.org/10.3390/app151910640>

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The pressure of shock waves generated by explosions constitutes a critical parameter for evaluating the destructive potential of weapons; consequently, precise measurement of these shock waves is of considerable importance [1,2]. Shock wave signals are characterized by transience, pronounced non-stationarity, and substantial energy fluctuations [3,4], thereby necessitating sampling rates in the megahertz range to faithfully capture critical physical features including the rising edge, peak, and decay. In specialized contexts such as maritime destructive experiments, a single explosion lasting only a few seconds can yield millions to tens of millions of data points, while multi-channel synchronous monitoring further amplifies the exponential increase in data volume. Moreover, in light of the inherent limitations of wireless transmission bandwidth at sea and the potential hazard of rapid vessel sinking, efficient data compression emerges as an indispensable prerequisite.

Current data compression methods are primarily categorized into two types: lossless compression and lossy compression [5]. The selection between these methods should be determined according to the specific application context, balancing fidelity with compression efficiency. Presently, widely utilized lossless compression algorithms, including Huffman coding [6], arithmetic coding [7], and Lempel–Ziv–Welch (LZW) dictionary coding [8], are capable of reducing data volume without any loss of original information. However, they are primarily based on modeling redundant characters or symbol frequencies and are suitable for data with stable structures or discrete values. For continuous real-valued data, such as shock wave signals, which exhibit significant non-stationarity, rapid fluctuations, and intricate spectral structures, lossless compression algorithms frequently encounter difficulties in attaining optimal compression. In contrast, lossy compression methods that focus on feature preservation exhibit greater potential. Among these, Compressive Sensing (CS) theory [9,10] offers a novel framework for efficiently reconstructing signals at sampling rates below the Nyquist rate, assuming that the signal possesses a sparse representation in a specific sparse basis or dictionary. The reconstruction performance is highly contingent upon the quality of the sparse representation basis design. Early sparse dictionaries primarily employed analytical construction methods, such as Gabor functions [11] and wavelet bases [12], which utilize fixed basis function libraries, while these methods offer high computational efficiency, their predefined basis functions constrain their capacity to effectively capture the complex feature structures inherent in shock wave signals.

To tackle this challenge, researchers have proposed adaptive dictionary learning approaches, among which the K Singular Value Decomposition (KSVD) algorithm is the most representative. This approach iteratively optimizes dictionary atoms and sparse coefficients, allowing the dictionary to adaptively capture the dominant feature patterns within training samples. It has been extensively applied in domains such as image classification [13,14] and video denoising [15–17]. Although the KSVD algorithm demonstrates strong performance in sparse representation, it remains constrained by high computational complexity and limited robustness. The existing studies have predominantly concentrated on the sparse coding stage. For instance, reference [18] proposed an adaptive sparsity estimation strategy, whereas reference [19] utilized segmented orthogonal matching pursuit to accelerate the solution of underdetermined equations, thereby enhancing training efficiency. However, these approaches exhibit unstable sparsity estimation in high-noise environments and are typically confined to optimizing the coding process, while neglecting the joint optimization of the dictionary update stage. Consequently, achieving a balance among sparsity, training efficiency, and reconstruction accuracy remains challenging. Reference [20] enhanced discriminative capability by jointly optimizing dictionary representation capabilities and classifier performance, but its computational complexity is high and it is highly dependent on labels. Reference [21] proposed a method based on signal feature segmentation training and dictionary concatenation, which reduced dictionary size and training time via classification learning; nevertheless, its performance relies heavily on prior sparsity knowledge and classification accuracy. In high-noise environments, sparsity estimation is unstable, affecting reconstruction quality.

The main contributions of this paper are summarized as follows:

1. A dynamic pruning mechanism is introduced to evaluate the importance of atoms based on their contribution. By adaptively removing redundant atoms while supplementing residual features, the compactness of the dictionary is enhanced and training efficiency is improved.
2. A hybrid sparse coding strategy is designed to integrate the strict sparsity of ℓ_0 -based Orthogonal Matching Pursuit (OMP) with the noise robustness of ℓ_1 -based Least

Absolute Shrinkage and Selection Operator (LASSO), dynamically balancing their weights to improve coding quality and reconstruction stability.

3. The proposed DH-KSVD algorithm achieves superior performance in both training efficiency and reconstruction accuracy under more aggressive compression, especially in noisy environments, offering an effective solution for transient signal compression such as shock waves.

The remainder of this paper is organized as follows: Section 2 presents the theoretical background, including the characteristics of shock wave signals and the fundamentals of CS and KSVD. Section 3 details the proposed DH-KSVD algorithm, including the dynamic pruning mechanism, hybrid sparse constraint strategy, and its integration into the CS framework. Section 4 provides experimental results and comparative analysis, including ablation studies and performance evaluation under various conditions. Section 5 discusses the advantages and limitations of the proposed method, along with insights into potential future research directions. Finally, Section 6 concludes the paper.

2. Theoretical Analysis

2.1. Shock Wave Characteristics

Shock wave signals constitute a representative class of transient, nonstationary signals, typically generated by violent events such as explosions, high-speed impacts, supersonic flight, and structural damage. These signals exhibit distinctive physical properties and temporal characteristics. In the time domain, shock wave signals manifest abrupt, short-lived high-amplitude responses, with waveforms characterized by steep rising edges, extremely brief durations, and rapidly decaying tails, thereby producing a strongly nonstationary structure [22]. In the frequency domain, these signals encompass abundant high-frequency components, with a broad spectral range typically extending from tens of hertz to several hundreds of kilohertz. Additionally, such signals are often accompanied by nonlinear effects and noise disturbances, exhibiting high complexity. To accurately capture the fine-grained characteristics of shock waves, engineering practices commonly employ oversampling strategies with sampling frequencies far surpassing the Nyquist criterion. Although this approach enables complete signal reconstruction, it produces substantial redundant data, particularly in scenarios involving high channel density, multi-dimensional acquisition, or remote transmission, thereby markedly increasing the costs of acquisition, storage, and processing. Traditional compression methods such as DCT and wavelet transform can reduce some redundancy, but they lack adaptability to shock waves, which are highly time-varying and exhibit local sudden changes, often resulting in limited compression rates and significant reconstruction distortions. However, CS theory offers a new approach to achieving compression at the acquisition end without relying on high-density sampling points, thereby providing a feasible theoretical and practical path for compressing shock wave signals.

2.2. Compressive Sensing Theory

CS constitutes a theoretical framework that simultaneously performs compression and modeling during the signal acquisition stage, thereby overcoming the constraints imposed by the traditional Nyquist sampling theorem. It enables efficient signal acquisition and reconstruction from a limited number of random observations, provided that the signal exhibits sparsity within a specific transform domain. For shock wave signals characterized by sparse spectra and pronounced locality, CS can substantially reduce sampling rates and storage overhead. Nevertheless, its application encounters two principal challenges: constructing sparse bases that align with the signal structure to ensure reconstruction accuracy, and efficiently and robustly estimating sparse coefficients in underdetermined systems.

The mathematical model of CS can be expressed as

$$y = \Phi x = \Phi \Psi \theta = \mathbf{A} \theta \quad (1)$$

where $x \in \mathbb{R}^N$ represents the original signal, $\Psi \in \mathbb{R}^{N \times K}$ is the sparse representation dictionary, and $\theta \in \mathbb{R}^K$ is the sparse coefficient vector. When θ satisfies both $\|\theta\|_0 = K$ and $K \ll N$, the signal is said to exhibit K -sparsity under the dictionary. $\|\cdot\|_0$ denotes the number of non-zero elements in the vector, used to measure sparsity. $\Phi \in \mathbb{R}^{M \times N}$ is the measurement matrix, y is the observation vector, $M \ll N$ indicates that the number of observations is far below the original dimension, and $\mathbf{A} = \Phi \Psi \in \mathbb{R}^{M \times K}$ is the equivalent perception matrix. The key to the entire system is whether signal x can be represented in sparse form under the dictionary, and how to recover the sparse coefficients θ from y under the underdetermined condition of $M \ll N$, thereby reconstructing the original signal $x = \Psi \theta$.

Under the CS model, the essence of signal reconstruction is to solve sparse optimization problems, which can be formally expressed as

$$\min_{\theta} \|\theta\|_0 \quad \text{s.t.} \quad y = \mathbf{A} \theta \quad (2)$$

Since solving the ℓ_0 -norm is NP-hard, an approximate optimization form is typically used:

$$\min_{\theta} \|\theta\|_1 \quad \text{s.t.} \quad y = \mathbf{A} \theta \quad (3)$$

When considering noise conditions, the error margin ε is introduced to obtain the following form:

$$\min_{\theta} \|\theta\|_1 \quad \text{s.t.} \quad \|y - \mathbf{A} \theta\|_2^2 \leq \varepsilon \quad (4)$$

Common solution methods include Basis Pursuit (BP), Matching Pursuit (MP), OMP [23], LASSO [24], etc. In traditional OMP, the selection of sparse supports is constructed gradually through a greedy strategy, i.e., in each step, the dictionary atoms most relevant to the current residual are selected, and the support set is updated. OMP can effectively identify atoms that are highly correlated with the signal, but in the presence of noise or dictionary redundancy, it may select redundant atoms, leading to unstable sparsity coefficients. Therefore, to further optimize the distribution of sparsity coefficients, a LASSO optimization stage can be introduced based on the sparse support set obtained by OMP. LASSO utilizes the convex relaxation property of the ℓ_1 norm to further optimize the coefficient distribution when the sparse support is known, suppressing noise effects while avoiding overfitting.

In practical applications, the quality of sparse representation, measured by sparsity level, reconstruction accuracy, and adaptability to local signal structures, is the key bottleneck constraining the compression efficiency of shock wave signals. Although predefined bases such as wavelets and discrete cosines have a certain degree of sparsity, they cannot effectively capture local structural features for sudden signals such as shock waves, resulting in poor sparse representation quality and affecting compression and reconstruction performance. Therefore, reliance solely on predefined transform domains is inadequate to fully exploit the sparse potential of shock wave signals.

To address this challenge, researchers have proposed data-driven sparse modeling methods, which involve learning to construct an adaptive dictionary $\mathbf{D}$ from training data to replace the predefined sparse basis Φ , thereby maximizing sparse representation capabilities. Compared to fixed bases, data-driven dictionaries offer greater flexibility and expressive power, making them particularly suitable for modeling complex, time-varying signals like shock waves.

2.3. KSVD Sparse Model

KSVD is one of the most commonly used dictionary learning algorithms today. It belongs to the iterative dictionary learning method, whose goal is to learn a dictionary $\mathbf{D} = [d_1, d_2, \dots, d_L] \in \mathbb{R}^{d \times L}$ and a sparse coding matrix $\Theta = [\theta_1, \theta_2, \dots, \theta_N] \in \mathbb{R}^{L \times N}$ under the condition of a given training sample matrix $\mathbf{X} = [X_1, X_2, \dots, X_N] \in \mathbb{R}^{d \times N}$, so that each sample X_i can be sparsely represented under $\mathbf{D}$ as

$$X_i \approx \mathbf{D}\theta_i \quad \text{with} \quad \|\theta_i\|_0 \leq T_0 \quad (5)$$

where T_0 is the preset upper limit of sparsity. The optimization objective function of KSVD is

$$\min_{\mathbf{D}, \Theta} \|\mathbf{X} - \mathbf{D}\Theta\|_F^2 \quad \text{s.t.} \quad \forall i, \|\theta_i\|_0 \leq T_0 \quad \text{for } i = 1, 2, \dots, N \quad (6)$$

KSVD employs an alternating minimization strategy comprising two iterative stages: fixing the dictionary to solve for sparse coefficients, followed by fixing the sparse coefficients to update the dictionary. First, the dictionary $\mathbf{D}$ is initialized, and algorithms such as OMP are applied to perform sparse coding on all samples, thereby obtaining the sparse coefficients θ_i under the current dictionary, as shown in Equation (7).

$$\theta_i = \arg \min_{\theta} \|X_i - \mathbf{D}\theta\|_2^2 \quad \text{s.t.} \quad \|\theta\|_0 \leq T_0 \quad (7)$$

During the dictionary update phase, the KSVD algorithm updates each atom d_k and its corresponding sparse coefficients in a column-wise manner. Let $\omega_k = \{i \mid \theta_{k,i} \neq 0\}$ be the set of sample indices that use atom d_k . The residual matrix is then constructed to remove the contribution of atom d_k : $\mathbf{E}_k = \mathbf{X} - \sum_{j \in \omega_k} d_j \theta_j$. Then extract the submatrix $\mathbf{E}_k^\omega$ under the support, perform SVD on the residuals: $\mathbf{E}_k^\omega = \mathbf{U}\mathbf{S}\mathbf{V}^T$. Set $d_k = \mathbf{U}(:, 1)$, and update the corresponding sparse coefficient to $\theta_{k,\omega} = \mathbf{S}(1, 1) \cdot \mathbf{V}(:, 1)^T$. Repeating these steps for each iteration ultimately yields the optimal representation dictionary $\mathbf{D}$. In this work, the dictionary is constructed to pursue a better linear representation of the signal, thereby attaining lower sparsity and facilitating the subsequent CS coding and reconstruction process.

3. Improved Algorithm DH-KSVD

3.1. Dynamic Dictionary Pruning Mechanism

In traditional KSVD algorithms, although the fixed dictionary structure guarantees the stability of sparse representations, efficiency issues become especially prominent when handling signals with abrupt local changes, such as shock waves. A substantial number of atoms are seldom activated during signal reconstruction, or are highly redundant with other atoms, thus consuming considerable computational resources while providing marginal improvements in reconstruction performance. To address this issue, this study introduces a dynamic pruning mechanism. By establishing a multi-dimensional atom evaluation system, the mechanism quantifies the usage efficiency and feature uniqueness of atoms, eliminating redundant or inefficient atoms to enable adaptive and lightweight dictionary optimization. Equation (8) defines the comprehensive contribution assessment function for the j -th atom, which integrates the representation strength and usage frequency of atoms to quantify their contributions to signal reconstruction.

$$C_j = \sum_{l=1}^L |\hat{\theta}_l(j)| + \partial \sum_{l=1}^L \Pi(\hat{\theta}_l(j) \neq 0) \quad (8)$$

In this equation, $\hat{\theta}_l(j)$ denotes the sparsity coefficient of the j -th atom in the l -th training sample, L represents the total number of training samples, and ∂ is a tuning parameter that balances the weights of representation strength and usage frequency. The first term quantifies the overall representation strength of the atom across all samples, whereas the second term enumerates the number of samples in which it is activated, thereby reflecting its usage frequency.

After evaluating the comprehensive contributions of atoms, an adaptive threshold strategy is employed for pruning, as illustrated in the following equation:

$$\text{Prune}(d_j) = \Pi \left(C_j < \tau (1 + \gamma \cdot \text{Uniq}(d_j)) \cdot \max_k C_k \right) \quad (9)$$

Pruning conditions compare the composite contribution C_j of an atom against a dynamic value determined jointly by the base pruning threshold τ and the maximum contribution $\max_k C_k$ in the current dictionary, thereby mitigating instability caused by scale variations. Building upon this, the uniqueness metric $\text{Uniq}(d_j)$ is introduced as a regulating factor, with its influence intensity controlled by the parameter γ . For atoms exhibiting high uniqueness, even when their comprehensive contribution is low, the pruning strictness should be appropriately relaxed to prevent the removal of key features.

The uniqueness of an atom is evaluated by computing its similarity to other atoms, as defined by the following equation:

$$\text{Uniq}(d_j) = 1 - \max_{k \neq j} \frac{|d_j d_k|}{\|d_j\|_2 \|d_k\|_2} \quad (10)$$

This equation evaluates the maximum correlation between an atom d_j and the other atoms in the dictionary. A higher degree of similarity corresponds to a lower level of uniqueness. In this manner, atoms possessing unique representational capacity within the dictionary can be identified and afforded special protection during pruning, thereby preventing a decline in the dictionary's representational capability as a consequence of pruning. To further enhance the stability of the pruning process, a sliding window mechanism was introduced, in which pruning operations were executed every five iterations and the statistics within the window were updated prior to each pruning step. This sliding window design ensures that pruning decisions are informed not only by the current iteration but also by the recent usage patterns of atoms, thereby mitigating misjudgments of their long-term value caused by short-term fluctuations.

Although the dynamic pruning mechanism enhances efficiency, it still carries the risk of over-pruning or inadvertently eliminating key atoms. To address this issue, a dynamic expansion mechanism was devised to introduce new atoms when necessary, thereby restoring the integrity of the dictionary. The core concept is to generate new atoms through the analysis of signal residuals when the reconstruction error increases markedly or when the dictionary contains too few atoms. Specifically, the following equation is employed to select new atoms:

$$d_{\text{new}} = \arg \min_{d \in \mathbf{D}_{\text{candidate}}} \sum_{l=1}^L \|X_l - \mathbf{D}_{\text{current}} \hat{\theta}_l - d \hat{\theta}_l(d)\|^2 \quad (11)$$

where $\mathbf{D}_{\text{candidate}}$ is a set of candidate atoms composed of residual signals that cannot be fully represented by the current dictionary $\mathbf{D}_{\text{current}}$. The construction process is as follows: first, calculate the reconstruction error $R_l = X_l - \mathbf{D}_{\text{current}} \hat{\theta}_l$ for all training samples X_l in the current dictionary $\mathbf{D}_{\text{current}}$. Subsequently, the residuals corresponding to the top 10% of samples with the largest reconstruction errors are selected as the initial residual set. This set undergoes normalization processing, and typical residual patterns are extracted via

K-means clustering. The resulting cluster centers are ultimately designated as the candidate atom set $D_{\text{candidate}}$. Finally, using Equation (11), a new atom d_{new} capable of maximally reducing the overall reconstruction error is selected for addition to the dictionary. This strategy ensures that new atoms originate from structural residuals that the system has long been unable to represent effectively, rather than random noise or transient perturbations. Simultaneously, the clustering operation enhances the representativeness and stability of candidate atoms. Furthermore, to prevent new atoms from being highly correlated with existing ones, which could increase dictionary coherence or cause collapse, we introduce a uniqueness screening mechanism during candidate atom incorporation. This mechanism requires that new atoms must exhibit a maximum correlation below 0.95 with all existing atoms in the dictionary. Those failing this criterion are discarded outright to preserve dictionary diversity and prevent cumulative coherence.

In addition, to avoid insufficient pruning or unstable pruning strategies, a dynamic adjustment mechanism is introduced to adjust the pruning threshold based on the performance of the current dictionary, as shown in the following equation:

$$\tau_{\text{new}} = \tau_{\text{old}} + \alpha (\text{Error}_{\text{current}} - \text{Error}_{\text{target}}) \quad (12)$$

where τ_{old} represents the current base threshold, α denotes the learning rate, which controls the step size or speed of adjusting the pruning threshold τ to ensure the adjustment process is neither too abrupt nor too slow. $\text{Error}_{\text{current}}$ is the current average reconstruction error, representing the average distortion level after applying sparse coding to a set of signals using the current dictionary. $\text{Error}_{\text{target}}$ is the target error, representing the minimum reconstruction error level to be achieved. This mechanism enables automatic adjustment of pruning strictness when reconstruction error deviates from the target, thereby achieving a dynamic balance between dictionary size and expressive capability. The specific equation for $\text{Error}_{\text{current}}$ is

$$\text{Error}_{\text{current}} = \frac{1}{N \cdot L} \sum_{l=1}^L \|X_l - D_{\text{current}} \theta_l\|^2 \quad (13)$$

When an atom remains inactive for several consecutive iterations, the system prioritizes its pruning; conversely, when the usage frequency of an atom increases substantially, the system postpones its pruning to preserve its potential contribution to signal representation.

3.2. Hybrid Sparse Constraint Strategy

In sparse coding, traditional methods such as OMP and LASSO each offer distinct advantages, yet are accompanied by inherent limitations. The OMP algorithm excels in delivering accurate sparse representations by progressively selecting the most relevant dictionary atoms to approximate signals. However, it is susceptible to overfitting when exposed to noise interference or dictionary redundancy, resulting in insufficient sparsity or increased reconstruction error. LASSO improves the model's robustness to noise through the incorporation of an ℓ_1 regularization term; however, the sparsity of its solution is typically contingent on the selection of the regularization parameter, and in some instances, precise control over sparsity cannot be assured. Therefore, to strike a balance between the accuracy and robustness of sparse representation, this study proposes a hybrid sparse constraint strategy that integrates the strengths of OMP and LASSO, thereby constructing a hierarchical optimization framework. The first stage utilizes the OMP algorithm for initial sparse selection, identifying the most relevant set of atoms in the signal representation. In contrast, the second stage formulates a LASSO optimization problem within this sparse subspace to further refine the coefficients, mitigate noise effects, and improve overall stability.

First, we define the sparse representation model of the signal $X \in \mathbb{R}^n$ as follows:

$$X_i = \mathbf{D}\theta_i + \varepsilon \quad (14)$$

where $\mathbf{D} \in \mathbb{R}^{n \times K}$ denotes a known overcomplete dictionary, $\theta \in \mathbb{R}^K$ represents a sparse coefficient vector, and ε accounts for additive noise.

In the first stage, the OMP algorithm is employed to identify the sparse support set $\Lambda \subseteq \{1, 2, \dots, K\}$ of the signal and obtain the initial sparse coefficient estimate θ_{OMP} . In the second stage, a sub-dictionary $\mathbf{D}_\Lambda$ is constructed based on the support set Λ , and the LASSO problem is subsequently solved within this subspace. In the specific implementation, the support selection process of OMP can be formalized through the following iterative steps:

Initialize the residual $r_0 = X$ and the support set $\Lambda_0 = \phi$.

For each iteration:

$$j_i = \arg \max_j |d_j r_{i-1}| \quad (15)$$

$$\Lambda_i = \Lambda_{i-1} \cup \{j_i\} \quad (16)$$

$$\theta_i = \arg \min_{\theta} \|X - \mathbf{D}_{\Lambda_i} \theta\|_2^2 \quad (17)$$

$$r_i = X - \mathbf{D}_{\Lambda_i} \theta_i \quad (18)$$

where d_i denotes the j -th column of dictionary $\mathbf{D}$, and X is the least-squares solution of θ_i on the support set Λ_i . Equation (15) specifies that during the i -th iteration, an atom d_i most relevant to the current residual r_{i-1} is selected from the dictionary $\mathbf{D}$. Specifically, this is achieved by calculating the absolute value of the inner product between each atom d_i in the dictionary and the current residual r_{i-1} , and selecting the index j_i of the atom yielding the maximum absolute inner product value. This ensures that the atom that most closely matches the current residual is chosen at each iteration. The newly selected atom index j_i is added to the support set Λ_i , as shown in Equation (16). Solving the least-squares problem on the current support set Λ_i yields the sparse coefficient vector θ_i , where $\mathbf{D}$ denotes the sub-dictionary composed of atoms from Λ_i . Finally, the residual r_i is updated using the new sparse coefficients θ_i .

OMP selects the most relevant atoms by maximizing the inner product between the dictionary atoms and the residuals, and updates the coefficients using the least squares method. Upon completion of the OMP stage, the LASSO optimization phase begins. To further improve the robustness of the sparse representation, the coefficients on the support set Λ are optimized through regularization:

$$\hat{\theta}_\Lambda = \arg \min_{\theta_\Lambda} \frac{1}{2} \|X - \mathbf{D}_\Lambda \theta_\Lambda\|_2^2 + \lambda \|\theta_\Lambda\|_1 \quad (19)$$

where $\mathbf{D}_\Lambda$ is a subset of columns corresponding to the support set Λ in dictionary $\mathbf{D}$, and θ_Λ denotes the corresponding sparse coefficient subset, λ is the regularization parameter, used to control sparsity. Through this hierarchical strategy, the introduction of LASSO not only retains the sparse support determined in the OMP stage, but also further compresses the coefficients via ℓ_1 regularization, suppressing noise effects and improving overall stability.

To dynamically adjust regularization parameter λ based on signal noise levels and dictionary redundancy, a noise level estimation and dynamic adjustment mechanism is introduced. First, the noise level is estimated using the median absolute deviation (MAD):

$$\hat{\sigma} = \frac{\text{MAD}(r)}{0.6745} \quad (20)$$

$\text{MAD}(r)$ denotes the median absolute deviation of the final residual vector r computed during the OMP phase. 0.6745 is a constant used to make MAD equal to the standard deviation under a Gaussian distribution. The regularization parameter λ is dynamically adjusted based on the estimated noise level $\hat{\sigma}$ and dictionary size Q :

$$\lambda = \beta \cdot \hat{\sigma} \cdot \sqrt{2 \log Q} \quad (21)$$

where β is an empirical adjustment coefficient, typically ranging between 0.5 and 1.0. $\sqrt{2 \log Q}$ is a factor related to the dictionary size that accounts for the impact of dictionary redundancy on regularization strength. As the dictionary size increases, the likelihood of selecting erroneous atoms also rises, necessitating stronger regularization to suppress this effect. This strategy enables the adaptive adjustment of regularization strength based on the signal-to-noise ratio, thereby ensuring the stability of sparse coding across various noise environments.

3.3. Algorithm Description of DH-KSVD

The DH-KSVD algorithm comprises four distinct stages: parameter initialization, hybrid sparse coding, dictionary updating, and dynamic pruning. Prior to the algorithm's initiation, the input variables encompass the training sample matrix $\mathbf{X}_{N \times L}$, the initial dictionary $\mathbf{D}_{N \times d_s}$, the pruning threshold τ , learning rate α , adjustment parameters ∂, γ and the maximum number of iterations $\max_{iter}$. For each iteration, the parameters are initialized, an empty set Λ_0 is defined to store the currently selected atom indices, and the initial sparse coefficient matrix $\hat{\theta}$ and atom frequency matrix F are set to zero matrices, respectively.

Consider the i -th iteration as an example; first, perform mixed sparse coding to find the optimal sparse representation of the signal. For each sample X_l , select the atoms from the dictionary that are most relevant to the current residual. Specifically, the inner product between each atom and the residual is calculated, and add the atom index j corresponding to the maximum value to the set of selected atoms Λ . The coefficients $\hat{\theta}_\Lambda$ of these selected atoms are solved using least squares, and the residual r is updated. After obtaining the preliminary sparse representation, the coefficients $\hat{\theta}$ are further optimized to ensure not only accurate data fitting but also high sparsity and robustness to noise. The sparse coefficient matrix $\hat{\theta}$ and the atom frequency matrix F reflect the results of the current iteration.

During the dictionary update phase, for each atom corresponding to a non-zero coefficient, calculate its corresponding error matrix E_n and update the atom using SVD. Dynamic pruning is employed to preserve the effectiveness and compactness of the dictionary. The decision to remove an atom is based on its contribution to the reconstructed signal C_j and its uniqueness index $\text{Uniq}(d_j)$. If an atom's contribution falls below the threshold τ plus the weighted sum of its uniqueness metric, the atom is removed from the dictionary. The algorithm also incorporates a dynamic expansion mechanism that automatically adds new atoms to the dictionary when the reconstruction error is high. Simultaneously, it adaptively adjusts the pruning threshold τ and regularization parameter λ based on the current error level to optimize algorithmic performance. After each iteration, the current residual $E(i)$ is evaluated to determine whether it is smaller than the residual from the previous iteration $E(i-1)$. If the condition is satisfied, the optimal dictionary $\mathbf{D}_{Min}$ is updated. When the number of iterations i reaches $\max_{iter}$, the iteration process terminates.

The steps of the DH-KSVD algorithm are shown in Algorithm 1.

Algorithm 1 DH-KSVD algorithm**Input:** $\mathbf{X}_{N \times L}$, $\mathbf{D}_{N \times ds}$, ∂ , τ , α , γ , λ , Q , $\max_{\text{iter}}$ **Output:** $\mathbf{D}_{\text{Min}}$, $\hat{\theta}$

```

1: Initialization:  $i = 1, r_0 = \mathbf{X}, \Lambda_0 = \emptyset, F = 0_{ds \times 1}, n = 1$ 
2: while  $i \leq \max_{\text{iter}}$  do
3:   Hybrid Sparse Coding (OMP+LASSO):
4:   for  $l = 1$  to  $L$  do
5:     OMP Stage:  $\Lambda_0 = \emptyset, r = X_l$ 
6:      $j_i = \arg \max_j |d_j r_{i-1}|, \Lambda_i = \Lambda_{i-1} \cup \{j_i\}, \theta_i = \arg \min_{\theta} \|X_l - \mathbf{D}_{\Lambda_i} \theta\|_2^2, r_i = X_l -$ 
        $\mathbf{D}_{\Lambda_i} \theta_i$ 
7:     LASSO Stage:  $\hat{\theta}_{\Lambda} = \arg \min_{\theta_{\Lambda}} \frac{1}{2} \|X_l - \mathbf{D}_{\Lambda} \theta_{\Lambda}\|_2^2 + \lambda \|\theta_{\Lambda}\|_1$ 
8:     Update  $\hat{\theta}$  and atomic frequency  $F_j \leftarrow F_j + 1$  for  $j \in \Lambda$ 
9:   end for
10:  Dictionary Update:
11:  SVD Update:
12:  for  $n = 1$  to  $ds$  do
13:     $\omega_n = \{i | \hat{\theta}_n(i) \neq 0\}, E_n = \mathbf{X}(\omega_n) - \sum_{j \neq n} \mathbf{D}_j \hat{\theta}_j(\omega_n), [U, \Sigma, V] = \text{SVD}(E_n),$ 
        $\mathbf{D}_n = U_1, \hat{\theta}_n(\omega_n) = \Sigma(1, 1) V_1^T$ 
14:  end for
15:  Dynamic Pruning (execute every five iterations):
16:  for  $j = 1$  to  $ds$  do
17:    Contribution Calculation:  $C_j = \sum_{l=1}^L |\hat{\theta}_j(l)| + \partial F_j, \forall j \in \{1, \dots, ds\}$ 
18:    Uniqueness Evaluation:  $\text{Uniq}(d_j) = 1 - \max_{k \neq j} \frac{|d_j d_k|}{\|d_j\|_2 \|d_k\|_2}$ 
19:    Prune Condition: Remove  $d_j$  if  $C_j < \tau(1 + \gamma \cdot \text{Uniq}(d_j)) \cdot \max_k C_k$ 
20:  end for
21:  Dynamic Expansion:
22:  Errorcurrent:  $E(i) = \frac{1}{N \cdot L} \sum_{l=1}^L \|X_l - \mathbf{D} \hat{\theta}_l\|_2^2$ 
23:  Add new atom:  $d_{\text{new}} = \arg \min_d \sum_{l=1}^L \|X_l - \mathbf{D} \hat{\theta}_l - d \hat{\theta}_l(d)\|_2^2$ 
24:  Parameter Adaptive Adjustment:  $\tau_{\text{new}} = \tau_{\text{old}} + \alpha (\text{Error}_{\text{current}} - \text{Error}_{\text{target}})$ 
25:  Estimate noise level:  $\hat{\sigma} = \frac{\text{MAD}(r)}{0.6745}$ 
26:  Adjust regularization parameter:  $\lambda = \beta \cdot \hat{\sigma} \cdot \sqrt{2 \log Q}$ 
27:  Residual and Termination:
28:  if  $E(i) < E(i-1)$  then
29:    set  $\mathbf{D}_{\text{Min}} = \mathbf{D}, i = i + 1$ 
30:  end if
31: end while

```

3.4. Application of DH-KSVD in CS Framework**3.4.1. A Compressed Sensing Framework Based on DH-KSVD**

The CS framework based on DH-KSVD, as depicted in Figure 1, comprises three key components: dictionary training, data compression, and signal reconstruction.

In the CS framework, the application of DH-KSVD is initiated during the dictionary training phase. Given the substantial volume of actual signal data, it is impractical to use the entire signal directly for dictionary learning. Therefore, the original signal must first be divided into blocks. According to Equation (22), the signal is divided into several signal blocks of appropriate length, which serve as training samples for redundant dictionary learning. Subsequently, an improved KSVD algorithm is employed to iteratively optimize the dictionary. In each iteration, a hybrid sparse coding strategy combining OMP and

LASSO is used to obtain the sparse representations of each signal block, and the dictionary atoms are updated via SVD. This process not only ensures the dictionary's efficient representation of local features, but also achieves adaptive optimization of the dictionary structure through dynamic pruning and expansion mechanisms, thereby enhancing its sparsity performance.

Upon completion of dictionary training, the data compression phase begins. Using the learned redundant dictionary $\mathbf{D}$, sparse decomposition is performed on each block of the original signal to obtain sparse coefficient vectors. Subsequently, these sparse or compressible signals are subjected to low-dimensional projection using the measurement matrix Φ , thereby achieving compressed sampling of the signals. The compressed measurement values reduce the data dimension, facilitating efficient transmission in bandwidth-constrained wireless sensor networks. This compression process does not require full knowledge of the original signal's information; instead, it relies on a small number of measurements to preserve the signal's key structural features, thereby effectively reducing communication overhead and conserving energy consumption.

When performing signal reconstruction at the management node, after receiving the compressed measurement data, the known dictionary $\mathbf{D}$ and measurement matrix Φ are used to solve the underdetermined linear system via a sparse reconstruction algorithm, thereby recovering the sparse coefficients θ for each signal block. Subsequently, these coefficients are multiplied by the dictionary to reconstruct the corresponding signal segments. Finally, all reconstructed signal blocks are concatenated and reshaped according to Equation (23) to restore the complete signal with the same length as the original signal. This reconstruction process fully utilizes the sparse prior of the signal under the learned dictionary to achieve high-precision signal recovery, thereby balancing the efficiency and reconstruction performance of DH-KSVD within the CS framework.

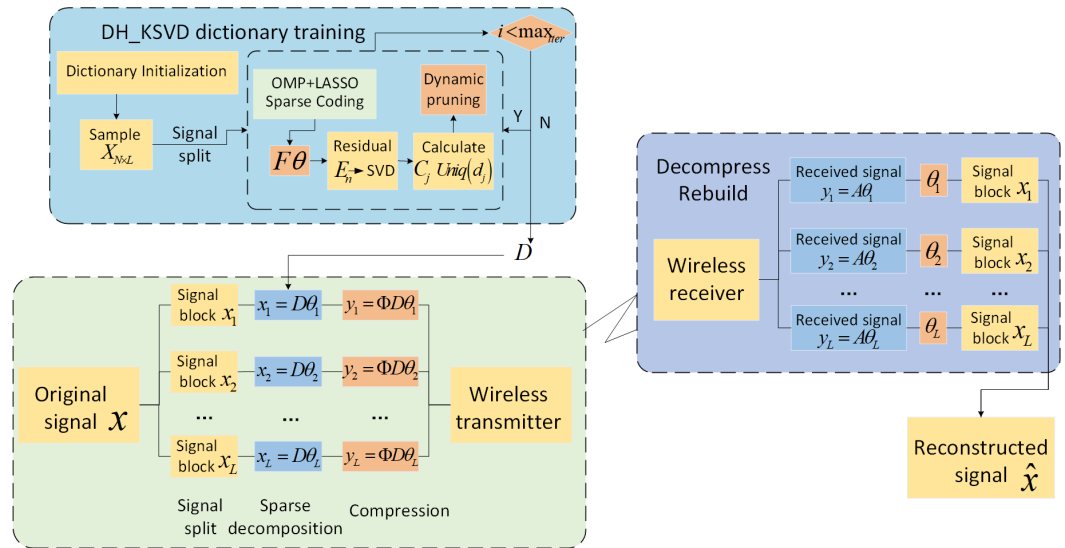


Figure 1. A CS framework based on DH-KSVD.

$$\mathbf{X}_{n \times t} = \begin{bmatrix} x_1 & x_{n+1} & \cdots & x_{(t-1)n+1} \\ x_2 & x_{n+2} & \cdots & x_{(t-1)n+2} \\ \vdots & \vdots & \ddots & \vdots \\ x_n & x_{2n} & \cdots & x_{tn} \end{bmatrix} \quad (22)$$

$$x_{tn \times 1} = \text{reshape}(\mathbf{X}(:, 1), \mathbf{X}(:, 2), \dots, \mathbf{X}(:, t)) = [x_1, x_2, \dots, x_{tn}]^T \quad (23)$$

3.4.2. Algorithm Complexity Analysis

To theoretically evaluate the computational efficiency of the DH-KSVD algorithm, an asymptotic analysis of its time complexity during training and encoding phases is conducted. The overall complexity of the algorithm is primarily determined by three core components: hybrid sparse coding, dictionary updating, and dynamic pruning.

During the training phase, the computational cost per iteration is as follows: first, the mixed sparse coding stage processes each of the L training samples sequentially. The OMP stage constructs the sparse support set by greedily selecting atoms, with a complexity of $O(Q \cdot N \cdot L)$, where Q is the number of dictionary atoms and N is the signal dimension. Subsequently, the LASSO phase performs coefficient optimization on the support set selected by OMP, with a complexity of approximately $O(T_0^3 + T_0^2 N)$, where T_0 denotes the sparsity factor. Although higher than pure OMP, it significantly improves noise robustness. Secondly, during the dictionary update phase, KSVD performs singular value decomposition (SVD) on each atom, with a complexity of $O(N|\omega_k|^2)$, where $|\omega_k|$ is the number of samples using that atom. Finally, the dynamic pruning mechanism executes periodically after several iterations, with a complexity of $O(Q^2 \cdot N + Q \cdot L)$ for evaluating atomic contributions and uniqueness. Overall, the asymptotic time complexity of a single iteration of DH-KSVD is on the same order as traditional KSVD, approximately $O(Q \cdot N \cdot L + Q \cdot N \cdot |\omega_k|^2)$. However, thanks to the dynamic pruning mechanism, redundant atoms are progressively eliminated during iteration, resulting in the effective number of atoms Q_{eff} being significantly smaller than the initial number Q . Consequently, the computational load of the algorithm decreases substantially during actual execution, particularly in later iterations. This provides the theoretical basis for its significantly reduced training time. During the encoding phase, the complexity of sparse representation for a single signal block is similar to that of the sparse coding phase in training, being $O(Q \cdot N + T_0^3)$, ensuring the efficiency of the data compression process.

4. Experimental Results and Analysis

4.1. Experimental Conditions Setting

Explosion tests were performed using a piezoelectric sensor with a range of 50 psi, and shock wave signals were collected at a sampling rate of 2 MSa/s. Each test duration was 6 s, generating 24 MB of data. However, the key physical characteristics of shock waves are primarily manifested during the millisecond-scale main pulse process, which includes rapid rise time, peak overpressure, and positive pressure duration. To evaluate the dictionary training efficiency and reconstruction performance of the proposed DH-KSVD algorithm within the CS framework, this study extracts 4096 consecutive sampling points containing the complete main pulse process from the original shock wave signal as the analysis unit, corresponding to a time window of 2.048 ms. A total of 150 independent datasets were collected to ensure diversity in explosion distance and environmental conditions. The datasets were divided into 120 training sets and 30 test sets through hierarchical random sampling. The training sets were employed for dictionary learning and algorithm parameter optimization, while the test sets were used to assess the algorithm's reconstruction performance and generalization ability.

In evaluating the algorithm's performance, the average iteration time ($\bar{T}_{\text{iter}}$), Compression Ratio (CR), Root Mean Square Error (RMSE), and Peak Signal-to-Noise Ratio (PSNR) were chosen as the evaluation metrics.

$\bar{T}_{\text{iter}}$ is defined as the time required to perform a maximum of 1000 dictionary iteration training steps.

CR is defined as

$$CR = \frac{N - M}{N} \quad (24)$$

where N represents the original data volume and M represents the number of measured values obtained by downsampling the original signal using measurement matrix Φ . A higher CR implies that less data is retained after compression, indicating a greater degree of compression.

$RMSE$ is employed to quantify the degree of difference between the reconstructed signal and the original signal, while $PSNR$ reflects the quality of the reconstructed signal, measured in dB. The higher the $PSNR$, the better the quality of the reconstruction. The definitions are provided as follows:

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^N (x_i - \hat{x}_i)^2} \quad (25)$$

$$PSNR = 10 \lg \left(\frac{x_{\max}^2}{\frac{1}{N} \sum_{i=1}^N (x_i - \hat{x}_i)^2} \right) \quad (26)$$

where x represents the original signal, $\hat{x}$ denotes the reconstructed signal, and $x_{\max}$ represents the maximum component of vector x .

All experiments were conducted on a laptop with an 11th Gen Intel(R) Core(TM) i7-1165G7 CPU @ 2.80 GHz, 16 GB RAM, and Windows 10. The code was implemented in Python 3.9 using NumPy 1.21.0, SciPy 1.7.0, and scikit-learn 0.24.2.

4.2. Algorithm Performance Analysis

To comprehensively evaluate the performance of the proposed DH-KSVD algorithm, this study compares it with the DCT, traditional KSVD, and CC-KSVD algorithms [21] under identical experimental conditions. All 150 sets of experimental data underwent the same preprocessing workflow: each set of 4096-point one-dimensional shock wave signals is processed segmentally according to Equation (22), divided into 32 consecutive non-overlapping sub-segments of length 128, forming a training sample matrix of size 128×32 . This segmentation strategy addresses the highly non-stationary nature of shockwave signals: their amplitude, frequency, and energy undergo drastic changes within extremely short time intervals. Direct sparse modeling of the entire signal segment poses challenges, as a single dictionary struggles to simultaneously capture distinctly different local features such as rising edges, peaks, and decay phases, often resulting in inefficient representation. By segmenting the signal into shorter chunks, local approximations can be made to be stationary. This allows the dictionary to focus on learning recurring key waveform patterns, thereby enhancing the accuracy and learning efficiency of sparse representations.

In addition to standardized data preprocessing, the following key parameters require initial configuration:

The dictionary learning process for all algorithms is terminated upon reaching a maximum of 1000 iterations.

The measurement matrix Φ was constructed using a Gaussian random matrix.

The initial dictionary size is set to 128×256 , meaning the dictionary initially contains 256 atoms. The dictionary's 128 rows correspond to the signal fragment length, ensuring each atom resides in the same R^{128} space as its signal fragment for direct linear sparse reconstruction. The column count of 256 represents the initial number of atoms, forming an overcomplete dictionary (redundancy ratio of 2:1). Moderately overfitted dictionaries enhance the flexibility and reconstruction accuracy of sparse representations, particularly in

dynamic pruning algorithms such as DH-KSVD. A larger initial dictionary provides a rich pool of candidate atoms, facilitating the selection of high-contribution, highly independent atoms to achieve superior sparse models. Conversely, an overly small initial dictionary suffers from insufficient diversity.

The sparsity upper bound T_0 was set to 12 in accordance with the fundamental requirement of compressed sensing theory, which stipulates $M \gtrsim CT_0 \lg N$ for stable reconstruction. This choice balances the trade-off between representation accuracy and computational efficiency: an excessively small T_0 (e.g., <8) compromises the fidelity of the reconstructed signal, while an excessively large T_0 (e.g., >16) increases computational complexity and may lead to overfitting.

Additionally, the parameter initialization for the DH-KSVD algorithm is shown in Table 1.

Table 1. Parameter initialization for the DH-KSVD algorithm.

Hyperparameter	Value	Description
τ	0.25	Base threshold for atom pruning.
α	0.1	Controls step size for adjusting τ based on reconstruction error.
∂	0.6	Weight for usage frequency in contribution score (higher values emphasize frequently used atoms).
γ	0.2	Balances atom uniqueness in pruning decision; helps avoid removing rare but informative atoms.
λ	0.01	Automatically adjusted within $[0.005, 0.02]$ based on estimated noise level during training.

4.2.1. Efficiency Analysis of Training Dictionary

To evaluate the improvement in dictionary training efficiency achieved by the dictionary pruning mechanism, 120 training samples were used in the dictionary learning stage. All algorithms used a uniform initial dictionary size of 128×256 , T_0 is 12, the maximum number of iterations was set to 1000. The experiment was conducted 200 times using Monte Carlo simulation. In each simulation, the initial dictionary $\mathbf{D}$ is randomly initialized from a standard normal distribution. Training samples are selected from the complete dataset via stratified random sampling, comprising 120 signal groups to ensure representative distributions of typical shock waveforms across all categories in each experiment. The average values of all performance indicators were taken as the final results to effectively suppress random deviations arising from initialization and sample selection.

The changes in the number of dictionaries throughout the iteration process are shown in Figure 2a. In the early stages of iteration, the dictionary has not yet fully adapted to the training data, resulting in a large number of ineffective or redundant atoms. At this stage, inefficient atoms are rapidly removed to accelerate the optimization of the dictionary structure. As the iteration progresses, the dictionary gradually approaches the optimal representation, and the pruning intensity is reduced to avoid deleting critical atoms, resulting in a more refined dictionary structure. Ultimately, the dictionary converges, and the number of atoms stabilizes, marking the completion of the optimization process and forming a compact and efficient sparse representation dictionary. Based on the analysis of the change in iteration time with the number of iterations in Figure 2b, compared to the fixed dictionary atom counts of the other three algorithms, the average training time of the DH-KSVD dictionary was reduced by 46.4%, 31%, and 13.7% compared to DCT, KSVD, and CC-KSVD, respectively.

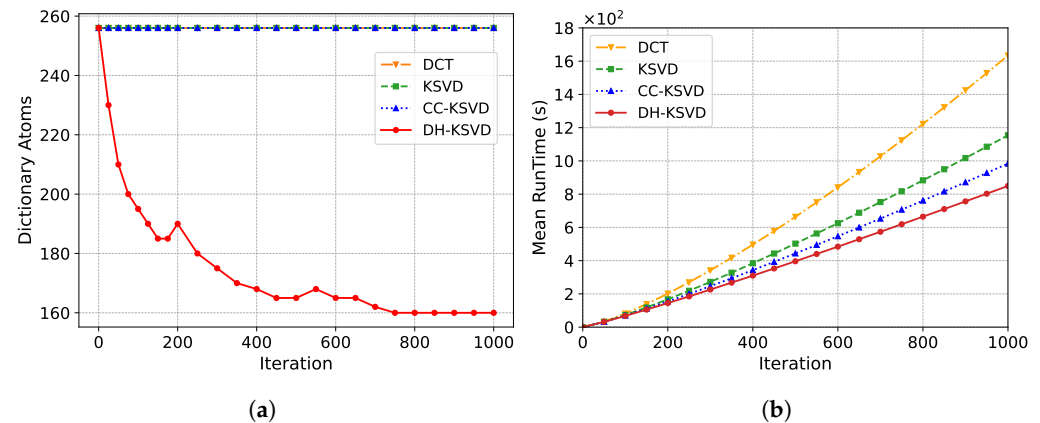


Figure 2. Dynamic dictionary atomic pruning effect. (a) The trend of the number of dictionary atoms changing with the number of iterations. (b) The variation in iteration time with the number of iterations.

4.2.2. Robustness Analysis of Algorithms

To assess the improvement in robustness of the algorithm in noisy environments under the mixed sparse constraint strategy, signal reconstruction experiments were conducted on 30 test sets with varying Signal-to-Noise Ratios (SNR) by adding Additive White Gaussian Noise (AWGN) to each test set at a CR of 0.7. In the experiment, noise was injected after the measurement process, i.e., the observation model was

$$y = \Phi x + n, n \sim \mathcal{N}(0, \sigma^2 \mathbf{I}) \quad (27)$$

where x is the original signal, Φ is the measurement matrix, y is the noisy observation vector, and n is independent and identically distributed Gaussian noise. The noise variance σ^2 is adjusted based on the preset SNR, defined as the ratio of signal power to noise power in the measurement vector, i.e.,

$$SNR = 10 \log_{10} \left(\frac{\|\Phi x\|^2}{\|n\|^2} \right). \quad (28)$$

This setting ensures comparability of performance across different noise levels and aligns with the physical context of real-world systems where noise primarily originates from measurement equipment or transmission channels. For each experimental group, the PSNR of reconstruction results is computed and averaged based on 200 independent Monte Carlo trials (each including a randomly generated measurement matrix Φ and independent noise realizations) to enhance statistical reliability.

As shown in Figure 3, the average PSNR of all algorithms increased with increasing SNR, indicating that lower noise levels contribute to enhanced reconstruction quality. Meanwhile, the error bars in the figure represent ± 1 standard deviation, reflecting fluctuations in reconstruction performance. It can be observed that in the low-SNR region, i.e., high-noise environments, the DH-KSVD algorithm not only achieves a significantly higher average PSNR than the other three algorithms but also exhibits shorter error bars, indicating lower performance variability and stronger noise resistance. Specifically, when SNR ranges from 5 to 10 dB, the average PSNR of DH-KSVD is 3 to 11 dB higher than that of the traditional KSVD algorithm, thereby fully validating its superior performance in reconstructing noisy signals.

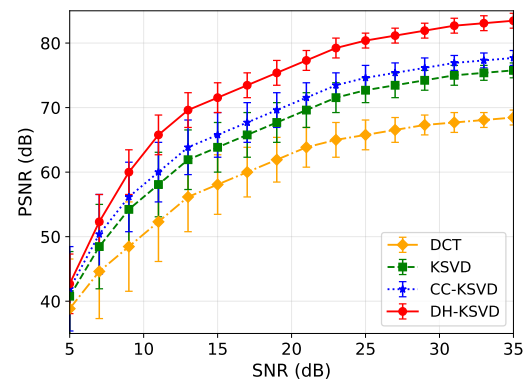


Figure 3. The influence of mixed sparse constraint strategy on signal reconstruction *PSNR* under different SNRs.

4.2.3. Performance Analysis of CS Reconstruction Algorithm

To validate the reconstruction performance of the proposed DH-KSVD algorithm within the CS framework, this study systematically evaluates its performance across different CRs and compares it with the DCT, traditional KSVD, and CC-KSVD algorithms, as depicted in Figure 4. CR is progressively increased from 0.1 to 0.9. For each CR, 200 independent samples and reconstructions are performed on 30 independent test set signals using a random Gaussian observation matrix. The random Gaussian measurement matrix $\Phi \in \mathbb{R}^{M \times N}$ has elements following a $\mathcal{N}(0, 1/M)$ distribution. Each Monte Carlo trial employs a brand-new, independently sampled measurement matrix. The test set signals are selected from the complete dataset via stratified random sampling, comprising 30 signal groups. The final performance metrics are the average of all repeated experiments to reduce the impact of randomness. The evaluation metrics include *RMSE* and *PSNR*, where a lower *RMSE* and higher *PSNR* indicate better reconstruction accuracy.

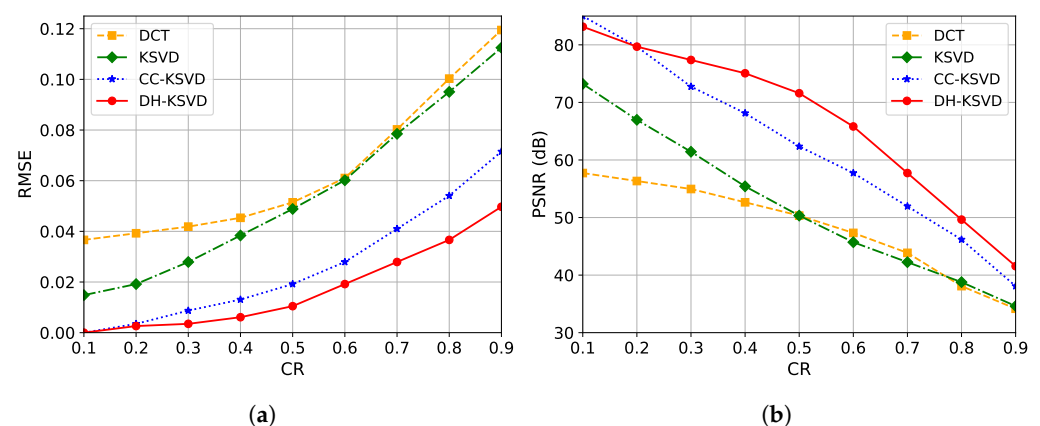


Figure 4. Comparison of reconstruction performance of different algorithms under CS with different CRs. (a) *RMSE* under different CR conditions. (b) *PSNR* under different CR conditions.

The results from different CRs show that the *RMSE* of DH-KSVD is lower than that of DCT, KSVD, and CC-KSVD, while the *PSNR* is higher than the other three categories, confirming the effectiveness of the proposed algorithm. The results indicate that the DH-KSVD algorithm maintains a low *RMSE* and high *PSNR* regardless of whether the CR is low or high. Specifically, at a CR of 0.7, the *RMSE* of the DH-KSVD algorithm is reduced by approximately 67.1%, 65.7%, and 36.2% compared to DCT, KSVD, and CC-KSVD, respectively, while the *PSNR* is improved by 35.5%, 39.8%, and 11.8%, respectively. These quantitative metrics confirm the proposed algorithm's advantages in the efficient

compression and reconstruction of shock wave signals and its potential for application within the CS framework.

Figure 5, shows the reconstruction results of four randomly selected signals from the test set under a CR of 0.7. As shown in the figure, the reconstruction errors of the four signal groups are all relatively low, with RMSE values ranging from 0.0203 to 0.0288 and PSNR values exceeding 64 dB for all cases. This indicates that the DH-KSVD algorithm achieves good reconstruction accuracy, high signal fidelity, and low error under a CR of 0.7.

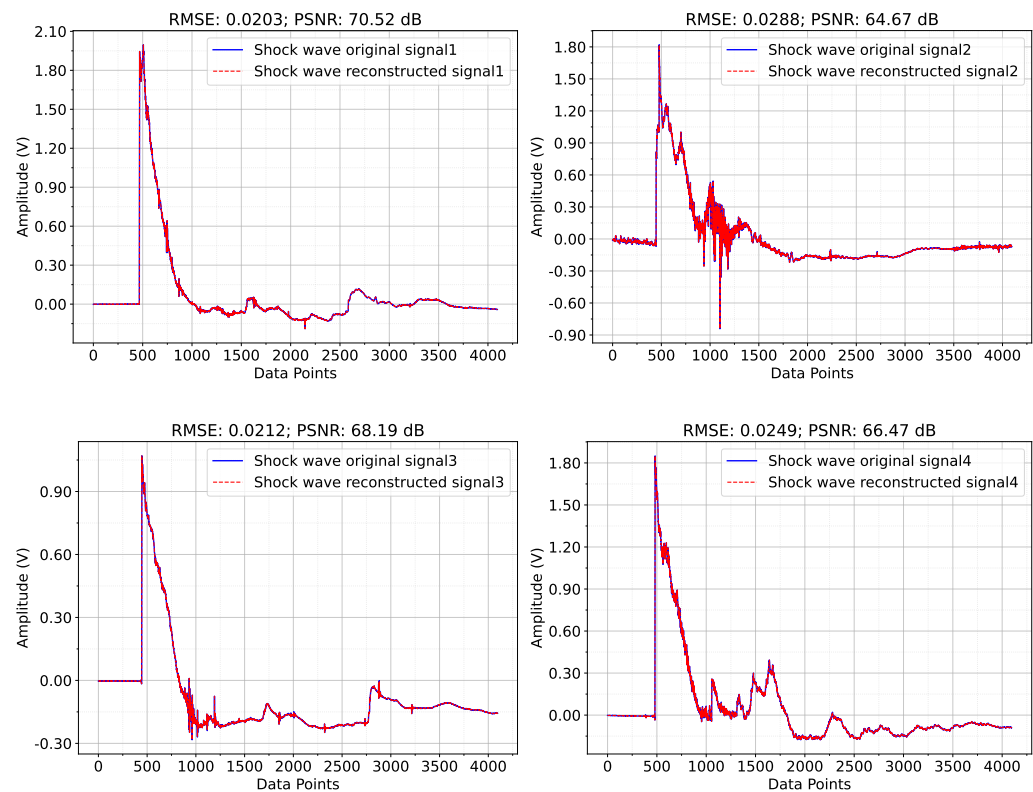


Figure 5. DH-KSVD training dictionary for shock wave compression perception reconstruction effect.

4.3. Ablation Experiment

To demonstrate the effectiveness of each improvement module, ablation experiments were conducted in this study. The proposed DH-KSVD integrates two key components: (i) a Dynamic Dictionary Pruning mechanism (D) that adaptively removes redundant atoms during dictionary learning to enhance compactness and training efficiency; and (ii) a Hybrid Sparse Coding strategy (H) that combines ℓ_0 -OMP and ℓ_1 -LASSO to improve coding robustness and reconstruction accuracy. For the ablation analysis, KSVD-A incorporates only the D module into the standard KSVD framework, retaining the conventional ℓ_0 -OMP sparse coding while enabling dynamic dictionary pruning. This variant evaluates the impact of dictionary compactness on training efficiency. KSVD-B integrates only the H module, replacing the single ℓ_0 -OMP coding with the hybrid ℓ_0 -OMP + ℓ_1 -LASSO strategy, but without applying dictionary pruning. This allows us to assess the contribution of enhanced sparse coding to reconstruction performance. DH-KSVD combines both D and H modules, representing the full proposed algorithm.

In the KSVD-A algorithm, the pruning threshold τ is initialized to 0.05 and subsequently updated adaptively via Equation (12). In KSVD-B, the regularization parameter λ is initialized to 0.01 and adjusted adaptively based on the residual norm of each iteration using Equation (21).

Table 2 shows the ablation experiment results of DH-KSVD, indicating that gradually introducing the improved module can significantly improve dictionary training efficiency and compression reconstruction performance, where $\times$ denotes the module is disabled and $\checkmark$ denotes it is enabled. After incorporating the dictionary pruning mechanism into KSVD-A, the dictionary training time was significantly reduced, decreasing by approximately 30% compared to traditional KSVD. Moreover, the standard deviation of timing decreased from ± 8.5 s to ± 4.2 s, with a significantly reduced fluctuation range. This not only demonstrates that the dynamic pruning mechanism effectively reduces redundant atoms and enhances the compactness of the dictionary, but also reflects that this strategy enhances the stability of the training process. This indicates that the dynamic pruning mechanism effectively enhances the compactness and training efficiency of the dictionary. KSVD-B significantly improves reconstruction quality by adopting a hybrid sparse coding strategy, with a substantial reduction in *RMSE* and a significant increase in *PSNR*. However, training time increases slightly, and the standard deviation is marginally higher than KSVD. This is due to the fact that the hybrid coding strategy demands additional computational resources during optimization to balance sparsity and robustness, resulting in slightly greater fluctuations in runtime. When the two improved modules were combined in DH-KSVD, the dictionary training time was reduced by 31% compared to traditional KSVD, *RMSE* decreased by 65.7%, and *PSNR* improved by 39.8%. Moreover, the smallest standard deviation in time indicates that the pruning mechanism effectively suppresses computational fluctuations. The *RMSE* standard deviation is only ± 0.0008 , demonstrating exceptionally high reconstruction stability. The *PSNR* standard deviation of ± 0.12 dB is the best among all models, indicating not only high reconstruction quality but also high reproducibility. This comprehensively validates the effectiveness of the modules working in tandem to both enhance dictionary training efficiency and achieve high-quality signal reconstruction within the CS framework.

Table 2. Ablation experiments of each improved module in DH-KSVD ($CR = 0.7$).

Model	D	H	$\bar{T}_{\text{iter}}$ (s)	<i>RMSE</i>	<i>PSNR</i> (dB)
KSVD	$\times$	$\times$	$11.71 \times 10^2 \pm 8.5$	0.0802 ± 0.0015	42.79 ± 0.32
KSVD-A	$\checkmark$	$\times$	$8.26 \times 10^2 \pm 4.2$	0.0790 ± 0.0012	45.88 ± 0.28
KSVD-B	$\times$	$\checkmark$	$12.24 \times 10^2 \pm 6.8$	0.0289 ± 0.0008	60.90 ± 0.15
DH-KSVD	$\checkmark$	$\checkmark$	$8.20 \times 10^2 \pm 3.9$	0.0228 ± 0.0008	61.87 ± 0.12

5. Discussion

The proposed DH-KSVD algorithm significantly enhances dictionary training efficiency and signal reconstruction accuracy when compressing shock wave data within the CS framework. The integration of a dynamic pruning mechanism with a hybrid sparse coding strategy effectively addresses key limitations of traditional KSVD algorithms. The following sections explore the implications, limitations, and potential future directions revealed by this research.

Implications of the proposed method:

1. Enhanced efficiency and practicality for transient signal compression: the core of the DH-KSVD algorithm lies in its “offline training, online compression” paradigm. The dictionary learning process itself is computationally intensive and performed offline, while the resulting highly compact and expressive dictionary enables efficient online compression on resource-constrained edge devices. This design makes the method particularly suitable for practical applications such as destructive testing at sea, where megahertz-level sampling generates massive data volumes but transmission bandwidth and storage space are extremely limited. Compared to the KSVD algorithm,

dictionary training time is significantly reduced (by 31%), while maintaining excellent reconstruction quality even at high compression ratios. This directly enhances the feasibility of long-term, multi-channel shock wave monitoring.

2. Robustness in noisy environments: the hybrid sparse constraint strategy combines the precise atom selection of ℓ_0 -OMP with the noise-resistant coefficient optimization of ℓ_1 -LASSO, endowing the DH-KSVD algorithm with enhanced resistance to noise interference. Experimental results confirm that the DH-KSVD algorithm consistently achieves higher *PSNR* values compared to various benchmark algorithms, demonstrating particularly outstanding performance in low-*SNR* scenarios. This robustness is crucial for shock wave measurements, which are frequently impacted by environmental and system noise contamination.

Limitations and future research:

1. The performance comparisons in this work are conducted within the traditional sparse representation framework. A valuable future direction is to benchmark DH-KSVD against modern reconstruction paradigms, such as reweighted- ℓ_1 or deep-learning-based methods, to further validate its generalization capability and robustness across diverse signal conditions.
2. The current version of the DH-KSVD algorithm is designed to handle additive noise but does not explicitly address the challenge of missing samples or structurally corrupted data, which is a common issue in harsh acquisition environments. Future work will therefore focus on enhancing the framework's resilience to such forms of data imperfection. Promising research directions include the integration of robust regression techniques capable of decomposing a signal into clean and sparse error components, as well as the development of preprocessing strategies based on data imputation or inpainting to effectively compensate for packet loss or sensor failures.
3. Although the online compression stage is designed for efficient operation, further optimization is needed to achieve real-time processing on low-power embedded platforms. Future work will focus on the co-design of algorithms and hardware, exploring strategies such as dictionary quantization and fixed-point computation implementation. This approach will maximize the algorithm's practical value in extreme edge computing scenarios.
4. The values of hyperparameters were determined through systematic analysis on an independent validation set. Although the proposed thresholding method has demonstrated effectiveness in the current task, optimal threshold selection remains a challenging and highly context-dependent problem. Future research will explore and integrate adaptive thresholding techniques from other signal processing domains. For instance, threshold determination for RD-HR maps in High-Frequency Surface Wave Radar systems presents a similar challenge, where recent data-driven strategies including geometry-based and experimentally derived adaptive algorithms [25,26] offer promising extensions to the current framework. Incorporating such approaches is expected to enhance the method's robustness, automation capability, and applicability across diverse scenarios.

6. Conclusions

This study proposes a DH-KSVD algorithm that incorporates dynamic pruning and hybrid coding to address the challenges of low dictionary training efficiency and poor signal reconstruction quality in traditional methods within the CS framework. By introducing a dynamic pruning mechanism to optimize the dictionary structure, the algorithm significantly enhances computational efficiency while improving the compactness of the dictionary. Additionally, a hybrid sparse constraint strategy, combining the advantages of

ℓ_0 -OMP and ℓ_1 -LASSO, is employed to improve coding quality and enhance reconstruction stability. Experimental results show that the average dictionary training time of DH-KSVD is significantly shorter than that of DCT, KSVD, and CC-KSVD, and it can significantly reduce RMSE and improve PSNR under high CRs. Furthermore, ablation experiments further validate the effectiveness of each improved module, demonstrating that the dynamic pruning mechanism and hybrid sparse coding strategy are crucial for enhancing the overall algorithm performance. Overall, the DH-KSVD algorithm demonstrates outstanding performance in both computational efficiency and reconstruction quality, particularly exhibiting stronger robustness and higher reconstruction accuracy when processing noisy signals. This provides an efficient and reliable solution for the data compression of shock wave signals, offering significant practical application value.

Author Contributions: Conceptualization, J.L. and Y.D.; methodology, J.L. and W.Y.; software, J.L. and W.Y.; validation, J.L. and Y.D.; formal analysis, J.L.; investigation, J.L.; resources, Y.D.; data curation, J.L.; writing—original draft preparation, J.L.; writing—review and editing, J.L. and Y.D.; visualization, J.L. and Y.D.; supervision, J.L.; project administration, Y.D.; funding acquisition, Y.D. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the National Natural Science Foundation of China grant number 61701445, the Open Research Foundation of the Key Laboratory of North University of China grant number DXMBJJ202007, and the “Fundamental Research Program” of Shanxi Province grant number 20210302124200.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author due to the sensitive nature of the experimental shock wave signals, which involve collaborations with the military and cannot be publicly shared at this time.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Ma, X.J.; Kong, D.R.; Shi, Y.C. Measurement and analysis of shock wave pressure in moving charge and stationary charge explosions. *Sensors* **2022**, *22*, 6582. [[CrossRef](#)] [[PubMed](#)]
2. You, W.B.; Ding, Y.H. W-MOPSO in adaptive circuits for blast wave measurements. *IEEE Sens. J.* **2021**, *21*, 9323–9332. [[CrossRef](#)]
3. Mou, Z.L.; Niu, X.B.; Wang, C. A precise feature extraction method for shock wave signal with improved CEEMD-HHT. *J. Ambient. Intell. Humaniz. Comput.* **2020**, 1–12. [[CrossRef](#)]
4. Ding, Y.H.; You, W.B. Sensor placement optimization based on an improved inertia and adaptive particle swarm algorithm during an explosion. *IEEE Access* **2020**, *8*, 207089–207096. [[CrossRef](#)]
5. Shi, Y.M.; Bai, S.H.; Wei, X.Y.; Gong, R.H.; Yang, J.L. Lossy and lossless (l2) post-training model size compression. In Proceedings of the IEEE/CVF International Conference on Computer Vision, Paris, France, 2–6 October 2023; pp. 17546–17556.
6. Sarker, P.; Rahman, M.L. A Huffman-based short message service compression technique using adjacent distance array. *Int. J. Inf. Commun. Technol.* **2024**, *25*, 118–136. [[CrossRef](#)]
7. Hassan, A.; Javed, S.; Hussain, S.; Ahmad, R.; Qazi, S. Arithmetic N-gram: An efficient data compression technique. *Discov. Comput.* **2024**, *27*, 1. [[CrossRef](#)]
8. Jrai, E.A.; Alsharari, S.; Almazaydeh, L.; Elleithy, K.; Abu-Hamdan, O. Improving LZW Compression of Unicode Arabic Text Using Multi-Level Encoding and a Variable-Length Phrase Code. *IEEE Access* **2023**, *11*, 51915–51929. [[CrossRef](#)]
9. Matsumoto, N.; Mazumdar, A. Binary iterative hard thresholding converges with optimal number of measurements for 1-bit compressed sensing. *J. ACM* **2024**, *71*, 1–64. [[CrossRef](#)]
10. Shi, L.; Qu, G.R. An improved reweighted method for optimizing the sensing matrix of compressed sensing. *IEEE Access* **2024**, *12*, 50841–50848. [[CrossRef](#)]
11. Průša, Z.; Holighaus, N.; Balazs, P. Fast matching pursuit with multi-Gabor dictionaries. *ACM Trans. Math. Softw. (TOMS)* **2021**, *47*, 1–20. [[CrossRef](#)]

12. Arya, A.S.; Mukhopadhyay, S. Adaptive sparse modeling in spectral & spatial domain for compressed image restoration. *Signal Process.* **2023**, *213*, 109191. [\[CrossRef\]](#)
13. Sun, Y.B.; Long, H.W.; Zhao, S.Y.; Zhang, Y.; Zhu, J.N.; Yang, X.H.; Fu, L.B. In Situ Monitoring and Defect Diagnosis Method Based on Synchronous Compression Short-Time Fourier Transform and K-Singular Value Decomposition for Al-Carbon Fiber-Reinforced Thermoplastic Friction Stir Lap Welding. *J. Mater. Eng. Perform.* **2024**, *34*, 8388–8397. [\[CrossRef\]](#)
14. Xie, M.L.; Ji, Z.X.; Zhang, G.Q.; Wang, T.; Sun, Q.S. Mutually exclusive-KSVD: Learning a discriminative dictionary for hyperspectral image classification. *Neurocomputing* **2018**, *315*, 177–189. [\[CrossRef\]](#)
15. Abubakar, A.; Zhao, X.J.; Takruri, M.; Bastaki, E.; Bermak, A. A hybrid denoising algorithm of BM3D and KSVD for Gaussian noise in DoFP polarization images. *IEEE Access* **2020**, *8*, 57451–57459. [\[CrossRef\]](#)
16. Jayakumar, R.; Dhandapani, S. Karhunen Loeve Transform with adaptive dictionary learning for coherent and random noise attenuation in seismic data. *Sādhanā* **2020**, *45*, 275. [\[CrossRef\]](#)
17. Leal, N.; Zurek, E.; Leal, E. Non-local SVD denoising of MRI based on sparse representations. *Sensors* **2020**, *20*, 1536. [\[CrossRef\]](#)
18. Ju, M.C.; Han, T.L.; Yang, R.K.; Zhao, M.; Liu, H.; Xu, B.; Liu, X. An adaptive sparsity estimation KSVD dictionary construction method for compressed sensing of transient signal. *Math. Probl. Eng.* **2022**, *1*, 6489917. [\[CrossRef\]](#)
19. Si, W.J.; Liu, Q.; Deng, Z.A. Adaptive Reconstruction Algorithm Based on Compressed Sensing Broadband Receiver. *Wirel. Commun. Mob. Comput.* **2021**, *1*, 6673235. [\[CrossRef\]](#)
20. Wang, H.Q.; Tang, Q.N.; Wang, Y.H.; Ren, S.L. Application of D-KSVD in compressed sensing based video coding. *Optik* **2021**, *226*, 165917. [\[CrossRef\]](#)
21. Zhang, A.L.; Han, T.L. Application of Sparse Dictionary Adaptive Compression Algorithm in Transient Signals. In *Journal of Physics: Conference Series*; IOP Publishing: Bristol, UK, 2019; Volume 1229, p. 012048. [\[CrossRef\]](#)
22. You, W.B.; Ding, Y.H.; Yao, Y. Static explosion field reconstruction based on the improved biharmonic spline interpolation. *IEEE Sens. J.* **2020**, *20*, 7235–7240. [\[CrossRef\]](#)
23. Wen, J.M.; Li, C.H.; Shu, Q.Y.; Zhou, Z.C. Randomized Orthogonal Matching Pursuit Algorithm with Adaptive Partial Selection for Sparse Signal Recovery. *SIAM J. Imaging Sci.* **2025**, *18*, 1028–1057. [\[CrossRef\]](#)
24. Lakshmi, M.V.; Winkler, J.R. Numerical properties of solutions of LASSO regression. *Appl. Numer. Math.* **2025**, *208*, 297–309. [\[CrossRef\]](#)
25. Golubović, D.; Erić, M.; Vukmirović, N. High-Resolution Method for Primary Signal Processing in HFSWR. In Proceedings of the 2022 30th European Signal Processing Conference (EUSIPCO), Belgrade, Serbia, 29 August–2 September 2022; IEEE: New York, NY, USA, 2022; pp. 912–916. [\[CrossRef\]](#)
26. Golubović, D.; Marjanović, D. An Experimentally-Based Method For Detection Threshold Determination in HFSWR's High-Resolution Range-Doppler Map Target Detection. In Proceedings of the 2025 24th International Symposium INFOTEH-JAHORINA (INFOTEH), East Sarajevo, Bosnia and Herzegovina, 19–21 March 2025; IEEE: New York, NY, USA, 2025; pp. 1–6. [\[CrossRef\]](#)

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.