ITC 1/55
Information Technology and Control
Vol. 55 / No. 1/ 2026
pp. 20-40
DOI 10.5755/j01.itc.55.1.42668

DIST-C-SEGAN: A Novel Distributed Dimension Reduction Neural Framework for Visual Data

Received 2025/08/29 | Accepted after revision 2025/10/05

HOW TO CITE: Wang, H. (2026). DIST-C-SEGAN: A Novel Distributed Dimension Reduction Neural Framework for Visual Data. *Information Technology and Control*, 55(1), 20-40. https://doi.org/10.5755/j01.itc.55.1.42668

DIST-C-SEGAN: A Novel Distributed Dimension Reduction Neural Framework for Visual Data

Hongren Wang
Leicester International Institute, Dalian University of Technology, Panjin, 124000, Liaoning, China

Corresponding author: 13796487571@mail.dlut.edu.cn

This paper proposes a distributed unsupervised dimensionality reduction network framework for visual data, referred to as DIST-C-SEGAN. It aims to achieve low-bandwidth, high-fidelity, scalable, and privacy-friendly unsupervised dimensionality reduction and compression of large-scale image data, laying a solid foundation for downstream tasks in resource-constrained distributed or edge computing environments. This method captures local spatial information through a lightweight convolutional network and explores deep feature information using a graph neural propagation structure that combines compression and activation techniques with multi-head self-attention mechanisms. At the same time, it introduces the Proximal Alignment Algorithm to generate low-dimensional embedding features, achieving end-to-end "near-lossless" preservation of local neighborhood structure and global manifold continuity. Experimental results show that DIST-C-SEGAN achieves an original variance retention rate of over 98% in all data tasks, demonstrating high reliability and strong continuity. Its structural evaluation metrics are significantly superior to those of classic dimensionality reduction techniques, such as Principal Component Analysis (PCA), Kernel Principal Component Analysis (KPCA), Autoencoder (AE), Variational Autoencoder (VAE), UMAP, and t-SNE. Sensitivity analysis determines the optimal range of key hyperparameters and confirms their robustness to random parameter initialization. Ablation experiments verify the necessity of multi-head self-attention mechanisms and geometric alignment, while downstream tasks further confirm their effectiveness. Moreover, DIST-C-SEGAN theoretically reduces communication costs and provides new solutions for high-fidelity and communication-efficient compression and recovery in distributed environments.

KEYWORDS: Distributed dimension reduction, near-lossless compression, Distributed learning.

1. Introduction

In contemporary distributed paradigms, including the Internet of Things, edge computing, and federated learning, vast volumes of high-dimensional image data are dispersed across heterogeneous nodes, often lacking annotation information [2]. To validate the generalizability of methodologies across diverse visual scenarios, unsupervised image benchmark datasets have emerged as pivotal research substrates [20]. Nevertheless, in practical implementations, the acquisition of labeled information poses substantial challenges, thereby confining exploratory analyses of raw data to exclusive reliance on end-to-end unsupervised dimensionality reduction techniques. This exigency delineates a clear trajectory for related research: the development of a framework capable of generating high-precision low-dimensional embeddings locally at distributed nodes, mitigating inter-node communication overhead, and concurrently enabling high-fidelity reconstruction of original images [1, 12]. Such a framework would furnish reliable data compression and recovery capabilities for resource-constrained distributed systems, while underpinning downstream tasks such as image clustering and anomaly detection tasks [10].

Since the inception of Distributed Principal Component Analysis (DPCA) [18], the theoretical underpinnings and algorithmic formulations of distributed linear dimensionality reduction have undergone continuous refinement. Early contributions by Qi et al. (2003) [17] proposed computing local sample statistics and the first k principal components at each node, followed by reconstructing the global covariance via weighted aggregation, which reduced the communication complexity from $O(np)$ to $O(p\sum_i k_i)$. Through the transmission of structured PCA involving singular values and singular vectors, they further attenuated the communication volume to $O(r^2d)$. Le Borgne, Roy, and Vetterli (2008) [13, 19] integrated power iteration with local linear projection, achieving decentralized estimation of global principal components through eigenvector exchange. Valcarcel Macua et al. (2010) [21] introduced CB-DPCA, which iteratively approximates the global covariance matrix via multi-round node interactions under decentralized architectures. In the same year, Fang and Li devised a compression matrix for bandwidth- and energy-constrained wireless sensor networks, effectively diminishing communication volume and node energy consumption, thereby laying the groundwork for efficient distributed PCA [6]. With the advancement of big data and cloud computing, a parallel processing paradigm based on Spark+HDFS was proposed, which transforms covariance computation into parallel summation of outer products of pixel vectors and executes centralized decomposition at the driver node, successfully accelerating PCA for hyperspectral imagery [11]. Researchers have also employed mergeable core set techniques, coupled with asynchronous differential privacy and streaming algorithms, to realize privacy-preserving and horizontally scalable distributed PCA [7, 8, 9, 24].

To address the demands of real-time control and prediction tasks, supervised information and model structural constraints have been progressively incorporated into the dimensionality reduction workflow. The POD+ANN hybrid framework extracts the first r modes through offline POD decomposition and integrates MPC optimizers and QP solvers to achieve millisecond-level online optimization and prediction for high-dimensional systems. Distributed sorted regression introduces adaptive nuclear norm penalties and SVD soft-threshold closed-form updates, facilitating rapid convergence of global low-rank regression coefficients with a communication overhead of $O(mpq)$ [3, 21]. Onder Tutsoy and Sumbul (2024) designed a parallel dimensionality reduction pipeline that eliminates redundant features and incorporates weighted hierarchical clustering to compress feature, label, and sample structures synergistically [20]. Recent investigations have integrated the Distributed Sparse Manifold-Constrained LDA (DSCLDA) with the SKR-DMKCF framework, applying l_2/l_0 sparse regularization to local projection matrices, introducing graph Laplacian manifold regularization and orthogonality constraints, and leveraging the ManPG algorithm for iterative optimization on the Stiefel manifold. It reduces the computational complexity from $O(nd^2+d^3)$ to $O(rd^2)$ and markedly enhances classification performance [4, 27].

As data structures grow increasingly intricate, the application of high-order structures and non-linear dimensionality reduction methodologies in distrib-

uted environments has garnered escalating attention. The DDR_S method proposed by Zhang et al. (2014) combines multi-layer random clustering networks with local PCA to achieve non-linear embedding that balances inter-class separability and intra-class compactness in the absence of communication [26]. The DDR framework, designed by Liang et al. (2016), employs a latent convex optimization approach with orthogonal constraints on projection vectors to estimate subspace projections in parallel by exchanging only covariates, thereby preserving tensor structures while eliminating high-dimensional centralized operations [15]. Subsequent research, building on initial Isomap embeddings, introduced distributed persistent homology (DPH) post-processing, combined with a topological mismatch gradient feedback mechanism and parallelized optimization strategies, to enhance the geometric and topological fidelity of low-dimensional mappings, outperforming UMAP and t-SNE [16, 23]. Additionally, some studies have integrated balanced truncation, extended Kalman filtering (EKF), and error covariance correction techniques, proposing a "peripheral compression + central refinement" two-tier architecture for IoT traffic anomaly detection. The peripheral layer utilizes an autoencoder (AE) for screening and compression, while the central layer employs global-local preserving projection (LPP) to refine embeddings, achieving efficient and controllable bandwidth utilization [22]. Li et al. (2025) expedited distributed wideband source localization by mapping sub-band covariances to a reference subspace and conducting rapid spectral peak searches, further augmenting system robustness [14].

However, existing methodologies exhibit significant limitations: PCA-based approaches encounter communication and computational bottlenecks when processing large-scale high-dimensional data. Balancing privacy preservation and efficiency in resource-constrained scenarios remains a formidable challenge. The traditional methods tend to suffer from slow or unstable convergence when handling heterogeneous or sparse data, and high-order structure and non-linear dimensionality reduction techniques are encumbered by excessive computational complexity. To redress these shortcomings, this paper presents the Distributed Convolutional SE Graph Alignment Network (DIST-C-SEGAN) and constructs a distributed computing framework that integrates local feature extraction and global information fusion. This methodology substantially reduces communication overhead by generating low-dimensional embeddings locally at nodes and exchanging only compact vectors, while retaining a large number of key image features during the compression process to achieve data availability. Furthermore, it captures spatio-temporal dependencies between nodes to minimize redundant transmissions, eschews reliance on a central node through incremental updates and asynchronous computation, and enhances convergence speed and stability in large-scale high-dimensional data scenarios via efficient graph convolution optimization algorithms. This approach offers an efficient and scalable solution for distributed environments with limited resources and massive datasets, thereby laying a robust foundation for a wide range of downstream tasks.

The remaining content of this paper is organized as follows: Section 2 elaborates on the proposed method, providing detailed explanations of the mathematical principles and implementation algorithms for each module. Section 3 presents data analysis, including comparative experiments, ablation experiments, and sensitivity analysis. Section 4 provides a summary of the entire paper.

2. Methodology

To address the challenges of efficient feature extraction and distributed learning in high-dimensional image processing, this study proposes a distributed dimensionality reduction framework formed by neural networks. This framework utilizes convolutional neural networks (CNN) and the channel recalibration of SE to extract local features locally, thereby enabling the "local processing" capability of distributed nodes. It uses Attention and graph neural networks (GNN) to capture the global structure across nodes or regions to achieve "node interaction" in a distributed environment. It uses Procrustes to align the local learning results into a global subspace, thereby achieving the "global consistency" requirement in the distributed system. The structural diagram is shown in Figure 1, and it is called DIST-C-SEGAN. This framework can capture both local spatial patterns and global context relationships. At the same time, it minimizes communication overhead in the distributed environment and transforms the classic distributed dimensionality

reduction problem into a form that neural networks can solve. Through the design of reasonable module combinations, it achieves dual optimization of performance and communication. The technical details are as follows.

2.1. Hierarchical Feature Extraction via CNN with Batch Normalization

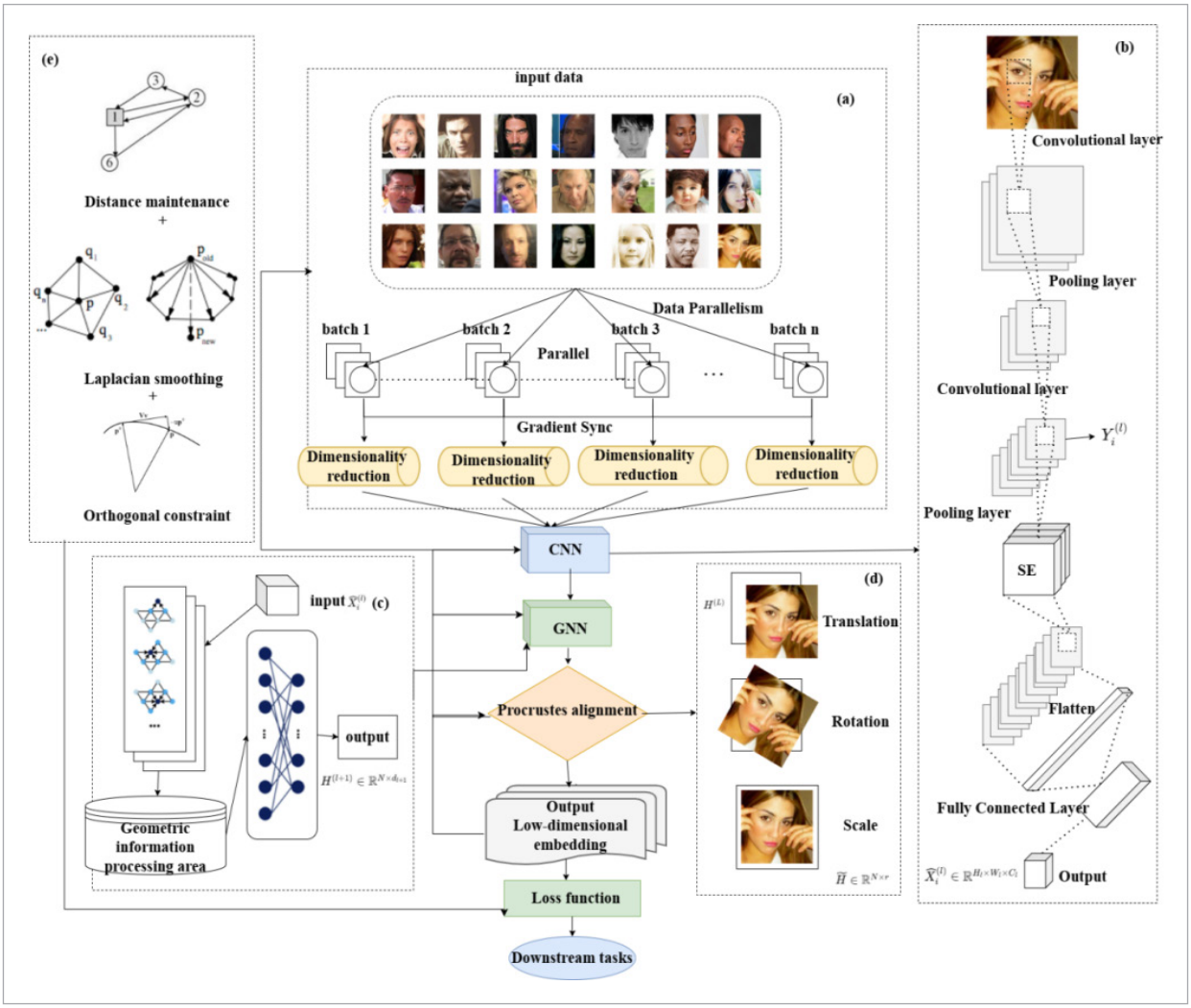
The study first uses a CNN to scan the image within a local window, extracting hierarchical feature representations from the input image as shown in Figure 1(b). For the input image $X_i \in R^{H\times W\times C}$ (where H and W represent the height and width, respectively, and C represents the number of channels). This network performs progressive feature transformation through stacked convolutional layers. At the *l*-th layer, the convolution operation is defined as:

$$Y_i^{(l)}[u,v,k]=\sum_{c=1}^{C_{l-1}}\sum_{m=-M}^{M}\sum_{n=-N}^{N}W_{m,n,c,k}^{(l)}X_i^{(l-1)}[u+m,v+n,c]+b_k^{(l)} \quad (1)$$

where $W^{(l)} \in R^{(2M+1)\times(2N+1)\times C_{l-1}\times C_l}$ is the convolution kernel with spatial dimensions 2M+1 and 2N+1, C_{l-1} and C_l are the input and output channel counts, respectively, and $b^{(l)} \in R^{C_l}$ is the bias term. To preserve spatial resolution, same-padding is applied such that:

Figure 1

Model framework: (a) Input layer: Data distribution (b) CNN (c) GNN (d) Alignment (e) Loss function.



$$L_{dist} = \frac{1}{B^2}\sum_{i,j}\left\|z_i - z_j\right\|_2^2 - \left\|\tilde{h}_i - \tilde{h}_j\right\|_2^2 . \quad (2)$$

Ensuring that a deeper layer focuses on abstract features without losing the spatial structure. After the convolution operation, batch normalization (BN) stabilizes the training process by reducing internal covariate variations, accelerates convergence, and suppresses internal covariate shifts:

$$\bar{Y}_i^{(I)}[u,v,k] = \frac{Y_i^{(l)}[u,v,k] - \mu_k^{(l)}}{\sqrt{\left(\sigma_k^{(l)}\right)^2 + Ú}}\gamma_k^{(l)} + \beta_k^{(l)} \quad (3)$$

where $\mu_k^{(l)}$ and $\sigma_k^{(l)}$ are the mean and variance of the k-th channel, $\gamma_k^{(l)}$ and $\beta_k^{(l)}$ are learnable scaling and shifting parameters, and ò ensures numerical stability. The normalized features are passed through a ReLU activation to introduce non-linearity:

$$X_i^{(I)}[u,V,k] = \max\left(0, \bar{Y}_i^{(I)}[u,V,k]\right). \quad (4)$$

This activation introduces non-linearity to the network, allowing it to learn more complex features.

2.2. Channel Recalibration with Squeeze-and-Excitation (SE) Module

Due to the varying importance of different channels (feature maps) for the final task, the "squeeze and excitation" (SE) module, as a "lightweight insertion block", is integrated immediately after each Conv-BN-ReLU and placed within the CNN architecture. It first performs global average pooling, then two layers of fully connected networks with Sigmoid activation, and finally weights the original features by channel to enable the network to learn to enhance proper channels and suppress redundant ones adaptively. The module consists of two stages. For the "squeeze" part, global average pooling will aggregate the spatial information of each channel into a compact descriptor:

$$s_k = \frac{1}{H_l W_l}\sum_{u=1}^{H_l}\sum_{v=1}^{W_l} X_i^{(I)}[u,v,k], \quad k = 1,2,\ldots,C_l . \quad (5)$$

This descriptor $s \in R^{C_l}$ captures the global importance of each channel. For the Excitation item, two fully connected (FC) layers learn attention weights to emphasize informative channels. The first FC layer compresses the descriptor to C_l/r dimensions (with reduction ratio r):

$$z = W_1 s + b_1, W_1 \in R^{(C_l/r)\times C_l}, b_1 \in R^{C_l/r} . \quad (6)$$

Followed by ReLU activation:

$$\delta(z)_i = \max\left(0, z_i\right), i = 1,2,\ldots,\frac{C_l}{r}. \quad (7)$$

The second FC layer expands the dimensionality back to C_l:

$$u = W_2\delta(z) + b_2, W_2 \in R^{C_l\times(C_l/r)}, b_2 \in R^{C_l} . \quad (8)$$

Sigmoid activation yields attention weights $e_k \in [0,1]$:

$$e_k = \frac{1}{1+\exp\left(-u_k\right)}. \quad (9)$$

For the Recalibration item, the feature map is scaled by the attention weights:

$$\hat{X}_i^{(I)}[u,v,k] = e_k \cdot X_i^{(I)}[u,v,k] . \quad (10)$$

This operation allows the qnetwork to emphasize more informative channels and suppress irrelevant ones, thereby improving feature representation.

2.3. Global Context Modeling via Multi-Head Self-Attention

After the channel-level recalibration of the SE module (Equation (10)), since CNN can only model a limited receptive field, while self-attention can capture the relationship between any two points across the entire image, applying the multi-head self-attention mechanism to the recalibrated features $\hat{X}^{(l)} \in R^{H_l\times W_l\times C_l}$ enables the model to focus on multiple patterns or semantics from different subspaces simultaneously. Firstly, it is reshaped into a sequence

of length. H_lW_l and embedding dimensi C_l. The projection calculations for queries, keys, and values are as follows:

$$Q = \hat{X}^{(I)}W^{Q}, K = \hat{X}^{(I)}W^{K}, V = \hat{X}^{(I)}W^{V}, \quad (11)$$

where $W^{Q}, W^{K}, W^{V} \in R^{C_l \times d_k}$. Scaled dot-product attention is then applied:

$$\text{Attention}(Q,K,V) = \text{softmax}\left(\frac{QK^{\cdot}}{\sqrt{d_k}}\right)V \cdot \quad (12)$$

and multi-head attention aggregates h parallel heads:

$$\begin{aligned}&\text{MultiHead}\left(\hat{X}^{(I)}\right) = \text{Concat}\left(\text{head}_1,\ldots,\text{head}_h\right)W^{0},\\ &\text{head}_i = \text{attention}\left(QW_i^{Q}, KW_i^{K}, VW_i^{V}\right)\end{aligned}, \quad (13)$$

where $W_i^{Q}, W_i^{K}, W_i^{V} \in R^{C_l \times d_k}$ and $W^{O} \in R^{hd_k \times C_l}$. The output $\tilde{X}^{(l)} \in R^{H_l \times W_l \times C_l}$ (after reshaping) fuses local channel importance and global spatial correlations.

2.4. Graph Neural Network (GNN) for Spatial Relationship Learning

The features after attention enhancement are input into a graph neural network, where each spatial position after attention enhancement is regarded as a graph node, facilitating the propagation of context information in a non-Euclidean space, as shown in Figure 1(c). It is to simulate the spatial relationships between local regions (graph nodes). The process flow is as follows. For affinity graph construction settings, the radial basis function (RBF) kernel is used to calculate the pairwise affinity between nodes:

$$\phi(i,j) = \exp\left(-\frac{\left\|\mathbf{z}_i - \mathbf{z}_j\right\|^2}{2\sigma_i^2}\right), \quad (14)$$

where σ_i is a measure of local density, computed based on the distances between the nodes and forming an affinity matrix $\Phi \in R^{N \times N}$:

$$\Phi = [\phi(i,j)]_{i,j=1}^{N} \in R^{N \times N}. \quad (15)$$

Sparsification via k-Nearest Neighbors (k-NN) retains top-k bidirectional edges:

$$\begin{aligned}N_k(i) = \{j |\ & j \in \text{top}-\text{k}(\phi(i,j))\\ & \text{and } i \in \text{top}-\text{k}(\phi(j,i))\}\end{aligned}. \quad (16)$$

Yielding a symmetric adjacency matrix W:

$$W_{ij} = \begin{cases}\sqrt{\phi(i,j)\cdot\phi(j,i)}, & j \in N_k(i)\\ 0, & \text{otherwise}\end{cases}. \quad (17)$$

Then, the adjacency matrix is normalized as:

$$\tilde{A} = D^{\frac{1}{2}}(W+I)D^{\frac{1}{2}}, \quad (18)$$

where D is the degree matrix and I is the identity matrix. Message aggregation and node feature update are defined as:

$$M^{(I)} = \tilde{A}H^{(I)}W_1^{(I)} \quad (19)$$

$$H^{(I+1)} = \sigma\left(H^{(I)}W_2^{(I)} + M^{(I)} + \mathbf{1}b^{(I)^{\cdot}}\right). \quad (20)$$

This process enables the network to aggregate features from neighboring nodes, capturing contextual information and refining the node embeddings in the process.

2.5. Global Subspace Estimation via Secure Federated PCA

To obtain the global target subspace $U \in R^{d_L \times r}$ in a distributed manner, we adopt a secure federated PCA protocol that never centralizes raw data.

At node i, let $H_i^{(L)} \in R^{n_i \times d_L}$ denote the final embedding matrix after CNN--SE--Attention--GNN, where n_i is the number of local samples and d_L is the feature dimension. Each node computes its local covariance.

$$S_i = \frac{1}{n_i}\left(H_i^{(L)}\right)^{ï} H_i^{(L)} \in \sim^{d_L \times d_L}. \quad (21)$$

Instead of transmittin S_i directly, each node participates in a secure aggregation (SA) protocol so that the server only receives the aggregated covariance.

$$S=\sum_{i=1}^{N}S_i=\frac{1}{n}\sum_{i=1}^{N}\left(H_i^{(L)}\right)^{T}H_i^{(L)},n=\sum_{i=1}^{N}n_i. \quad (22)$$

without observing any individual S_i or raw samples.

Thus no local statistics are leaked.

Computing the eigenvectors of S directly would require sharing the full $d_L \times d_L$ matrix. Instead, we approximate the global subspace U through power iteration with SA. With a public random seed, initialize.

$$q^{(0)}\in \sim^{d_L} \text{ with } \left\|q^{(0)}\right\|_2=1. \quad (23)$$

At each iteration t:

$$u_i^{(t)}=S_iq^{(t)}=\frac{1}{n_i}\left(H_i^{(L)}\right)^{ï}\left(H_i^{(L)}q^{(t)}\right) \quad (24)$$

$$u^{(t)}=SA\left(\sum_{i=1}^{N}u_i^{(t)}\right),q^{(t+1)}=\frac{u^{(t)}}{\left\|u^{(t)}\right\|_2}. \quad (25)$$

After convergence, the leading eigenvector $q^{(\infty)}$ is obtained. Deflating S and repeating this procedure yields the top-r eigenvectors, which form the global subspace

$$U=\left[q_1,q_2,\ldots,q_r\right]\in \sim^{d_L\times r}. \quad (26)$$

Each iteration only requires transmitting $u_i^{(t)}\in R^{d_L}$, so the communication cost is $O(d_L)$ per round under SA.

Since $r \ll d_L$, the overall bandwidth for estimating U is negligible compared to raw feature exchange.

Moreover, secure aggregation guarantees that the server only learns the summed vector $u^{(t)}$, not any individual contribution.

2.6. Procrustes Alignment and Loss Function

The final graph neural network embedding $H^{(L)}\in R^{timesd_L}$ is aligned with the target subspace $U\in R^{N\times r}$ Through the Procrustes alignment method. This enables the output low-dimensional node embedding to not only precisely align with the predefined subspace U, but also to maintain the geometric relationships among the original high-dimensional features while enhancing the smooth consistency of adjacent nodes on the graph and suppressing the redundancy of the embedding dimensions. The aim is to find a transformation. $R^*\in R^{d_L\times r}$ such that the following equation reaches its minimum value as shown in Figure 1(d):

$$R^*=\arg\min_{R^{ï}\ R=I_r}\|HR-U\|_F^2. \quad (27)$$

The optimal transformation via Singular Value Decomposition (SVD) of $C=H^*U$ is:

$$R^*=VW^{ï}. \quad (28)$$

Yielding aligned embeddings:

$$H=HR^*. \quad (29)$$

The loss function for training the model consists of three components: distance preservation, Laplacian smoothing, and orthogonality regularization as shown in Figure 1(e).

1 Distance Preservation ensures that the distances between the original high-dimensional features and the aligned low-dimensional features are preserved:

$$L_{dist}=\frac{1}{B^2}\sum_{i,j}\left\|z_i-z_j\right\|_2^2-\left\|\tilde{h}_i-\tilde{h}_j\right\|_2^2. \quad (30)$$

2 Laplacian Smoothing encourages smoothness in the graph structure by penalizing the features' inconsistency across the graph:

$$L_{smooth}=\frac{1}{B^2}\text{trace}\left(\tilde{H}^{ï}\ L\tilde{H}\right). \quad (31)$$

where L is the normalized Laplacian matrix.

3 Orthogonality Regularization enforces the orthogonality of the aligned feature matrix to avoid redundancy:

$$L_{ortho}=\frac{1}{r^2}\left\|\tilde{H}^{ï}\ \tilde{H}-I_r\right\|_F^2. \quad (32)$$

Then, the total compression loss is:

$$L_{comp}=\lambda_{dist}L_{dist}+\lambda_{smooth}L_{smooth}+\lambda_{ortho}L_{ortho}. \quad (33)$$

2.7. Distributed Learning with Federated Averaging

Since each node completes the computationally intensive CNN+SE feature extraction locally and only exchanges compact embedding vectors through low-bandwidth channels, it then performs GNN aggregation, Procrustes alignment, and linear reconstruction locally. Consequently, there is no need to store any original or intermediate high-dimensional features in a central location. Only small-scale vectors need to be transmitted, which fully conforms to the distributed learning paradigm. The message passing of GNN and Procrustes alignment can be achieved either through asynchronous peer-to-peer (P2P) communication or with the help of a lightweight central server, thus enabling multi-node collaboration without the centralized aggregation of all data. Under this mode, the research method minimizes communication overhead while maintaining the performance of dimensionality reduction and reconstruction, fully meeting the requirements of efficiency and scalability in distributed environments, as shown in Figure 2. Therefore, the study first defines that each client or node optimizes its local compression loss:

$$\mathrm{L}_i\left(\theta_i\right)=\lambda_{\text{dist}}\,\mathrm{L}_{\text{dist}}^{(i)}+\lambda_{\text{smooth}}\,\mathrm{L}_{\text{smooth}}^{(i)}+\lambda_{\text{ortho}}\,\mathrm{L}_{\text{ortho}}^{(i)}, \quad (34)$$

where θ_i are local parameters. The global objective aggregates local losses:

$$\mathrm{L}_{\text{global}}\left(\{\theta_i\}\right)=\sum_{i=1}^{N}\frac{n_i}{n}\mathrm{L}_i\left(\theta_i\right), \quad (35)$$

with $n=\sum n_i$. At each communication round, parameters are updated via Federated Averaging:

$$\theta^{t+1}=\sum_{i=1}^{N}\frac{n_i}{n}\theta_i^t, \quad (36)$$

and GNN message passing only requires exchanging neighbor embeddings:

$$H_i^{(I+1)}=\sigma\left(W_2^{(I)}H_i^{(I)}+\sum_{j\in N_k(I)}\tilde{A}_{ij}W_1^{(I)}H_j^{(I)}+b^{(I)}\right). \quad (37)$$

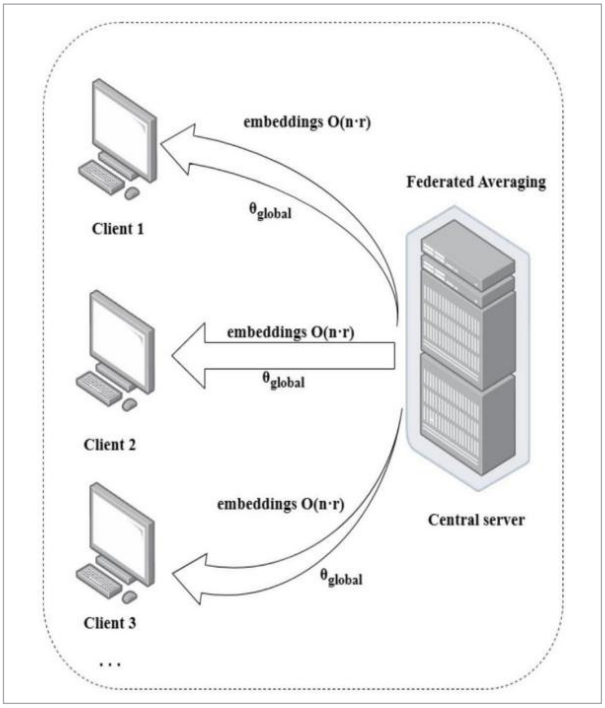
Communication savings are quantified as:

$$\text{Comm}_{\text{ours}}=O(nr)=\frac{r}{p}O(np)=\frac{r}{p}\,\text{Comm}_{\text{orig}}. \quad (38)$$

In practice, the global subspace U is refreshed every T_U local epochs using the secure federated PCA protocol (Section 2.5). Each power-iteration step communicates only $O(d_L)$ values per node, so the overall complexity is $O(Md_LRr)$ when M nodes participate. Meanwhile, the embedding exchange in federated averaging costs only O(Mr), which is negligible since $r \ll d_L$. Thus, even in the regime $M \gg r$, the communication bottleneck scales linearly with M but only weakly with r, and estimating U does not violate the no-central-data premise. With r"p reducing bandwidth by p/r. The distributed training process of DIST-C-SEGAN is summarized in Algorithm 1, encompassing local feature learning (CNN+SE, attention, GNN), alignment, loss computation, and federated parameter aggregation. This end-to-end framework optimizes all components jointly, achieving efficient feature compression and distributed collaboration with minimal communication overhead.

Figure 2

The communication process of DIST-C-SEGAN in a distributed environment.



Algorithm 1: Distributed DIST-C-SEGAN

Require: H, W, C, input X_i, CNN params (K, S, P), BN params ($\gamma, \beta, \in$), SE ratio r, attention heads h, d_k, GNN params (k), Procrustes target U, optimizer (η, λ_{wd}), loss weights (λ_{dist}, λ_{smooth}, λ_{ortho})

Ensure: Aligned embeddings $\hat{X}_i$

Init: initialise local parameters θ_i

for all epoch do

for all node i in parallel do

(a) CNN+SE :

Feature extraction via Equations (1)-(4)

Channel recalibration via Equations (5)-(10)

(b) Self-Attention:

Compute attention via Equations (11)-(13) to obtain $\widetilde{X}^{(l)}$

(c) GNN Propagation:

Build affinity graph via Equations (14)-(18)

Message aggregation via Equations (19)-(20)

(d) Procrustes Alignment:

Solve Equations (21)-(23) for R* and compute $\hat{X}_i$

(e) Local Loss & Update:

Compute compression loss L_i via Equation (24)-(27)

Update local θ_i by optimizer

end for

(f) Federated Averaging:

$$\theta \leftarrow \sum_{i=1}^{N} \frac{n_i}{n} \theta_i (\text{Equations (30)})$$

Broadcast θ to all nodes

end for

In conclusion, the DIST-C-SEGAN framework proposed in this study aims to achieve efficient and secure image feature compression and collaborative learning in a multi-node environment. Firstly, each node locally extracts hierarchical local features through CNN+BN+ReLU. Then, the SE module is inserted to recalibrate the channel weights adaptively. Next, multi-head self-attention is utilized to capture global spatial dependencies, and the recalibrated features are regarded as graph nodes. Through GNN message passing, the structured relationships between different regions are explicitly modeled. Subsequently, the Procrustes alignment is used to map the GNN low-dimensional embeddings to a predefined subspace. Through the joint constraints of distance preservation, Laplacian smoothing, and orthogonal regularization, the low-dimensional representation retains the original geometric structure without redundancy. Finally, in the federated learning framework, nodes only exchange small-scale r-dimensional embedding vectors (the communication volume is reduced from mathcalO(np) to mathcalO(nr)). Through Federated Averaging, the global model is updated via aggregation, achieving an organic integration of local and international feature complementation, channel adaptation, structured propagation, and geometric decoupling, which significantly improves the efficiency and scalability of distributed high-dimensional image processing.

3. Data Analysis

To systematically evaluate the performance of DIST-C-SEGAN from multiple dimensions, we designed three complementary experiments, focusing on data selection, model components, and downstream applications. The aim was to identify the optimal test scenarios at the macro level (data distribution differences), delve into the internal mechanisms of the model at the micro level (hyperparameters and module contributions), and verify the practical value of the framework at the application level (clustering downstream tasks), thereby comprehensively demonstrating the outstanding performance of DIST-C-SEGAN in distributed, high-dimensional image compression and collaborative learning. Firstly, in the "Dataset Comparison" experiment, we evaluated four representative image benchmarks - AffectNet-master, CIFAR-10, SVHN, and Places365 - under a fixed compression ratio. By comparing indicators such as credibility, continuity, and variances of explanations, the study determined the best dataset to be used in subsequent research. Secondly, in the "Hyperparameter Sensitivity and Ablation" experiment, we changed key parameters, including the distance-preserving weight λ, the number of graph neighbors k, and the Laplacian smoothing coefficient λ_{smooth}, and the orthogonality regularization coefficient λ_{ortho}. Additionally, we conducted two ablation analyses by removing the SE module and the Procrustes alignment layer, quantifying the impact of each hyperparameter and key module on the model performance. Finally,

in the baseline comparison and downstream clustering experiments, we compared DIST-C-SEGAN with classic linear methods (such as Principal Component Analysis and Kernel Principal Component Analysis) and popular non-linear techniques (such as Autoencoders, Variational Autoencoders, UMAP, and t-SNE). We also applied clustering algorithms to the low-dimensional embeddings of the selected datasets (evaluated using the silhouette coefficient, ARI, and NMI metrics), fully verifying the relative advantages of the research method and its practical application value in downstream analysis.

In this study, the CNN architecture serves not only for dimensionality reduction but also effectively extracts hierarchical feature representations from images. Initially, the input image is represented as a tensor with three channels (i.e., an RGB image). This image tensor is processed through different layers of the CNN, progressively extracting features from low-level to high-level image representations. At each layer, the CNN applies convolution operations to process the feature maps from the previous layer. The convolution kernel at the l-th layer has a height of h_l and a width of w_l, where h_l and w_l are the kernel's height and width, respectively. The stride is set to 1, and padding is also set to 1, ensuring that the spatial dimensions of the output feature map match those of the input, thereby maintaining the spatial structure of the features at each layer. After the convolution operation, the ReLU activation function is applied to introduce non-linearity, which helps increase the network's capacity to learn more complex image features. Subsequently, a pooling layer (Max Pooling) is used to reduce the spatial dimensions of the feature map, thereby decreasing computational complexity. A 2× max pooling kernel with a stride of 2 is employed, which halves the spatial dimensions of the feature map while retaining the most significant features. Furthermore, to enhance inter-channel dependencies, a Squeeze-and-Excitation (SE) module is introduced. In the Squeeze stage of the SE module, each channel of the image is compressed into a scalar via global average pooling, representing the global information of each channel. In the Excitation stage, two fully connected (FC) layers are used to learn the attention weights for each channel: the first layer compresses the descriptors, and the second layer expands them back to the original number of channels. The Sigmoid activation function is applied to compute the attention coefficient for each channel, and these weights are then applied to the feature maps to dynamically reweight each channel, allowing the network to adjust the importance of each channel adaptively.

Initially, the input image undergoes convolution operations through each layer, where the weights of each convolution kernel. $W^{(l)}$ are updated during training using the Adam optimizer. This optimizer automatically adjusts the learning rate for each weight based on its gradient, an adaptive learning rate optimization strategy that accelerates network convergence and ensures a more stable training process. The weight update equations are as follows:

$$m_t = \beta_1 m_{t-1} + (1-\beta_1)\frac{\partial \mathrm{L}}{\partial W^{(l)}} \tag{39}$$

$$v_t = \beta_2 v_{t-1} + (1-\beta_2)\left(\frac{\partial \mathrm{L}}{\partial W^{(l)}}\right)^2 \tag{40}$$

$$\hat{m}_t = \frac{m_t}{1-\beta_1^t}, \hat{v}_t = \frac{v_t}{1-\beta_2^t} \tag{41}$$

$$W^{(l)} \leftarrow W^{(l)} - \eta \frac{\hat{m}_t}{\sqrt{\hat{v}_t} + \acute{\mathrm{U}}}, \tag{42}$$

where m_t and v_t are the first and second moment estimates of the gradients, β_1 and β_2 are the decay rates, and Ú is a small constant to prevent division by zero. Through this approach, the Adam optimizer efficiently updates the weights and automatically adjusts the learning rate for each parameter during training.

To prevent overfitting during training, Elastic Net regularization is introduced, which combines both L1 regularization (Lasso) and L2 regularization (Ridge) by applying penalties to the weights to control the model's complexity. L1 regularization helps produce sparse models by pushing less important weights toward zero, while L2 regularization prevents weights from becoming too large, ensuring network stability. By employing these methods, we effectively extract hierarchical features from images and ensure feature sparsity and importance through weight adjustment and regularization. Furthermore, the introduction of

channel attention allows the network to reweight features across multiple layers, improving the model's performance and generalization ability. The weights of each convolution kernel and fully connected layer are optimized via backpropagation and gradient descent algorithms, ultimately achieving efficient image feature extraction and learning.

3.1. Dataset Comparison Experiment

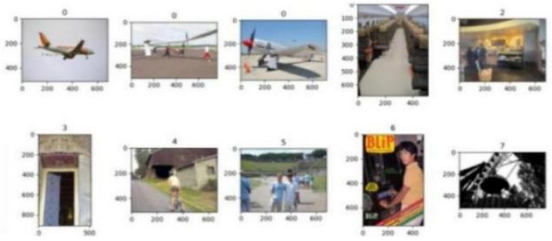
The study selected four representative image benchmarks: AffectNet-master (expression recognition), CIFAR-10 (small object classification), SVHN (street scene digitization), and Places365 (scene recognition), as shown in Figure 3. Under a unified compression ratio (CompRatio=588), the three indicators of Trustworthiness (local neighborhood structure fidelity), Continuity (global manifold continuity), and Explained Variance (information retention) were calculated, respectively. By comparing the performance on these datasets, we can determine which type of image distribution is most conducive to the high-fidelity reconstruction of DIST-C-SEGAN while considering both local and global structures. We can evaluate the model's generalization ability in various visual task scenarios.

As shown in Table 1, we first evaluated the original DIST-C-SEGAN model on four benchmark image datasets. With the compression ratio remaining constant at 588, there were significant differences in reliability and continuity: SVHN had the highest re-

Figure 3
Types of test data.

(a) The CIFAR-10 dataset

(b) SVHN (Street View House Numbers)



(c) Places365

(d) AffectNet-master

Table 1
Comparison results of different datasets under the original model.

Data type	Trustworthiness	Continuity	CompRatio	Explained Variance
AffectNet-master	0.8578	0.8119	588	0.9853
The CIFAR-10 dataset	0.8351	0.7781	588	0.9729
SVHN	0.8627	0.7694	588	0.9681
Places365	0.8306	0.7589	588	0.9794

liability (0.8627), but was slightly behind in continuity (0.7694), indicating that although the local neighborhood structure was well preserved, the global manifold was somewhat distorted. In contrast, AffectNet-master achieved an almost optimal balance (reliability = 0.8578, continuity = 0.8119), and the explanation variance also reached the highest value (0.9853), suggesting that it performed better in preserving both fine and coarse features. The two datasets, CIFAR-10 and Places365, which represent medium complexity and high scene diversity, respectively, had slightly lower reliability (0.8351 and 0.8306), but still maintained a strong overall variance retention ability (0.9729 and 0.9794). These results indicate that DIST-C-SEGAN can consistently provide high-quality, near-lossless compression across various image domains; therefore, AffectNet-master was selected as the test set for the following experiments.

3.2. Hyperparameter Sensitivity, Baseline Comparison and Ablation Experiments

In this experiment, the research first adjusted four key hyperparameters the distance-preserving weight λ, the graph adjacency number k, the Laplacian smoothing coefficient. λ_{smooth}, and the orthogonal regularization coefficient λ_{ortho}, and recorded the changes in credibility and continuity, as well as explained the variance brought about by these adjustments. It enabled us to quantify the impact of each parameter on the model performance and determine the optimal combination. We compared DIST-C-SEGAN with eight classic linear and non-linear dimensionality reduction methods, such as PCA, Kernel PCA, AE, VAE, Denoising AE, UMAP, t-SNE, and LLE, focusing on the comprehensive advantages of the model in terms of structural fidelity (Trustworthiness, Continuity) and information retention (Explained Variance). Subsequently, we conducted ablation studies by removing the squeezing and excitation modules ("-SE") and the Procrustes alignment layer ("-al"). By comparing the significant decline in performance before and after these removal operations, we confirmed the necessity of channel attention and geometric alignment for maintaining the consistency of high-dimensional structures and the reconstruction accuracy.

In Table 2, the research systematically investigated the sensitivity of the model performance to key hyperparameters: distance-preserving weight (λ), graph connectivity index (k), smooth regularizer (λ_{smooth}),

Table 2
Hyperparameter experimentation results.

Experiment		Trustworthiness	Continuity	Explained Variance
λ	λ=1	0.8071	0.8004	0.9794
	λ=2	0.8578	0.8119	0.9853
	λ=4	0.7907	0.7585	0.9527
k	k=13	0.7716	0.8005	0.9652
	k=15	0.8578	0.8119	0.9853
	k=17	0.7907	0.7585	0.9172
λ_{smooth}	λ_{smooth}=0.5	0.7822	0.8102	0.9527
	λ_{smooth}=1	0.8578	0.8119	0.9853
	λ_{smooth}=2	0.7727	0.7743	0.9362
λ_{ortho}	λ_{ortho}=0.005	0.8141	0.5635	0.9601
	λ_{ortho}=0.01	0.8578	0.8119	0.9853
	λ_{ortho}=0.1	0.8269	0.7923	0.9545

and orthogonality regularizer (λ_{ortho}). Under the settings of λ=2, k=15 and λ_{smooth}=1, λ_{ortho}=0,01 the best performance was achieved (confidence =0.8578, continuity =0.8119, explained variance =0.9853). Deviating from these values would lead to a significant decline in performance. For instance, λ=0.005 reduced the continuity to 0.5635, highlighting the necessity of a sufficiently strong orthogonality constraint to maintain the coherence of the global manifold; similarly, k=13 or k=17 reduced the explained variance by up to 2 percentage points, revealing that the optimal range of graph sparsity is relatively narrow. By comparing the significantly declined performance with other values, the model's sensitivity to these hyperparameters was verified, providing a reliable basis for parameter tuning in practical applications.

Table 3

Results of controlled comparison experiments.

Dimensionality reduction methods	Trustworthiness	Continuity	Explained Variance
AE	0.6206	0.5902	0.7536
VAE	0.549	0.5437	0.8142
Denoising AE	0.5097	0.4708	0.8632
PCA	0.7354	0.7581	0.8195
Kernel PCA	0.6122	0.6338	0.7367
UMAP	0.8029	0.8119	0.7456
t-SNE	0.8351	0.7781	0.7355
LLE	0.8306	0.7589	0.7039
UDRN	0.8473	0.8105	0..8364
DIST-C-SEGAN	0.8578	0.8119	0.9853

Table 3 compares DIST-C-SEGAN against eight classical and deep non-linear dimensionality reduction methods on CIFAR-10 embeddings. While PCA variants (PCA and Kernel PCA) achieved moderate Explained Variance (0.8195 and 0.7367), their Trustworthiness and Continuity were inferior (≤0.7581), indicating limited capacity for structure preservation. Autoencoder-based methods (AE, VAE, Denoising AE) prioritized reconstruction, with Explained Variance up to 0.8632, but compromised severely on manifold integrity (Trustworthiness ≤0.6206, Continuity ≤0.5902). Popular manifold learners such as UMAP and t-SNE improved Trustworthiness (0.8029 and 0.8351) but still capped Explained Variance below 0.746. The framework UDRN, which has a similar structure to DIST-C-SEGAN, effectively retains a large amount of information. However, in comparison, DIST-C-SEGAN simultaneously achieves the highest credibility (0.8578), continuity

Table 4

Ablation study results.

Experiment	Trustworthiness	Continuity	Explained Variance
Original	0.858	0.812	0.985
-SE (no Squeeze-and-Excitation)	0.571	0.601	0.797
-al (no Procrustes alignment)	0.582	0.563	0.701
-GNN (no Squeeze-and-Excitation)	0.329	0.416	0.698
-CNN (no Squeeze-and-Excitation)	0.451	0.532	0.721

(0.8119), and interpretability variance (0.9853) values, validating its capacity for "near-lossless" compression that neither linear nor existing non-linear methods can match.

As summarized in Table 4, removing any single component degrades all three metrics relative to the full model, with the steepest declines observed when ablating the GNN or the CNN—highlighting that relational propagation and hierarchical local descriptors are primary drivers of neighborhood fidelity and manifold continuity. Eliminating Procrustes alignment produces a broader erosion, especially in explained variance, underscoring the importance of anchoring embeddings to a shared subspace; removing the SE block yields a milder yet consistent drop, indicating that channel recalibration remains beneficial. Overall, the contributions rank as GNN > CNN > alignment > SE for trustworthiness, GNN > CNN > alignment > SE for continuity, and GNN ≈ alignment > CNN > SE for explained variance, confirming the complementary roles of local feature extraction, channel reweighting, global relation propagation, and geometric alignment in sustaining compact yet faithful embeddings.

Table 5
Edge-device inference profile (RTX 4070 Laptop GPU, 8 GB).

Metric	Value	Notes
Mean latency (ms / image)	350.58	CUDA events, synchronized
p50 / p90 / p95 (ms)	350.58 / 378.63 / 403.17	Long-tail jitter captured
Std. dev. (ms)	0.06	Across measured batches
Peak GPU memory (MB)	1999.95	From torch.cuda.max_memory_allocated
Peak system R.AM (MB)	5966.14	From psutil RSS

After verifying the validity of the model, in order to validate the deployability of this method on ordinary edge devices, the inference performance and memory consumption were tested on a commercial notebook-level GPU (NVIDIA GeForce RTX 4070 Laptop, 8 GB). The test environment was Windows 10, PyTorch 2.2.2 + cu118, with a 32-core Intel processor as the CPU and 15.63 GB of system memory. We used CUDA events for precise timing and separately counted the GPU memory and system RAM peak values through torch.cuda.max_memory_allocated and psutil. The experimental results (Table 5) show that the average inference delay for a single image was 350.58 ms, with p50 = 350.58 ms, p90 = 378.63 ms, p95 = 403.17 ms, and the standard deviation was only 0.06 ms, indicating a stable delay distribution. The peak GPU memory usage during the inference stage was 1999.95 MB, accounting for approximately 25% of the total 8 GB GPU memory, and the system RAM peak was 5966.14 MB. These results indicate that DIST-C-SEGAN has low memory and memory consumption on edge-level devices, and the inference delay is mainly affected by affinity construction. Overall, it is feasible to deploy in resource-constrained environments.

In addition, the study quantify bandwidth by counting transmitted scalars per round. On a validation batch with n=36 images of size 224×224×3 (original dimension p=150,528) and an aligned embedding of r=256, DIST-C-SEGAN transmits nr=9,216 values (≈0,04 MB at 32-bit), whereas FedPCA/FedAE each require np=5,419,008 values (≈20.67 MB). This corresponds to a 99.8 reduction in per-round bandwidth. The result aligns closely with the theoretical ratio $1 - \frac{r}{p}$: with r/p = 256/150,528 ≈ 0.0017, the predicted saving is 99.83, closely matching the observed 99.8. Since the cost scales as O(nr) for DIST-C-SEGAN versus O(np) for the baselines, the savings grow linearly with n and are substantial whenever $r \ll p$. In terms of privacy resilience, PSNR = 21.9546 dB and SSIM = 0.6973 indicate a good balance between feature preservation for downstream tasks and privacy protection. The reconstructed images are sufficiently distorted to provide strong privacy while retaining enough structure for practical use. This suggests that DIST-C-SEGAN is on the right track in terms of privacy. Although further lowering PSNR and SSIM could enhance privacy, it may negatively impact downstream task performance, reflecting the trade-off between privacy protection and task efficacy.

3.3. Repeated Experiment and Downstream Clustering

In this section of the experiment, we first examined the robustness of DIST-C-SEGAN to weight initialization by repeating the training with five different

random seeds: in multiple runs, the mean and varianc of Trustworthiness, Continuity, and Explained Variance remained at extremely low levels, indicating that the model performance is stable and reproducible under different initial conditions; subsequently, the lowdimensional embeddings generated by it were used for downstream clustering tasks, and indicators such as Silhouette, ARI, and NMI were calculated, and compared with classical methods (AE, VAE, PCA, UMAP, t-SNE, etc.) and ablation versions to verify that DIST-C-SEGAN can significantly enhance clustering cohesion and classification effectiveness while maintaining high-fidelity reconstruction (97.62% Explained Variance), and by removing the SE module or performing Procrustes alignment, the clustering performance significantly declined, further demonstrating the necessity of channel attention and geometric alignment for generating low-dimensional representations with high discriminative power.

The robustness of the model to initialization is quantified in Table 6, where the training process is repeated with five distinct random seeds. The mean values of Trustworthiness (0.86158), Continuity (0.81246), and Explained Variance (0.98134) exhibit extremely low variance, which demonstrates that the performance of DIST-C-SEGAN is stable and reproducible regardless of the weight initialization. This stability is of great significance for practical deployment, as it ensures consistent compression quality without the need for repeated hyperparameter searches or ensemble averaging.

Table 6
Results of repeated experiments with different random seeds.

Seed	Trustworthiness	Continuity	Explained Variance
0	0.8578	0.8119	0.9853
1	0.8684	0.8022	0.9898
2	0.8597	0.8214	0.9774
3	0.8757	0.8187	0.9821
4	0.8463	0.8081	0.9721
Mean	0.86158	0.81246	0.98134
Variance	0.00009944	0.00004869	0.00003775

Finally, Table 7 presents the downstream clustering results of the complete DIST-C-SEGAN model

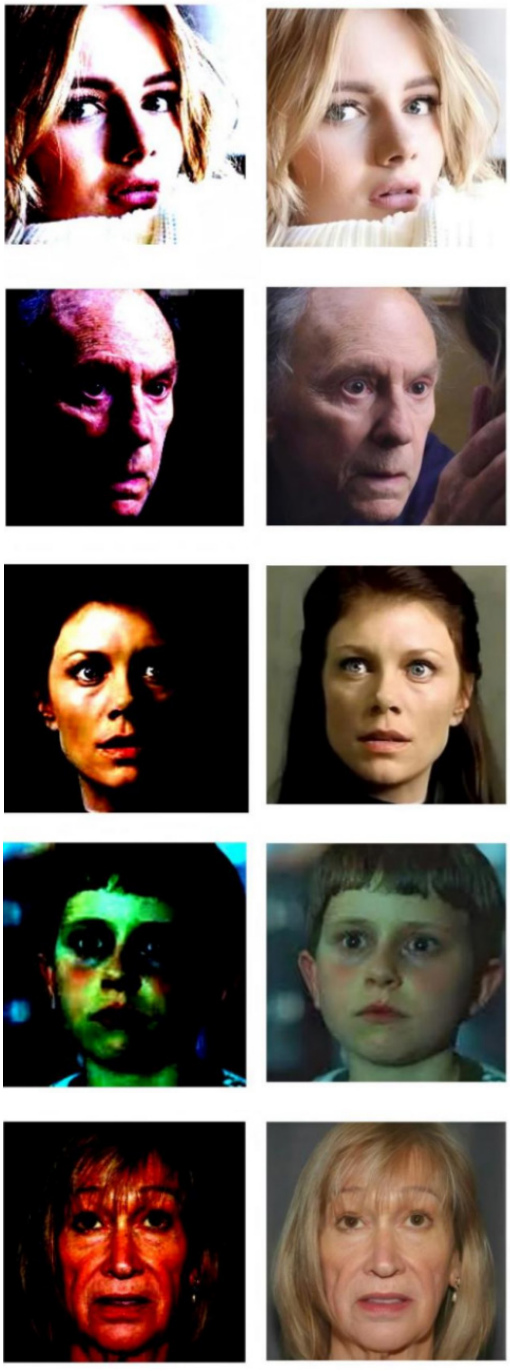
Table 7
Cluster analysis results for downstream tasks.

Experiment	Model	Silhouette	ARI	NMI	Explained Variance
Controlled experiment	DIST-C-SEGAN	0.1305	0.0939	0.2966	0.9762
	AE	0.2135	0.0432	0.1585	0.6786
	VAE	-0.2129	-0.0173	0.0599	0.7893
	Denoising AE	-0.1653	-0.0043	0.1571	0.7962
	PCA	-0.1647	-0.007	0.1567	0.8001
	Kernel PCA	-0.1954	-0.0066	0.0810	0.8360
	UMAP	0.3762	0.0838	0.0756	0.8574
	t-SNE	0.5439	0.0736	0.0757	0.7981
	LLE	0.7584	0.006	0.1644	0.9136
Ablation experiment	baf	0.7598	0.0076	0.147	0.9105
	SE	0.761	0.0079	0.2842	0.9377
	al	0.7596	0.0082	0.0731	0.8564

and the four ablation variants. In the control experiment, DIST-C-SEGAN achieved the best balance between intra-cluster cohesion (coefficient of variation =0.1305) and external validity (adjusted mutual information index =0.0939, normalized mutual information index =0.2966), while retaining 97.62% of the information. Compared with AE/VAE and linear methods (which often produce negative coefficient of variation values or ARI values lower than 0.1), our method significantly improves the separability of clustering without sacrificing the fidelity of reconstruction. The ablation study further indicates that deleting the SE module or Procrustes alignment layer leads to a 10-20% decrease in clustering metrics, highlighting the indispensable role of channel attention and geometric alignment in capturing local and global structures.

Figure 4 illustrates five representative examples of our "near-lossless" compression and reconstruction pipeline. This term "near-lossless" refers to the minimal loss of visual fidelity (less than 2% of the variance) that is imperceptible to human observers and does not degrade the utility of the data for downstream tasks. In each row, the left sub-figure shows the image reconstructed from its low-dimensional embedding generated by DIST-C-SEGAN, and the right sub-figure shows the corresponding original input. Across all examples, the reconstructed images retain salient facial structures, such as the contours of eyes, nose, and mouth, with high fidelity. Fine details, including skin texture and hair strands, remain largely intact, which demonstrates that the lightweight linear projection layer is sufficient to-recover critical visual information. Minor artifacts can be observed in areas with very low contrast (e.g., forehead highlights in row 2) or complex textures (e.g., hair in row 5), but these do not affect the overall perceptual quality. Quantitatively, these qualitative results are consistent with the high Explained Variance (97.62%) and strong manifold metrics reported in Tables 1-6. Together, they confirm that DIST-C-SEGAN achieves near-near-lossless reconstruction while facilitating practical downstream analysis. Collectively, these rigorous analyses confirm that DIST-C-SEGAN outperforms existing techniques in terms of compression quality, structural fidelity, robustness, and downstream utility, thereby establishing a new benchmark for dimensionality reduction of distributed, near-lossless images.

Figure 4
Comparison between original images and images after dimensionality reduction.



(a) Image after dimensionality reduction (b) Original image

4. Conclusions

The proposed DIST-C-SEGAN framework demonstrates an end-to-end 'near-lossless' dimensionality reduction strategy for distributed visual data, integrating convolutional feature extraction, channel attention, graph propagation, and Procrustes alignment. Extensive evaluations on multiple image datasets confirm that the method simultaneously preserves local neighborhood structure (Trustworthiness; 0.05), global manifold continuity (Continuity; 0.08), and nearly all original variance (Explained Variance; 0.08), while enabling high-fidelity reconstruction via a lightweight linear projection. Hyperparameter sensitivity analyses, comparisons with classical linear and non-linear baselines, robustness tests under different random seeds, and downstream clustering studies all verify that each architectural component is indispensable and that the overall approach establishes a new benchmark for distributed, near-lossless image compression and recovery. However, it should be noted that the current framework is mainly designed for visual data. When applied to non-image modalities (such as audio or sensor time series), the extensibility of the framework allows the CNN + SE block to be replaced with modality-specific encoders (such as 1D convolutional neural networks, spectral-based convolutional neural networks, or transformers), while the subsequent attention, graph neural networks, and alignment modules are still generally applicable, thereby preserving high fidelity embeddings.

Future research will extend DIST-C-SEGAN to a true multi-modal distributed setting by replacing the pipeline with modality-specific modules. For text streams, pre-trained Transformer encoders (such as BERT or RoBERTa) or lightweight recurrent blocks (such as BiLSTM, GRU) can replace the CNN backbone, while the SE module recalibrates token-level representations. For time series data, modules like Time Convolutional Networks (TCN) or BRL-STM can capture long-range dependencies before graph propagation. In multi-modal scenarios, cross-attention layers or co-graph embeddings can be introduced to fuse heterogeneous features (e.g., image-text, sensor-speech) into a shared low-dimensional space. Regarding privacy protection, specific implementations will combine multiple techniques. Secure aggregation can be achieved through additive secret sharing, allowing the server to observe only the aggregated vector $u^{(t)}$ without seeing individual $u_i^{(t)}$. Differential privacy can be enforced by adding calibrated Gaussian or Laplace noise to local embeddings before transmission, providing $(\grave{o},\delta)$-DP guarantees. Lightweight homomorphic encryption (e.g., Paillier or CKKS) can further protect sensitive linear operations while maintaining manageable communication overhead. In practice, a hybrid design—using secure aggregation for linear updates and differential privacy for nonlinear layers—will strike a balance between accuracy and privacy budget. Graph construction will also be optimized by implementing learnable sparsification modules (capable of adaptively pruning edges), attention-based topology selection (assigning weights to neighbors), and dynamic neighborhood adjustment driven by task-specific confidence. These strategies reduce redundant communication while maintaining the graph's expressive power. Finally, DIST-C-SEGAN is planned to be deployed on large-scale IoT and edge platforms. Specific experiments will measure bandwidth consumption, inference latency, and downstream performance (e.g., anomaly detection or clustering) under constrained computational and network budgets. These directions will expand the applicability of DIST-C-SEGAN while providing concrete privacy and efficiency guarantees in real multi-modal distributed environments.

Declaration of Conflicting Interests

The author(s) declared no potential conflicts of interest with respect to the research, author-ship, and/or publication of this article.

Data Sharing Agreement

The datasets used and/or analyzed during the current study are available from the corresponding author on reasonable request.

Funding

The author(s) received no financial support for the research, authorship, and/or publication of this article.

Appendix

1 Convergence Guarantees

Throughout this subsection, N denotes the number of participating clients (nodes in the federated setting), n_i the local sample size at client i, $n = \sum_{i=1}^{N} n_i$ the total sample size, d_L the feature dimension of the final embedding, $H_i^{(L)} \in R^{n_i \times d_L}$ the local embedding matrix, $S_i = \frac{1}{n_i}(H_i^{(L)})^{\ddot{\imath}} H_i^{(L)} \in R^{d_L \times d_L}$ the local covariance, and $S = \sum_{i=1}^{N} S_i = \frac{1}{n}\sum_{i=1}^{N}(H_i^{(L)})^{\ddot{\imath}} H_i^{(L)}$ the global covariance. The power-iteration vector is $q^{(t)} \in R^{d_L}$, and the target principal subspace is $U \in R^{d_L \times r}$, consistent with Eqs.,(27)–(29).

Lemma (Algebraic equivalence of the distributed update).

At iteration t, each client computes $u_i^{(t)} = S_i q^{(t)}$ and secure aggregation returns $u^{(t)} = \sum_{i=1}^{N} u_i^{(t)} = S q^{(t)}$, followed by normalization $q^{(t+1)} = u^{(t)} / \| u^{(t)} \|_2$. Hence the distributed procedure is algebraically identical to centralized multiplication by S at every step.

Theorem (Convergence to the dominant eigenvector).

Let $S \in R^{d_L \times d_L}$ be symmetric positive semidefinite with eigenvalues $\lambda_1 > \lambda_2 \geq \cdots \geq \lambda_{d_L} \geq 0$ and corresponding orthonormal eigenvectors $u_1, \ldots, u_{d_L}$. Suppose $q^{(0)}$ has a nonzero component along u_1, i.e., $\alpha_1 := u_1^{\ddot{\imath}} q^{(0)} \neq 0$. Then the distributed power iteration of the lemma satisfies

$$\lim_{t\to\infty} q^{(t)} = \pm u_1, \| q^{(t)} - u_1 \|_2 \leq C(\frac{\lambda_2}{\lambda_1})^t,$$

for some C depending only on the initialization.

Proof.

By spectral decomposition, write $q^{(0)} = \sum_{j=1}^{d_L} \alpha_j u_j$ with $\alpha_1 \neq 0$. Using the lemma, before normalization

$$S^t q^{(0)} = \sum_{j=1}^{d_L} \alpha_j \lambda_j^t u_j.$$

Let $v^{(t)} = S^t q^{(0)}$ and $q^{(t)} = v^{(t)} / \| v^{(t)} \|_2$. Then

$$\frac{\| (I - u_1 u_1^{\ddot{\imath}}) v^{(t)} \|_2}{| u_1^{\ddot{\imath}} v^{(t)} |} = \frac{\sum_{j\geq 2} \alpha_j \lambda_j^t u_{j2}}{|\alpha_1| \lambda_1^t} \leq \frac{\sqrt{\sum_{j\geq 2} \alpha_j^2}}{|\alpha_1|} (\frac{\lambda_2}{\lambda_1})^t.$$

This bounds the tangent (hence sine) of the angle between $q^{(t)}$ and u_1, yielding the stated geometric rate after normalization.

Corollary (Top-r principal subspace via deflation or block power).

Let $U = [u_1, \ldots, u_r] \in R^{d_L \times r}$ be the matrix of leading eigenvectors of S, with eigenvalues $\lambda_1 \geq \cdots \geq \lambda_r > \lambda_{r+1}$. Applying either (i) exact deflation after each eigenvector is obtained, or (ii) block power iteration with $Q^{(t)} \in R^{d_L \times r}$ followed by orthonormalization, yields

$$\sin\Theta\left(\mathrm{span}\left(Q^{(t)}\right), \mathrm{span}(U)\right) \leq C'(\frac{\lambda_{r+1}}{\lambda_r})^t,$$

where Θ denotes the vector of principal angles and C'. depends only on the initialization of $Q^{(0)}$. .

Remark (Compatibility with Sec.2.6).

The iteration used here is exactly the one described in Sec2.6: $u_i^{(t)} = S_i q^{(t)}$, secure aggregation $u^{(t)} = SA\left(\sum_i u_i^{(t)}\right) = S q^{(t)}$, and normalization $q^{(t+1)} = u^{(t)} / \| u^{(t)} \|_2$. Consequently, the distributed protocol inherits the centralized convergence rates in the theorem and corollary above.

Proposition (Optimality of Procrustes alignment, Eqs.,(27)–(29)).

Given $U \in R^{d_L \times r}$ and $H \in R^{B \times d_L}$, the orthogonal Procrustes problem

$$R^* = \arg\min_{R^{\ddot{\imath}} R = I_r} \| HR - U \|_F^2$$

has the unique minimizer $R^* = VW^{\ddot{\imath}}$, where $H^{\ddot{\imath}} U = W\Sigma V^{\ddot{\imath}}$ is the SVD. Hence the aligned embedding $\tilde{H} = HR^*$ used in Eqs., (22)–(23) is globally optimal.

2 Error Bounds

Let the eigen-decomposition of S be $S = \sum_{j=1}^{d_L} \lambda_j u_j u_j^{\ddot{\imath}}$. with $\lambda_1 \geq \cdots \geq \lambda_{d_L} \geq 0$ and $U = [u_1, \ldots, u_r] \in R^{d_L \times r}$.

Let $\hat{U} \in R^{d_L \times r}$ denote the subspace produced by the distributed protocol, and write the principal-angle matrix between span($\hat{U}$) and span(U) as Θ, with Θ, with $\| \sin\Theta \|_2 \leq \varepsilon$.

Proposition (PCA reconstruction risk and excess error).

For any $X \in R^{n \times d_L}$ with centered rows and sample covariance S, the optimal rank-r linear reconstruction (projection onto U) attains

$$\min_{\operatorname{rank}(P)=r,\ P^{\intercal}=P} \| X - XP\|_F^2 = n \sum_{j=r+1}^{d_L} \lambda_j.$$

If $\hat{P} = \hat{U}\hat{U}^{\intercal}$, then the excess reconstruction error ounded by

$$\| X - X\hat{P}\|_F^2 - \| X - XUU^{\intercal}\|_F^2 \le 2n \| S\|_2 \| \sin\Theta\|_F^2$$

Sketch of Proof.

The first identity is classical PCA (Eckart–Young–Msky). For the excess term, use $\| X - X\hat{P}\|_F^2 = \operatorname{trace}\left(X^{\intercal} X\right) - \operatorname{trace}\left(\hat{U}^{\intercal} S\hat{U}\right)$ and likewise for U. Then

$e\left(U^{\intercal} SU\right) - \operatorname{trace}\left(\hat{U}^{\intercal} S\hat{U}\right) \le \| S\|_2 \| U_{\perp}^{\intercal} \hat{U}\|_F^2 = \|\|$, cf. subspace perturbation bounds. Multiply by n to switch from covariance to Frobenius risk.

Corollary (Explained variance lower bound).

Let $\mathrm{EV}\left(\hat{U}\right) := \operatorname{trace}\left(\hat{U}^{\intercal} S\hat{U}\right)$. Then

$$\mathrm{EV}\left(\hat{U}\right) \ge \sum_{j=1}^{r} \lambda_j - \| S\|_2 \| \sin\Theta\|_F^2 \ge \sum_{j=1}^{r} \lambda_j - \lambda_1 \| \sin\Theta\|_F^2.$$

Consequently, the explained-variance ratio satisfies

$$\mathrm{EVR}\left(\hat{U}\right) \ge \frac{\sum_{j=1}^{r} \lambda_j - \lambda_1 \| \sin\Theta\|_F^2}{\sum_{j=1}^{d_L} \lambda_j}.$$

Theorem (Subspace error via Davis–Kahan).

Let $\hat{S}$ be the (possibly perturbed) covariance implicitly used by the distributed power iteration due to finite sampling or stochasticity, and let $E = \hat{S} - S$..

If $\lambda_r > \lambda_{r+l}$, then

$$\|\sin\Theta\|_2 \le \frac{\| E\|_2}{\lambda_r - \lambda_{r+1}}, \|\sin\Theta\|_F \le \frac{\sqrt{r}\| E\|_2}{\lambda_r - \lambda_{r+1}}$$

Remark (From subspace error to reconstruction & variance).

Combining Theorem 1 with Proposition 1 and Corollary 1 yields explicit bounds on excess reconstruction error and explained variance in terms of the spectral gap $\left(\lambda_r - \lambda_{r+1}\right)$ and perturbation $\| E\|_2$.

Distance-distortion bound induced by the training loss.

Recall the distance-preservation term (Equation (30)):

$$L_{\mathrm{dist}} = \frac{1}{B^2} \sum_{i,j} \left(\| z_i - z_j\|_2^2 - \| \tilde{h}_i - \tilde{h}_j\|_2^2 \right),$$

where $\tilde{H} = HR^*$ is the aligned embedding (Eqs.,(28)–(29)).

If $\left|L_{\mathrm{dist}}\right| \le \delta.$, then the average pairwise squared-distance distortion is at most δ. Moreover, by Markov's inequality, for any $\eta \in (0,1)$, at least a $1-\eta$ fraction of pairs satisfy

$$\left| \| z_i - z_j\|_2^2 - \| \tilde{h}_i - \tilde{h}_j\|_2^2 \right| \le \frac{\delta}{\eta}.$$

Since R^* is orthogonal (Proposition1), $\| \tilde{h}_i - \tilde{h}_j\|_2 = \| h_i - h_j\|_2$; hence the distortion is entirely controlled by the learned mapping $X \mapsto H$. and the empirical loss δ.

Putting it together.

Under a nontrivial spectral gap and small perturbation $\| E\|_2$, DIST-C-SEGAN yields

$$\left\| X - X\hat{U}\hat{U}^{\intercal} \right\|_F^2 \le n \sum_{j=r+1}^{d_L} \lambda_j + \frac{2n\lambda_1 r \| E\|_2^2}{\left(\lambda_r - \lambda_{r+1}\right)^2}$$

$$\mathrm{EV}\left(\hat{U}\right) \ge \sum_{j=1}^{r} \lambda_j - \frac{\lambda_1\ r \| E\|_2^2}{\left(\lambda_r - \lambda_{r+1}\right)^2}.$$

The empirical neighborhood-structure distortion is further upper-bounded in average by $\left|L_{\mathrm{dist}}\right|$.

3 Communication Complexity

Let $|\theta|$ denote the number of trainable parameters in the model broadcast/aggregated by FedAvg,

d_L the embedding dimension before alignment, r the aligned embedding dimension,

R the number of power-iteration steps per subspace refresh, and T_U the refresh period in local epochs.

Let R be the number of communication rounds of FedAvg.

a Secure federated PCA (Sec.2.6)

At each power-iteration step, client i uploads $u_i^{(t)} = S_i q^{(t)} \in R^{d_L}$ (uplink d_L scalars),

and receives the aggregated $u^{(t)} \in R^{d_L}$ (downlink d_L scalars).

With R steps per refresh and T/T_U refreshes in total, the overall complexity is

$$\mathrm{Comm}_{\mathrm{PCA}} = 2Nd_L R \cdot \frac{T}{T_U}.$$

b Federated averaging of parameters

Each round aggregates (uplink) $|\theta|$ scalars from each of the N clients and broadcasts (downlink) $|\theta|$ scalars back. Thus

$$\mathrm{Comm}_{\mathrm{FedAvg}} = 2N\left|\theta\right| T.$$

c GNN message passing across clients (Equation (31))

When $N_k(i)$ indexes client-level neighbors, node i sends its current r-dimensional embedding (after

alignment) to each of its k neighbors and receives k embeddings in return per message-passing exchange.

Hence per exchange the cost is 2kr scalars per client, i.e.,

$$\mathrm{Comm}_{\mathrm{GNN/exch}} = 2\mathrm{Nkr} \times \#\mathrm{exchanges}.$$

If message passing is performed once per round, $\mathrm{Comm}_{\mathrm{GNN/exch}} = 2\mathrm{NkrT}$.

d Total cost and comparison to centralized baselines

Summing (A)–(C),

$$\mathrm{Comm}_{\mathrm{total}} = 2\mathrm{N}|\theta|\mathrm{T} + 2\mathrm{Nd}_{\mathrm{L}}\mathrm{R} \cdot \frac{\mathrm{T}}{\mathrm{T}_{\mathrm{U}}} + 2\mathrm{Nkr} \times \#\mathrm{exchanges}$$

For a centralized baseline that uploads all raw features of dimension p for n samples, the one-shot cost is O(np).

In DIST-C-SEGAN, only $r \ll p$ -dimensional embeddings are exchanged across clients for topology-aware aggregation, yielding the reduction

$$\mathrm{Comm}_{\mathrm{ours(embedding\ exchange)}} = \mathrm{O}(\mathrm{nr}) = \frac{\mathrm{r}}{\mathrm{p}}\mathrm{O}(\mathrm{np}) = \frac{\mathrm{r}}{\mathrm{p}}\mathrm{Comm}_{\mathrm{raw}},$$

consistent with Equation (38). The additional costs $\mathrm{Comm}_{\mathrm{PCA}}$ and $\mathrm{Comm}_{\mathrm{FedAvg}}$ scale with d_L, $|\theta|$, and T but remain independent of n; with $r \ll p$ and moderate d_L, they are negligible compared to centralized raw-data transfer.

Remark (Parameter choices for tight budgets).

Choosing a larger refresh period T_U and a small number of power steps R (leveraging the spectral gap) reduces $\mathrm{Comm}_{\mathrm{PCA}}$. Using sparse client graphs (small k) and aligned embeddings (dimension r) keeps $\mathrm{Comm}_{\mathrm{GNN/exch}}$ linear in r and k, preserving the $\frac{\mathrm{r}}{\mathrm{p}}$ gain.

References

1. Ali, M. F., Khan, R. Z. Distributed Computing: An Overview. International Journal of Advanced Networking and Applications, 2015, 7(1), 2630.
2. Cao, X., Başar, T., Diggavi, S., Eldar, Y. C., Letaief, K. B., Poor, H. V., Zhang, J. Communication-Efficient Distributed Learning: An Overview. IEEE Journal on Selected Areas in Communications, 2023, 41(4), 851-873. https://doi.org/10.1109/JSAC.2023.3242710
3. Chen, C., Xu, W., Zhu, L. Distributed Estimation in Heterogeneous Reduced Rank Regression: With Application to Order Determination in Sufficient Dimension Reduction. Journal of Multivariate Analysis, 2022, 190, 104991. https://doi.org/10.1016/j.jmva.2022.104991
4. Dhinakaran, D., Srinivasan, L., Raja, S. E., Valarmathi, K., Nayagam, M. G. Synergistic Feature Selection and Distributed Classification Framework for High-Dimensional Medical Data Analysis. MethodsX, 2025, 14, 103219. https://doi.org/10.1016/j.mex.2025.103219
5. Duan, S., Wang, D., Ren, J., Lyu, F., Zhang, Y., Wu, H. Distributed Artificial Intelligence Empowered by End-Edge-Cloud Computing: A Survey. IEEE Communications Surveys & Tutorials, 2022, 25(1), 591-624. https://doi.org/10.1109/COMST.2022.3218527
6. Fang, J., Li, H. Optimal/Near-Optimal Dimensionality Reduction for Distributed Estimation in Homogeneous and Certain Inhomogeneous Scenarios. IEEE Transactions on Signal Processing, 2010, 58(8), 4339-4353. https://doi.org/10.1109/TSP.2010.2048213
7. Feldman, D., Schmidt, M., Sohler, C. Turning Big Data Into Tiny Data: Constant-Size Coresets for K-Means, PCA, and Projective Clustering. SIAM Journal on Computing, 2020, 49(3), 601-657. https://doi.org/10.1137/18M1209854
8. Gahar, R. M., Arfaoui, O., Hidri, M. S., Hadj-Alouane, N. B. A Distributed Approach for High-Dimensionality Heterogeneous Data Reduction. IEEE Access, 2019, 7, 151006-151022. https://doi.org/10.1109/ACCESS.2019.2945889
9. Grammenos, A., Mendoza-Smith, R., Crowcroft, J., Mascolo, C. Federated Principal Component Analysis. Advances in Neural Information Processing Systems, 2020, 33, 6453-6464.
10. Schizas, I. D., Aduroja, A. A Distributed Framework for Dimensionality Reduction and Denoising. IEEE Transactions on Signal Processing, 2015, 63(23), 6379-6394. https://doi.org/10.1109/TSP.2015.2465300
11. Kannan, R., Vempala, S. S., Woodruff, D. P. Principal Component Analysis and Higher Correlations for Distributed Data. Conference on Learning Theory, 2014, 1040-1057.
12. Kshemkalyani, A. D., Singhal, M. Distributed Computing: Principles, Algorithms, and Systems. Cambridge University Press, 2011.
13. Le Borgne, Y.-A., Raybaud, S., Bontempi, G. Distributed Principal Component Analysis for Wireless Sensor Networks. Sensors, 2008, 8(8), 4821-4850. https://doi.org/10.3390/s8084821

14. Li, D., Li, J., You, M., Tang, W., Zhang, X., Qin, W. Direct Localization of Wideband Sources Using Distributed Arrays: A Subspace Focusing and Dimension Reduction Approach. Signal Processing, 2025, 232, 109902. https://doi.org/10.1016/j.sigpro.2025.109902

15. Liang, J., Yu, G., Chen, B., Zhao, M. Decentralized Dimensionality Reduction for Distributed Tensor Data Across Sensor Networks. IEEE Transactions on Neural Networks and Learning Systems, 2015, 27(11), 2174-2186. https://doi.org/10.1109/TNNLS.2015.2469100

16. Piroozmand, P., Brenner, O., Jenny, P. Dimensionality Reduction for Regularization of Sparse Data-Driven RANS Simulations. Journal of Computational Physics, 2023, 492, 112404. https://doi.org/10.1016/j.jcp.2023.112404

17. Qi, H., Wang, T.-W., Birdwell, J. D. Global Principal Component Analysis for Dimensionality Reduction in Distributed Data Mining. Statistical Data Mining and Knowledge Discovery, 2004, 327-342. https://doi.org/10.1201/9780203497159.ch19

18. Qu, Y., Ostrouchov, G., Samatova, N., Geist, A. Principal Component Analysis for Dimension Reduction in Massive Distributed Data Sets. Proceedings of the IEEE International Conference on Data Mining, 2002, 1784-1788. https://doi.org/10.1109/ICDM.2002.1184046

19. Roy, O., Vetterli, M. Dimensionality Reduction for Distributed Estimation in the Infinite Dimensional Regime. IEEE Transactions on Information Theory, 2008, 54(4), 1655-1669. https://doi.org/10.1109/TIT.2008.917635

20. Tutsoy, O., Sumbul, H. E. A Novel Deep Machine Learning Algorithm with Dimensionality and Size Reduction Approaches for Feature Elimination: Thyroid Cancer Diagnoses with Randomly Missing Data. Briefings in Bioinformatics, 2024, 25(4). https://doi.org/10.1093/bib/bbae344

21. Valcarcel Macua, S., Belanovic, P., Zazo, S. Consensus-Based Distributed Principal Component Analysis in Wireless Sensor Networks. IEEE SPAWC, 2010, 1-5. https://doi.org/10.1109/SPAWC.2010.5671089

22. Vrlić, M., Pernsteiner, D., Schirrer, A., Hametner, C., Jakubek, S. Reduced-Dimensionality Nonlinear Distributed-Parameter Observer for Fuel Cell Systems. Energy Reports, 2023, 10, 1-14. https://doi.org/10.1016/j.egyr.2023.06.006

23. Wagner, A., Solomon, E., Bendich, P. Improving Metric Dimensionality Reduction with Distributed Topology. arXiv, 2021, arXiv:2106.07613. https://doi.org/10.48550/arXiv.2106.07613

24. Wu, Z., Lü, Y., Plaza, A., Li, J., Xiao, F., Wei, Z. Parallel and Distributed Dimensionality Reduction of Hyperspectral Data on Cloud Computing Architectures. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 2016, 9(6), 2270-2278. https://doi.org/10.1109/JSTARS.2016.2542193

25. Xie, W., Bonis, I., Theodoropoulos, C. Data-Driven Model Reduction-Based Nonlinear MPC for Large-Scale Distributed Parameter Systems. Journal of Process Control, 2015, 35, 50-58. https://doi.org/10.1016/j.jprocont.2015.07.009

26. Zhang, X.-L. Nonlinear Dimensionality Reduction of Data by Deep Distributed Random Samplings. Asian Conference on Machine Learning, 2014, 221-233.

27. Zhang, Y., Chen, X., Feng, M., Liu, J. Distributed Sparse Manifold-Constrained Optimization Algorithm in Linear Discriminant Analysis. Journal of Imaging, 2025, 11(3), 81. https://doi.org/10.3390/jimaging11030081