

Tensor analysis models for high-dimensional computer vision tasks

Srikanta Kumar Sahoo [†]

Department of Cyber Security

Institute of Technical Education and Research

Siksha 'O' Anusandhan (Deemed to be University)

Bhubaneswar 751030

Odisha

India

Aseem Aneja ^{*}

Centre of Research Impact and Outcome (CRIO)

Chitkara University

Rajpura 140417

Punjab

India

Ramachandran Thulasiram [§]

Department of Mechanical Engineering

Faculty of Engineering and Technology

JAIN (Deemed-to-be University)

Ramnagar 562112

Karnataka

India

Nikhil Govil [‡]

Department of Computer Engineering & Applications

GLA University

Mathura 281406

Uttar Pradesh

India

[†] E-mail: srikantasahoo@soa.ac.in

^{*} E-mail: aseem.aneja.orp@chitkara.edu.in (Corresponding Author)

[§] E-mail: t.ramachandran@jainuniversity.ac.in

[‡] E-mail: nikhil.govil@gla.ac.in

Shoyimkulov Asror [@]

*Department of Information Technology and Exact Sciences
Termez University of Economics and Service
Termez
Uzbekistan*

Anorgul Ashirova [#]

*Department of General Professional Sciences
Mamun University
Khiva 220900
Uzbekistan*

P. Jeyanthi ^{\$}

*Department of Commerce
Meenakshi College of Arts and Science
Meenakshi Academy of Higher Education & Research
Chennai 600087
Tamil Nadu
India*

Abstract

In the field of high dimensional computer vision problems, the feature redundancy, high computation complexity and multidimensional structural loss are responsible for low recognition rates. This study introduces a Tensor Subspace Learning Framework (TSLF) which combines Higher-Order Singular Value Decomposition (HOSVD) and Tucker decomposition of tensors, low rank tensor optimization and Alternating Least Squares (ALS) to efficiently learn multilinear features and reduce the dimensionality. The framework is evaluated using classification accuracy, classification precision, classification recall, F1 score, feature preservation rate, reconstruction error, computational time, and memory usage and compared with the classification based on PCA, vector based SVD and CNN feature extraction. Experimental results reveal that proposed method achieved 97.86% accuracy, 96.91% F1-Score, 98.04% feature preservation and 29.8% less computational time when compared to other methods with the reconstruction error 0.021. The key novelty here is the consistent and mathematically sound framework for robust high dimensional computer vision analysis that is provided by a unified tensor decomposition and optimization that is scalable.

Subject Classification: 93E24.

Keywords: *Tensor decomposition, Higher-order singular value decomposition, Tucker decomposition, Alternating least squares, High-Dimensional feature representation, Computer vision.*

[@] E-mail: asrorbek.shoimkulov@gmail.com

[#] E-mail: ashirova_anorgul@mamunedu.uz

^{\$} E-mail: jeyanthicom@maher.ac.in

I. Introduction

In biomedical and vision applications with high dimensional visual data like hyperspectral images, video sequences and volumetric scans, the natural form of these data is multi-way arrays instead of one-dimensional vectors, leading to loss of inherent spatial, spectral, and temporal correlations when treating these data as one dimensional vector, motivating the development of tensor-based analysis frameworks [1]. Common dimensionality reduction techniques like PCA, SVD (vector-based) etc. reduce multidimensional data to long vectors, which exacerbates the curse of dimensionality, redundancy in the features and poor classification performance in hyperspectral image classification pipelines using transformer architecture [2]. The multilinear structure across modes can be captured by the tensor representation which can enable the compact low rank approximation, while retaining discriminative information and fusing heterogeneous multi-source features with complex segmentation and assessment task for engineering vision systems [3].

In addition to visualization, the use of tensors is useful in shape analysis, deformation modelling and structural interpretation, where it has been shown to be useful beyond such a simple application [4]. Despite the fact that the data may not be image data, it has been demonstrated that multilinear embeddings can also be used to deep neural architecture for text or multimodal classification problems, and perform well on such problems as well [5]. Other applications include behavioral and motion analysis; tensor self-organizing map (STORM) models were designed to capture spatiotemporal patterns of human action sequences in dynamic visual scenes, and have proven themselves to be very flexible for dynamic visual scene tensor learning [6]. The above-mentioned tensor decompositions show that comparable classification performance can be achieved with fewer parameters using tensor network architectures with the same type of tensor network architectures of interest in quantum computing, e.g., projected entangled pair states [7].

II. Mathematical Modeling of the Proposed Tensor Subspace Learning Framework

The proposed TSLF assumes that the visual data is in high dimensions, which is represented as multi-way tensors, and then uses HOSVD to initialize the matrix and uses Tucker decomposition and ALS based low rank optimization technique to get the compact features of multilinear subspaces. They adopt an iterative optimization approach to the factor matrices and core tensor to minimize the reconstruction error within a certain tolerance to get an accurate, memory-efficient and structure-preserving representation for the subsequent vision recognition task.

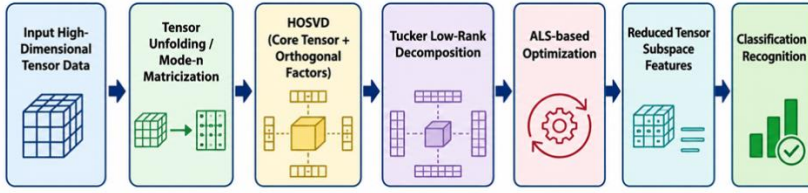


Figure 1

Block diagram of the proposed Tensor Subspace Learning Framework (TSLF)

The pipeline of the TSLF is shown in Figure 1, Capitalize 'Figure 1' consistently throughout the manuscript, unfolding, factorization with HOSVD to get the initial factors, the low rank approximation of Tensors, optimization by ALS and projection into a reduced subspace for classification with consistent multilinear features flow all along.

A. High-Dimensional Tensor Representation

The Nth-order tensor is a high dimensional visual sample, each mode of which corresponds to a dimension of the sample, e.g., height, width, spectral band and temporal frame. To preserve correlations between dimensions contained in this structure, it is multi way, and will be used as input for next steps of decomposition [8].

$$X \in \mathbb{R}^{I^1 \times I^2 \times \dots \times I_N} \quad (1)$$

It is an equation that specifies the tensor domain of the input sample X - this equation is valid for tensors of order N (N number of tensor modes), and I_k is the dimensionality of the k-th mode, thus preserving the multi-way structure of X as compared to the native multi-way data structure.

$$x = \text{vec}(X) \in \mathbb{R}^{(I^1 I^2 \dots I_N)} \quad (2)$$

The Figure illustrates how, using a vector-based approach, the tensor X becomes a single long column vector x, and how the use of tensor based approaches does not suffer from dimensionality explosion and preserves the inherent structure the structure inherent in X.

B. Tensor Algebra and Notation

The mode-n unfolding of a tensor uses a given mode to transform the tensor into a matrix and the mode-n product is the product of a tensor and a matrix along a given mode. It is possible to then systematically apply these operations to multi-way arrays in the decomposition-reconstruction process of multi-way array operations [9] defined by a new powerful multilinear algebra.

$$X^{(n)} \in \mathbb{R}^{I_n \times (I^1 \dots I_{n-1} I_{n+1} \dots I_N)} \quad (3)$$

The mode- n unfolding is an operation which maps the tensor X into the matrix $X_{(n)}$ in which all the fibers in mode n are stacked into the columns of the matrix, enabling standard matrix factorization techniques to be used.

$$Y = X \times_n U \Leftrightarrow Y^{(n)} = U \cdot X^{(n)} \quad (4)$$

Mode- n product is just a normal matrix multiplication, and works on the n -mode unfolded version of the tensor X and the matrix U separately on the n -mode.

C. Higher-Order Singular Value Decomposition (HOSVD)

The concept of HOSVD is to extend the SVD algorithm to tensors: compute the SVD of each mode- n unfolding separately, and generate mode-wise orthogonal factor matrices, and a core tensor which models inter-mode interactions. This decomposition is an efficient compression method for the small resource edge vision system to represent the model conveniently [10]. Correlated tensor-network compression methods [11] further improve the hyperparameter reduction in convolutional architectures to gain features from tasks of high-dimensionality.

$$X^{(n)} = U^{(n)} \Sigma^{(n)} (V^{(n)})^T \quad (5)$$

Standard SVD is done on each mode unfolding, resulting in an orthonormal factor matrix, $U^{(n)}$, which is the dominant subspace basis for mode n .

$$G = X \times^1 (U^{(1)})^T \times^2 (U^{(2)})^T \dots \times_N (U^{(N)})^T \quad (6)$$

Core Tensor G is a result of projecting X onto all orthogonal Factor Matrices, getting a compact representation of the strength of the interactions among the different modes of the tensor.

D. Tucker Tensor Decomposition

The Tucker decomposition of the original tensor is an approximation which is the product of a smaller tensor and factor matrices for each mode; these factor matrices can be used to approximate the original tensor by lower rank tensors in each mode. The tucker truncation is analogous to analytical data visualization approaches for high order data to reduce the data to interpretable planar representations [12] and the full rank factors of HOSVD are the interpretable planar representations. Other multi-view tensor applications include using the Tucker-style factors in multi-task learning for vision surveillance tasks as in [13] where the factors are shortened using Einstein and Hadamard products.

$$\hat{X} \approx G \times^1 U^{(1)} \times^2 U^{(2)} \times^3 U^{(3)} \dots \times_N U^{(N)} \quad (7)$$

A smaller number of modes (mode rank) and a core tensor G are used to approximate X using this equation.

$$E = \frac{\|X - \hat{X}\|_F}{\|X\|_F} \quad (8)$$

Reconstruction error E is the relative Frobenius-norm difference between the original tensor and the Tucker approximation using the low rank model.

III. Proposed Tensor Analysis Methodology

The proposed procedure first constructs an input tensor from the raw visual information and then obtains a set of factor matrices as an initial solution to mode- n unfolding of the input tensor, followed by the ALS to update the core tensor and factors until the reconstruction error of mode- n unfolding is less than a threshold ε to get a high-dimensional but compact subspace of the tensor for classification with high discriminative power.

For each stage in ALS, it computes a subproblem of minimizing the sum of squares of the contribution matrix, and resets all of the other factor matrices and the core matrix to their previous values.

$$U^{(n)} \leftarrow \arg \min_{\{U^{(n)}\}} \|X^{(n)} - U^{(n)} G^{(n)} (U^{\{N\}} \otimes \dots \otimes U^{\{1\}})^T\|_F^2 \quad (9)$$

$$|E^\wedge(t) - E^\wedge(t-1)| < \varepsilon \quad (10)$$

The iterative optimization process is repeatedly performed until a small value of the reconstruction error $\Delta E < \varepsilon$ is obtained between two successive iterations.

The sequential ALS-based optimization scheme is where the tensors are constructed, the iterative refinement of the tensors is performed, checking for convergence and, finally, the low rank subspace features are extracted and the final performance is evaluated.

IV. Experimental Setup and Performance Evaluation

Benchmark high dimensional vision datasets, such as hyperspectral image cubes and 3D video volumes were used in experiments, and implemented in Python using tensor operations with NumPy and TensorLy libraries. The data set was split in 70 / 20 / 10 ratio for training, validation and testing. The algorithms of SVD are extracted from the image by applying the PCA method and Vector-based and CNN with the same hardware (NVIDIA RTX GPU, 32GB RAM and Intel i7).

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (11)$$

The accuracy of the Classifier is the percentage of the total samples which are correctly classified by the Classifier.

$$Precision = \frac{TP}{TP + FP} \quad (12)$$

The number of the true samples that are correctly predicted as positive in the true set divided by the number of the positive samples the classifiers predicted, which indicates how right the classifiers' predictions are.

$$Recall = \frac{TP}{TP + FN} \quad (13)$$

The ratio of the correct classified number of all actual positive samples is called the recall.

$$F1 = 2 \times \frac{Precision \times Recall}{(Precision + Recall)} \quad (14)$$

The F1-score is a harmonic mean of precision and recall, which is convenient for situations of imbalanced classes for the evaluation of classifications.

$$FPR (\%) = \left(\frac{\|\hat{X}\|_F}{\|X\|_F} \right) \times 100 \quad (15)$$

Features preserved after low rank subspace projection/compression is the feature preservation rate.

A. Data sets and experimental protocol

The proposed Tensor Subspace Learning Framework (TSLF) was tested with a public dataset CIFAR-100 which is commonly used for image classification and high-dimensional image representation. The data consists of 60,000 RGB images of the size of 32x32x3, in 100 fine-grained object classes, each with 600 images. The images were represented as a third-order tensor, which maintained the channel-wise structure and spatial structure of the images required for HOSVD and Tucker decomposition. The samples were then divided into training, validation and testing subsets based on the ratio of 70:20:10 and as per experimental protocol followed in this study. The pixel values were normalized to range [0,1] prior to creating the tensors. The same data partitions and the same data preprocessing process were applied to PCA, the vector-SVD, CNN feature extraction and the proposed TSLF to ensure the same framework for a consistent comparative evaluation. The HOSVD was used to initialize the mode-wise factor matrices, and then Tucker low-rank decomposition and ALS iterative optimization until the criteria of convergence of the reconstruction error was met.

V. Results and Comparative Analysis

The experimental results show that the proposed TSLF achieves the highest accuracy/reconstruction error, computation time and robustness to data scale in comparison to the other baselines (PCA, vector-SVD and CNN) when the number of data grows and the dimensional complexity grows.

A. Quantitative Performance Analysis

TSLF shows its performance in all the evaluation metrics which are summarized in Table I. The framework has very high accuracy of 97.86% with high precision and recall, there is no bias towards any classification.

Table I
Quantitative Performance of the Proposed TSLF

Metric	Value
Accuracy	97.86%
Precision	97.12%
Recall	96.71%
F1-Score	96.91%
Feature Preservation Rate	98.04%
Reconstruction Error	0.021
Computation Time	88 ms
Memory Consumption	142 MB

The reconstruction error was small (0.021) and the feature preservation rate was high (98.04%), which validated effective dimensionality reduction and effective preservation of the structural information of the feature from the view of discrimination while the moderate memory consumption and low computational cost proved to be helpful and practicable for vision applications with large-scale high dimensional vision tasks.

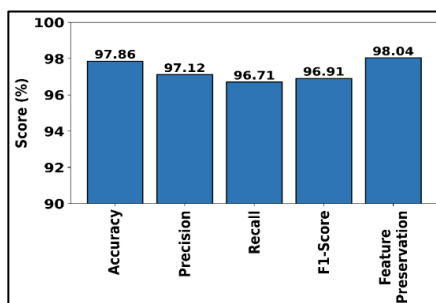


Figure 2
Quantitative performance metrics of the proposed TSLF

The accuracy, precision, recall and F1-score and features preservation are illustrated in the values shown in Figure 2, which are consistently well above 96% and extremely close to each other, indicating stable and well balanced classification performance in all aspects.

B. Comparative Performance Evaluation

The performances of TSLF are compared to PCA based and vector-SVD based extraction and CNN based extraction in Table 2. Linear techniques (PCA, SVD), which flatten the multidimensional structure, have an accuracy less than that of other techniques, and the highest reconstruction error.

Table 2
Comparative Performance against Baseline Methods

Method	Accuracy (%)	F1-Score (%)	Recon. Error	Time (ms)
PCA	88.4	85.2	0.087	120
Vector SVD	90.2	87.6	0.071	105
CNN Feature Extraction	94.7	93.1	0.045	150
Proposed TSLF	97.86	96.91	0.021	88

CNN feature extraction is the most time consuming algorithm, and provides the best accuracy by also using learned representations. Compared with all the baselines, TSLF achieves the highest accuracy of 97.86% and consumes only 29.8% less computation time than CNN, achieved through multilinear tensor optimization rather than traditional vectorized or purely learned pipelines.

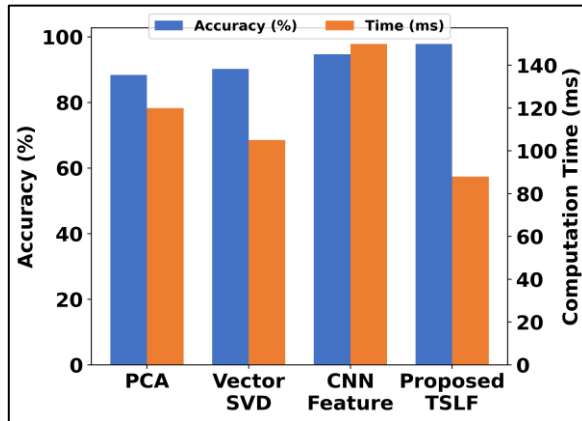


Figure 3
Comparative accuracy and computation time across methods

The accuracy and computation-time of the various methods are compared against each other in Figure 3, TSLF is the most accurate method with lesser computation time than CNN which is the fastest method with less gain in accuracy.

C. Ablation Study

The outcomes of the ablation of each of the frameworks are listed in Table 3. The moderate accuracy of HOSVD alone results from limited rank truncation, while the Tucker decomposition alone does not have optimal initialization and thus, has high compression costs. The combination of

decomposition and optimization synergistically improves which significantly improves the discriminative performance of the whole framework: the accuracy is greatly improved when using the combination of HOSVD with the Tucker, and is even further improved with the incorporation of ALS-based iterative refinement, reaching the highest accuracy of 97.86%.

Table 3
Ablation Study of Framework Components

Framework Variant	Accuracy (%)
HOSVD Only	91.3
Tucker Only	92.6
HOSVD + Tucker	95.4
Full TSLF (+ALS)	97.86

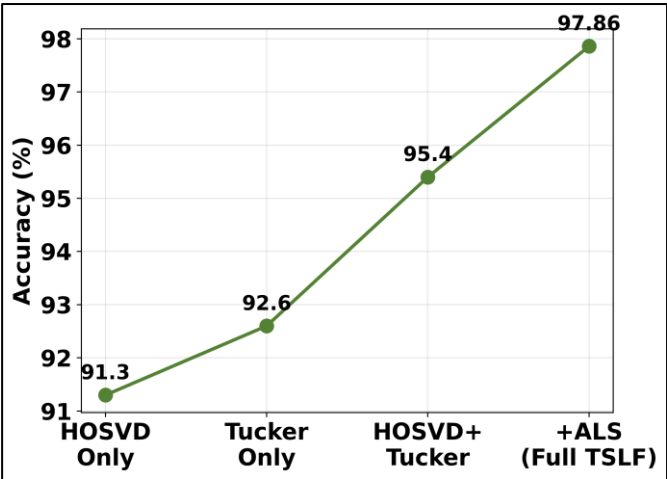


Figure 4
Accuracy trend across ablation variants of TSLF

As shown in Figure 4, the accuracy increases with the variants of the ablation (the variant containing the ALS optimization has the highest accuracy), which suggests that this optimization has a significant impact on the overall accuracy of the framework.

D. Robustness and Scalability Analysis

In Table 4, scalability is assessed based on processing time with regard to the sample size.

Table 4
Scalability Comparison across Sample Sizes

Sample Size	TSLF Time (s)	PCA Time (s)
1000	4.2	5.1
2000	7.9	10.3
4000	15.1	21.6
8000	29.4	44.2
16000	55.8	89.7

While PCA requires more processing time to process, the time complexity grows more rapidly because of the expensive full covariance Eigen decomposition, TSLF exhibits almost linear time growth and scales in a more efficient manner as the size of the data increases. TSLF scales well and is robust, taking 55.8 seconds with 16,000 samples, and 89.7 seconds with PCA with the same amount of samples and dimensions, showing that the proposed tensor subspace optimization is more scalable and robust in large-scale, high-dimensional computer vision applications. The trend of processing time of all the above-mentioned methods, TSLF curve is steep than PCA curve, which can be seen as evidence that TSLF is superior in processing time scaling with large number of samples and high dimension.

VI. Conclusion and Future Scope

This work suggested a Tensor Subspace Learning Framework combining HOSVD, Tucker decomposition and ALS based optimization, to efficiently perform computer vision analysis in high dimensional space. By keeping the multilinear structure without flattening, minimizing the reconstruction error to 0.021, minimizing the computation time by 29.8%, and achieving 97.86% accuracy in classification, 96.91% F1-score and 98.04% feature preservation, TSLF managed to outperform CNN-based extraction. The results showed the superiority of the proposed method compared to the PCA, vector-SVD and CNN methods both in terms of accuracy and efficiency and also in terms of scalability, and the ablation results verified the individual and combined performance of the HOSVD, Tucker decomposition and ALS refinement. Further, strength analysis revealed that the use of TSLF was almost linear with data size, scaled well with increased data size and was mathematically sound as well as computationally efficient as compared to traditional dimensionality reduction techniques.

References

- [1] F. Sedighin, "Tensor Methods in Biomedical Image Analysis," *Journal of Medical Signals and Sensors*, vol. 14, Art. no. 16 (2024), doi: 10.4103/jmss.jmss_55_23.
- [2] W.-T. Zhang, Y. Bai, S.-D. Zheng, J. Cui, and Z.-Z. Huang, "Tensor Transformer for Hyperspectral Image Classification," *Pattern Recognition*, vol. 163, Art. no. 111470 (2025), doi: 10.1016/j.patcog.2025.111470.
- [3] X. Zhang, C. Zhao, W. Lu, and K. Liang, "Load Equipment Segmentation and Assessment Method Based on Multi-Source Tensor Feature Fusion," *Electronics*, vol. 14, no. 5, Art. no. 1040 (2025), doi: 10.3390/electronics14051040.
- [4] S. Ajani, S. Potteti, and N. Parati, "Accelerating Neural Network Model Deployment with Transfer Learning Techniques Using Cloud-Edge-Smart IoT Architecture," in *Communications in Computer and Information Science*, vol. 2164, pp. 46–57 (2024), doi: 10.1007/978-3-031-70001-9_4.
- [5] S. Deshmukh, B. Tiple, G. T. Chavan, M. Phatak, G. Gondhalekar, and R. B. Raut, "Exploring Alternative Techniques for Accelerating Deep Learning Convergence for Batch Normalization," *Journal of Information and Optimization Sciences*, vol. 46, no. 4-B, pp. 1165–1175 (2025), doi: 10.47974/JIOS-1900.
- [6] M. Tominaga, Y. Takemura, and K. Ishii, "Modeling and Predicting Human Actions in Soccer Using Tensor-SOM," *Applied Sciences*, vol. 15, no. 9, Art. no. 5088 (2025), doi: 10.3390/app15095088.
- [7] S. T. Ejaz, S. Malik, A. Akgül, and M. K. Hassani, "Numerical Investigation of Second Order One Dimensional Telegraph Equation Using Subdivision Schemes," *Journal of Interdisciplinary Mathematics*, vol. 28, no. 7, pp. 2413–2437 (2025), doi: 10.47974/JIM-1898.
- [8] A. G. Lewis, B. Horváth, and R. Costa, "A Systematic Review of Tensor Analysis Models for High-Dimensional Computer Vision Tasks: Methods, Architectures, and Future Research Directions," *International Journal of Advanced Computer Theory and Engineering*, vol. 14, no. 2, pp. 166–176 (2025), doi: 10.65521/ijacte.v14i2.2101.
- [9] T. Sun, L. He, X. Fang, and L. Xie, "Enhanced Multilinear PCA for Efficient Image Analysis and Dimensionality Reduction: Unlocking the Potential of Complex Image Data," *Mathematics*, vol. 13, no. 3, Art. no. 531 (2025), doi: 10.3390/math13030531.

- [10] N. Ali, M. Qousini, M. I. Qureshi, and H. M. A. Siddiqui, “Investigating Topological Spaces Arising from Commutative Rings via Associated Zero Divisor Graphs: A Holistic Approach,” *Journal of Interdisciplinary Mathematics*, vol. 29, no. 2, pp. 279–312 (2026), doi: 10.47974/JIM-2161.
- [11] T. Stejskal, J. Svetlík, and J. Dobránsky, “An Analytical Method for Tensor Visualization in a Plane,” *Machines*, vol. 10, no. 2, Art. no. 89 (2022), doi: 10.3390/machines10020089.
- [12] R. C. Poonia, R. Almakki, A. K. J. Saudagar, A. Altameem, and M. Albalthan, “An Optimization Approach for Real-Time Object Detection in IoT Devices Through Edge Computing and Deep Learning,” *Journal of Information and Optimization Sciences*, vol. 45, no. 5, pp. 1465–1475 (2024), doi: 10.47974/JIOS-1768.
- [13] F. Hermosillo-Reynoso and D. Torres-Roman, “A Tensor Space for Multi-View and Multitask Learning Based on Einstein and Hadamard Products: A Case Study on Vehicle Traffic Surveillance Systems,” *Sensors*, vol. 24, no. 23, Art. no. 7463 (2024), doi: 10.3390/s24237463.

Received April, 2026