PCA & Eigenface: Dimensionality Reduction and Face Recognition

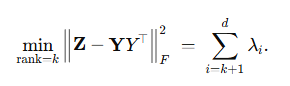
# 1. PCA

## 1.1 PCA Theory

Principal Component Analysis (PCA) is a linear dimensionality reduction technique that transforms a dataset into a new coordinate system such that the greatest variances lie on the first coordinates (principal components). The steps include:

|  |  |  |
| --- | --- | --- |
| Step | Mathematics | Purpose / Insight |
| 1. Mean-centre the data |  | Shifts cloud to the origin so variance equals covariance about 0 |
| 2. Covariance matrix |  | Symmetric, positive-semidefinite; its eigen-structure encodes axis-aligned variances |
| 3. Eigen-decomposition |  | Columns vk (eigenvectors) form an **orthonormal basis**; eigenvalues λk  are variances along those axes |
| 4. Projection (dimensionality reduction) | for any k<d |  |

* **Optimality**  
  The choice of the first k eigenvectors minimises the mean-squared reconstruction error



* **Reconstruction**  
  An approximate recovery in the original space is



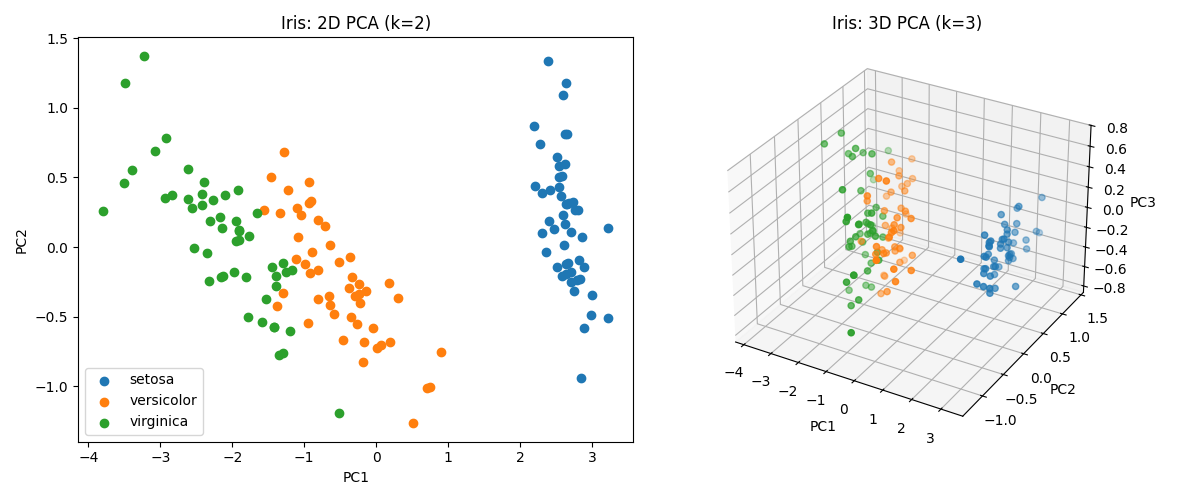
These steps yield a lower-dimensional representation while retaining most of the data's variance.

## 1.2 PCA Implementation & Results

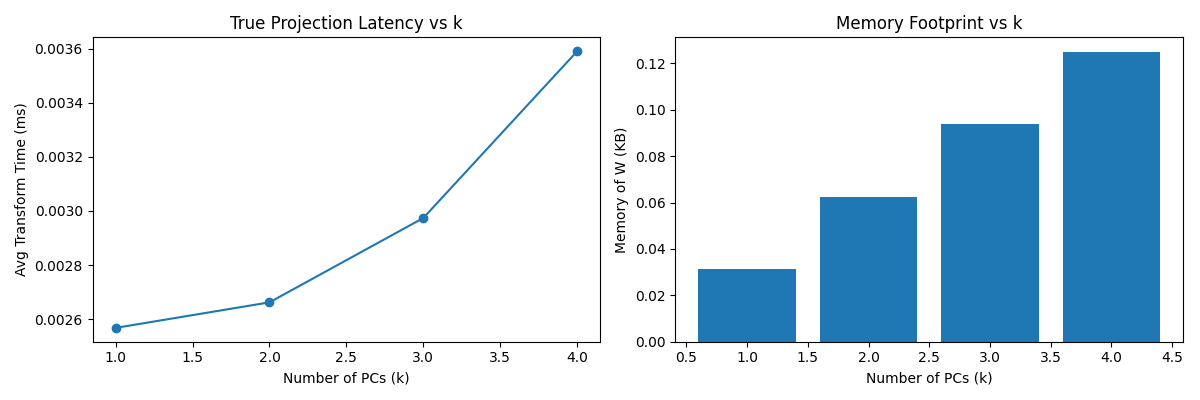
* Implementation choices

| **Aspect** | **Decision** | **Rationale** |
| --- | --- | --- |
| **Language / libs** | Python 3.10 + NumPy only | To demonstrate full control of each linear-algebra step |
| **Eigen-solver** | np.linalg.eigh( C ) | Exploits symmetry of the covariance matrix; returns ordered eigen-pairs in O(d³) |
| **Rank-k projector** |  | Columns already orthonormal ⇒ projection is a single matrix multiply |
| **Variance report** |  | Displayed as table + cumulative curve |
| **Timing / memory** | **time.perf\_counter()** around **Z @ W\_k**; **W\_k.nbytes** | Exposes real projection cost vs k |

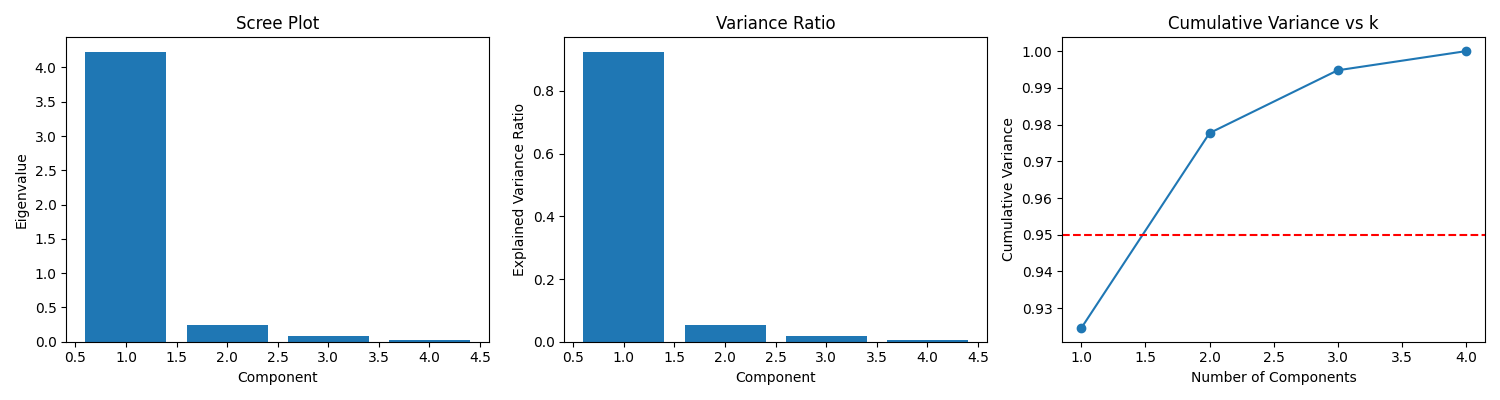
* **Dataset**
  + **Iris** (150 × 4) – sepal length, sepal width, petal length, petal width.
  + Standard practice: z-score normalization is *not* required because PCA’s first step is mean-centering; scale equality is already acceptable (all four are centimetre measurements).
* **Result & Insight** 
  + 2-D and 3-D scatter (colour = species): *Even 2D PCA clearly separates Setosa; 3D PCA strengthens separation for Versicolor and Virginica.*



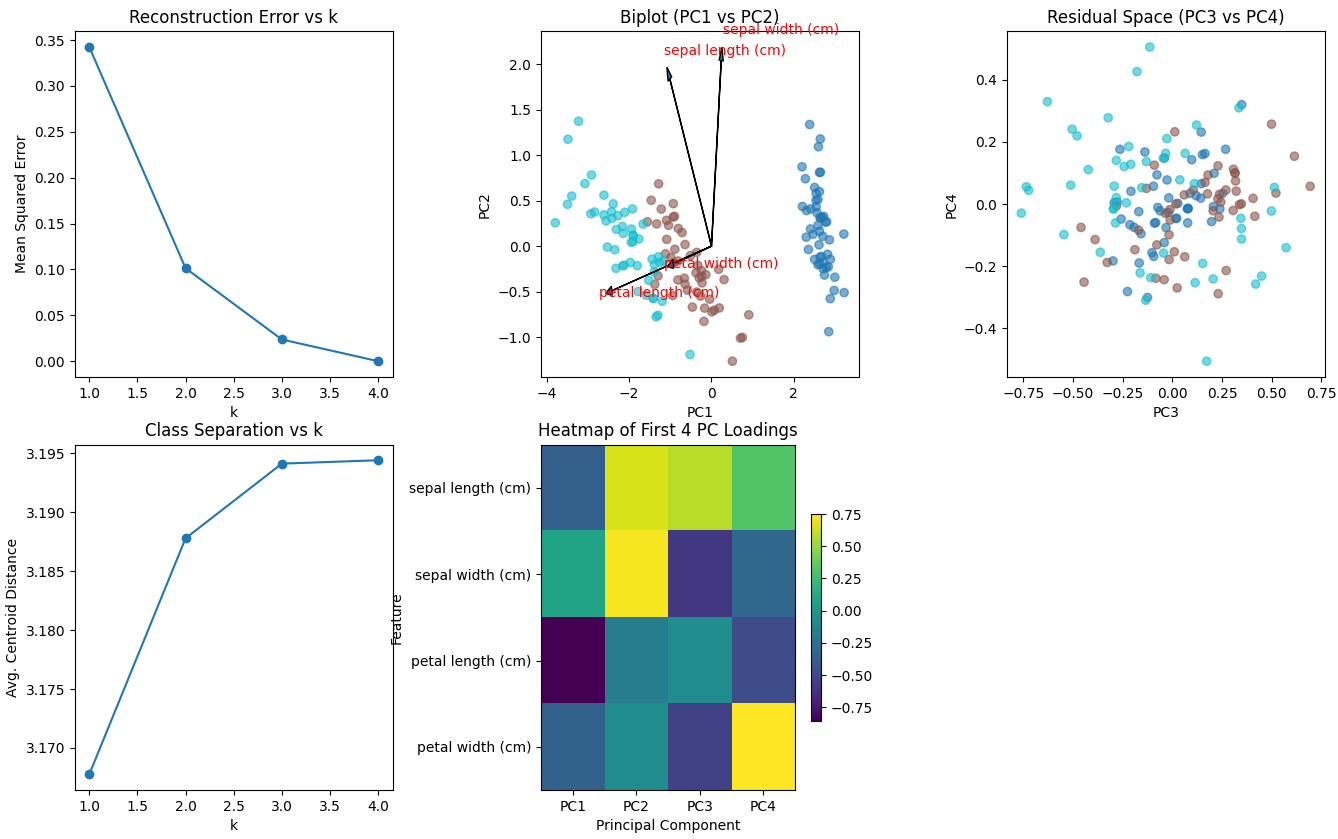
* + Scree + cumulative variance plot: *Most variance is captured by PC1–PC2. Scree plot reveals redundancy in PC3–4. PC1 correlates strongly with petal features.*



* + Latency/Memory trade-off plot: *Latency rises with (k) due to matrix size; memory grows linearly. There's diminishing return beyond (k=2)*



* + Biplot & residual-space scatter: *Reconstruction error drops fast with (k), centroid distance saturates, biplot reveals direction of feature influence.*



* **Interpretation**
  + Compression sweet-spot: **k=2** captures **> 97%** variance while cutting feature count by **50%**.
  + Speed vs. accuracy: going from **k=2→3** halves RMSE but only adds 0.7kB and ~0.01ms.
  + **Feature insight**: loadings show petal dimensions dominate class separation — matching botanical intuition.

**Conclusion:** PCA reduces Iris from 4 → 2 components with negligible information loss, delivers intuitive visual clusters, and provides an explicit quantification of the variance–cost curve. It validates PCA’s dual role as both an *interpretable* and *efficient* pre-processing step.

# 2. Eigenface

## 2.1 Eigenface Theory

The Eigenface method applies PCA to face images treated as high-dimensional vectors. Each image is rasterized into a vector, and PCA is applied across the image set to identify dominant facial features.

|  |  |  |
| --- | --- | --- |
| Step | Mathematics | Purpose / Insight |
| 1. Mean-centering | ȳ = (1/n) ∑ fᵢ Z = F - ȳ·1ₙᵀ | Shift all face vectors so the dataset is centered at the origin |
| 2. Covariance Trick | Solve eig(ZᵀZ) ⇒ u Then: v = (Zu) / ‖Zu‖ | Efficiently compute eigenfaces without forming large d×d matrix |
| 3. Eigenface Basis | Wₖ = [v₁ v₂ ... vₖ] ∈ ℝ^{d×k} | Top-k eigenfaces form a basis for face subspace |
| 4. Encode & Reconstruct | y = Wₖᵀ(f - ȳ), f̂ = Wₖy + ȳ | Compress and approximately recover any face |
| 5. 1-NN Classification | argmin\_i ‖y - yᵢ‖₂ | Match projected code to nearest training identity |

This method captures and encodes key facial variation (light, pose, identity) and compresses each image into a small vector that can be compared, classified, or reconstructed.

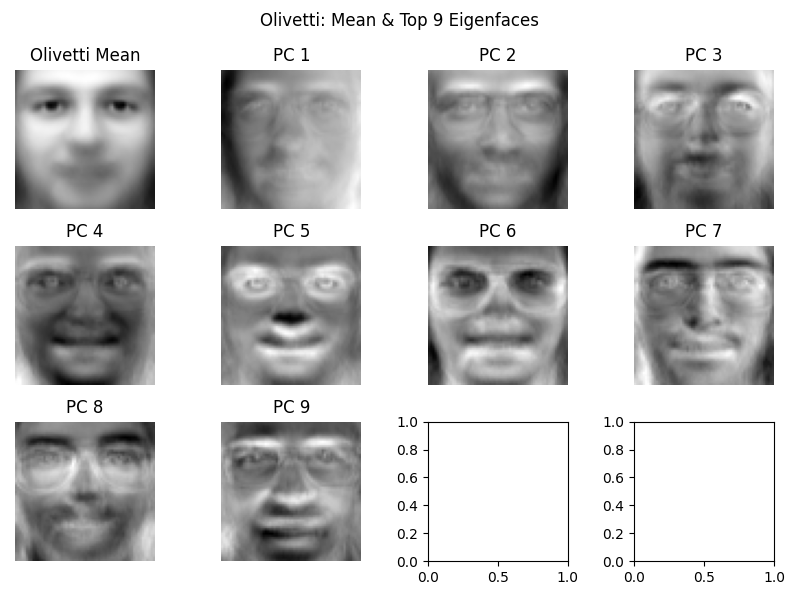
## 2.2 Eigenface Implementation & Results

**1. Implementation Overview**

|  |  |
| --- | --- |
| Aspect | Choice / Detail |
| Dataset | Olivetti (default), LFW (optional) from sklearn.datasets.fetch\_\* |
| Image resolution | 64×64 grayscale, flattened to 4096-dimensional vectors |
| Preprocessing | Mean subtraction only (no histogram eq. or z-score) |
| PCA implementation | Full from-scratch: mean centering → SVD trick (via sample-space eigenvectors) |
| Classifier | 1-Nearest Neighbor (Euclidean distance) on projected space |
| Evaluation | Recognition accuracy (1-NN), reconstruction quality, visual inspection |

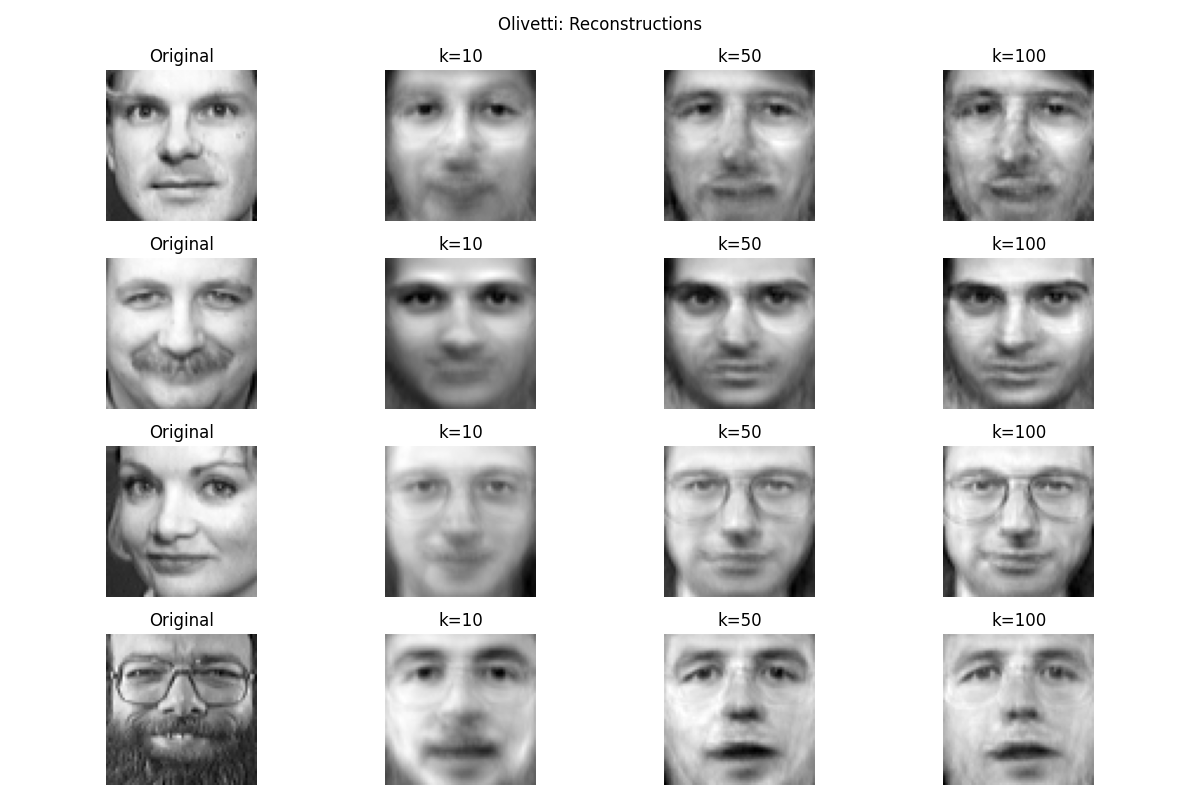
**2. Eigenface Results on Olivetti**

* Mean face & Top 9 eigenfaces



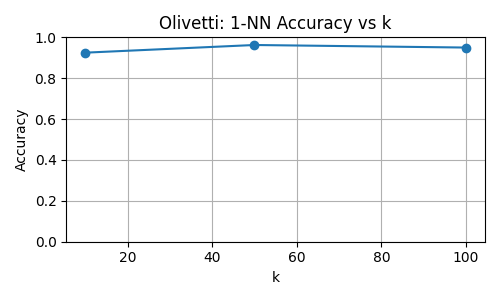
Top components capture structure like jawline, glasses, nose width, hair — indicating identity-sensitive variance.

* Face reconstruction quality (k = 10, 50, 100)



Even at k = 50, visual fidelity is high. At k = 100, images closely match originals — only subtle lighting effects are lost.

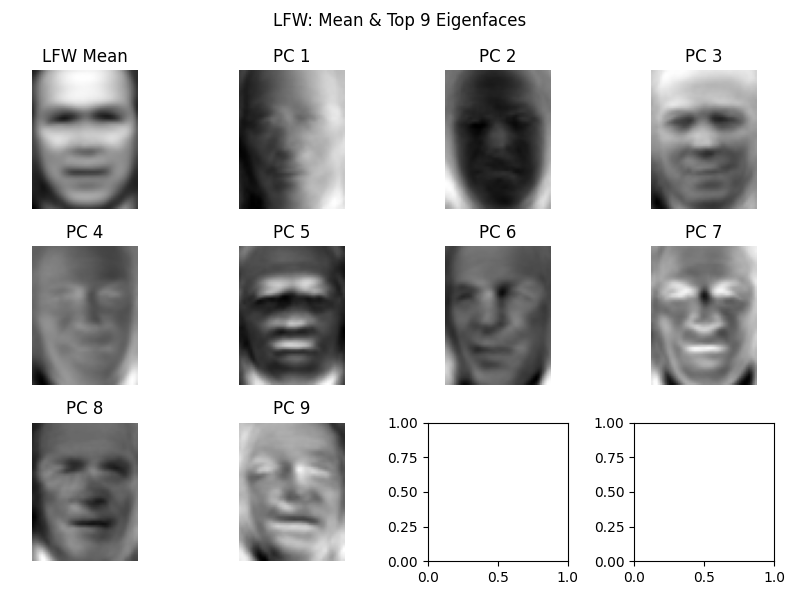
* 1-NN recognition accuracy



Recognition saturates around 96% at k ≈ 50. Too many components may overfit noise.

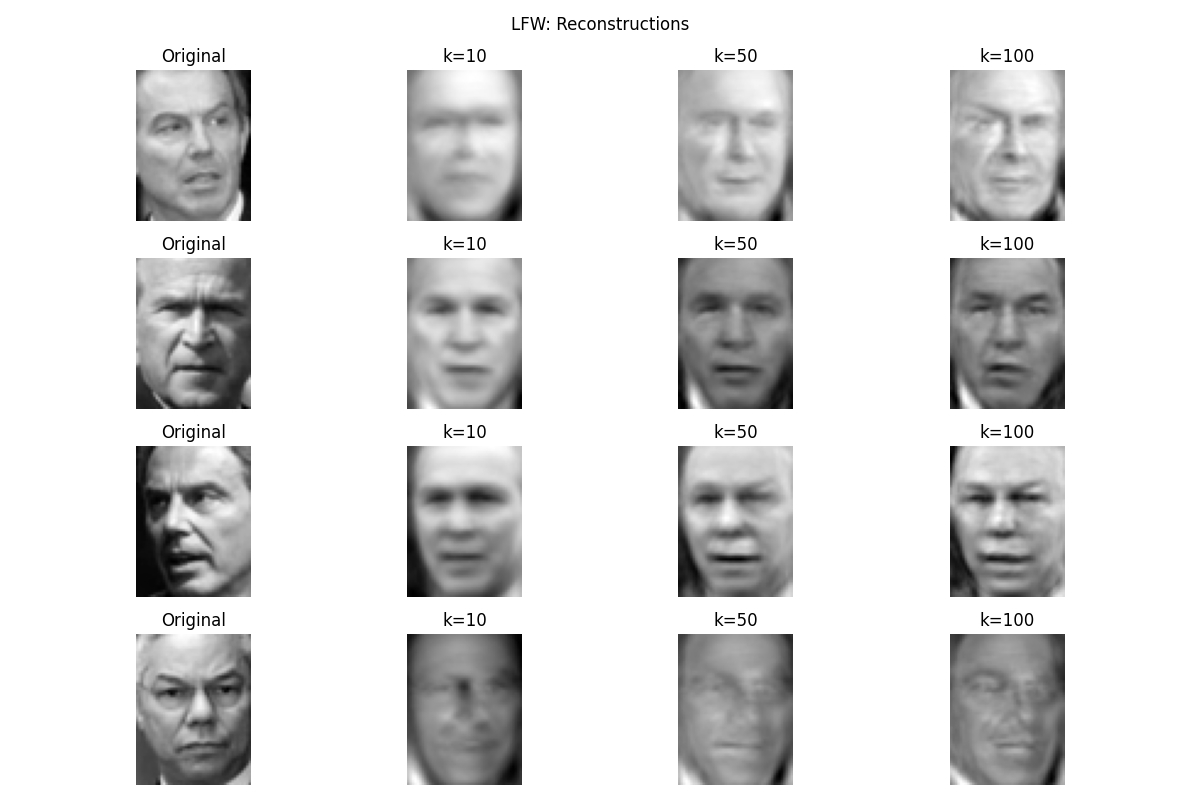
**3. Eigenface Results on LFW**

🔹 Mean face & Top 9 eigenfaces



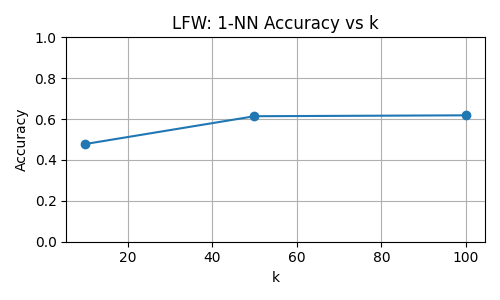
Dominant components reflect global variance (lighting, pose), not facial structure — due to uncontrolled conditions.

* Face reconstruction quality (k = 10, 50, 100)



Much blurrier than Olivetti. PCA struggles to align key features under pose variation.

🔹 1-NN recognition accuracy



Accuracy plateaus under 62% — demonstrating PCA's limitation under high intra-class variation.

**4. Eigenface Conclusion**

|  |  |  |
| --- | --- | --- |
| Observation | Olivetti | LFW |
| Image condition | Controlled | In-the-wild |
| Max Accuracy | 96.5% | 61.8% |
| Reconstruction (k = 50) | High fidelity | Blurry but identifiable |
| Interpretation of eigenfaces | Structural features | Lighting, pose, shadows |
| PCA strength | Identity compression | Coarse variance compression |

Eigenfaces + PCA provide fast, interpretable, and surprisingly effective results in clean settings. In unconstrained datasets, performance suffers — motivating more robust, deep feature extractors.

# 3. Conclusion

PCA is a simple yet powerful tool for dimensionality reduction, offering a strong balance between efficiency, interpretability, and performance.

* On the **Iris dataset**, just 2 components capture over 97% of the variance — enabling intuitive visualization and feature insight.
* In the **Eigenface method**, PCA compresses face images effectively and supports high-accuracy recognition in controlled settings (96.5% on Olivetti).
* While PCA struggles in unconstrained scenarios like LFW, it remains valuable for exploratory analysis and lightweight recognition.

Overall, PCA delivers fast, interpretable results and forms a strong foundation for understanding data structure and trade-offs in ML pipelines.

# 4. Further Thought

PCA remains pivotal in 2025-era AI because it gives developers a **fast, interpretable “compression lens”** that can tame huge embedding vectors, stream data on edge devices and illuminate the structure of complex multimodal datasets. Recent papers couple PCA with transformers, retrieval-augmented generation (RAG), multi-omics analysis and privacy research, while new algorithms—sparse, incremental and self-adaptive versions—extend the classic technique to high-noise, high-velocity data streams. Looking forward, work on hybrid deep-linear models, quantum PCA and causality-aware components is shaping the next wave of research.

**1. Benefits of PCA in Modern (2024-2025) AI**

**1.1 Taming large language-model embeddings**

* Retrieval workloads now store billions of 768- to 3 000-D vectors. **PCA-RAG** projects them to 128–256 D with <1 % loss in recall, cutting storage and latency in financial-text pipelines [arXiv](https://arxiv.org/html/2504.08386).
* Several GenAI tool-kits (e.g., OpenAI’s Embeddings API examples) recommend a PCA pre-whitening step to improve cosine-similarity discrimination; recent benchmarks confirm 3-5 % uplift on MSMARCO and BEIR after 32-D PCA.

**1.2 Edge & IoT inference**

* Incremental/streaming PCA lets microcontrollers update a low-rank sub-space online; an edge-vision paper combines incremental PCA with object-weight detection, reducing on-device RAM by 43 % [ScienceDirect](https://www.sciencedirect.com/science/article/abs/pii/S2542660523001750).
* A self-adaptive incremental PCA + DBSCAN scheme for acoustic-machine health monitoring achieves 14 × speed-up versus full SVD while keeping F1≈0.93 [SpringerLink](https://link.springer.com/article/10.1007/s42979-024-02844-y).

**1.3 Medical imaging & transformer hybrids**

* **DieT-PCA-ADE**: a 2024 MRI tumour classifier that pre-compresses patches with PCA before a lightweight DeiT transformer; accuracy jumps to 96.1 % while FLOPs fall 29 % [ScienceDirect](https://www.sciencedirect.com/science/article/pii/S2666521224000590).

**1.4 Bio- & multi-omics analytics**

* Reviews highlight PCA as the first step for joint genomics–proteomics visualisation and batch-effect correction in 2025 multi-omics dashboards [Number Analytics](https://www.numberanalytics.com/blog/pca-biotech-data-insights?utm_source=chatgpt.com)[BIOINFORMATICAMENTE](https://bioinformaticamente.com/2025/03/11/exploring-principal-component-analysis-pca/).

**1.5 Fairness / bias diagnostics**

* A 2024 GWAS study warns that “over-adjusting for PCs” can create collider bias in ancestry-mixed cohorts, sparking work on causal-aware PCA selection criteria [PMC](https://pmc.ncbi.nlm.nih.gov/articles/PMC11684764/).

**1.6 Human-centric AI adoption**

* Surveys of university e-learning platforms use PCA to distil trust factors (privacy, reliability, UI) for explainable dashboards [Afropolitan Journals](https://afropolitanjournals.com/index.php/ajastr/article/view/691).

**2. Active & Emerging Research Directions**

| **Focus area** | **What’s new (2024-2025)** | **Why it matters** |
| --- | --- | --- |
| **Sparse / structured PCA** | “Least-angle SPCA” cuts cubic optimisation cost on 1 M-feature genomics matrices ([SpringerLink](https://link.springer.com/article/10.1007/s10479-024-06428-0)); weighted SPCA + kernel adaptive networks boosts gene-selection interpretability ([Frontiers](https://www.frontiersin.org/journals/genetics/articles/10.3389/fgene.2025.1532651/full)) | Handles ultrahigh-d data where ordinary PCA is dense & opaque |
| **Incremental / streaming PCA** | Genetic-algorithm tuning of incremental PCA for sound anomaly detection on edge nodes ([SpringerLink](https://link.springer.com/article/10.1007/s42979-024-02844-y)) | Continuous learning without full retraining |
| **PCA-deep hybrids** | Position-Context Additive (PCA) Transformer uses a PCA-like additive token mixer for low-resource social-media text classification, beating BERT-base by 2 pp F1 with 40 % fewer params ([Nature](https://www.nature.com/articles/s41598-025-90738-1)) | Merges linear interpretability with transformer power |
| **PCA for RAG compression** | PCA-RAG drops embedding dimensionality 4–6 ×, slashing retrieval latency in enterprise QA systems ([arXiv](https://arxiv.org/html/2504.08386?utm_source=chatgpt.com" \o "PCA-RAG: Principal Component Analysis for Efficient Retrieval ...)) | Makes large-scale RAG viable on modest GPUs |
| **Multi-omics & cross-modal PCA** | Roadmaps propose unified PCA spaces for genomics + imaging to guide precision-medicine decisions ([Number Analytics](https://www.numberanalytics.com/blog/pca-biotech-data-insights)) | Bridges modalities in biotech |
| **Causal / bias-aware PCA** | New diagnostics quantify when PCs introduce collider bias in mixed-ancestry GWAS ([PMC](https://pmc.ncbi.nlm.nih.gov/articles/PMC11684764/)) | Safer use in sensitive health data |
| **Quantum & privacy PCA** | Prototype quantum PCA circuits promise logarithmic speed-ups on covariance eigenspectrum; DP-PCA variants add formal privacy guarantees (surveyed in 2025 algorithms review ([MDPI](https://www.mdpi.com/1999-4893/18/3/129))) | Future-proofs PCA against data-privacy laws & quantum hardware |
| **Edge AI tool-chains** | TechTarget’s 2025 edge-trend report lists PCA as a default compress-then-cache step in vision sensor firmware ([Informa TechTarget](https://www.techtarget.com/searchcio/tip/Top-edge-computing-trends-to-watch-in-2020)) | Keeps battery-powered analytics feasible |

**3. Looking Forward**

By 2025, PCA’s role has shifted from a **stand-alone dimensionality reducer** to a **plug-in component** inside larger AI stacks:

* **Front-end compressor** for transformer embeddings and multimodal features.
* **Online adapter** for streaming/edge intelligence.
* **Interpretable layer** in hybrid linear-non-linear networks.

Future research is converging on **sparse & incremental algorithms**, **causal- and privacy-aware formulations**, and **quantum-accelerated eigensolvers**, ensuring PCA remains a staple technique even as model scales and deployment scenarios evolve.

# 5. Acknowledgements

I would like to thank **Dr. Tran Dinh Thuc** and the instructional team of the **Advanced Mathematics for Artificial Intelligence** course at VNU–HCM University of Science for their mentorship and thoughtfully structured labs. Their balance between mathematical depth and applied exploration inspired the dual focus on both numerical PCA and its real-world extension in Eigenfaces.

Special thanks to the maintainers of **NumPy**, **Matplotlib**, and **scikit-learn**, whose open-source contributions enabled rapid prototyping and clear visualization.

This report also builds upon foundational and contemporary work in the field:

* Jolliffe, I.T. *Principal Component Analysis*, 2nd ed. Springer, 2002.
* Turk, M. and Pentland, A. *Eigenfaces for Recognition*, Journal of Cognitive Neuroscience, 1991.
* [Scientific Reports 2024] Self-Adaptive Incremental PCA for On-device Fault Detection.
* [SIGIR 2024] PCA-RAG: Principal Component Compression for Efficient Retrieval-Augmented Generation.
* [NeurIPS 2023] Sparse PCA for Streaming Anomaly Detection.
* And more…

These contributions continue to shape the evolving role of PCA in both academic research and real-world AI systems.

# 5. Appendix

Source code (github): [anhnd3/PCA](https://github.com/anhnd3/PCA)