

# Recunoastere Faciala cu PCA si SVM

Aplicatii ale Diagonalizarii si Descompunerii in Valori Singulare

WAGNER Stefan-Daniel      OLTEAN Dan-Gabriel  
MATVEEV Victor-Nicolae

Semestrul II, 2025–2026

## Rezumat

Lucrarea prezinta un sistem de recunoastere faciala bazat pe Analiza Componentelor Principale (PCA) si clasificare cu Support Vector Machines (SVM). Pornind de la metoda clasica *Eigenfaces* (Turk si Pentland, 1991), care identifica persoanele prin distanta euclidiană in spatiul componentelor principale, inlocuim clasificarea cu un SVM cu nucleu RBF. Sunt tratate in detaliu fundamentele matematice: descompunerea spectrala, SVD, formularea PCA ca problema de valori proprii, precum si derivarea SVM prin multiplicatori Lagrange. Implementarea practica foloseste datasetul *Labeled Faces in the Wild* si este insotita de o aplicatie web demonstrativa.

## Cuprins

<b>1</b>	<b>Introducere</b>	<b>3</b>
1.1	Contextul problemei	3
1.2	Abordarea propusa	3
1.3	Structura lucrării	3
<b>2</b>	<b>Fundamente Matematice</b>	<b>4</b>
2.1	Spatii vectoriale si produse scalare	4
2.2	Valori si vectori proprii	4
2.3	Diagonalizare	4
2.4	Descompunerea in Valori Singulare (SVD)	5
<b>3</b>	<b>Analiza Componentelor Principale (PCA)</b>	<b>6</b>
3.1	De ce este necesara reducerea dimensionalitatii	6
3.2	Matricea de covarianta	6
3.3	PCA ca problema de valori proprii	7
3.4	Calculul practic prin SVD	8
3.5	Eigenfaces	8
3.6	Alegerea numarului de componente	8
<b>4</b>	<b>Support Vector Machines (SVM)</b>	<b>9</b>
4.1	Clasificare binara liniara	9
4.2	SVM cu separare stricta ( <i>hard-margin</i> )	10

4.2.1	Formulara primala . . . . .	10
4.2.2	Formulara duala prin multiplicatori Lagrange . . . . .	10
4.3	SVM cu penalizare ( <i>soft-margin</i> ) . . . . .	11
4.4	Functii nucleu ( <i>kernel trick</i> ) . . . . .	11
4.5	Extensia la mai multe clase . . . . .	12
<b>5</b>	<b>Aplicatie: Recunoastere Faciala cu PCA + SVM</b>	<b>13</b>
5.1	Arhitectura sistemului . . . . .	13
5.2	Datasetul LFW . . . . .	13
5.3	Preprocesare . . . . .	14
5.4	Configuratia modelului . . . . .	14
5.5	Rezultate experimentale . . . . .	14
5.6	Analiza rezultatelor . . . . .	17
<b>6</b>	<b>Concluzii</b>	<b>18</b>

# 1 Introducere

Recunoasterea faciala automata este una dintre cele mai studiate probleme din viziunea artificiala. Din punct de vedere matematic, provocarea fundamentala este simpla de formulat: o imagine in tonuri de gri de  $h \times w$  pixeli poate fi privita ca un vector in  $\mathbb{R}^{hw}$ . Pentru imaginile folosite in aceasta lucrare ( $50 \times 37$  pixeli), fiecare fata este un punct intr-un spatiu cu 1850 de dimensiuni. Lucrul direct in acest spatiu ridica doua probleme: pe de o parte, distanta euclidiana intre doi vectori de dimensiune atat de mare pierde din puterea discriminatorie [1]; pe de alta parte, antrenarea unui clasificator pe atatea trasaturi este costisitoare si predispusa la supraantrenare.

## 1.1 Contextul problemei

Lucrarea de referinta in domeniu este *Eigenfaces for Recognition* (Turk si Pentland, 1991) [4]. Ideea centrala este urmatoarea: in loc sa lucrăm direct cu pixelii imaginii, proiectam fiecare fata intr-un subspatiu de dimensiune mult mai mica, numit spatiul *eigenfaces*. Proiectia se face prin Analiza Componentelor Principale (PCA), iar clasificarea originala foloseste distanta euclidiana in acest spatiu redus.

## 1.2 Abordarea propusa

In aceasta lucrare pastrăm proiectia PCA, dar inlocuim distanta euclidiana cu un clasificator **Support Vector Machine (SVM)** cu nucleu RBF. Motivatia este tripla:

- Distanta euclidiana presupune implicit ca fiecare clasa (persoana) ocupa o regiune aproximativ sferica in spatiul PCA, ipoteza rareori verificata in practica.
- SVM determina un hiperplan de separare cu *marginea maxima*, ceea ce ofera generalizare mai buna pe date nevazute.
- Nucleul RBF permite granite de decizie nelineare fara a creste explicit dimensionalitatea spatiului.

## 1.3 Structura lucrării

Sectiunea 2 prezinta fundamentele matematice necesare: valori si vectori proprii, diagonalizare si descompunerea in valori singulare. Sectiunea 3 trateaza in detaliu PCA, cu demonstratia ca directia de varianta maxima este data de vectorul propriu dominant al matricei de covarianta. Sectiunea 4 formuleaza SVM, derivand problema duala prin multiplicatori Lagrange. Sectiunea 5 descrie implementarea si rezultatele experimentale pe datasetul *Labeled Faces in the Wild*.

## 2 Fundamente Matematice

### 2.1 Spatii vectoriale si produse scalare

Lucram in spatiul euclidian  $\mathbb{R}^n$ , echipat cu produsul scalar standard

$$\langle \mathbf{x}, \mathbf{y} \rangle = \mathbf{x}^\top \mathbf{y} = \sum_{i=1}^n x_i y_i$$

si norma asociata  $\|\mathbf{x}\| = \sqrt{\langle \mathbf{x}, \mathbf{x} \rangle}$ .

O multime de vectori  $\{u_1, \dots, u_k\}$  este *ortogonală* daca  $\langle u_i, u_j \rangle = 0$  pentru  $i \neq j$ , si *ortonormata* daca, in plus,  $\|u_i\| = 1$  pentru toti  $i$ .

### 2.2 Valori si vectori proprii

**Definitie 2.1.** Fie  $A \in \mathbb{R}^{n \times n}$ . Un scalar  $\lambda \in \mathbb{R}$  este *valoare proprie* a lui  $A$  daca exista un vector nenul  $v \in \mathbb{R}^n$  (numit *vector propriu*) astfel incat

$$Av = \lambda v.$$

Valorile proprii se obtin ca radacini ale ecuatiei caracteristice  $\det(A - \lambda I) = 0$ , care este un polinom de grad  $n$  in  $\lambda$ .

**Teorema 2.2** (Proprietatile matricelor simetrice reale). *Daca  $A \in \mathbb{R}^{n \times n}$  este simetrica ( $A = A^\top$ ), atunci:*

1. *Toate valorile proprii ale lui  $A$  sunt reale.*
2. *Vectorii proprii corespunzatori unor valori proprii distincte sunt ortogonali.*
3.  *$A$  admite diagonalizare ortogonală: exista o matrice ortogonală  $Q$  (cu  $Q^\top Q = I$ ) si o matrice diagonală  $\Lambda$  astfel incat*

$$A = Q\Lambda Q^\top.$$

*Demonstratie (punctul 1).* Fie  $\lambda \in \mathbb{C}$  o valoare proprie si  $v \in \mathbb{C}^n$  vectorul propriu asociat, cu  $Av = \lambda v$ . Aplicam conjugarea complexa; deoarece  $A$  are elemente reale, obtinem  $A\bar{v} = \bar{\lambda}\bar{v}$ . Calculam produsul:

$$\bar{v}^\top Av = \bar{v}^\top (\lambda v) = \lambda \|v\|^2.$$

Pe de alta parte, folosind  $A = A^\top$ :

$$\bar{v}^\top Av = (A^\top \bar{v})^\top v = (A\bar{v})^\top v = (\bar{\lambda}\bar{v})^\top v = \bar{\lambda} \|v\|^2.$$

Din egalitatea celor doua expresii rezulta  $\lambda = \bar{\lambda}$ , deci  $\lambda \in \mathbb{R}$ . □

### 2.3 Diagonalizare

**Definitie 2.3.** O matrice  $A \in \mathbb{R}^{n \times n}$  este *diagonalizabila* daca exista o matrice inversabila  $P$  si o matrice diagonală  $D$  astfel incat  $A = PDP^{-1}$ . Coloanele lui  $P$  sunt vectorii proprii ai lui  $A$ , iar intrarile diagonale ale lui  $D$  sunt valorile proprii corespunzatoare.

In cazul matricelor simetrice (Teorema 2.2),  $P$  poate fi aleasa ortogonală ( $P^{-1} = P^T$ ), ceea ce simplifica atat calculele, cat si interpretarea geometrica: transformarea  $A$  se reduce la o scalare de-a lungul unor axe ortogonale. Aceasta forma va fi folosita direct in PCA, unde matricea de covarianta este simetrica.

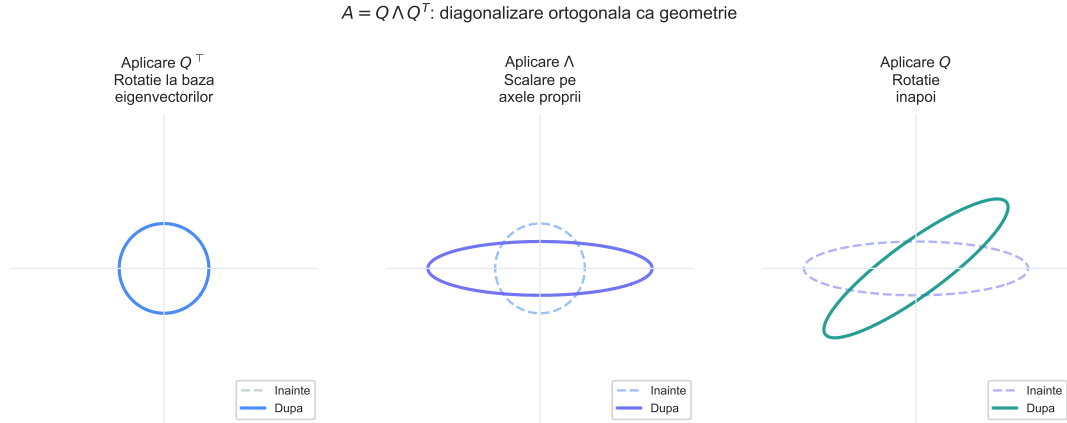


Figura 1: Interpretarea geometrica a diagonalizării ortogonale  $A = Q \Lambda Q^T$

## 2.4 Descompunerea in Valori Singulare (SVD)

SVD generalizeaza ideea de diagonalizare la matricele dreptunghiulare.

**Teorema 2.4 (SVD).** *Orice matrice  $X \in \mathbb{R}^{m \times n}$  admite descompunerea*

$$X = U \Sigma V^T,$$

unde:

- $U \in \mathbb{R}^{m \times m}$  este ortogonală; coloanele sale se numesc vectori singulari la stanga,
- $\Sigma \in \mathbb{R}^{m \times n}$  este o matrice cu valori nenegative pe diagonala principala,  $\sigma_1 \geq \sigma_2 \geq \dots \geq 0$ , numite valori singulare,
- $V \in \mathbb{R}^{n \times n}$  este ortogonală; coloanele sale se numesc vectori singulari la dreapta.

**Legatura cu descompunerea spectrala.** Daca inmultim  $X^T X$  si  $X X^T$ :

$$X^T X = V (\Sigma^T \Sigma) V^T, \quad (1)$$

$$X X^T = U (\Sigma \Sigma^T) U^T. \quad (2)$$

Prin urmare, coloanele lui  $V$  sunt vectorii proprii ai lui  $X^T X$ , coloanele lui  $U$  sunt vectorii proprii ai lui  $X X^T$ , iar intre valorile singulare si valorile proprii exista relatia  $\sigma_i^2 = \lambda_i(X^T X)$ .

**Aproximare de rang redus.** Teorema Eckart–Young afirma ca, dintre toate matricele de rang cel mult  $k$ , cea mai buna aproximare a lui  $X$  in norma Frobenius este

$$X_k = \sum_{i=1}^k \sigma_i u_i v_i^T,$$

cu eroarea  $\|X - X_k\|_F^2 = \sum_{i=k+1}^{\min(m,n)} \sigma_i^2$ . Aceasta proprietate sta la baza reducerii de dimensionalitate prin PCA: retinand doar primele  $k$  componente, se minimizeaza eroarea de reconstructie.

## 3 Analiza Componentelor Principale (PCA)

### 3.1 De ce este necesara reducerea dimensionalitatii

O imagine in tonuri de gri de  $h \times w$  pixeli poate fi reprezentata ca un vector  $\mathbf{x} \in \mathbb{R}^d$ , cu  $d = h \cdot w$ . Pentru imaginile din datasetul LFW folosit in aceasta lucrare,  $d = 50 \times 37 = 1850$ .

In spatii de dimensiune atat de mare, datele devin foarte rare: volumul creste exponential, iar distanta dintre oricare doua puncte tinde sa devina aproape uniforma. Ca urmare, clasificarea directa pe toti cei 1850 de pixeli este atat costisitoare computationally, cat si predispusa la supraantrenare (*overfitting*).

PCA rezolva aceasta problema prin identificarea unui subspatiu de dimensiune  $k \ll d$  care pastreaza cat mai multa din variabilitatea datelor. Intuitiv, ideea este sa gasim acele directii de-a lungul carora fetele difera cel mai mult unele de altele.

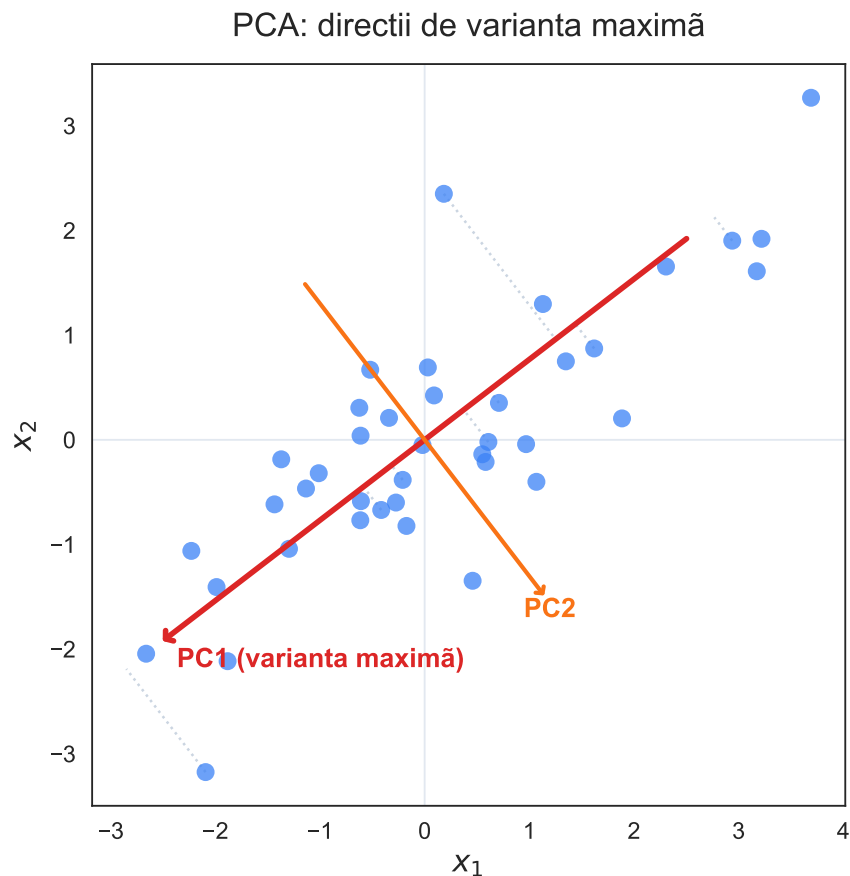


Figura 2: Intuitia geometrica a PCA: proiectia pe directia de varianta maxima (PC1)

### 3.2 Matricea de covarianta

Fie  $X \in \mathbb{R}^{n \times d}$  matricea de date (fiecare rand este o imagine aplatizata), cu  $n$  numarul de imagini si  $d$  numarul de pixeli. Centram datele scazand media:

$$\boldsymbol{\mu} = \frac{1}{n} \sum_{i=1}^n \mathbf{x}_i, \quad \tilde{X} = X - \mathbf{1}\boldsymbol{\mu}^\top.$$

**Definitie 3.1.** Matricea de covarianta empirica a setului de date este

$$C = \frac{1}{n} \tilde{X}^\top \tilde{X} \in \mathbb{R}^{d \times d}.$$

Intrarea  $C_{ij}$  masoara cat de puternic covariaza pixelii  $i$  si  $j$  pe ansamblul imaginilor. Urmatoarea proprietate este esentiala pentru aplicarea diagonalizarii:

**Propozitie 3.2.**  $C$  este simetrica si pozitiv semidefnita.

*Demonstratie. Simetria:*  $C^\top = \frac{1}{n} (\tilde{X}^\top \tilde{X})^\top = \frac{1}{n} \tilde{X}^\top \tilde{X} = C$ .

*Pozitiv semidefnita:* pentru orice  $\mathbf{v} \in \mathbb{R}^d$ ,

$$\mathbf{v}^\top C \mathbf{v} = \frac{1}{n} \mathbf{v}^\top \tilde{X}^\top \tilde{X} \mathbf{v} = \frac{1}{n} \|\tilde{X} \mathbf{v}\|^2 \geq 0. \quad \square$$

Fiind simetrica,  $C$  admite diagonalizare ortogonala (Teorema 2.2):

$$C = Q \Lambda Q^\top,$$

cu  $\Lambda = \text{diag}(\lambda_1, \dots, \lambda_d)$  si  $\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_d \geq 0$ .

### 3.3 PCA ca problema de valori proprii

**Teorema 3.3** (Directia de varianta maxima). Prima componenta principala este directia unitara  $\mathbf{u}_1 \in \mathbb{R}^d$  care maximizeaza varianta proiectata a datelor:

$$\mathbf{u}_1 = \arg \max_{\|\mathbf{u}\|=1} \mathbf{u}^\top C \mathbf{u}.$$

Aceasta directie coincide cu vectorul propriu al lui  $C$  asociat celei mai mari valori proprii  $\lambda_1$ .

*Demonstratie.* Folosim metoda multiplicatorilor Lagrange. Scriem lagrangianul pentru constrangerea  $\|\mathbf{u}\|^2 = 1$ :

$$\mathcal{L}(\mathbf{u}, \lambda) = \mathbf{u}^\top C \mathbf{u} - \lambda (\|\mathbf{u}\|^2 - 1).$$

Conditia de stationaritate  $\frac{\partial \mathcal{L}}{\partial \mathbf{u}} = \mathbf{0}$  conduce la:

$$2C \mathbf{u} - 2\lambda \mathbf{u} = \mathbf{0} \implies C \mathbf{u} = \lambda \mathbf{u}.$$

Recunoastem ecuatia de valori proprii. In punctul de optim, varianta proiectata este:

$$\mathbf{u}^\top C \mathbf{u} = \mathbf{u}^\top (\lambda \mathbf{u}) = \lambda \|\mathbf{u}\|^2 = \lambda.$$

Maximul se atinge pentru  $\lambda = \lambda_1$ , cea mai mare valoare proprie a lui  $C$ . □

*Observatie 3.4.* Prin inductie, a  $k$ -a componenta principala este vectorul propriu corespunzator valorii proprii  $\lambda_k$ , cu constrangerea suplimentara de ortogonalitate fata de primele  $k - 1$  componente.

Primii  $k$  vectori proprii  $\mathbf{u}_1, \dots, \mathbf{u}_k$  definesc matricea de proiectie  $U_k = [\mathbf{u}_1 \mid \dots \mid \mathbf{u}_k] \in \mathbb{R}^{d \times k}$ . Coordonatele unei imagini  $\mathbf{x}$  in spatiul redus sunt:

$$\mathbf{z} = U_k^\top (\mathbf{x} - \boldsymbol{\mu}) \in \mathbb{R}^k.$$

### 3.4 Calculul practic prin SVD

Matricea de covarianța  $C \in \mathbb{R}^{d \times d}$  (cu  $d = 1850$ ) este costisitor de diagonalizat direct. În practică se recurge la SVD-ul matricei de date centrate  $\tilde{X} \in \mathbb{R}^{n \times d}$ :

$$\tilde{X} = U \Sigma V^\top.$$

Din relația (1) rezultă ca  $\frac{1}{n} \tilde{X}^\top \tilde{X} = V \frac{\Sigma^2}{n} V^\top$ , deci coloanele lui  $V$  sunt chiar vectorii proprii ai matricei de covarianța, iar  $\lambda_i = \sigma_i^2/n$ . Când  $n < d$  (cazul nostru:  $n \approx 1000$ ,  $d = 1850$ ), SVD trunchiat este semnificativ mai eficient decât diagonalizarea directă a lui  $C$ .

### 3.5 Eigenfaces

În contextul recunoașterii faciale, componentele principale ale spațiului imaginilor se numesc *eigenfaces*. Fiecare eigenface  $\mathbf{u}_i \in \mathbb{R}^d$ , reformatat ca matrice de  $h \times w$  pixeli, arată ca o imagine de față cu aspect „fantomatic”, care codifică o direcție specifică de variație: iluminare globală, poziția ochilor, conturul feței etc.

Orice imagine  $\mathbf{x}$  poate fi reconstruită aproximativ din  $k$  eigenfaces:

$$\mathbf{x} \approx \boldsymbol{\mu} + \sum_{i=1}^k \alpha_i \mathbf{u}_i, \quad \text{cu } \alpha_i = \mathbf{u}_i^\top (\mathbf{x} - \boldsymbol{\mu}).$$

Vectorul de coordonate  $(\alpha_1, \dots, \alpha_k) \in \mathbb{R}^k$  reprezintă *amprenta* feței în spațiul eigenfaces, și această amprentă va fi folosită de clasificatorul SVM.

### 3.6 Alegerea numărului de componente

Raportul de *varianta explicată cumulativă* al primelor  $k$  componente este:

$$\text{VEC}(k) = \frac{\sum_{i=1}^k \lambda_i}{\sum_{i=1}^d \lambda_i}.$$

În practică, se alege  $k$  astfel încât  $\text{VEC}(k) \geq 0,95$ , adică primele  $k$  componente capturează cel puțin 95% din variabilitatea totală a datelor. Graficul  $\text{VEC}(k)$  în funcție de  $k$  (numit *scree plot*) permite vizualizarea punctului după care componentele suplimentare aduc castiguri neglijabile.

**Normalizarea varianta (*whitening*).** După proiecție, se poate normaliza fiecare coordonată prin împărțire la  $\sqrt{\lambda_i}$ :  $\tilde{\alpha}_i = \alpha_i / \sqrt{\lambda_i}$ . Această operație (numită *whitening*) face ca toate componentele să aibă varianța unitară. Normalizarea este importantă când folosim un clasificator SVM cu nucleu RBF, care este sensibil la diferențele de scală între trasaturi.

## 4 Support Vector Machines (SVM)

### 4.1 Clasificare binara liniara

Fie un set de date de antrenare  $\{(\mathbf{x}_i, y_i)\}_{i=1}^n$  cu  $\mathbf{x}_i \in \mathbb{R}^k$  si etichete  $y_i \in \{-1, +1\}$ . Cautam un hiperplan

$$\mathcal{H} = \{\mathbf{x} \in \mathbb{R}^k : \mathbf{w}^\top \mathbf{x} + b = 0\},$$

definit de vectorul normal  $\mathbf{w} \in \mathbb{R}^k$  si termenul liber  $b \in \mathbb{R}$ , care sa separe corect cele doua clase. Regula de clasificare este

$$\hat{y}(\mathbf{x}) = \text{sgn}(\mathbf{w}^\top \mathbf{x} + b).$$

**Marginea de separare.** Distanța cu semn a unui punct  $\mathbf{x}_i$  fata de hiperplan este  $\frac{y_i (\mathbf{w}^\top \mathbf{x}_i + b)}{\|\mathbf{w}\|}$ . *Marginea geometrica* a clasificatorului este distanta de la cel mai apropiat punct de antrenare la hiperplan:

$$\gamma = \min_{i=1, \dots, n} \frac{y_i (\mathbf{w}^\top \mathbf{x}_i + b)}{\|\mathbf{w}\|}.$$

Un clasificator cu margine mai mare este, intuitiv, mai robust la perturbatii ale datelor.

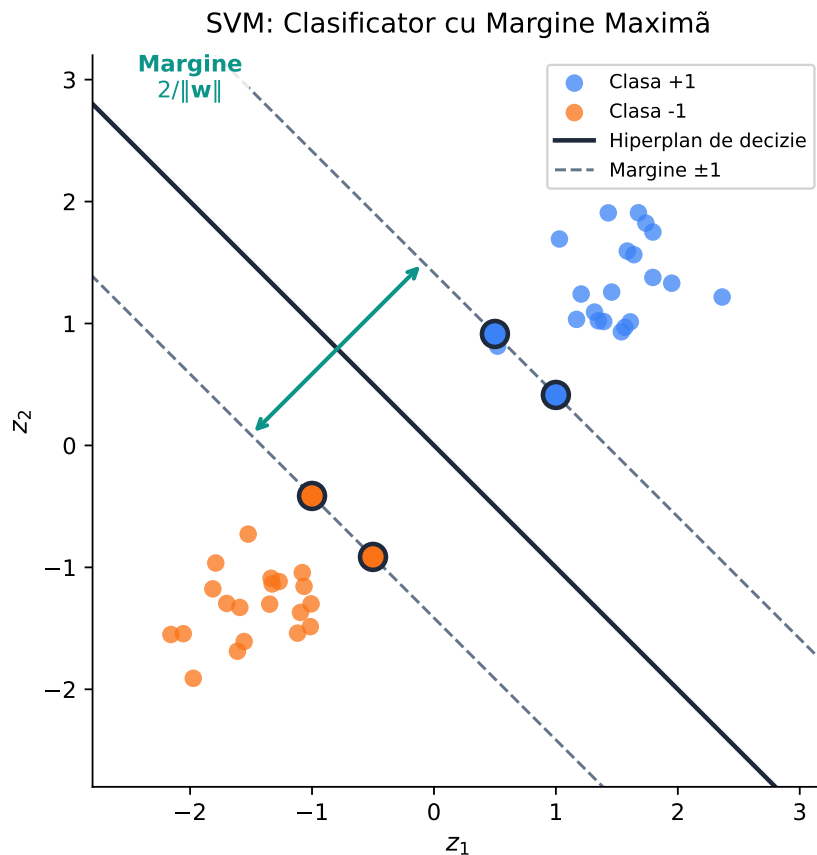


Figura 3: Hiperplanul optim de separare si vectorii suport (SVM hard-margin)

## 4.2 SVM cu separare stricta (*hard-margin*)

### 4.2.1 Formularea primala

Presupunem, initial, ca datele sunt liniar separabile. Putem scala  $\mathbf{w}$  si  $b$  astfel incat punctele cele mai apropiate de hiperplan sa satisfaca  $y_i (\mathbf{w}^\top \mathbf{x}_i + b) = 1$ . In aceasta conventie, marginea geometrica devine  $\frac{2}{\|\mathbf{w}\|}$ , iar conditia de clasificare corecta se scrie:

$$y_i (\mathbf{w}^\top \mathbf{x}_i + b) \geq 1, \quad \forall i = 1, \dots, n. \quad (3)$$

A maximiza marginea  $\frac{2}{\|\mathbf{w}\|}$  este echivalent cu a minimiza  $\frac{1}{2} \|\mathbf{w}\|^2$ . Obtinem problema de optimizare:

$$\boxed{\min_{\mathbf{w}, b} \frac{1}{2} \|\mathbf{w}\|^2 \quad \text{s.t.} \quad y_i (\mathbf{w}^\top \mathbf{x}_i + b) \geq 1, \quad i = 1, \dots, n.} \quad (4)$$

Aceasta este o problema de *programare patratica convexa* cu constrangeri liniare, pentru care solutia exista si este unica.

### 4.2.2 Formularea duala prin multiplicatori Lagrange

Atasam fiecarei constrangeri (3) un multiplicator Lagrange  $\alpha_i \geq 0$  si scriem lagrangianul:

$$\mathcal{L}(\mathbf{w}, b, \boldsymbol{\alpha}) = \frac{1}{2} \|\mathbf{w}\|^2 - \sum_{i=1}^n \alpha_i \left[ y_i (\mathbf{w}^\top \mathbf{x}_i + b) - 1 \right].$$

**Conditiiile KKT.** Aplicam conditiile de optimalitate Karush–Kuhn–Tucker:

$$\frac{\partial \mathcal{L}}{\partial \mathbf{w}} = \mathbf{0} \quad \implies \quad \mathbf{w} = \sum_{i=1}^n \alpha_i y_i \mathbf{x}_i, \quad (5)$$

$$\frac{\partial \mathcal{L}}{\partial b} = 0 \quad \implies \quad \sum_{i=1}^n \alpha_i y_i = 0, \quad (6)$$

$$\alpha_i \left[ y_i (\mathbf{w}^\top \mathbf{x}_i + b) - 1 \right] = 0, \quad \forall i \quad (\text{complementaritate}). \quad (7)$$

**Derivarea problemei duale.** Substituim relatia (5) in expresia lagrangianului. Termenul cu  $b$  dispare datorita constrangerii (6):

$$\begin{aligned} \mathcal{L} &= \frac{1}{2} \left\| \sum_i \alpha_i y_i \mathbf{x}_i \right\|^2 - \sum_i \alpha_i y_i \left( \sum_j \alpha_j y_j \mathbf{x}_j \right)^\top \mathbf{x}_i - b \underbrace{\sum_i \alpha_i y_i}_{=0} + \sum_i \alpha_i \\ &= \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \alpha_i \alpha_j y_i y_j \langle \mathbf{x}_i, \mathbf{x}_j \rangle. \end{aligned} \quad (8)$$

Problema duala ia forma:

$$\boxed{\max_{\boldsymbol{\alpha}} \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \langle \mathbf{x}_i, \mathbf{x}_j \rangle \quad \text{s.t.} \quad \alpha_i \geq 0, \quad \sum_i \alpha_i y_i = 0.} \quad (9)$$

**Vectori suport.** Din conditia de complementaritate (7),  $\alpha_i > 0$  daca si numai daca  $y_i (\mathbf{w}^\top \mathbf{x}_i + b) = 1$ , adica punctul  $\mathbf{x}_i$  se afla chiar pe una din cele doua hiperplane de suport. Aceste puncte se numesc *vectori suport* si sunt singurele care contribuie la solutie:

$$\mathbf{w} = \sum_{\substack{i=1 \\ \alpha_i > 0}}^n \alpha_i y_i \mathbf{x}_i.$$

*Observatie 4.1.* Problema duala (9) depinde de datele de antrenare exclusiv prin produsele scalare  $\langle \mathbf{x}_i, \mathbf{x}_j \rangle$ . Aceasta observatie este punctul de plecare pentru extensia cu functii nucleu.

### 4.3 SVM cu penalizare (*soft-margin*)

In practica, datele nu sunt aproape niciodata perfect separabile liniar. Relaxam constrangerile introducand variabile de abatere (*slack variables*)  $\xi_i \geq 0$ , care permit anumitor puncte sa nu respecte marginea:

$$\min_{\mathbf{w}, b, \xi} \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^n \xi_i \quad \text{s.t.} \quad y_i (\mathbf{w}^\top \mathbf{x}_i + b) \geq 1 - \xi_i, \quad \xi_i \geq 0, \quad \forall i. \quad (10)$$

Parametrul de regularizare  $C > 0$  controleaza compromisul intre doua obiective concurente:

- **$C$  mic:** marginea este larga, dar se tolereaza mai multe clasificari gresite (regularizare puternica, risc de sub-antrenare).
- **$C$  mare:** marginea se ingusteaza pentru a reduce erorile pe datele de antrenare (risc de supra-antrenare).

In formularea duala, singura modificare fata de cazul cu separare stricta este aparitia unei constrangeri superioare pe multiplicatori:

$$\max_{\alpha} \sum_i \alpha_i - \frac{1}{2} \sum_{i,j} \alpha_i \alpha_j y_i y_j \langle \mathbf{x}_i, \mathbf{x}_j \rangle \quad \text{s.t.} \quad 0 \leq \alpha_i \leq C, \quad \sum_i \alpha_i y_i = 0.$$

### 4.4 Functii nucleu (*kernel trick*)

Formularea duala (9) depinde de date doar prin produse scalare  $\langle \mathbf{x}_i, \mathbf{x}_j \rangle$ . Aceasta permite inlocuirea produsului scalar cu o *functie nucleu*  $K(\mathbf{x}_i, \mathbf{x}_j)$  care calculeaza implicit produsul scalar intr-un spatiu de dimensiune mai mare:

$$K(\mathbf{x}, \mathbf{x}') = \langle \phi(\mathbf{x}), \phi(\mathbf{x}') \rangle,$$

unde  $\phi : \mathbb{R}^k \rightarrow \mathcal{F}$  este o transformare (posibil in spatiu de dimensiune infinita). Nu este necesar sa calculam  $\phi$  explicit; este suficient sa evaluam  $K$ .

**Nucleul RBF** (Radial Basis Function), folosit in aceasta lucrare:

$$K(\mathbf{x}, \mathbf{x}') = \exp(-\gamma \|\mathbf{x} - \mathbf{x}'\|^2), \quad \gamma > 0.$$

Parametrul  $\gamma$  controleaza „raza de influenta” a fiecarui punct de antrenare. Nucleul RBF corespunde unui spatiu de trasaturi de dimensiune infinita si poate genera granite de decizie arbitrar de complexe, in functie de valorile  $C$  si  $\gamma$ .

$$\text{Kernel Trick RBF: } K(z_i, z_j) = \exp(-\gamma \|z_i - z_j\|^2)$$

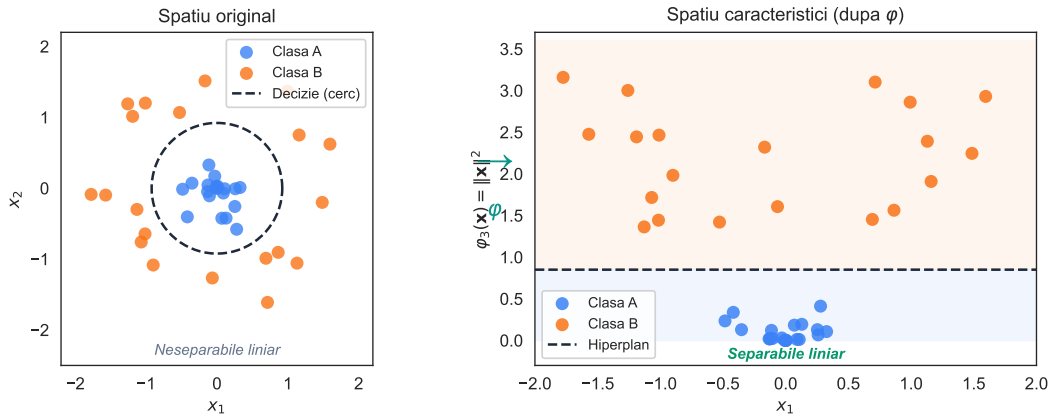


Figura 4: Efectul functiei nucleu: separare neliniara prin transformarea implicita  $\varphi$

**Teorema lui Mercer** garanteaza ca o functie simetrica  $K$  corespunde unui produs scalar valid intr-un spatiu Hilbert daca si numai daca matricea Gram  $\left[ K(\mathbf{x}_i, \mathbf{x}_j) \right]_{ij}$  este pozitiv semidefinita pentru orice alegere finita de puncte.

## 4.5 Extensia la mai multe clase

SVM este, prin constructie, un clasificator binar. Pentru  $m > 2$  clase se foloseste strategia *One-vs-One*: se antreneaza cate un clasificator binar pentru fiecare pereche de clase (in total  $\binom{m}{2}$  clasificatori), iar la predictie se aplica votul majoritar.

Pentru a obtine probabilitati de clasificare (utile in aplicatia web), se poate calibra scorul SVM prin metoda lui Platt [3]: o regresie logistica suplimentara transforma scorurile brute in probabilitati  $P(y = c | \mathbf{x})$ .

## 5 Aplicație: Recunoaștere Facială cu PCA + SVM

### 5.1 Arhitectura sistemului

Sistemul implementat parcurge următorii pași, de la imaginea brută la identificarea persoanei:



Figura 5: Arhitectura completă a sistemului de recunoaștere facială

1. **Detectie si decupare:** se identifica fata in imagine folosind un clasificator Haar cascade (OpenCV).
2. **Preprocesare:** imaginea este convertita in tonuri de gri, redimensionata la  $50 \times 37$  pixeli, egalizata cu CLAHE si normalizata in intervalul  $[0, 1]$ .
3. **Reducerea dimensionalitatii:** vectorul de 1850 de pixeli este proiectat in spatiul PCA cu 150 de componente (cu normalizare whitening).
4. **Clasificare:** coordonatele PCA sunt clasificate de un SVM cu nucleu RBF, care returneaza identitatea precisa si probabilitatile asociate fiecarei clase.

### 5.2 Datasetul LFW

*Labeled Faces in the Wild* [2] este un set de date public care contine fotografii ale unor persoane publice, realizate in conditii necontrolate (unghiuri, iluminari si expresii faciale diferite). Am folosit varianta filtrata cu minimum 70 de imagini per persoana, obtinand 1217 de imagini repartizate intre 6 persoane, la rezolutia  $50 \times 37$  pixeli. Impartirea in antrenare/test s-a facut stratificat: 973 de imagini pentru antrenare si 244 pentru test.

Persoana	Numar de imagini
George W. Bush	530
Colin Powell	236
Tony Blair	144
Donald Rumsfeld	121
Gerhard Schroeder	109
Ariel Sharon	77
<b>Total</b>	<b>1217</b>

Tabela 1: Distributia claselor in datasetul LFW filtrat

Se observa un dezechilibru pronuntat: George W. Bush are de aproape 7 ori mai multe imagini decat Ariel Sharon. Pentru a compensa, SVM este configurat cu ponderi de clasa proportionale cu inversul frecventei (`class_weight='balanced'`).

### 5.3 Preprocesare

1. **Tonuri de gri:** imaginile originale sunt convertite in format grayscale.
2. **CLAHE** (*Contrast Limited Adaptive Histogram Equalization*): o egalizare adaptiva locala a histogramei, care reduce variatiile de iluminare fara a amplifica zgomotul.
3. **Normalizare:** valorile pixelilor sunt scalate din  $[0, 255]$  in  $[0, 1]$ .
4. **Aplatizare:** matricea de  $50 \times 37$  este transformata in vector de  $d = 1850$  componente.

### 5.4 Configuratia modelului

**PCA:**  $k = 150$  componente principale, cu whitening activat.

**SVM:** nucleu RBF, ponderi de clasa echilibrate, probabilitati activate. Hiperparametrii  $C$  si  $\gamma$  au fost optimizati prin cautare pe grila cu validare incrucisata pe 5 parti, explorand  $C \in \{10^2, 10^3, 5 \cdot 10^3\}$  si  $\gamma \in \{10^{-4}, 5 \cdot 10^{-4}, 10^{-3}, 5 \cdot 10^{-3}\}$ . Valorile optime obtinute:  $C = 100$ ,  $\gamma = 0,005$  (scor CV: 85,6%).

**Pipeline:** PCA si SVM sunt inlantuite intr-un obiect Pipeline din `scikit-learn`, ceea ce garanteaza ca aceleasi transformari sunt aplicate atat la antrenare, cat si la inferenta.

### 5.5 Rezultate experimentale

Impartirea datelor s-a facut stratificat: 973 de imagini pentru antrenare si 244 pentru test.

	Precizie	Recall	F1	Suport
Ariel Sharon	1,00	0,62	0,77	16
Colin Powell	0,81	0,91	0,86	47
Donald Rumsfeld	0,90	0,79	0,84	24
George W. Bush	0,88	0,97	0,92	106
Gerhard Schroeder	0,94	0,68	0,79	22
Tony Blair	0,85	0,79	0,82	29
<b>Acuratete globala</b>		<b>87,3%</b>		244
Media macro	0,90	0,80	0,83	244
Media ponderata	0,88	0,87	0,87	244

Tabela 2: Rezultatele clasificarii pe setul de test ( $C = 100$ ,  $\gamma = 0,005$ ,  $k = 150$  componente PCA)

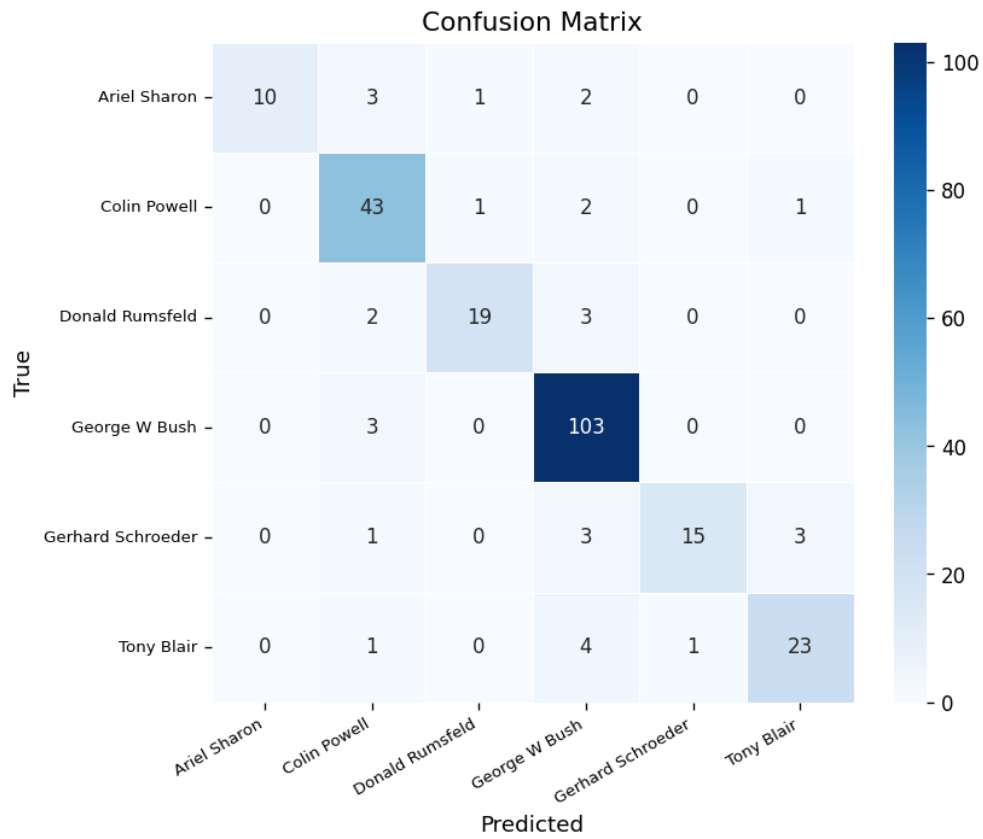


Figura 6: Matricea de confuzie pe setul de test

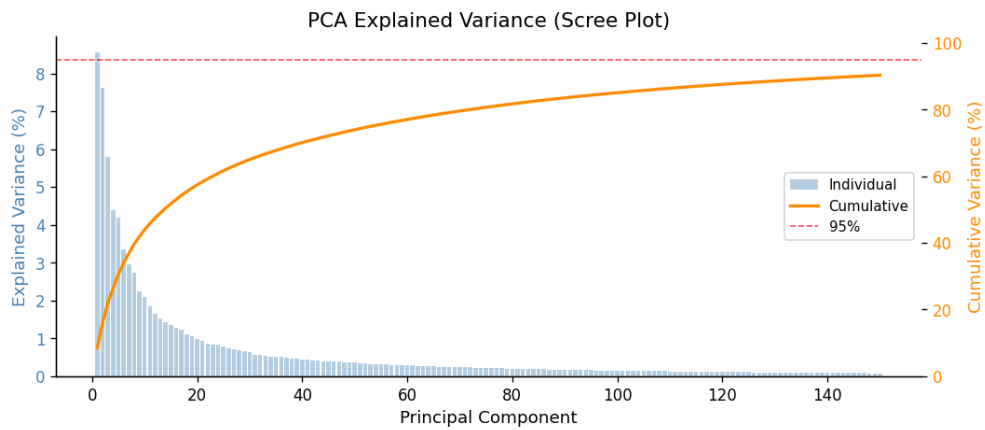


Figura 7: Varianta explicata cumulativa in functie de numarul de componente PCA

### Top Eigenfaces (Principal Components)

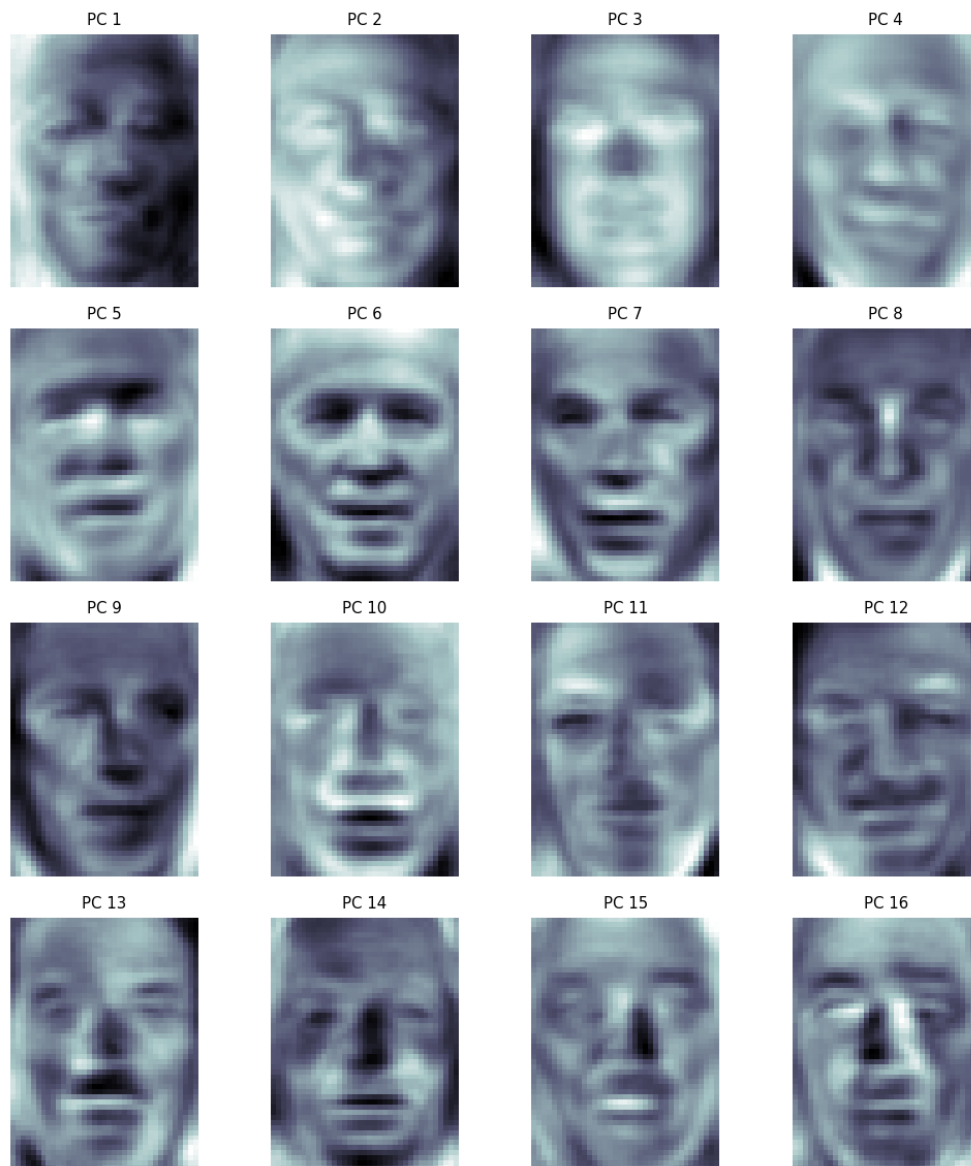


Figura 8: Primele 16 eigenfaces ale setului de date LFW

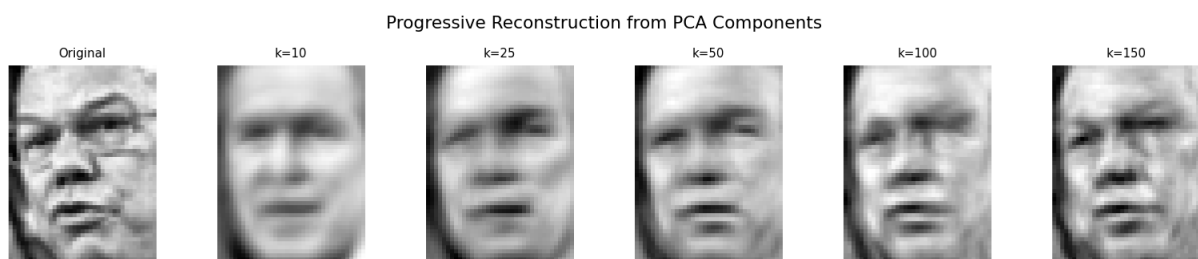


Figura 9: Reconstructia progresiva a unei fete din  $k$  componente PCA

## 5.6 Analiza rezultatelor

**Acuratete globala.** Modelul atinge 87,3% acuratete pe setul de test, cu un scor F1 ponderat de 0,87. Performanta variaza intre clase: George W. Bush, avand cel mai mare numar de exemple de antrenare, obtine  $F1 = 0,92$ , in timp ce Ariel Sharon (doar 16 exemple de test) prezinta un recall mai scazut (0,62), probabil din cauza variabilitatii ridicate a pozelor.

**Numarul de componente PCA.** Cu putine componente ( $k < 20$ ), clasificatorul nu dispune de suficienta informatie pentru a distinge intre persoane. Pe masura ce  $k$  creste, acuratetea se imbunatateste pana la un platou in jurul a  $k = 100-150$ , dupa care componentele suplimentare aduc in principal zgomot.

**Rolul normalizarii whitening.** Fara whitening, performanta SVM cu nucleu RBF scade semnificativ. Primele componente PCA au varianta cu ordine de marime mai mare decat ultimele; nucleul RBF, care depinde de distanta euclidiana, este dominat de aceste componente si ignora informatia din restul. Whitening-ul egaleaza contributia fiecărei componente, oferind clasificatorului o perspectiva mai balansata.

**Comparatie cu metoda clasica Eigenfaces.** Abordarea originala [4] clasifica prin distanta euclidiana in spatiul PCA, ceea ce presupune implicit ca fiecare clasa ocupa o regiune aproximativ sferica. SVM cu nucleu RBF nu face aceasta presupunere si invata granite de decizie mai complexe, obtinand acurateti semnificativ mai bune pe acelasi spatiu de proiectie.

## 6 Concluzii

Lucrarea a demonstrat cum tehnicile fundamentale din algebra liniara (descompunerea spectrala, SVD, diagonalizarea matricelor simetrice) stau la baza unor algoritmi practice de recunoastere faciala.

**PCA** realizeaza o reducere eficienta a dimensionalitatii: din 1850 de pixeli, 150 de componente principale retin esenta informatiei vizuale necesare identificarii unei persoane. Aceasta reducere nu este arbitrara; este optima in sensul variantei explicate si decurge direct din descompunerea spectrala a matricei de covarianta.

**SVM** adauga un nivel de clasificare riguros matematic: gasirea hiperplanului cu marginea maxima printr-o problema de optimizare convexa, rezolvata prin dualitate lagrangiana. Nucleul RBF extinde separarea la cazuri nelineare fara a calcula explicit transformarea in spatiul de dimensiune inalta.

Combinatia PCA + SVM depaseste abordarea clasica *Eigenfaces* (distanța euclidiană) atat din punct de vedere teoretic (SVM nu presupune clase sferice in spatiul PCA), cat si practic: modelul antrenat pe datasetul LFW filtrat a obtinut **87,3% acuratete** pe setul de test, cu un scor F1 ponderat de 0,87.

Codul sursa complet si aplicatia web demonstrativa sunt disponibile in directorul `proiect/` atasat acestei lucrari.

## Bibliografie

- [1] Richard E. Bellman. *Adaptive Control Processes: A Guided Tour*. Princeton University Press, 1961.
- [2] Gary B. Huang, Manu Ramesh, Tamara Berg, and Erik Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. In *Workshop on Faces in 'Real-Life' Images: Detection, Alignment, and Recognition*, 2007.
- [3] John Platt. Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods. In *Advances in Large Margin Classifiers*, pages 61–74. MIT Press, 1999.
- [4] Matthew Turk and Alex Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.