Principle Component Analysis

Ziv Yaniv

School of Engineering and Computer Science
The Hebrew University, Jerusalem, Israel.

The subject of this lecture is principle component analysis (PCA). Originally developed as a tool for the reduction of data dimensionality (Hotelling 1933) PCA has found many other uses than its original intended one, several of which will be discussed in this lecture. Before starting this lecture it should be noted that PCA appears in the literature under several names. The ones I am aware of are:

1. Discrete Karhunen-Loéve Transform (KLT).
2. Principle Axis Transform (PAX/PAT).
3. Hotelling Transform.
4. Eigenvector Transform.
5. Principle Component Analysis (PCA).

This summary is divided into two sections: (1) The mathematics underlying PCA. (2) Applications of PCA.

1 Mathematical Background

PCA was originally developed as a multivariate analysis tool. Multivariate analysis deals with the study of vectors of random variables, for example n random vectors $y_1, y_2, \ldots, y_n$, each of dimension d. Typically these n vectors arise from taking measurements on d variables or characteristics for each of the n observations.

To analyze the relationships between these random variables we must generalize the concepts of expectation and covariance from uni-variate statistics (see appendix) to the multi-variate setting.

Let A be a matrix of random variables the expectation operator is defined by:

$$\mathcal{E} \equiv [(E[A_{ij}])]$$

Going on to the covariance: Let x and y be two vectors of random variables, not necessarily of the same dimensions then the covariance operator is

$$\begin{aligned} \mathcal{C}[x,y] &= [(cov(x_i,y_j])] \\ &= \mathcal{E}[(x-\mu_x)(y-\mu_y)^T] \\ &= \mathcal{E}[xy^T] - \mu_x\mu_y^T \end{aligned}$$

If x and y are statistically independent then $\mathcal{C}[x,y] = O$.
When $x = y$ then $\Sigma = \mathcal{C}[x,x] = var[x]$ is called the covariance matrix of x. In statistics texts this matrix is also called the variance-covariance or dispersion matrix.

We are now ready to discuss the original idea behind PCA, dimension reduction. As previously stated multivariate analysis studies vectors of random variables. The dimension, d, of these vectors is often very large. Unfortunately when $d \gg$ studying the interrelationships between the random variables is very hard. What we would like to do is reduce the dimension of our data set by looking at a subset ($p \ll$) of derived variables while retaining as much as possible of the *variation* found in the original data set. This subset of variables should be a linear combination of the original ones.

The first step towards deriving this set of variables is to look for a linear function $\alpha_1^T y$ which has maximum variance where $var[\alpha_1^T y] = \alpha_1^T \Sigma \alpha_1$, α_1 is the first principle component. Next look for a linear function $\alpha_2^T y$, uncorrelated with $\alpha_1^T y$ which has maximum variance, and so on. There are up to d principle components, hopefully the first $p \ll d$ will account for most of the variance in the original data.

To derive the form of the principle components we start by looking at

$$\max(var[\alpha_1^T y]) = \max(\alpha_1^T \Sigma \alpha_1)$$

It is clear that the maximum is not attained with a finite α_1 so we will impose a normalization. We now formulate the problem as:

$$\max_{\alpha_1} \left\{ \frac{\alpha_1^T \Sigma \alpha_1}{\alpha_1^T \alpha_1} \right\}$$

The maximum is attained when α_1 is the eigenvector of Σ which corresponds to the largest eigenvalue (Theorem A.3(1)).

We now derive the second principle component which is the solution to the following problem:

$$\max_{\alpha_1 \alpha_2 = 0} \left\{ \frac{\alpha_2^T \Sigma \alpha_2}{\alpha_2^T \alpha_2} \right\}$$

The maximum is now attained when α_2 is the eigenvector of Σ which corresponds to the second largest eigenvalue (Theorem A.3(2)).

We continue the above construction until we obtain the required number of principle components. Essentially we are changing the original basis, which the data is represented in, to a basis in which the data is uncorrelated. The original covariance matrix was of the form YY^T with $SVD(YY^T) = X_d S^2 X_d^T$ and X_d the eigenvectors of the data matrix Y (see appendix A.1). The covariance matrix of the data represented in the new basis X_d is diagonal, meaning the data is uncorrelated:

$$(X_d^T Y)(X_d^T Y)^T = X_d^T Y Y^T X_d = X_d^T X_d S^2 X_d^T X_d = S^2$$

As our goal is data reduction we will only use $p \ll d$ eigenvectors, thus projecting the original d dimensional data onto a p dimensional subspace while retaining most of the variation. This is done by subtracting the mean from the original data and then projection using the matrix:

$$X_p = \begin{bmatrix} -x_1- \\ \vdots \\ -x_p- \end{bmatrix}$$

where $x_1, \cdots, x_p$ are eigenvectors of the covariance matrix Σ corresponding to the eigenvalues $\lambda_1 \geq \cdots \geq \lambda_p$.

The previous construction was done using the knowledge about the dimensionality of the subspace p. We now consider the choice of p. This subspace was chosen so that it retains as much of the variability found in the original data, but what is this quantity?

We compute p from the percentage of variability we want the projected data to account for:

$$\frac{\sum_{i=1}^{p} \lambda_i}{\sum_{i=1}^{n} \lambda_i} > t$$

where $\lambda_1 \geq \cdots \geq \lambda_n$ are the eigenvalues of Σ and $t \in [0, 1]$ is the percentage of variability accounted for in the p dimensional subspace.

1.1 Computing the Eigenvalues/Eigenvectors

Given $y_1, \cdots, y_n$, d dimensional observations with mean μ we have:

$$\Sigma = \frac{\sum_{i=1}^{n} (y_i - \mu)(y_i - \mu)^T}{n}$$

We define the following matrix:

$$Y = \begin{bmatrix} (y_1 - \mu) & \cdots & (y_n - \mu) \end{bmatrix}$$

which gives us the following representation of the covariance matrix:

$$\Sigma = YY^T$$

We now describe four approaches to computing the eigenvalues/eigenvectors of the covariance matrix.

The first approach is to use standard numerical techniques to compute the eigenvalues and eigenvectors of Σ.

Unfortunately if $d >>$ then Σ is a very large matrix $d \times d$ (this is the usual situation). This brings us to the second approach, instead of working with YY^T we can look at Y^TY which is an $n \times n$ matrix. We now look at the eigenvectors of this new matrix:

$$Y^TYv_i = a_iv_i$$

Premultiplying both sides by Y will give us the eigenvectors of YY^T which we seek:

$$(YY^T)(Yv_i) = a_i(Yv_i)$$

The eigenvectors of YY^T are just the eigenvectors of Y^TY premultiplied by Y.

The third approach utilizes the Singular Value Decomposition (SVD). Given the matrix $Y = U\Sigma V^T$ we have:

$$YY^T = (U\Sigma V^T)(U\Sigma V^T)^T = U\Sigma V^TV\Sigma U^T = U\Sigma^2U^T$$

This is exactly an eigenvector decomposition of the matrix YY^T, the columns of U are the eigenvectors of YY^T and the entries of Σ^2 the corresponding eigenvalues.

The fourth approach is rarely used but it is the best choice in the following case: If the matrix Σ is of size $d \times d$ where $d \leq 4$ there is a closed form solution to the characteristic equation which gives us the eigenvalues. Once we have the eigenvalues we solve the linear equations $(YY^T - \lambda_iI)x = 0$ to yield the eigenvectors (see appendix B for the 2×2 case).

1.2 Robustness

Computing the eigenvectors from the covariance matrix as described in the previous section will result in undesirable results in the presence of outliers. The covariance matrix is affected by the presence of outliers, so if your input contains outliers apply robust estimation of the covariance matrix prior to PCA.

2 Applications

The applications described in this section use PCA either for data reduction or for finding direction of variation.

2.1 Eigen Images

Object recognition has been one of the fundamental goals of computer vision. One approach towards this task has been to model an object by its images. When a novel

image is introduced it is compared to all existing images and its nearest neighbor is chosen as the best candidate object.

More formally: Given a database of images $\{I_i\}$ of k different objects and a novel image J we identify it with object j if the image corresponding to $min(\|\{I_i\} - J\|^2)$ belongs to object j (Sum of Squared Distances, SSD, distance).

Representing an object using only images requires large amounts of high dimensional data (images). If we view an image as a vector of random variables we can apply PCA to our data set and store it in a compact representation. When the novel image is presented we project it onto the eigen-image space and search for the nearest match there.

Again formally: An image in the data base can be represented by its eigenvectors and the mean image:

$$I_m = \sum_{i=1}^{N} w_i e_i + \mu$$

where w_i are the projections of the images onto the eigenvector space and μ is the mean image. An approximate of the image is given using only the first K eigenvectors:

$$I_m \simeq \sum_{i=1}^{K} w_i e_i + \mu$$

Now the SSD distance is computed as follows:

$$\|I_m - I_n\|^2 \simeq \|\sum_{i=1}^{K} w_{m,i} e_i - \sum_{i=1}^{K} w_{n,i} e_i\|^2 = \|\sum_{i=1}^{K} (w_{m,i} - w_{n,i}) e_i\|^2 = \|w_m - w_n\|^2$$

where the last simplification is due to the orthonormality of the eigenvectors.

This approach was applied by [11, 6] for face recognition and by [4] for general object recognition. It is quit surprising that this scheme is applicable to general object recognition. Figure 1 demonstrates data reconstruction using different sized eigen-spaces.

(a)

(b) (c) (d)

Figure 1: (a) Image database (b) Original image (c) Reconstruction using four eigenvectors, accounting for 0.5 of variation. (d) Reconstruction using ten eigenvectors, accounting for 0.95 of variation. Data was taken from columbia university [5].

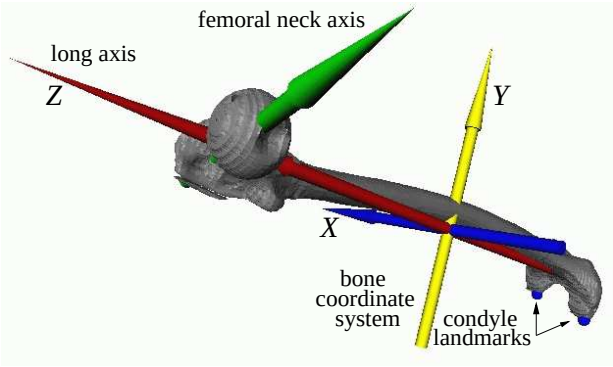

Figure 2: Bone axis were computed from the covariance matrix of the model vertex points.

2.2 Data Axis

There are applications where all we seek is the direction of maximal variance. This direction is given by the eigenvector of the covariance matrix which corresponds to the largest eigenvalue. An example of this use in an application is finding the main axis of bones in an orthopedic applications Figure 2 [7]. The axis with maximal variance is also used for data partitioning in the spatial decomposition data structures described in [9] and in [3].

2.3 Orthogonal Least Squares

Given d dimensional points we want to fit them to a hyperplane such that their orthogonal distance from the plane is minimized.

A hyperplane is defined by the equation:

$$n \cdot (p - a) = 0$$

where a is a point on the hyperplane and n is the hyperplane's normal.

Given a point p_i it can be written as follows (see Figure 3):

$$p_i = a + (n \cdot (p_i - a))n + s_i n_i^{\perp}$$

where $n_i^{\perp}$ is a unit vector perpendicular to n and s_i is the appropriate scale factor.

The sum of squared distances we want to minimize is given by:

$$\begin{aligned} \Delta &= \Sigma_{i=1}^{m}[(p_i - a) \cdot n]^2 \\ &= \Sigma_{i=1}^{m}((p_i - a)^T n)(n^T(p_i - a)) \\ &= n^T[\Sigma_{i=1}^{m}(p_i - a)(p_i - a)^T]n \end{aligned}$$

Deriving Δ with respect to a we get:

$$\frac{\partial \Delta}{\partial a} = -2[n^T n]\Sigma_{i=1}^{m}(p_i - a)$$

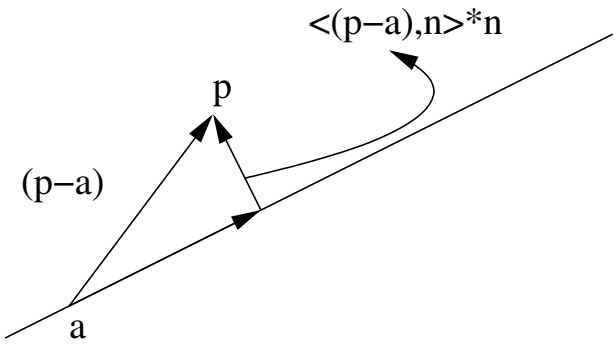

Figure 3: Orthogonal least squares, point from plane distance.

equating this to zero yields:

$$\Sigma_{i=1}^{m}(p_i - a) = 0$$
$$\Downarrow$$
$$a = \frac{1}{m}\Sigma_{i=1}^{m} p_i$$

a is simply the average of the sample points.

Looking again at our optimization problem we have:

$$\min n^T[\Sigma_{i=1}^{m}(p_i - a)(p_i - a)^T]n$$
$$\Downarrow$$
$$\min n^T \Sigma n$$

where Σ is simply the data covariance matrix. The solution to this minimization problem is when n is the eigenvector corresponding to the smallest eigenvalue.

A Symmetric Matrices

Theorem A.1 *The eigenvalues of a real symmetric matrix are real.*

Proof Given the symmetric real matrix A we have:

$$Ax = \lambda x \tag{1}$$

Where λ is an eigenvalue and x the corresponding eigenvector. Until we prove the theorem we must assume that λ might be a complex number ($\lambda = a + ib$) and x might contain components which are complex too. Remembering that $\overline{\lambda x} = \overline{\lambda}\overline{x}$ and that $A = \overline{A} = A^T$ we take the conjugates of equation 1:
$\overline{Ax} = \overline{\lambda x}$ leads to $A\overline{x} = \overline{\lambda}\overline{x}$. Transpose this to get $\overline{x}^T A = \overline{x}^T\overline{\lambda}$ Now taking the dot product of the first equation with $\overline{x}$ and the last equation with x we get:
$\overline{x}^T Ax = \overline{x}^T \lambda x$ and $\overline{x}^T Ax = \overline{x}^T\overline{\lambda}x$
Which gives us: $\overline{x}^T \lambda x = \overline{x}^T\overline{\lambda}x$
Therefore $\lambda = \overline{\lambda}$ proving that λ is real ($a + bi = a - bi$ so the complex coefficient is equal to zero).

Theorem A.2 *The eigenvectors of a real symmetric matrix which correspond to different* [1] *eigenvalues are perpendicular.*

Proof Let λ_1 and λ_2 be two different eigenvalues and x_1 and x_2 the corresponding eigenvectors. This gives us the following two equations:

$$Ax_1 = \lambda_1 x_1$$
$$Ax_2 = \lambda_2 x_2$$

Taking the dot product with x_2 we get:

$$(\lambda_1 x_1)^T x_2 = (Ax_1)^T x_2 = x_1^T A^T x_2 = x_1^T A x_2 = x_1^T \lambda_2 x_2$$

The left side is $x_1^T \lambda_1 x_2$ and the right side is $x_1^T \lambda_2 x_2$. Since $\lambda_1 \neq \lambda_2$ this proves that $x_1^T x_2 = 0$. The eigenvectors are perpendicular.

Theorem A.3 *Let A be a real nxn symmetric matrix with eigenvalues $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_n$ and corresponding orthonormal eigenvectors $x_1, x_2, \ldots, x_n$, as proven in A.2. Define $X_k = (x_1, x_2, \ldots, x_k)$ $(k = 1, 2, \ldots, n-1)$ and $X = (x_1, x_2, \ldots, x_n)$. Then if we assume that $\alpha \neq 0$, we have the following:*

[1] Proof that *all eigenvectors of a real symmetric matrix are orthogonal to each other can be found in [1]*

1.

$$\sup_{\alpha}\left\{\frac{\alpha^T A\alpha}{\alpha^T\alpha}\right\} = \lambda_1$$

and the supremum is attained if $\alpha = x_1$.

2.

$$\sup_{X_k^T\alpha=0}\left\{\frac{\alpha^T A\alpha}{\alpha^T\alpha}\right\} = \lambda_{k+1}$$

and the supreumum is attained if $\alpha = x_{k+1}$.

3.

$$\inf_{\alpha}\left\{\frac{\alpha^T A\alpha}{\alpha^T\alpha}\right\} = \lambda_n$$

and the infimum is attained if $\alpha = x_n$.

4. If $X_{n-k} = (x_{n-k+1}, x_{n-k+2}, \ldots, x_n)$ *then*

$$\inf_{X_{n-k}^T\alpha=0}\left\{\frac{\alpha^T A\alpha}{\alpha^T\alpha}\right\} = \lambda_{n-k}$$

and the infimum is attained if $\alpha = x_{n-k}$.

Proof 1. Let $\alpha = Xy = y_1x_1 + y_2x_2 + \cdots + y_nx_n$ and $\Lambda = \begin{bmatrix} \lambda_1 & & \\ & \ddots & \\ & & \lambda_n \end{bmatrix}$.
Then

$$\frac{\alpha^T A\alpha}{\alpha^T\alpha} = \frac{y^T X^T AXy}{y^T y} = \frac{y^T X^T X\Lambda y}{y^T y} = \frac{(\sum_i \lambda_i y_i^2)}{y^T y} \leq \frac{\lambda_1 y^T y}{y^T y} = \lambda_1$$

with equality when $y_1 = 1, y_2 = y_3 = \cdots = y_n = 0$, that is, when $\alpha = x_1$.

2. If $\alpha \perp x_1, x_2, \ldots, x_k$, then $y_1 = y_2 = \cdots = y_k = 0$. The result then follows with the same argument as 1.

3. Same proof as 1 but with the inequality reversed.

4. Same proof as 2 but with inequality reversed.

A.1 Matrices of the form AA^T

Theorem A.4 *Matrices of the form AA^T, where A is non-singular, have the following properties:*

1. *They are positive definite.*
2. *They have positive eigenvalues.*
3. *The Singular Value Decomposition (SVD) of A yields the eigenvalues and eigenvectors of AA^T.*

Proof 1. A matrix B is positive definite if:

$$\forall x,\ x \neq 0 \quad x^T B x > 0$$

Given a matrix B of the form AA^T we have:

$$x^T B x = x^T A A^T x = (A^T x)^T (A^T x) = \|A^T x\|^2 > 0$$

2. Given a matrix B of the form AA^T with eigenvalue λ and corresponding eigenvector x we have:

$$Bx = \lambda x$$

Premultiplying by x^T we get:

$$x^T B x = \lambda x^T x = \lambda \|x\|^2$$

As B is positive definite we have:

$$\begin{aligned} \lambda \|x\|^2 &> 0 \\ \Downarrow & \\ \lambda &> 0 \end{aligned}$$

3. The Singular Value Decomposition of the matrix A is given by

$$A_{m\times n} = U_{m\times n} S_{n\times n} V^T_{n\times n}$$

where the columns of U are an orthonormal basis for the column space of A, the rows of V are an orthonormal basis for the row space of A and S is a diagonal matrix. We now look at the matrix AA^T:

$$AA^T = (USV^T)(VSU^T) = US^2U^T$$

Postmultipying this equation by U yields the following equation:

$$(AA^T)U = US^2$$

The eigenvectors of AA^T are the columns of U with corrosponding eigenvalues in S.

B Eigenvalues/Eigenvectors of a 2×2 Symmetric Matrix

Given a symmetric matrix:

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{12} & a_{22} \end{bmatrix}$$

We obtain the eigenvalues of A by solving the characteristic equation:

$$det(A - \lambda I) = 0$$

For the matrix A this is a quadratic equation:

$$(a_{11} - \lambda)(a_{22} - \lambda) - a_{12}^2 = \lambda^2 - (a_{11} + a_{22})\lambda + (a_{11}a_{22} - a_{12}^2)$$

whose solution yields the eigenvalues:

$$\lambda_1 = \frac{a_{11} + a_{22} + \sqrt{(a_{11} - a_{22})^2 + 4a_{12}^2}}{2} \qquad \lambda_2 = \frac{a_{11} + a_{22} - \sqrt{(a_{11} - a_{22})^2 + 4a_{12}^2}}{2}$$

and corresponding eigenvectors:

$$v_1 = [\lambda_1 - a_{22}, a12] \qquad v_2 = [-a12, \lambda_1 - a_{22}]$$

Note that when $a_{12} \rightarrow 0$ the eigenvectors are:

$$\begin{array}{cc} a_{11} \rightarrow 0 & a_{22} \rightarrow 0 \\ v_1 = [0, 1] \; v_2 = [1, 0] & v_1 = [1, 0] \; v_2 = [0, 1] \end{array}$$

C Statistical Review

This appendix gives a short set of definitions and equations from uni-variate statistics. This will, hopefully, help the reader in the understanding of the construction of the multi-variate sample co-variance matrix and the subject of Principle Component Analysis.

C.1 Expectation,Variance/Covariance

Definitions:
X, Y - discrete random variables. a - a constant. $f(X)$ - a function of X.

Given the above definitions the following definitions and equations hold:

$$
\begin{aligned}
E(X) &= \sum_x xP(x) \\
E(f(X)) &= \sum_x f(x)P(f(x)) \\
&= \sum_x f(x)P(x) \\
E(a) &= \sum_x aP(x) \\
&= a\sum_x P(x) \\
&= a \\
E(aX) &= \sum_x axP(x) \\
&= a\sum_x xP(x) \\
&= aE(X) \\
E(X+Y) &= \sum_x \sum_y (x+y)P(x,y) \\
&= \sum_x \sum_y xP(x,y) + \sum_x \sum_y yP(x,y) \\
&= \sum_x xP(x) + \sum_y yP(y) \\
&= E(X) + E(Y)
\end{aligned}
$$

$$
\begin{aligned}
var(X) &= E(X-\mu)^2 \\
&= E(X^2 - 2\mu X + \mu^2) \\
&= E(X^2) - 2\mu E(X) + \mu^2 \\
&= E(X^2) - 2E(X)^2 + E(X)^2 \\
&= E(X^2) - E(X)^2 \\
var(aX) &= E[(aX)^2] - [E(aX)]^2 \\
&= a^2E(X)^2 - a^2[E(X)]^2 \\
&= a^2 var(X) \\
var(X+a) &= E[(X+a) - E(X+a)]^2 \\
&= E[X - E(X)]^2 \\
&= var(X) \\
var(X+Y) &= E[(X+Y)^2] - [E(X+Y)]^2 \\
&= E(X^2) + 2E(XY) + E(Y^2) - [E(X)^2 + 2E(X)E(Y) + E(Y)^2] \\
&= var(X) + var(Y) + 2cov(XY) \\
cov(XY) &= E[(X-\mu_x)(Y-\mu_y)] \\
&= E(XY) - E(X)E(Y)
\end{aligned}
$$

C.2 Point Estimation

Point estimation consists of using a single sample statistic (estimator) to infer a single value which is used as the estimate of a population quantity.

The following are criteria on which the goodness of a point estimator is judged:

1. Bias - An estimator G for θ is said to be unbiased if $E(G) = \theta$.
2. Consistency - An estimator G for θ is said to be consistent if
$$P(|G - \theta| < \epsilon) \underset{N\to\infty}{\longrightarrow} 1$$
The larger the sample size N the smaller the error in estimation.
3. Relative efficiency - Given two estimators G and H for θ where both are unbiased we define:
$$Relative\ efficiancy\ = \frac{\sigma_H^2}{\sigma_G^2}$$
The more efficient estimator has a smaller deviation.

When estimating the variance our natural tendency would be to use the following statistic:

$$\sum_i \frac{(x_i - M)^2}{N} = \sum_i \frac{x_i^2}{N} - M^2 \tag{2}$$

where M is the sample mean and N the number of samples.
The right side of the equation is obtained with a bit of algebra and the fact that M is constant in all summations.

Unfortunately Equation 2 is a biased estimator for the variance. An unbiased estimator is given by:

$$\sum_i \frac{(x_i - M)^2}{N - 1}$$

References

[1] Golub G.H., Van Loan C.F., *Matrix Computations, third edition*, Jhons Hopkins University Press,1996.

[2] Jolliffe I.T., *Principle Component Analysis*, Springer-Verlag, 1986.

[3] McNames J., "A Fast Nearest-Neighbor Algorithm Based on a Principal Axis Search Tree", IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI), Vol. 23(9), pp. 964–976, 2001.

[4] Nayar S.K., Murase H., Nene S.A., "Parametric Appearance Representation", Early Visual Learning, Oxford University Press, 1996.

[5] Nene S.A., Nayar S.K., Murase H., "Columbia Object Image Library (COIL-20)", Technical Report CUCS-005-96, 1996.
http://www.cs.columbia.edu/CAVE/research/softlib/coil-20.html

[6] Pentland A., Moghaddam B., Starner T., "View-Based and Modular Eigenspaces for Face Recognition", IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 1994.

[7] Ron O., Joskowicz L., Simkin A., Milgrom C., "Computer-Based Periaxial Rotation Measurement for Aligning Fractured Femur Fragments", Computer Aided Radiology and Surgery (CARS), 2001.

[8] Seber G.A.F., *Multivariate Observations*, Wiley, 1984.

[9] Sproull R.L., "Refinements to nearest-neighbor searching", Algorithmica, Vol.6, pp. 579–589, 1991.

[10] Strang G., *Introduction To Linear Algebra*, Wellesley-Cambridge Press, 1993.

[11] Turk M., Pentland A., "Eigenfaces for recognition", Journal of Cognitive Neuroscience, Vol.3(1), 1991.