3.4 Calculating the eigenfaces numerically.

The basic idea for obtaining the eigenfaces could not be simpler: calculate the eigenvectors of the covariance matrix of the faces in the training set. The problem is a computational one. For an image of size $N \times N$ the covariance matrix is of size $N^2 \times n^2$. We therefore need to compute N^2 eigenvectors, of which only a very small number will have nonzero eigenvalues. There are different ways in which this computational problem can be solved, auch as using a singular value decomposition. Here we follow a direct approach proposed by Sirovich and Kirby [**??**].

The basic idea is to assume that each eigenvector is a linear combination of the faces in the training set. This is reasonable because we are looking for a basis of the space spanned by the faces in the training set. Accordingly we assume,

$$\boldsymbol{u}_l = \sum_{k=1}^{M} a_{lk} \boldsymbol{X}_k \,. \tag{3.14}$$

Since the $\boldsymbol{u}_l$'s are eigenvectors of C, they also satisfy $\lambda_l \boldsymbol{u}_l = C\boldsymbol{u}_l$ and (3.14) becomes

$$\lambda_l \sum_{k=1}^{M} a_{lk} \mathbf{X}_k = C \sum_{k=1}^{M} a_{lk} \mathbf{X}_k .$$

Using the definition of the covariance matrix (3.5, it follows that

$$\begin{aligned} \lambda_l \sum_{k=1}^{M} a_{lk} \boldsymbol{X}_k &= \frac{1}{M} \sum_{n=1}^{M} \boldsymbol{X}_n \boldsymbol{X}_n^T \sum_{k=1}^{M} a_{lk} \boldsymbol{X}_k \\ &= \frac{1}{M} \sum_{k=1}^{M} a_{lk} \sum_{n=1}^{M} \boldsymbol{X}_n \boldsymbol{X}_n^T \mathbf{X}_k \,. \end{aligned} \tag{3.15}$$

Defining the scalar quantities

$$K_{nk=} \mathbf{X}_n^T \mathbf{X}_k$$

then (3.15)becomes

$$\lambda_l \sum_{k=1}^{M} a_{lk} \boldsymbol{X}_k = \frac{1}{M} \sum_{n=1}^{M} \boldsymbol{X}_n \sum_{k=1}^{M} a_{lk} K_{nk}$$

and equating the coefficients of the $\boldsymbol{X}_k$'s (assuming the $\boldsymbol{X}_k$'s are independent), we arrive at

$$\lambda_l a_{lk} = \frac{1}{M} \sum_{n=1}^{M} a_{ln} K_{kn} \quad \forall\, k \in [1 \ldots M] \,. \tag{3.16}$$

Finally, if we define $\boldsymbol{a}_l = \begin{bmatrix} a_{l1} & a_{l2} & \cdots & a_{lM} \end{bmatrix}^T$ and $\boldsymbol{K}_k = \begin{bmatrix} K_{1M} & \cdots & K_{kM} \end{bmatrix}^T$, then

$$\sum_{n=1}^{M} a_{ln} K_{nk} = \boldsymbol{K}_k^T \boldsymbol{a}_l$$

and (3.16) becomes

$$M \lambda_l \mathbf{a}_l = K \mathbf{a}_l \tag{3.17}$$

where

$$K = \begin{bmatrix} \boldsymbol{K}_1^T \\ \boldsymbol{K}_2^T \\ \vdots \\ \boldsymbol{K}_M^T \end{bmatrix} .$$

Thus $\boldsymbol{a}_l$ is an eigenvector of the $M \times M$ matrix K with eigenvalue $M\lambda_l$. Note that if the $\boldsymbol{X}_k$'s are not normalized, we can get extremely large values in K, which can lead to numerical instabilities while calculating the eigenvalues and eigenvectors.

From (3.17), it follows that the problem of finding the eigenvectors of C is reduced to one of finding the eigenvectors of K. Since K is $M \times M$ and M is usually much smaller than N, the magnitude of the problem has been reduced to manageable proportions.

Since this method can only produce at most M non-zero eigenvalues, we will only obtain a maximum of M basis vectors for our subspace. This is quite reasonable since there are only M images describing the subspace we are interested in. In any case, we have already seen that most of these carry no information about the face subspace and can be safely discarded.