5. Descrição dos Experimentos

Os capítulos 3 e 4 anteriores apresentaram e discutiram as técnicas de redução de dimensionalidade e os modelos de classificadores utilizados neste trabalho. O presente capítulo tem por objetivo descrever como essas técnicas foram aplicadas ao problema do reconhecimento de faces, apresentando detalhadamente tanto o banco de imagens utilizado no trabalho como os experimentos realizados.

5.1 Banco de Faces

A escolha do banco de imagens foi feita levando-se em consideração, basicamente, dois pontos. O primeiro se refere ao número de fotos de face por indivíduo. Este número deveria ser suficiente para poder retratar um ambiente relativamente próximo da realidade, onde as fotos de um mesmo indivíduo possuíssem, pois, variações relativas à iluminação, escala, translação, rotação e a outros detalhes inerentes às imagens de face em geral. O segundo ponto diz respeito à quantidade de trabalhos anteriores que utilizaram o mesmo banco de faces. O ideal seria que pudesse se escolher um banco de imagens que já tivesse sido utilizado em outros trabalhos desta área de reconhecimento, permitindo que os resultados obtidos nesta tese, principalmente aqueles relativos ao novo modelo de classificador proposto, pudessem ser comparados com trabalhos anteriores.

A partir destas considerações, verificou-se que o banco de imagens que melhor atendia essas especificações é o banco de faces conhecido como *ORL face database*. Este banco de faces foi desenvolvido pelo Laboratório de Pesquisas da Olivetti (*Olivetti Research Laboratory – ORL*) que se encontra em Cambridge, na Inglaterra. Por ser um banco de faces

de fácil acesso e obtenção - o mesmo está disponivel de forma gratuita[1] na Internet – pode-se encontrar na literatura de reconhecimento de faces alguns trabalhos que fizeram uso deste banco de imagens para teste e geração de resultados [Sama94, Lawr97].

O banco de faces ORL é um banco de imagens formado por um conjunto de fotos de face retiradas entre o período de abril de 1992 e abril de 1994 no próprio laboratório de pesquisas da Olivetti. Este banco de dados contém 10 fotos para cada um dos 40 indivíduos disponíveis, totalizando um número de 400 imagens. Em alguns casos essas fotos foram retiradas em momentos distintos, permitindo variações de iluminação, de expressões faciais como olhos abertos/fechados, sorrindo/não sorrindo, e variações nos detalhes da face como usando/não usando óculos. Além disso, todas as fotos foram retiradas contra um fundo homogêneo escuro e as faces das pessoas foram posicionadas frontalmente, possibilitando, no entanto, variações de rotação da ordem de 20 graus e variações de escala da ordem de 10 graus. Todas as imagens possuem resolução de 92x112 pixels, e cada pixel contém 256 tons de cinza. Por conveniência de implementação, neste trabalho reduziu-se a resolução original dessas imagens para 64x64 pixels, representando, por conseguinte, essas imagens como matrizes de 64 linhas e 64 colunas, totalizando um total de 4096 valores compreendidos entre 0 e 255. Um exemplo do conjunto de 10 imagens de um indivíduo pode ser visto na figura 5.1 a seguir, e todo o banco de faces pode ser observado no apêndice.

[1] O endereço para *download* é http:// www.cam-orl.co.uk/facedatabase.html

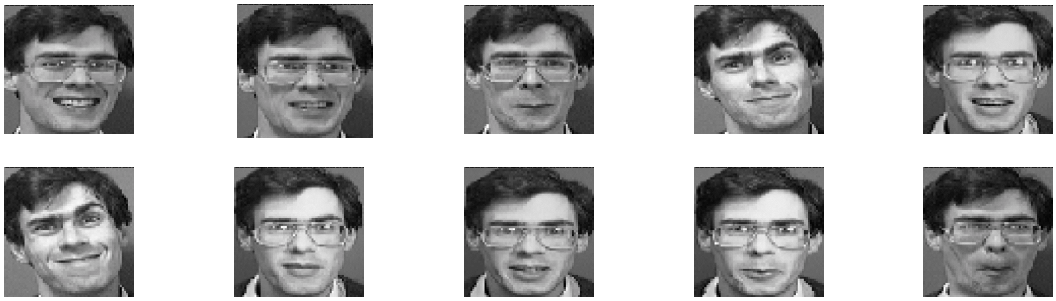

Figura 5.1 – O conjunto de 10 imagens de um indivíduo do banco de faces ORL.

5.2 Os Experimentos

Antes que as imagens do banco de faces ORL pudessem ser utilizadas nos experimentos, as mesmas foram transformadas em vetores multi-dimensionais. Esta transformação é realizada através da simples concatenação das 64 linhas de cada foto, representando cada imagem de face de 64x64 pixels em um vetor de 4096 elementos. Um exemplo dessa transformação é apresentado na figura 5.2. Em termos de classificação, tal transformação pressupõe que duas imagens parecidas de um mesmo indivíduo possam ser representadas por dois pontos próximos um do outro no espaço multi-dimensional, ou espaço imagem. Esta suposição é verdadeira para a aplicação em questão.

A partir desta transformação, cada experimento foi realizado em basicamente três passos: criação da matriz de transformação do PCA, treinamento dos classificadores e teste dos classificadores. Um diagrama desses passos pode ser visualizado na figura 5.3.

$$= \begin{bmatrix} 150 & 152 & \dots & 151 \\ 153 & 154 & \dots & 155 \\ \dots & \dots & \dots & \dots \\ 254 & 255 & \dots & 255 \end{bmatrix}_{64 \times 64 \text{ pixels}}$$

Imagem (matriz de pixels)

Concatenação das linhas da matriz de pixels, transformando a imagem de face em um vetor multi-dimensional

Vetor multi-dimensional

$$\begin{bmatrix} 150 & 152 & \dots & 151 & 153 & 154 & \dots & 155 & \dots & 254 & 255 & \dots & 255 \end{bmatrix}_{4096 \text{ elementos}}$$

Figura 5.2 – Representação das imagens em vetores multi-dimensionais.

5.2.1 Criação da matriz de transformação do PCA

No primeiro passo, a matriz de transformação do PCA é gerada. Esta matriz de transformação tem por finalidade reduzir a dimensionalidade das imagens de face, projetando-as em um novo espaço composto pelas componentes principais do experimento. Seleciona-se, para isto, um conjunto de faces de treinamento. Este conjunto de treinamento é constituído por 5 imagens de cada indivíduo, as quais são escolhidas aleatoriamente. As 5 imagens restantes de cada pessoa são utilizadas posteriormente, no terceiro passo, para teste. A razão de se escolher 5 imagens para treinamento e 5 imagens para teste se deve ao fato de se querer adotar um procedimento de geração de resultados bastante próximo dos demais trabalhos de reconhecimento de faces que utilizaram esse mesmo banco de faces ORL.

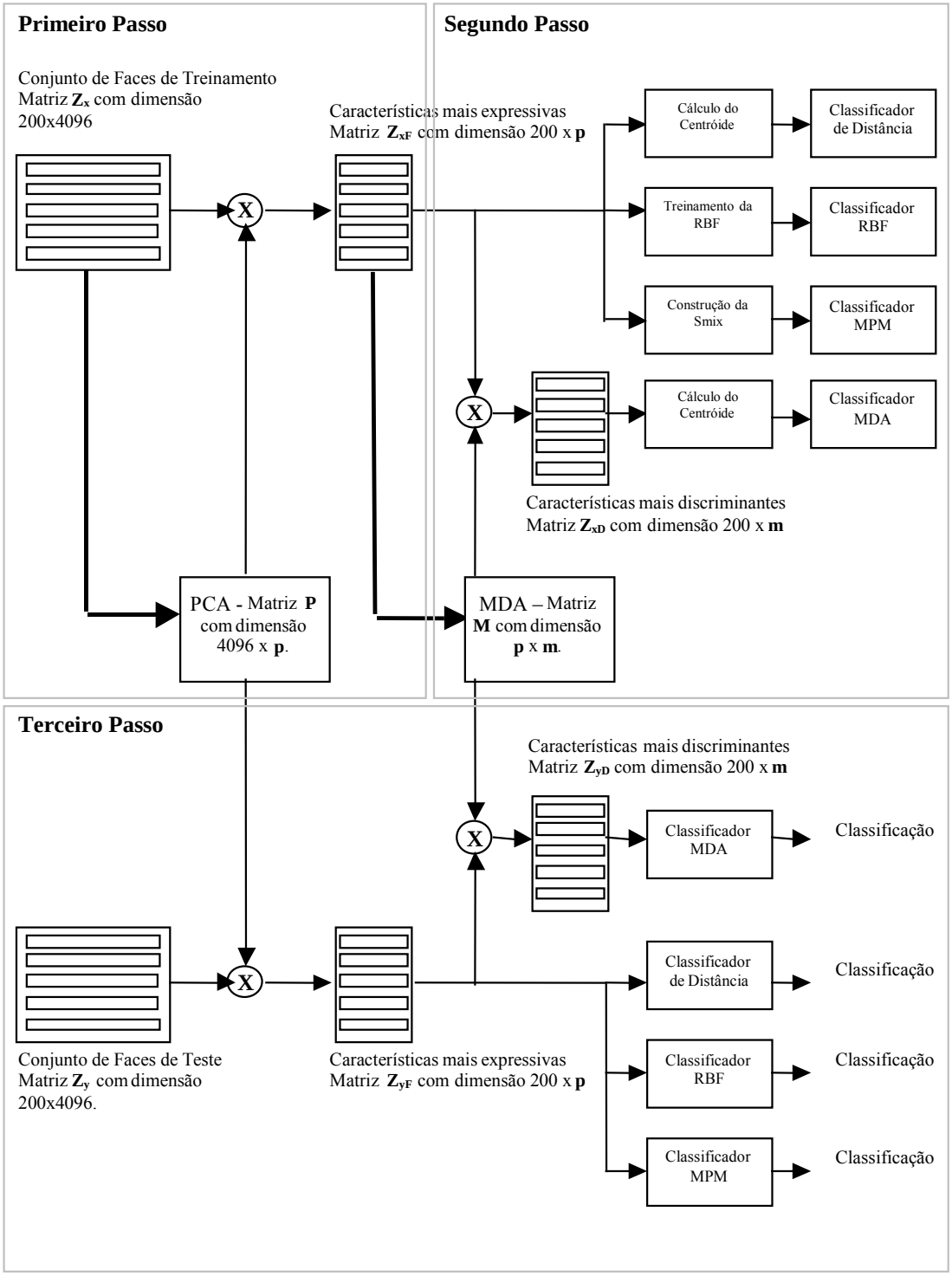

Figura 5.3 – Diagrama dos experimentos

Como cada imagem de face é transformada em um vetor de 4096 valores (figura 5.2) e existem 40 indivíduos distintos no banco de imagens, o conjunto de treinamento é

representado por uma matriz $\mathbf{X}$ com dimensão 200x4096 e o conjunto de teste por uma matriz $\mathbf{Y}$ de mesma dimensão. Estes valores correspondem, respectivamente, a 200 linhas ou imagens de face e 4096 colunas ou pixels, conforme a matriz de treinamento $\mathbf{X}$ explicitada abaixo

$$\mathbf{X} = \begin{bmatrix} \mathbf{x_1}^T \\ \vdots \\ \mathbf{x_{200}}^T \end{bmatrix} = \begin{bmatrix} x_1(1) & \cdots & x_1(4096) \\ \vdots & \ddots & \vdots \\ x_{200}(1) & \cdots & x_{200}(4096) \end{bmatrix}.$$

Pretende-se, pois, calcular as componentes principais deste conjunto de treinamento. Conforme visto na seção 3.1.2 do capítulo 3, tal cálculo implica em encontrar os autovetores e autovalores do sistema de variáveis em questão. Como, no caso, todas as variáveis representam pixels que podem assumir valores entre 0 e 255 (as imagens possuem pixels de 256 tons de cinza) e, por consequência, possuem a mesma ordem de grandeza, o cálculo das componentes principais é realizado a partir da matriz de covariância $\mathbf{S}$. Esta matriz $\mathbf{S}$ é construída da seguinte forma. Inicialmente, calcula-se o vetor $\overline{\mathbf{X}}$ que corresponde à média de todas as imagens de face de treinamento, ou seja,

$$\overline{\mathbf{X}} = \left[\overline{\mathrm{x}}(1) \quad \cdots \quad \overline{\mathrm{x}}(4096)\right]$$

onde $\overline{\mathrm{x}}(n)$ é o valor médio de um pixel n, calculado como

$$\overline{\mathrm{x}}(n) = \frac{1}{200} \sum_{\mathrm{i}=1}^{200} \mathrm{x}_{\mathrm{i}}(n), \quad \text{para } \mathrm{n} = 1,2,...,4096 \text{ pixels.}$$

Depois, diminui-se de cada face, pertencente ao conjunto de treinamento $\mathbf{X}$ e ao conjunto de teste $\mathbf{Y}$, esse mesmo vetor $\overline{\mathbf{X}}$, criando-se as matrizes $\mathbf{Z_x}$ e $\mathbf{Z_y}$

$$\mathbf{Z_x} = \begin{bmatrix} \mathbf{z}_{x_1}^T \\ \vdots \\ \mathbf{z}_{x_{200}}^T \end{bmatrix} = \begin{bmatrix} x_1(1) - \overline{x}(1) & \cdots & x_1(4096) - \overline{x}(4096) \\ \vdots & \ddots & \vdots \\ x_{200}(1) - \overline{x}(1) & \cdots & x_{200}(4096) - \overline{x}(4096) \end{bmatrix}$$

$$\mathbf{Z_y} = \begin{bmatrix} \mathbf{z}_{y_1}^T \\ \vdots \\ \mathbf{z}_{y_{200}}^T \end{bmatrix} = \begin{bmatrix} y_1(1) - \overline{x}(1) & \cdots & y_1(4096) - \overline{x}(4096) \\ \vdots & \ddots & \vdots \\ y_{200}(1) - \overline{x}(1) & \cdots & y_{200}(4096) - \overline{x}(4096) \end{bmatrix}$$

A matriz $\mathbf{Z_x}$, cujas colunas possuem médias iguais a zero, é utilizada como entrada para o cálculo das componentes principais (linha tracejada na figura 5.3 – primeiro passo), enquanto a matriz $\mathbf{Z_y}$ é guardada para ser utilizada no terceiro passo, o passo de teste dos classificadores.

Para se encontrar as componentes principais da matriz $\mathbf{Z_x}$, deve-se computar a matriz de covariância $\mathbf{S}$ que é dada por $\mathbf{S} = \mathbf{Z_x}^{\mathbf{T}}\mathbf{Z_x}$ e tem dimensão 4096 x 4096. No entanto, em situações onde tem-se o número n de variáveis maior do que o número k de exemplos de treinamento - nesta aplicação tem-se $n = 4096$ variáveis e $k = 200$ exemplos de treinamento - pode-se encontrar as componentes principais, ou seja, os autovetores e autovalores do sistema, a partir da matriz $\mathbf{Z_x}\mathbf{Z_x}^{\mathbf{T}}$ de dimensão $k \times k$ - no caso, uma matriz bem menor de dimensão 200x200. Isto implica em obter a solução da equação fundamental do problema

$$\mathbf{Z}_x \mathbf{Z}_x^{\,T} \mathbf{w}_i = \lambda_i \mathbf{w}_i \,, \qquad (5.2.1.1)$$

onde o autovalor λ_i está associado ao autovetor $\mathbf{w_i}$. A premultiplicação da equação (5.2.1.1) por $\mathbf{Z_x^T}$ resulta em

$$\mathbf{Z}_x{}^T\mathbf{Z}_x(\mathbf{Z}_x{}^T\mathbf{w}_i) = \lambda_i(\mathbf{Z}_x{}^T\mathbf{w}_i). \qquad (5.2.1.2)$$

Logo, pode-se verificar pela equação (5.2.1.2) que $\mathbf{e_i} = \mathbf{Z_x^T w_i}$ é o autovetor de $\mathbf{S} = \mathbf{Z_x^T Z_x}$ com autovalor igual a λ_i e estes pares de autovetores e autovalores podem, assim, ser calculados através de uma manipulação matemática que requer bem menos recursos computacionais. Os autovetores podem ser ordenados segundo a ordem descrescente dos correspondentes autovalores. A partir desta consideração, constrói-se, enfim, a matriz $\mathbf{P}$ de transformação do PCA que é formada por um conjunto de p autovetores distintos, conforme descrito abaixo

$$\mathbf{P} = \begin{bmatrix}\mathbf{e_1} & \cdots & \mathbf{e_p}\end{bmatrix} = \begin{bmatrix} e_1(1) & \cdots & e_p(1) \\ \vdots & \ddots & \vdots \\ e_1(4096) & \cdots & e_p(4096) \end{bmatrix}_{4096 \text{ x } p}$$

O valor máximo do parâmetro p pode ser determinado considerando-se a variância λ_i associada a cada componente ou autovetor $\mathbf{e_i}$ ignorado. O erro residual dado pela utilização de p características, de tal forma que $p < (k-1)$, é simplesmente a soma dos autovalores cujos autovetores não foram utilizados, ou seja,

$$\sum_{i=p+1}^{k} \lambda_i \ . \qquad (5.2.1.3)$$

Pode-se, portanto, escolher p de tal forma que essa soma seja menor do que um percentual F da variância total dada por todas as componentes principais. Assim, seleciona-se p que satisfaça a inequação

$$\left(\sum_{i=p+1}^{k} \lambda_i \right) \Big/ \left(\sum_{i=1}^{k} \lambda_i \right) < \mathrm{F}. \qquad (5.2.1.4)$$

Por exemplo, para $\mathrm{F} = 10\%$ pode-se ter uma expressiva redução de dimensionalidade dos vetores originais enquanto preserva-se 90% da variância total dos dados. Em muitas situações, este valor de F é mais do que suficiente para a classificação dos dados indicando, como visto na seção 3.3 do capítulo 3, que diminuindo-se esse percentual não necessariamente tem-se resultados melhores. Assim, o número de componentes principais p pode variar de aplicação para aplicação e uma forma de determiná-lo é variá-lo experimentalmente. Este foi o procedimento adotado neste trabalho.

Uma vez que a matriz $\mathbf{P}$ de autovetores já esteja calculada, projetam-se as imagens de face que se encontram no espaço original de dimensão 4096, o espaço de imagens, no novo espaço formado pelas p componentes principais do conjunto de treinamento. Este espaço e as respectivas componentes principais são chamados em diversos trabalhos de espaço de faces e de autofaces, respectivamente. Um exemplo das 20 primeiras autofaces (p =20) calculadas a partir de um conjunto de treinamento de 200 imagens escolhido aleatoriamente pode ser visto na figura 5.4.

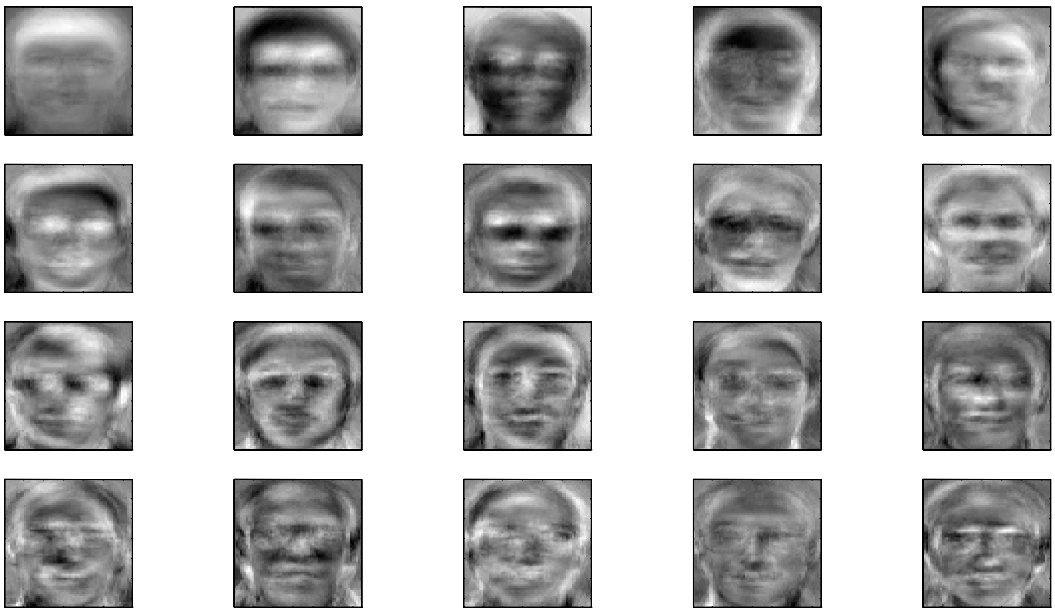

Figura 5.4 – Exemplo de 20 autofaces calculadas a partir de um conjunto de treinamento

A projeção no espaço de faces, como mostra a figura 5.3, se faz através de uma simples transformação linear, dada pela fórmula $\mathbf{Z}_{\mathbf{xF}} = \mathbf{Z}_{\mathbf{x}}\mathbf{P}$. A matriz $\mathbf{Z}_{\mathbf{xF}}$ possui, portanto, dimensão 200 x p e representa as características mais expressivas das 200 imagens de face de treinamento projetadas no novo espaço, cujos eixos são dados pelas p autofaces escolhidas. Esta nova matriz $\mathbf{Z}_{\mathbf{xF}}$ é, então, utilizada como entrada para o segundo passo dos experimentos.

5.2.2 Treinamento dos Classificadores

No segundo passo dos experimentos os classificadores são treinados.

5.2.2.1 Classificadores de Distância

Como pode-se observar pela figura 5.3, o treinamento dos classificadores de distância, tanto da distância Euclideana quanto da distância de Mahalanobis, consiste meramente no cálculo dos centróides de cada classe. Este cálculo é feito tomando-se a média das 5 imagens de cada indivíduo utilizadas como treinamento e projetadas no espaço de faces. Os centróides ou vetores média $\boldsymbol{\mu}_i$ de cada indivíduo, onde $i = 1,2,...40$, tem, pois, dimensão p e são calculados a partir da matriz $\mathbf{Z}_{\mathbf{xF}}$. Estes valores são utilizados posteriormente para a classificação das imagens de teste, no terceiro e último passo.

5.2.2.2 Classificador RBF

Para se treinar o classificador RBF, foram utilizados dois algoritmos de aprendizado denominados Método Ortogonal dos Mínimos Quadrados ou *Forward Selection* (FS) e Método de Mistura de Gaussianas ou *Mixture Model* (MM), examinados no capítulo 4. Ambas as redes RBF possuem o mesmo número de entradas e o mesmo número de saídas, tendo, no entanto, os números de neurônios na camada escondida distintos para cada algoritmo de aprendizado. O número de entradas é determinado pelas p características de cada face que são dadas pela matriz de transformação do PCA, e o número de neurônios na camada de saída é sempre igual a 40, pois existem 40 indivíduos ou grupos para se classificar. Os dois algoritmos de aprendizado foram implementados para responder com o valor 1 no neurônio correspondente ao indivíduo cujo vetor de características da face foi apresentado na entrada e 0 nos demais neurônios de saída.

Para o treinamento do classificador RBF com algoritmo de aprendizado FS, dividiram-se as 5 imagens de face de cada indivíduo, previamente escolhidas no primeiro passo, em dois sub-conjuntos: 3 imagens para treinamento e 2 imagens para validação. Esta é uma abordagem chamada de "*hold out method*" [Bish97] e costuma ser utilizada em algoritmos de aprendizado de redes neurais para se evitar o sobre-treinamento das mesmas, prejudicando suas respectivas generalizações. Depois de serem projetados no espaço de faces, tanto o sub-conjunto formado pelas imagens de treinamento quanto o sub-conjunto formado pelas imagens de validação são utilizados para se treinar a rede RBF. As funções de ativação dos neurônios da camada escondida são dadas por funções base radiais gaussianas, conforme fórmula (4.2.2) descrita no capítulo 4. Os centros destas funções são definidos pelo sub-conjunto de treinamento, utilizado como conjunto de dados de entrada, e cada função base possui a mesma variânça, todas elas iguais a 1. Então, conforme descrito no capítulo 4, o processo de treinamento FS, dado pelo método ortogonal dos mínimos quadrados, cria iterativamente a rede RBF, incrementando-se em cada novo passo de treinamento o número de neurônios pertencentes à camada escondida da rede. Estes neurônios são adicionados até que a soma do erro quadrático referente ao sub-conjunto de validação atinja o seu mínimo. Quando isto acontece, a rede é então considerada como treinada e pronta para ser testada no terceiro passo.

O treinamento da rede RBF com algoritmo de aprendizado *Mixture Model* é um pouco diferente do anterior. Neste caso, como a finalidade é estimar a densidade de probabilidade do conjunto de dados de treinamento, todas as 5 projeções das imagens de face de cada indivíduo, ou seja, toda a matriz $\mathbf{Z}_{\mathbf{xF}}$ é utilizada como entrada da rede. O número de neurônios Q da camada escondida passa a ser um outro parâmetro de entrada do modelo de tal forma que o procedimento não-supervisionado faz a otimização dos parâmetros das Q

funções de ativação através apenas do conjunto de treinamento. Os centros das funções base gaussianas são determinados pelo método iterativo referente ao algoritmo EM (*expectation-maximization*), citado na seção 4.2.2.2 do capítulo 4, e as respectivas variâncias são feitas iguais a máxima distância quadrada entre esses centros otimizados. Depois da determinação dos parâmetros das funções gaussianas, o treinamento termina com o cálculo dos pesos existentes entre a camada escondida de Q neurônios e a camada de saída de 40 neurônios. Estes pesos são dados pela pseudo-inversa, conforme discussão do capítulo 4.

5.2.2.3 Classificador MDA

O terceiro classificador implementado foi aquele aqui denominado de classificador MDA (*Multivariate Discrimant Analysis*), ou classificador de Análise Multivariada de Discriminantes. Este classificador nada mais é do que um classificador de distância Euclideana onde as faces a serem comparadas com os centróides de cada grupo são antes projetadas em um outro espaço, o espaço formado pelos eixos discriminantes das características mais expressivas de cada face. Estes eixos discriminantes são calculados de forma análoga às componentes principais do primeiro passo, com o detalhe de serem originados a partir da matriz $\mathbf{Z}_{\mathbf{xF}}$ de dimensão 200 x p e serem calculados a partir do produto de matrizes $\mathbf{W}^{-1}\mathbf{B}$, onde $\mathbf{W}$ é a matriz intra-grupos e $\mathbf{B}$ a matriz inter-grupos definidas no capítulo 3. Primeiro calculam-se, então, as matrizes simétricas $\mathbf{B}$ e $\mathbf{W}$ que possuem dimensão m x m de acordo com as fórmulas (3.2.2.1) e (3.2.2.2) apresentadas no capítulo 3. Naquele capítulo, mostrou-se que para se encontrar os eixos discriminantes, deve-se calcular os autovetores e os autovalores do produto de matrizes $\mathbf{W}^{-1}\mathbf{B}$. Isto implica em resolver a equação fundamental

$$\mathbf{W}^{-1}\mathbf{B}\mathbf{v}_{\mathbf{i}} = \lambda_i \mathbf{v}_{\mathbf{i}}, \qquad (5.2.2.3.1)$$

onde, o autovalor λ_i está associado ao autovetor $\mathbf{v}_{\mathbf{i}}$. No entanto, como não necessariamente a matriz $\mathbf{W}^{-1}\mathbf{B}$ é simétrica, o cálculo computacional desses autovalores e autovetores pode se tornar instável, gerando vetores com valores imaginários. Para evitar este problema, adota-se o seguinte método algébrico que diagonaliza duas matrizes simétricas e produz um sistema de variáveis estável. Computa-se primeiramente as matrizes $\mathbf{H}$ e $\mathbf{\Lambda}$ de tal forma que

$$\mathbf{W} = \mathbf{H}\mathbf{\Lambda}\mathbf{H}^{\mathbf{T}}, \qquad (5.2.2.3.2)$$

onde $\mathbf{H}$ e $\mathbf{\Lambda}$ são, respectivamente, uma matriz ortonormal e diagonal que correspondem aos autovetores e autovalores de $\mathbf{W}$. Logo, $\left(\mathbf{H}\mathbf{\Lambda}^{-\frac{1}{2}}\right)^{\mathbf{T}}\mathbf{W}\mathbf{H}\mathbf{\Lambda}^{\frac{1}{2}} = \mathbf{I}$. Calcula-se, agora, $\mathbf{U}$ e $\mathbf{\Sigma}$ de tal forma que $\left(\mathbf{H}\mathbf{\Lambda}^{-\frac{1}{2}}\right)^{\mathbf{T}}\mathbf{B}\mathbf{H}\mathbf{\Lambda}^{\frac{1}{2}} = \mathbf{U}\mathbf{\Sigma}\mathbf{U}^{\mathbf{T}}$, onde $\mathbf{U}$ e $\mathbf{\Sigma}$ são também, respectivamente, uma matriz ortonormal e diagonal que correspondem aos autovetores e autovalores de $\left(\mathbf{H}\mathbf{\Lambda}^{-\frac{1}{2}}\right)^{\mathbf{T}}\mathbf{B}\mathbf{H}\mathbf{\Lambda}^{\frac{1}{2}}$. Assim, pode-se escrever

$$\mathbf{B} = \mathbf{H}\mathbf{\Lambda}^{\frac{1}{2}}\mathbf{U}\mathbf{\Sigma}\mathbf{U}^{\mathbf{T}}\mathbf{\Lambda}^{\frac{1}{2}}\mathbf{H}^{\mathbf{T}} \qquad (5.2.2.3.3)$$

$$\mathbf{W} = \mathbf{H}\mathbf{\Lambda}^{\frac{1}{2}}\mathbf{U}\mathbf{I}\mathbf{U}^{\mathbf{T}}\mathbf{\Lambda}^{\frac{1}{2}}\mathbf{H}^{\mathbf{T}}. \qquad (5.2.2.3.4)$$

Através da equação (5.2.2.3.2), tem-se que

$$\mathbf{W}^{-1} = \mathbf{H}\boldsymbol{\Lambda}^{-1}\mathbf{H}^{\mathbf{T}}, \qquad (5.2.2.3.5)$$

e utilizando as equações (5.2.2.3.5) e (5.2.2.3.3), obtém-se

$$\begin{aligned} \mathbf{W}^{-1}\mathbf{B} &= \mathbf{H}\boldsymbol{\Lambda}^{-1}\mathbf{H}^{\mathbf{T}}\mathbf{H}\boldsymbol{\Lambda}^{\frac{1}{2}}\mathbf{U}\boldsymbol{\Sigma}\mathbf{U}^{\mathbf{T}}\boldsymbol{\Lambda}^{\frac{1}{2}}\mathbf{H}^{\mathbf{T}} \\ &= \mathbf{H}\boldsymbol{\Lambda}^{-\frac{1}{2}}\mathbf{U}\boldsymbol{\Sigma}\mathbf{U}^{\mathbf{T}}\boldsymbol{\Lambda}^{\frac{1}{2}}\mathbf{H}^{\mathbf{T}} \\ &= (\mathbf{H}\boldsymbol{\Lambda}^{-\frac{1}{2}}\mathbf{U})\boldsymbol{\Sigma}(\mathbf{H}\boldsymbol{\Lambda}^{-\frac{1}{2}}\mathbf{U})^{-1} \end{aligned} \qquad (5.2.2.3.6)$$

Pode-se verificar pela equação (5.2.2.3.6) que o produto de matrizes $\mathbf{H}\boldsymbol{\Lambda}^{-\frac{1}{2}}\mathbf{U}$ representa os autovetores de $\mathbf{W}^{-1}\mathbf{B}$ e $\boldsymbol{\Sigma}$ os seus autovalores correspondentes. Assim, utilizando as propriedades de simetria das matrizes $\mathbf{W}$ e $\mathbf{B}$ tem-se um método para calcular os autovetores e autovalores de $\mathbf{W}^{-1}\mathbf{B}$ de forma computacionalmente estável. A partir desta consideração algébrica, constrói-se, enfim, a matriz $\mathbf{M}$ de transformação do MDA que é formada por um conjunto de m autovetores distintos, conforme descrito abaixo

$$\mathbf{M} = \begin{bmatrix} \mathbf{v_1} & \cdots & \mathbf{v_m} \end{bmatrix} = \begin{bmatrix} v_1(1) & \cdots & v_m(1) \\ \vdots & \ddots & \vdots \\ v_1(p) & \cdots & v_m(p) \end{bmatrix}_{p \times m}$$

Como existem no máximo $g-1$ autovalores não nulos de $\mathbf{W}^{-1}\mathbf{B}$, onde g é o número de grupos ou indivíduos, o valor máximo do parâmetro m é, neste caso, igual a 39.

Com a matriz $\mathbf{M}$ de transformação do MDA calculada, projetam-se as características mais expressivas de cada face, que se encontram no espaço de faces de dimensão p, no novo

espaço formado pelas m características discriminantes do conjunto de treinamento $\mathbf{Z}_{\mathbf{xF}}$. Esta projeção é similar à projeção do PCA e é dada, como mostra a figura 5.3, por uma simples transformação linear, dada pela fórmula $\mathbf{Z}_{\mathbf{xD}} = \mathbf{Z}_{\mathbf{xF}}\ \mathbf{M}$. A matriz $\mathbf{Z}_{\mathbf{xD}}$ possui, portanto, dimensão 200 x m e representa as características mais discriminantes das características mais expressivas das imagens de face originais. Por fim, tomam-se as linhas dessa matriz $\mathbf{Z}_{\mathbf{xD}}$ para se computar os centróides ou vetores média $\boldsymbol{\mu}_i$ de cada indivíduo nesta nova base, onde $i = 1,2,...40$, os quais serão usados posteriormente para a classificação das imagens de teste, no terceiro e último passo.

5.2.2.4 Classificador MPM

O quarto e último classificador treinado foi justamente o novo modelo de classificação proposto neste trabalho, o classificador gaussiano de Máxima Probabilidade com matrizes de covariância "Misturadas" (MPM).

O treinamento deste classificador é bastante simples e se refere apenas à computação das matrizes de covariância $\mathbf{Smix_i}$ de cada indivíduo ou grupo, onde $i = 1,2,...40$. Para isto, conforme o detalhamento do modelo na seção 4.3.2.3 do capítulo 4, precisa-se calcular, a partir da matriz de treinamento $\mathbf{Z}_{\mathbf{xF}}$, as matrizes de covariância $\mathbf{S_i}$ dada pelos 5 vetores de características mais expressivas de cada grupo e a matriz de covariância $\mathbf{S_{pooled}}$. Estas matrizes são, pois, combinadas linearmente de acordo com a fórmula (4.3.2.3.1) do capítulo 4 através de um fator a, de tal forma que $0 < a < 1$. Este fator a é um parâmetro de entrada para o modelo e é considerado igual para todas as matrizes $\mathbf{Smix_i}$ calculadas.

Para completar o treinamento, computam-se os centróides ou vetores média $\boldsymbol{\mu}_i$ de cada indivíduo através da matriz $\mathbf{Z}_{\mathbf{xF}}$, e estes vetores, juntamente com as matrizes $\mathbf{Smix_i}$ já calculadas, são utilizados posteriormente para a classificação das imagens de teste, no último passo dos experimentos.

5.2.3 Teste dos Classificadores

No terceiro e último passo, o desempenho de cada um dos classificadores treinados é avaliado. Cada imagem de teste é projetada no espaço de faces, descrito pela matriz $\mathbf{P}$ formada pelas componentes principais geradas no primeiro passo, e estas projeções são utilizadas como entrada para os classificadores de distância, os classificadores RBF e o classificador MPM (figura 5.3). No caso do classificador MDA, as suas respectivas entradas são o resultado de duas projeções, uma no espaço do PCA através da matriz $\mathbf{P}$ e a outra no espaço do MDA cujos eixos ou autovetores se referem a matriz $\mathbf{M}$ calculada no segundo passo. Para cada imagem de teste é contabilizado o número de reconhecimentos corretos e o número de reconhecimentos falsos, para cada um dos classificadores, e este valor é armazenado para o cálculo da taxa de reconhecimento de cada modelo de classificador implementado.

Este procedimento em três passos foi repetido 25 vezes, utilizando-se diferentes conjuntos de treinamento e de teste escolhidos aleatoriamente. O número de componentes principais usado para representar as imagens no espaço de faces (p) foi variado ao longo dos experimentos e assumiu diversos valores distintos. Outros parâmetros que foram variados nos experimentos foram o número de neurônios Q da camada escondida da rede RBF MM (*Mixture Model*), o número de componentes discriminantes (m) do classificador MDA, e o

fator de regressão a do classificador MPM. Para cada uma das p componentes principais, esses parâmetros de entrada assumiram valores distintos que serão explicitados no próximo capítulo.

Todos os experimentos foram implementados no *software* matemático MatLab. Para o treinamento da rede RBF FS foi utilizado como base o algoritmo disponível na caixa de ferramentas de redes neurais do próprio Matlab. O treinamento da rede RBF MM foi realizado utilizando-se rotinas prontas previamente desenvolvidas pelo pesquisador C.L.Bishop [Bish97].