DETECÇÃO DE PONTOS FIDUCIAIS EM FACES USANDO FILTRAGEM LINEAR

Felipe Moreira Lopes Ribeiro

Dissertação de Mestrado apresentada ao Programa de Pós-graduação em Engenharia Elétrica, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Mestre em Engenharia Elétrica.

Orientador: Eduardo Antônio Barros da Silva

Rio de Janeiro
Fevereiro de 2014

DETECÇÃO DE PONTOS FIDUCIAIS EM FACES USANDO FILTRAGEM LINEAR

Felipe Moreira Lopes Ribeiro

DISSERTAÇÃO SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA (COPPE) DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS EM ENGENHARIA ELÉTRICA.

Examinada por:

Prof. Eduardo Antônio Barros da Silva, Ph.D.

Prof. Sérgio Lima Netto, Ph.D.

Prof. Lisandro Lovisolo, D.Sc.

RIO DE JANEIRO, RJ – BRASIL
FEVEREIRO DE 2014

Ribeiro, Felipe Moreira Lopes

Detecção de Pontos Fiduciais em Faces Usando Filtragem Linear/Felipe Moreira Lopes Ribeiro. – Rio de Janeiro: UFRJ/COPPE, 2014.

XII, 71 p.: il.; 29, 7cm.

Orientador: Eduardo Antônio Barros da Silva

Dissertação (mestrado) – UFRJ/COPPE/Programa de Engenharia Elétrica, 2014.

Referências Bibliográficas: p. 64 – 69.

1. Reconhecimento de padrões. 2. Processamento de imagens. 3. Visão computacional. I. Silva, Eduardo Antônio Barros da. II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia Elétrica. III. Título.

À minha família

Agradecimentos

Escrever é uma arte. Escrever agradecimentos uma arte ainda maior. Não estou à altura de tamanha arte, mas já que me instaram a escrever um agradecimento, tentarei o meu melhor.

Existe muito a agradecer, mas antes de tudo, gostaria de agradecer à minha família, sem a qual este trabalho não seria possível, incluindo aqueles que desafortunadamente não puderam estar presente. Obrigado pelo suporte, carinho, amor e pelos exemplos de vida.

Em seguida aos meus amigos. Obrigado pela amizade concedida. Vida longa e próspera. Apesar dos encontros e desencontros. Apesar das despedidas. Que possamos ainda compartilhar muitas memórias e cultivar nossa amizade.

Dentre os meus amigos, agradeço novamente, também como colegas, aos amigos do recém formado SMT. Não citarei nomes, para que ninguém se sinta culpado pelo texto que segue, e para que as gerações futuras se sintam parte dos agradecimentos. Obrigado pelo apoio, descontração e conselhos. Que continuemos trabalhando juntos em futuras empreitadas.

Agradeço ao meu orientador, Eduardo, e aos meus colegas, Gabriel e José Fernando. Os conselhos e ajuda proporcionados foram de fundamental importância ao longo desta jornada. Muito obrigado pela oportunidade e pelos exemplos.

Aos membros da banca, agradeço pelos conselhos e pela presença em um momento importante. Espero que o trabalho esteja à altura da ajuda despendida.

Aos futuros leitores, muito obrigado por terem lido até os agradecimentos. Que a atenção recebida seja recompensada.

Resumo da Dissertação apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Mestre em Ciências (M.Sc.)

DETECÇÃO DE PONTOS FIDUCIAIS EM FACES USANDO FILTRAGEM LINEAR

Felipe Moreira Lopes Ribeiro

Fevereiro/2014

Orientador: Eduardo Antônio Barros da Silva

Programa: Engenharia Elétrica

Este trabalho têm três contribuições principais. A primeira contribuição deste trabalho é a implementação de um arcabouço para treino e validação de detectores. Para prover flexibilidade, este arcabouço foi projetado de modo que seus blocos constituintes são blocos permutáveis. Desta forma, outros métodos podem ser aplicados sobre diversas bases da dados e avaliados no arcabouço proposto. A segunda contribuição é a avaliação do desempenho de diferentes técnicas de filtragem linear quando aplicadas a detecção de pontos fiduciais faciais. Detectores lineares foram escolhidos devido a rapidez, simplicidade e eficiência, com taxas de acertos comparáveis a métodos mais complexos. Adotou-se o contexto de detecção de pontos fiduciais faciais em razão de sua relevância para uma gama de aplicações, como interfaces homem-máquina, sistemas de entretenimento, sistemas de segurança, dentre outros. Como terceira contribuição são propostas novas abordagens para alguns dos métodos já existentes, seja no treinamento ou no formato de extração de características, tendo como objetivo aumentar a robustez e o desempenho dos detectores. O sistema resultante é capaz de realizar a detecção de pontos fiduciais em indivíduos não pertencentes à base de treinamento em tempo-real, sobre várias condições de iluminação, com resultados competitivos.

Abstract of Dissertation presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Master of Science (M.Sc.)

FIDUCIAL LANDMARKS DETECTION ON FACES USING LINEAR FILTERING

Felipe Moreira Lopes Ribeiro

February/2014

Advisor: Eduardo Antônio Barros da Silva

Department: Electrical Engineering

This work has three main contributions. First, implementing a framework for training and validation of detectors. In order to provide flexibility, the framework was designed as interchangeable building blocks. Thus, different methods applied to the same and/or other databases can be used and evaluated inside the proposed framework. Second, evaluating the performance of different linear filtering techniques when applied to human fiducial points detection. Linear detectors were selected by their speed and low complexity, with hit rates comparable to more complex methods. The facial fiducial points detection context was chosen due to its relevance for a great range of applications, such as human-machine interfaces, entertainment systems, security systems, as well as many others. The third contribution consist of new approaches to some existing methods, either in the training or in the features extraction, with the objective of increasing the detectors robustness and overall performance. The proposed framework is able to recognize facial landmarks on subjects not belonging to the training database in real-time, under various lighting conditions, with competitive performance.

Sumário

Lista de Figuras

Lista de Tabelas

Lista de Algoritmos

Capítulo 1

Introdução

Recentemente o problema de extração de informações e reconhecimento de padrões em imagens tem recebido grande atenção, dado o crescente interesse em formas alternativas de interação com dispositivos eletrônicos com diferentes usos em múltiplas aplicações, como sistemas de segurança, sistemas de realidade aumentada, interface homem-máquina, dentre outras [1, 2].

Muitas dessas aplicações têm como objetivo a interação e o reconhecimento de seres humanos, onde o conjunto de características mais marcantes são localizadas sobre a face. Nesse caso, a realização de tarefas pode ser feita através da detecção de *pontos fiduciais faciais*. Pontos fiduciais são pontos de controle sobre regiões características de um objeto. No caso da face, são pontos definem uma face humana, como ponta do nariz, cantos da boca e centro dos olhos.

Existem inúmeras técnicas para realizar essa detecção, principalmente para a região dos olhos [3–5]. Como exemplo, em [5] é realizada a detecção de pontos fiduciais da região dos olhos através de detectores locais *Inner Product Detector* (IPD) [6, 7] em conjunto com o detector de faces *Viola-Jones* [8], utilizados em conjunto com um esquema de consistência temporal, tendo o auxílio do algoritmo de fluxo óptico *Lucas-Kanade* [9], e geométrica, trabalho continuado em [10]. Em [1] é apresentado um método que combina a saída de detectores locais com um modelo geométrico de face. Apesar dos bons resultados, o tempo de detecção torna o sistema inviável para o uso em tempo real. Está restrição é superada em [2], onde é apresentado um método que utiliza classificadores *random trees*, e em [11], onde a detecção é realizada utilizando cascata de detectores por produto interno.

Dentro da limitação de tempo real, uma família de detectores se destaca: os detectores[1] lineares [3, 11], onde o detector é capaz de indicar, através de operações lineares, se a imagem de teste contém o objeto alvo. Estes detectores possuem, como qualidades, simplicidade, robustez e baixa complexidade, necessárias quando

[1]Devido a natureza do trabalho, os termos filtros, classificadores e detectores serão utilizados de forma equivalente ao longo do texto.

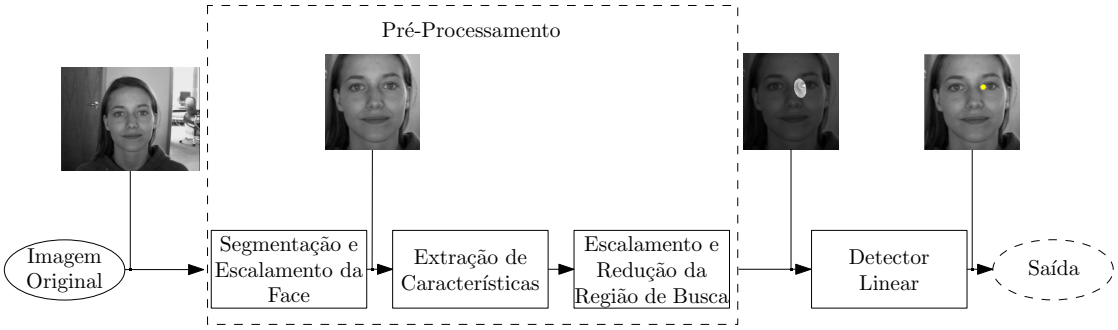


Figura 1.1: Arcabouço implementado

considerada a dimensão do problema de detecção em imagens, tornando-os objetos de interesse desta dissertação.

A primeira contribuição deste trabalho é a implementação de um arcabouço de treinamento e validação de detectores implementado em linguagem de programação *C++* em conjunto com a biblioteca de programação *OpenCV* [12]. O sistema funciona de forma modular com blocos permutáveis e permite a detecção em tempo real. Um diagrama em blocos do sistema implementado é apresentado na Figura 1.1.

A segunda contribuição deste trabalho é a avaliação dos diferentes classificadores lineares utilizando uma base de dados com imagens e marcações manuais disponíveis [13]. Cinco métodos foram avaliados: detectores IPD [6, 7], filtros discriminativos [6], e os filtros conhecidos como *Unconstrained Average Correlation Energy* (UMACE) [14], *Average of Synthetic Exact Filters* (ASEF) [3], e *Minimum Output Sum of Squared Error* (MOSSE) [15].

A terceira contribuição são variações do treinamento do detector IPD com o objetivo de melhorar o desempenho do mesmo ou permitir o treinamento online. Cinco novas abordagens são propostas. A primeira, denominada *Regressor IPD*, o transforma o problema de classificação dicotômico (entre duas classes) em um problema de regressão. Essa modificação gera detectores mais robusto ao ponderar as amostras segundo sua proximidade com o ponto fiducial alvo. Isto permite, por exemplo, que regiões que contenham o padrão parcialmente ocluso participem do processo de treinamento. Proposta similar é apresentada no segundo método, *Multiple Instance Learning* IPD (MIL IPD), inspirada em um novo paradigma de treinamento que permite ao sistema lidar com ambiguidades automaticamente através do treino e avaliação das amostras em conjuntos denominados *bags* [16, 17]. As abordagens *Boosting* e *Bagging* IPD partem de princípios distintos ao tentar gerar um classificador de melhor desempenho reunido classificadores. A última metodologia é uma adaptação do método tradicional de treinamento dos classificadores IPD para treinamento online, denominada *Online* IPD.

1.1 Organização

No Capítulo 2 são apresentado os classificadores utilizados neste trabalho, com ênfase no classificador IPD, para o qual serão apresentadas propostas de novos métodos de treinamento.

No Capítulo 3 são descritos os métodos de pré-processamento e extração de características utilizados ao longo deste trabalho.

No Capítulo 4 apresenta-se um estudo de caso em que os classificadores descritos anteriormente são empregados no contexto de detecção de pontos fiduciais em imagens estáticas. Para isso descreve-se a metodologia utilizada e são apresentados os resultados obtidos.

Finalizando, no Capítulo 5, são realizadas as considerações finais sobre os resultados obtidos e são apresentados possíveis desdobramentos.

Capítulo 2

Classificadores Lineares

2.1 Introdução

Seja um problema de classificação tal que exista um conjunto de amostras rotuladas $\{\mathbf{x}_i\}$, e seus respectivos rótulos $\{y_i\}$, onde o objetivo é encontrar uma função tal que

$$f(\mathbf{x}_i) = y_i, \quad \forall i. \tag{2.1}$$

A função que realiza este mapeamento é conhecida como *classificador*. Um exemplo de função de classificação é o classificador linear, definido como

$$\tilde{y}_i = f\left(\mathbf{x}_i\right) = \langle \mathbf{w}, \mathbf{x}_i \rangle + b, \tag{2.2}$$

onde $\mathbf{w}$ é o vetor de pesos e b o fator de viés do detector.

O processo de aprendizado dos parâmetros $\mathbf{w}$ e b é conhecido como treinamento do classificador. Este processo pode ser realizado, por exemplo, minimizando o risco empírico regularizado [18, 19] sobre o conjunto de amostras de treinamento:

$$\min_{\mathbf{w},b} I_{emp}[f] = \sum_{i=1}^{N_s} L\left(y_i, f\left(\mathbf{x}_i\right)\right) + \lambda \|f\|^2, \tag{2.3}$$

onde $L\left(y_i, f\left(\mathbf{x}_i\right)\right)$ é uma função custo, N_s o número de amostras e λ o fator de regularização[1].

O algoritmo de aprendizado é definido a partir da escolha da função de custo $L\left(y_i, f\left(\mathbf{x}_i\right)\right)$. Como exemplo, escolhendo L como a função *hinge loss* (Eq. 2.4) [20]

$$L\left(y_i, f\left(\mathbf{x}_i\right)\right) = \max\left(0, 1 - y_i f\left(\mathbf{x}_i\right)\right), \tag{2.4}$$

[1] O fator de viés (*bias* b) não é utilizado neste trabalho pois há interesse apenas a relação de intensidade entre os valores, que se mantém com ou sem viés.

produz classificadores lineares denominados SVM – *Support Vector Machines* (Máquinas de Vetores Suporte) [19–22]:

O treinamento de classificadores SVM leva a um problema de otimização convexa em N_s variáveis, com bom desempenho em diferentes contextos [18, 19]. Uma alternativa que produz resultados equivalentes em problemas práticos [19] é conhecida como RLS – *Regularized Least Square* (Mínimos Quadrados Regularizado), que utiliza a função de custo quadrática

$$L\left(y_i, f\left(\mathbf{x}_i\right)\right) = \left(y_i - f\left(\mathbf{x}_i\right)\right)^2. \tag{2.5}$$

É possível demonstrar que a solução $f^*\left(\mathbf{x}\right)$, onde $\mathbf{x}$ é uma amostra de teste arbitrária, para o problema de regularização, sem perda de generalidade, admite uma solução na forma [18, 19]

$$f^*\left(\mathbf{x}\right) = \sum_{i}^{N_s} a_i \kappa\left(\mathbf{x}, \mathbf{x}_i\right), \tag{2.6}$$

onde a função $\kappa\left(\mathbf{x}, \mathbf{x}_i\right)$ é denominada *kernel*, mapeando as entradas em um espaço de características definido por

$$\kappa\left(\mathbf{x}_j, \mathbf{x}\right) = \left\langle \varphi\left(\mathbf{x}\right), \varphi\left(\mathbf{x}_i\right) \right\rangle, \tag{2.7}$$

onde $\varphi\left(\mathbf{x}\right)$ é uma função de mapeamento. Neste espaço, a norma de $f\left(\mathbf{x}\right)$ é

$$\|f\|_\kappa^2 = \mathbf{a}^T \mathbf{K} \mathbf{a}, \tag{2.8}$$

onde $\mathbf{K}$ denota a matriz $N_s \times N_s$ na qual o (i, j)-ésimo elemento é $\kappa\left(\mathbf{x}_i, \mathbf{x}_j\right)$.

Substituindo os resultados das equações (2.5), (2.6) e (2.8) na Equação (2.3), o problema de classificação pode ser escrito como

$$\min_{\mathbf{a}} I_{emp}[f] = \left(\mathbf{y} - \mathbf{K}\mathbf{a}\right)^T \left(\mathbf{y} - \mathbf{K}\mathbf{a}\right) + \lambda \mathbf{a}^T \mathbf{K} \mathbf{a}. \tag{2.9}$$

Esta é uma função convexa diferenciável que pode ser minimizada apenas derivando-a em respeito a $\mathbf{a}$:

$$\begin{aligned} \nabla_{I_\mathbf{a}} &= -\mathbf{K}^T \left(\mathbf{y} - \mathbf{K}\mathbf{a}\right) + \lambda \mathbf{K}\mathbf{a}, \\ 0 &= -\mathbf{K}\mathbf{y} + \mathbf{K}\left(\mathbf{K} + \lambda I\right)\mathbf{a}. \end{aligned} \tag{2.10}$$

Assim, os coeficientes $\mathbf{a}$ podem ser obtidos resolvendo

$$\mathbf{a} = \left(\mathbf{K} + \lambda I\right)^{-1} \mathbf{y}. \tag{2.11}$$

Para o classificador linear, em particular, substituindo os coeficientes encontrados

na Equação (2.6), temos que

$$\mathbf{w} = \sum_{i}^{N_s} a_i \kappa\left(\mathbf{x}, \mathbf{x}_i\right). \tag{2.12}$$

Este classificador é denominado classificador RLSC – *Regularized Least Square Classifier* [19].

Nas seções a seguir serão apresentadas variações do classificador RLSC para casos particulares. Apresenta-se na Seção 2.2 os classificadores IPD, classificadores RLSC baseados em produto interno e, como contribuições desenvolvidas ao longo desta dissertação, variações do algoritmo de aprendizado. Na Seção 2.3 são descritos os classificadores baseados em filtragem correlativa/discriminativa, vertentes do classificador RLSC para o problema de correlação. Na Seção 2.4 são delimitados alguns possíveis critérios de decisão dicotômicos. Na Seção 2.5 é feito um breve resumo do capítulo e são apresentadas algumas das abordagens recentes na área.

2.2 Classificador IPD

Dado um conjunto de amostras $\{\mathbf{x}_i\}$ com rótulos $\{y_i\} \in \{0, 1\}$, pode-se definir um classificador linear $\mathbf{h}$ tal que

$$\mathbf{h}^T\mathbf{x}_i = \begin{cases} 1, & \text{se } y_i = 1 \\ 0, & \text{se } y_i = 0 \end{cases} \tag{2.13}$$

O classificador IPD (*Inner Product Detector*) [6, 7] é o classificador obtido através da solução LMMSE para o problema de classificação acima. O erro de classificação é definido como

$$e_i = \mathbf{h}^T\mathbf{x}_i - y_i; \tag{2.14}$$

Para amostras reais, a estimativa LMMSE do classificador $\mathbf{h}$ é obtida resolvendo o problema de minimização do valor esperado do erro quadrático

$$\begin{aligned} E\left[e^2\right] &= E\left[(\mathbf{h}^T\mathbf{x} - y)^T(\mathbf{h}^T\mathbf{x} - y)\right], \\ &= E\left[\mathbf{h}^T\mathbf{x}\mathbf{x}^T\mathbf{h} + y^2 - 2y\mathbf{h}^T\mathbf{x}\right], \\ &= \mathbf{h}^T E\left[\mathbf{x}\mathbf{x}^T\right]\mathbf{h} + E\left[y^2\right] - 2\mathbf{h}^T E\left[\mathbf{x}y\right]. \end{aligned} \tag{2.15}$$

Minimizando em função de $\mathbf{h}$

$$\frac{\partial E\left[e^2\right]}{\partial \mathbf{h}} = 2E\left[\mathbf{x}\mathbf{x}^T\right]\mathbf{h} - 2E\left[\mathbf{x}y\right] = 0, \tag{2.16}$$

obtemos uma solução para $\mathbf{h}$ que minimiza o erro

$$\mathbf{h} = \left(E\left[\mathbf{x}\mathbf{x}^T\right]\right)^{-1} E\left[\mathbf{x}y\right]; \tag{2.17}$$

onde $E\left[\mathbf{x}\mathbf{x}^T\right]$ é a matriz de covariância de $\mathbf{x}$, $\mathbf{R}_{\mathbf{xx}}$, e $E\left[\mathbf{x}y\right]$ é o vetor de correlação cruzada entre os rótulos y e as amostras $\mathbf{x}$, $\mathbf{r}_{y\mathbf{x}}$. Reescrevendo a Equação (2.17) com estas definições temos:

$$\mathbf{h} = \mathbf{R}_{\mathbf{xx}}^{-1}\mathbf{r}_{y\mathbf{x}}, \tag{2.18}$$

observamos que o classificador IPD dicotômico é um caso particular do classificador RLS (vide Equação (2.11)). Definindo $\mathcal{X}_P$ e $\mathcal{X}_N$ como o conjunto de elementos de treinamento das classes positiva e negativa, respectivamente, o vetor de correlação cruzada pode ser escrito como

$$\mathbf{r}_{y\mathbf{x}} = E\left[\mathbf{x}y\right] = E\left[\mathbf{x}y|\,\mathcal{X}_P\right]p(\mathcal{X}_P) + E\left[\mathbf{x}y|\,\mathcal{X}_N\right](p(\mathcal{X}_N)). \tag{2.19}$$

Onde $p(\mathcal{X}_P)$ e $p(\mathcal{X}_N)$ são, respectivamente, as probabilidades das classes positivas e negativas. Como deseja-se $y = 0$ para $\mathbf{x} \in \mathcal{X}_N$, o vetor de correlação cruzada reduz-se a

$$\mathbf{r}_{y\mathbf{x}} = E\left[\mathbf{x}|\,\mathcal{X}_P\right]p(\mathcal{X}_P). \tag{2.20}$$

Substituindo o operador valor esperado pela sua estimativa, o detector IPD pode ser escrito como

$$\begin{aligned}\mathbf{h} &= \left(\sum_{i=1}^{N_s}\mathbf{x}_i\mathbf{x}_i^T\right)^{-1}\left(\sum_{i=1}^{N_s}y_i\mathbf{x}_i\right), \\ &= \hat{\mathbf{R}}_{\mathbf{xx}}\hat{\mathbf{m}}_{\mathcal{X}_P},\end{aligned} \tag{2.21}$$

onde $\hat{\mathbf{m}}_{\mathcal{X}_P}$ é a média amostral sobre os elementos da classe positiva.

2.2.1 Weighted IPD

Retornemos ao problema anterior, considerando apenas a estimativa do valor esperado. Se adicionarmos um peso w_i, relativo à importância da i-ésima amostra, para cada amostra teremos a tríade $(\mathbf{x_i}, w_i, y_i)$ e o problema de classificação pode ser rescrito como

$$\min_{\mathbf{h}} \varepsilon(\mathbf{h}) = \sum_{i=1}^{N_s} w_i(\mathbf{h}^T\mathbf{x}_i - y_i)^2, \tag{2.22}$$

onde $\varepsilon(\mathbf{h})$ é a função custo ou erro sobre o conjunto de treinamento, sendo o peso w_i sujeito à

$$\sum_{i=1}^{N_s} w_i = 1, \tag{2.23}$$

e

$$w_i \geq 0. \tag{2.24}$$

Diferenciando a função custo $\varepsilon(\mathbf{h})$ em $\mathbf{h}$, encontramos

$$\frac{\partial \varepsilon(\mathbf{h})}{\partial \mathbf{h}} = \sum_{i=1}^{N_s} 2w_i \mathbf{x}_i \mathbf{x}_i^T \mathbf{h} - 2w_i y_i \mathbf{x}_i = 0, \tag{2.25}$$

O que leva a uma solução em $\mathbf{h}$ tal que

$$\mathbf{h} = \left(\sum_{i=1}^{N_s} w_i \mathbf{x}_i \mathbf{x}_i^T \right)^{-1} \left(\sum_{i=1}^{N_s} w_i y_i \mathbf{x}_i \right). \tag{2.26}$$

Esta formulação é equivalente a encontrada em na Equação (2.21) para o caso $w_i = {}^1/_{N_s}$ e será utilizada como base para alguns dos métodos descritos a seguir.

2.2.2 Regressor IPD

Nesta abordagem, abandona-se a restrição original sobre os rótulos onde $y_i \in \{0, 1\}$. Adota-se então novos rótulos $y_i \in \mathbb{R}$, de tal forma que o problema de classificação, onde objetiva-se encontrar o rótulo correspondente a classe alvo, torna-se um problema de regressão, onde deseja-se aproximar a resposta da função contínua geratriz dos rótulos. Neste caso, o detector $\mathbf{h}$ resultante é escrito de forma semelhante à Equação 2.21:

$$\begin{aligned} \mathbf{h} &= \left(\sum_{i=1}^{N_s} \mathbf{x}_i \mathbf{x}_i^T \right)^{-1} \left(\sum_{i=1}^{N_s} y_i \mathbf{x}_i \right), \\ &= \hat{\mathbf{R}}_{\mathbf{xx}} \hat{\mathbf{m}}_{\mathbf{x}}, \end{aligned} \tag{2.27}$$

onde o rótulo y_i passa a representar a pertinência da amostra à classe desejada e $\hat{\mathbf{m}}_{\mathbf{x}}$ representa a média ponderada das amostras pela sua respectiva pertinência.

2.2.3 Multiple Instance Learning IPD

Como descrito na Seção 2.1, em um problema de classificação tradicional o objetivo é encontrar uma função conhecida como *classificador* que mapeia cada amostra a seu rótulo.

Entretanto, definir rótulos para cada amostra é um processo muitas vezes complexo, demorado e ambíguo. Um exemplo é o processo de rotulação de amostra para classificações de objetos em imagens. Muitas vezes esse processo é realizado por operadores humanos de forma manual, responsáveis por marcar o centro do objeto ou regiões consideradas importantes, um processo intrinsecamente ambíguo devido

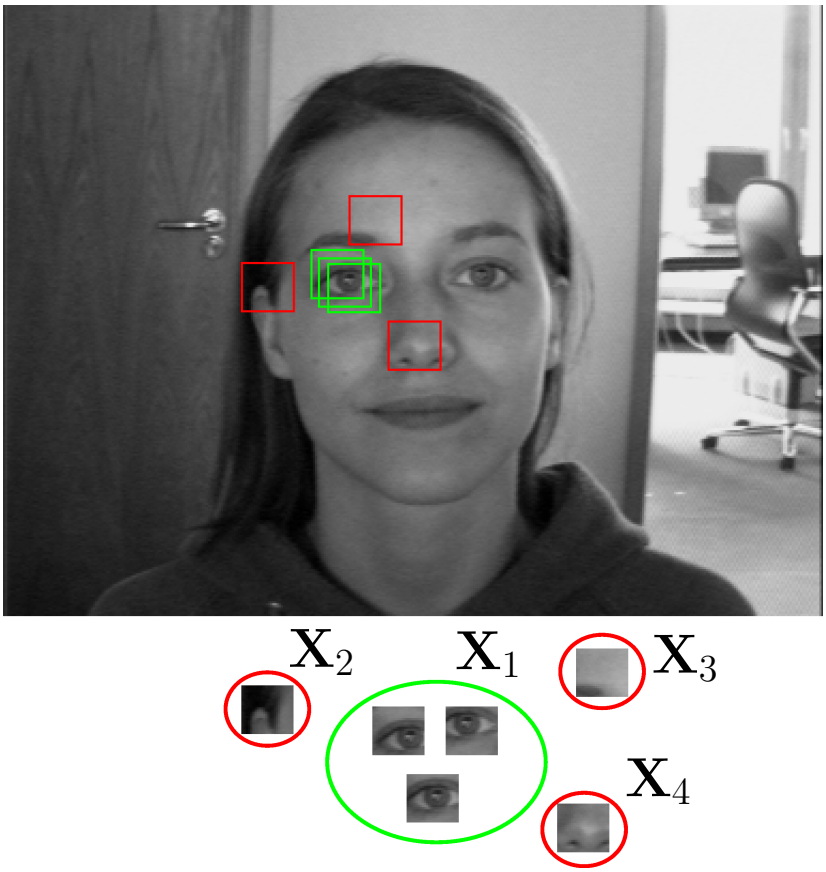


Figura 2.1: Exemplo de *bags* para um detector de olhos. Blocos vermelhos correspondem à instâncias negativas; blocos verdes à instâncias positivas. Abaixo, instâncias separadas pelos seus respectivos *bags*: $\mathbf{X}_1$ é um *bag* positivo e, $\mathbf{X}_2$, $\mathbf{X}_3$ e $\mathbf{X}_4$ são *bags* negativos.

à subjetividade da tarefa. Essa ambiguidade pode acarretar problemas durante o treinamento do classificador, podendo ser modelada como um ruído inerente.

Como exemplo, considerando o problema de classificação de objetos, é razoável assumir que o operador, apesar das marcações ambíguas, está próximo do correto. Podemos então transmitir esta ambiguidade inerente do problema para o classificador, informando ao mesmo que na região marcada existe ao menos uma amostra positiva, no caso positivo, ou, que todas são negativas, no caso negativo. As amostras organizadas desta forma são denominadas *instâncias* e não possuem rótulos individuais, mas compartilham o rótulo do conjunto que as contém, denominado "*bag*" (Figura 2.1).

O rótulo do *bag* é definido da seguinte forma: caso exista ao menos uma amostra positiva, o conjunto é rotulado como positivo. Senão, recebe rótulo negativo. Desta forma, dado um conjunto de amostras $\{\mathbf{x}_{ij}\}$ reunidas em um *bag* $\mathbf{X}_i$, que possui

rótulo y_i, o classificador ideal $g(\mathbf{X}_i)$ pode ser descrito como

$$g(\mathbf{X}_i) = \max_{\mathbf{x}_{ij} \in \mathbf{X}_i} [f(\mathbf{x}_{ij})] = y_i, \quad \forall i,$$

onde se $\mathbf{x} = \{x_1, x_2, \ldots, x_N\}$, $\max(\mathbf{x}) = x_k$ tal que $x_k \geq x_j \forall j$, e o rótulo y_i é igual a 1 para o caso positivo e 0 no caso contrário. Este método de treinamento é conhecido como MIL – *Multiple Instance Learning* [16, 17].

Nesta seção são propostas extensões do classificador IPD dentro do paradigma de aprendizado MIL. Sendo $\mathbf{h}$ o detector, podemos definir o problema de classificação dos *bags* como um problema de mínimos quadrados na forma

$$\min_{\mathbf{h}} \varepsilon(\mathbf{h}) = \sum_{i=1}^{N_B} \left[\max_{\mathbf{x}_{ij} \in \mathbf{X}_i} (\mathbf{h}^T \mathbf{x}_{ij}) - y_i \right]^2, \tag{2.28}$$

onde N_B é o número de *bags* de treinamento. Esta função não é diferenciável devido ao operador $\max(\cdot)$. Para tornar o problema diferenciável, definimos uma variável escalar auxiliar t_i tal que

$$t_i \geq \max_{\mathbf{x}_{ij} \in \mathbf{X}_i} (\mathbf{h}^T \mathbf{x}_{ij}) - y_i. \tag{2.29}$$

Dado que

$$\max_{\mathbf{x}_{ij} \in \mathbf{X}_i} (\mathbf{h}^T \mathbf{x}_{ij}) - y_i = \max_{\mathbf{x}_{ij} \in \mathbf{X}_i} (\mathbf{h}^T \mathbf{x}_{ij} - y_i),$$

podemos escrever a Equação (2.28) como um problema de otimização com restrições da forma

$$\begin{aligned} \min_{\mathbf{h}} \varepsilon(\mathbf{h}) &= \sum_{i=1}^{N_B} t_i^2, \\ \text{s.t} \quad t_i &\geq \mathbf{h}^T \mathbf{x}_{ij} - y_i, \forall j \in \mathbf{X}_i, \\ j &= \{1, \ldots, N_i\}. \end{aligned} \tag{2.30}$$

Existem múltiplas formas de solução deste problema. Abaixo são apresentadas as abordagens propostas.

Solução Heurística

Esta primeira proposta de abordagem é um algoritmo iterativo produzido de forma heurística a partir das características do problema, solução inspirada em método semelhante utilizado para resolver o problema de otimização de classificadores SVM com restrições MIL [16].

Seja $\mathbf{h}(n)$ o detector gerado na iteração n. Nesta iteração as amostras $\mathbf{x}_{iM(i)}$, onde $M(i)$ é o índice da amostra com máximo produto interno pertencente ao i-ésimo

bag, são definidas como:

$$\mathbf{x}_{iM(i)}(n) = \arg\max_{\mathbf{x}_{ij}}(\mathbf{x}_{ij}^T \mathbf{h}(n)), \tag{2.31}$$

e as variáveis $t_i(n)$ são computadas a partir da Equação (2.29) como

$$t_i(n) = \mathbf{x}_{iM(i)}^T(n)\mathbf{h}(n) - y_i, \quad \forall i. \tag{2.32}$$

Definido o valor de $t_i(n)$, encontramos o classificador $\mathbf{h}(n+1)$ minimizando a função custo erro

$$\begin{aligned} \min_{\mathbf{h}} \varepsilon(\mathbf{h}) &= \sum_{i=1}^{N_B} t_i^2, \\ &= \sum_{i=1}^{N_B} (\mathbf{x}_{iM(i)}^T(n)\mathbf{h}(n+1) - y_i)^2. \end{aligned} \tag{2.33}$$

Obtém-se solução semelhante à Equação (2.21):

$$\mathbf{h}(n+1) = \left(\sum_{i=1}^{N_B} \mathbf{x}_{iM(i)}(n)\mathbf{x}_{iM(i)}^T(n) \right)^{-1} \left(\sum_{i=1}^{N_B} y_i \mathbf{x}_{iM(i)}(n) \right). \tag{2.34}$$

Este processo é iterado até que os índices $M(i)$ em todos os *bags* permaneçam constantes. Entretanto, essa solução não tem garantias de convergência. O seu pseudocódigo está descrito no Algoritmo(2.1).

Algoritmo 2.1 Iterative Multiple Instance Learning Inner Product Detector

$\mathbf{h}(0) = \mathbf{mean}(y_i \mathbf{x_{ij}})$
for $n = \{1, ..., N_{\max}\}$ **do**
  for $i = \{1, ..., N_B\}$ **do**
    $\mathbf{x}_{iM(i)}(n) = \arg\max_{\mathbf{x}_{ij}}(\mathbf{x}_{ij}^T \mathbf{h}(n)), \quad j = \{1, \ldots, N_i\}$
    $t_i(n) = \mathbf{x}_{iM(i)}^T(n)\mathbf{h}(n) - y_i$
  end for
  if $[\mathbf{x}_{iM(i)}(n) \equiv \mathbf{x}_{iM(i)}(n-1), \forall i]$ **then**
    return $\mathbf{h}(n)$
  end if
  $\mathbf{R_{xx}}(n) = \sum_{i=1}^{N_B} \mathbf{x}_{iM(i)}(n)\mathbf{x}_{iM(i)}^T(n)$
  $\mathbf{r}_{y\mathbf{x}}(n) = \sum_{i=1}^{N_B} y_i \mathbf{x}_{iM(i)}(n)$
  $\mathbf{h}(n+1) = \mathbf{R}_{\mathbf{xx}}^{-1}(n)\mathbf{r}_{y\mathbf{x}}(n)$
end for
return $\mathbf{h}(n)$

Aproximação Softmax

Esta outra abordagem desenvolvida utiliza-se de aproximações diferenciáveis da função *max*, denominadas *softmax* [17]. Uma aproximação *softmax* é uma função $g(\mathbf{z})$ tal que

$$g(\mathbf{z}) \approx \max_l(z_l) = z_* \tag{2.35}$$

$$\frac{\partial g(\mathbf{z})}{\partial z_i} \approx \frac{u(z_i - z_*)}{\sum_l u(z_l - z_*)} \tag{2.36}$$

onde $u(\cdot)$ é a função degrau unitário.

No caso em que z_i seja o único valor máximo em $\mathbf{z} = (z_1, \ldots, z_d)^T$, ou seja, $z_i = z_*$, alterações em z_i causam alterações no máximo na mesma proporção. Senão, mudanças em z_i não alteram o máximo, e a derivada não é afetada.

Dentre do conjunto de aproximações propostas para o máximo, neste trabalho será utilizada uma variante da função *log-sum-exponential* (logaritmo-somatório-exponencial) conhecida como LSE [17, 23]. Essa escolha foi feita devido ao domínio dessa função conter todo o $\mathbb{R}^n$, da mesma forma que a saída do IPD. Para um sumário incluindo outras possíveis aproximações vide [17].

Escolhida a aproximação para a função *max*, a função LSE $g(z_l)$ e sua derivada são descritas, respectivamente, nas Equações (2.37) e (2.38):

$$g(\mathbf{z}) = \frac{1}{r} \ln\left(\frac{1}{d}\sum_l \mathrm{e}^{(rz_l)}\right), \tag{2.37}$$

$$\frac{\partial g(\mathbf{z})}{\partial z_i} = \frac{\mathrm{e}^{(rz_i)}}{\sum_l \mathrm{e}^{(rz_l)}}, \tag{2.38}$$

onde d é a dimensão do vetor e r é um fator que define a precisão da aproximação: $g(\mathbf{z}) \to z_*$ quando $r \to \infty$. Todavia, valores altos de r podem causar instabilidade numérica [17].

Dada a aproximação *softmax* LSE, podemos reescrever a Equação (2.28) como:

$$\min_{\mathbf{h}} \hat{\varepsilon}(\mathbf{h}) = \sum_{i=1}^{N_B} \left[g\left(\mathbf{X}_i^T\mathbf{h}\right) - y_i\right]^2, \tag{2.39}$$

onde $\mathbf{X}_i = (\mathbf{x}_{i1}, \ldots, \mathbf{x}_{iN_i})$ é o i-ésimo *bag* composto por N_i amostras.

Diferenciando a função custo $\hat{\varepsilon}(\mathbf{h})$ em $\mathbf{h}$ encontramos o gradiente $\Delta\mathbf{h}$

$$\Delta\mathbf{h} = \frac{\partial\hat{\varepsilon}(\mathbf{h})}{\partial\mathbf{h}} = 2\sum_{i=1}^{N_B}\left[g\left(\mathbf{X}_i^T\mathbf{h}\right) - y_i\right]\frac{\partial g\left(\mathbf{X}_i^T\mathbf{h}\right)}{\partial\mathbf{h}}, \tag{2.40}$$

onde a derivada da função *softmax* em função do detector $\mathbf{h}$ é

$$\frac{\partial g\left(\mathbf{X}_i^T\mathbf{h}\right)}{\partial \mathbf{h}} = \frac{\partial\left(\mathbf{X}_i^T\mathbf{h}\right)}{\partial \mathbf{h}} \frac{\partial g\left(\mathbf{X}_i^T\mathbf{h}\right)}{\partial\left(\mathbf{X}_i^T\mathbf{h}\right)}, \tag{2.41}$$

onde

$$\frac{\partial\left(\mathbf{X}_i^T\mathbf{h}\right)}{\partial \mathbf{h}} = \mathbf{X}_i, \tag{2.42}$$

e, assim,

$$\frac{\partial g\left(\mathbf{X}_i^T\mathbf{h}\right)}{\partial\left(\mathbf{X}_i^T\mathbf{h}\right)} = \begin{bmatrix} \mathrm{e}^{\left(r\mathbf{h}^T\mathbf{x}_{i1}\right)} \\ \mathrm{e}^{\left(r\mathbf{h}^T\mathbf{x}_{i2}\right)} \\ \vdots \\ \mathrm{e}^{\left(r\mathbf{h}^T\mathbf{x}_{iN_1}\right)} \end{bmatrix} \frac{1}{\sum_j \mathrm{e}^{(r\mathbf{h}^T\mathbf{x}_{ij})}}. \tag{2.43}$$

Substituindo este resultado na Equação (2.40), obtemos

$$\Delta\mathbf{h} = 2\sum_{i=1}^{N_B}\left[\frac{1}{r}\ln\left(\frac{1}{N_i}\sum_j^{N_i}\mathrm{e}^{\left(r\mathbf{h}^T\mathbf{x}_{ij}\right)}\right) - y_i\right]\frac{\sum_l^{N_i}\mathbf{x}_{il}\mathrm{e}^{\left(r\mathbf{h}^T\mathbf{x}_{il}\right)}}{\sum_k^{N_i}\mathrm{e}^{(r\mathbf{h}^T\mathbf{x}_{ik})}}, \tag{2.44}$$

onde N_i é o número de amostras no i-ésimo *bag*.

Podemos então aplicar uma estratégia gradiente descendente [24], neste problema, atualizando o detector na iteração $n+1$ como:

$$\mathbf{h}(n+1) = \mathbf{h}(n) - \mu_n\Delta\mathbf{h}(n), \tag{2.45}$$

onde μ_n é o passo de atualização.

O valor de μ_n em cada iteração pode ser obtido através de uma busca em linha. Seja a aproximação em série de Taylor de $\hat{\varepsilon}(\mathbf{h})$

$$\hat{\varepsilon}(\mathbf{h}(n)-\mu_n\Delta\mathbf{h}(n)) \approx \hat{\varepsilon}(\mathbf{h}(n))-\mu_n\Delta\mathbf{h}^T(n)\Delta\mathbf{h}(n)+\frac{1}{2}\mu_n^2\Delta\mathbf{h}^T(n)\mathbf{H}_\mathbf{h}(n)\Delta\mathbf{h}(n), \tag{2.46}$$

onde $\mathbf{H}_\mathbf{h}(n)$ é a matriz Hessiana definida como [24]

$$\mathbf{H}_\mathbf{h}(n) = \frac{\partial}{\partial\mathbf{h}(n)}\left(\frac{\partial\hat{\varepsilon}\left[\mathbf{h}(n)\right]}{\partial\mathbf{h}(n)}\right) \tag{2.47}$$

Diferenciando a Equação (2.46) e igualando o resultado a zero, obtemos

$$\frac{\partial\hat{\varepsilon}(\mathbf{h}(n)-\mu_n\Delta\mathbf{h}(n))}{\partial\mu_n} \approx -\Delta\mathbf{h}^T(n)\Delta\mathbf{h}(n) + \mu_n\Delta\mathbf{h}^T(n)\mathbf{H}_\mathbf{h}(n)\Delta\mathbf{h}(n) = 0 \tag{2.48}$$

ou,

$$\mu_n = \frac{\Delta\mathbf{h}^T(n)\Delta\mathbf{h}(n)}{\Delta\mathbf{h}^T(n)\mathbf{H}_\mathbf{h}(n)\Delta\mathbf{h}(n)}. \tag{2.49}$$

Desta forma é possível obter o passo ótimo μ_n que minimiza $\hat{\varepsilon}(\mathbf{h}(n) - \mu_n \Delta\mathbf{h}(n))$.

Contudo, a avaliação da função $\hat{\varepsilon}(\mathbf{h})$ pode ser bastante custosa, devido à quantidade de amostras, a dimensionalidade do problema e o cálculo da Hessiana. Optou-se assim por um algoritmo de gradiente descendente sem busca em linha [24], onde o passo obtido depende do valor ótimo de μ_{n-1} obtido na iteração anterior.

$$\mu_n \approx \frac{\mu_{n-1}^2 \|\Delta\mathbf{h}(n)\|_2^2}{2\left(\hat{\varepsilon}' - \hat{\varepsilon}[\mathbf{h}(n)] + \mu_{n-1}\|\Delta\mathbf{h}(n)\|_2^2\right)}, \tag{2.50}$$

e

$$\hat{\varepsilon}' = \hat{\varepsilon}\left[\mathbf{h}(n) - \mu_{n-1}\Delta\mathbf{h}(n)\right] \tag{2.51}$$

É interessante observar que, para $N_i = 1$, o gradiente obtido na Equação (2.44) assume forma idêntica à da formulação do IPD original:

$$\frac{\partial\hat{\varepsilon}(\mathbf{h})}{\partial\mathbf{h}} = \sum_{i=1}^{N_B} 2\mathbf{x}_{i1}\mathbf{x}_{i1}^T\mathbf{h} - 2y_i\mathbf{x}_{i1}. \tag{2.52}$$

O processo de obtenção dos classificadores utilizando a aproximação *softmax* é apresentado no Algoritmo 2.2.

Algoritmo 2.2 Softmax Multiple Instance Learning Inner Product Detector

$\mathbf{h}(0) = \mathbf{mean}(y_i\mathbf{x_{ij}})$, $\mu_0 = 1$
for $n = \{1, ..., N_{\max}\}$ **do**
  $\hat{\varepsilon}\left[\mathbf{h}(n)\right] = \sum_{i=1}^{N_B}\left[g_j\left(\mathbf{x}_{ij}^T\mathbf{h}(n)\right) - y_i\right]^2$
  $\Delta\mathbf{h}(n) = \frac{\partial\hat{\varepsilon}[\mathbf{h}(n)]}{\partial\mathbf{h}(n)} = 2\sum_{i=1}^{N_B}\left[g_j\left(\mathbf{x}_{ij}^T\mathbf{h}(n)\right) - y_i\right]\frac{\partial g_j\left(\mathbf{x}_{ij}^T\mathbf{h}(n)\right)}{\partial\mathbf{h}}$
  $\hat{\varepsilon}' = \hat{\varepsilon}\left[\mathbf{h}(n) - \mu_{n-1}\Delta\mathbf{h}(n)\right]$
  $\mu_n = \frac{\mu_{n-1}^2\|\Delta\mathbf{h}(n)\|_2^2}{2\left(\hat{\varepsilon}' - \hat{\varepsilon}[\mathbf{h}(n)] + \mu_{n-1}\|\Delta\mathbf{h}(n)\|_2^2\right)}$
  $\mathbf{h}(n+1) = \mathbf{h}(n) - \mu_n\Delta\mathbf{h}(n)$
  if $\|\mu_n\Delta\mathbf{h}(n)\|_2^2 < \delta_{thrs}$ **then**
    return $\mathbf{h}(n+1)$
  end if
end for
return $\mathbf{h}(n)$

Programação Semidefinida

Definindo o vetor $\mathbf{t} = (t_1, ..., t_{N_B})^T$, podemos reescrever o problema de minimização como

$$\begin{aligned} \min_{\mathbf{h}} \varepsilon(\mathbf{h}) &= \sum_{i=1}^{N_B} t_i^2 = \mathbf{t}^T\mathbf{t} = \mathrm{tr}(\mathbf{t}\mathbf{t}^T), \\ \text{s.t} \quad t_i &\geq \mathbf{h}^T\mathbf{x}_{ij} - y_i, \forall j \in \mathbf{X}_i, \\ j &= \{1, \ldots, N_i\} \end{aligned} \tag{2.53}$$

As restrições em t_i podem ser reescritas como

$$\mathbf{t} \succeq \begin{bmatrix} \max(\mathbf{X}_1^T\mathbf{h} - y_1\mathbf{1}) \\ \vdots \\ \max(\mathbf{X}_{N_B}^T\mathbf{h} - y_{N_B}\mathbf{1}) \end{bmatrix}. \tag{2.54}$$

Restringindo todos as *bags* a possuírem o mesmo número de amostras ($N_i = N_s, \forall i$), e definindo o vetor de rótulos $\mathbf{y} = (y_1, \ldots, y_{N_B})^T$, podemos transformar as restrições t_i em restrições lineares através do produto de Kronecker

$$\mathbf{t} \otimes \mathbf{1}_{N_s} + \mathbf{y} \otimes \mathbf{1}_{N_s} \succeq \begin{bmatrix} \mathbf{h}^T\mathbf{x}_{11} \\ \mathbf{h}^T\mathbf{x}_{12} \\ \vdots \\ \mathbf{h}^T\mathbf{x}_{1N_s} \\ \mathbf{h}^T\mathbf{x}_{21} \\ \vdots \\ \mathbf{h}^T\mathbf{x}_{N_BN_s-1} \\ \mathbf{h}^T\mathbf{x}_{N_BN_s} \end{bmatrix} = \begin{bmatrix} \mathbf{x}_{11}^T \\ \mathbf{x}_{12}^T \\ \vdots \\ \mathbf{x}_{1N_s}^T \\ \mathbf{x}_{21}^T \\ \vdots \\ \mathbf{x}_{N_BN_s-1}^T \\ \mathbf{x}_{N_BN_s}^T \end{bmatrix} \mathbf{h} = \mathbf{X}^T\mathbf{h}, \tag{2.55}$$

onde $\mathbf{1}_{N_s}$ é um vetor de 1 com dimensões $N_s \times 1$ e $\mathbf{X} = (\mathbf{x}_{11}^T, \ldots \mathbf{x}_{N_BN_s}^T)^T$, o conjunto de amostras de treinamento.

Realizando alguns algebrismos na Equação (2.55), podemos escrever as restrições como

$$(\mathbf{t} - \mathbf{y}) \otimes \mathbf{1}_{N_s} - \mathbf{X}^T\mathbf{h} \succeq \mathbf{0}.$$

Através do operador diag$(\cdot)$, podemos transformar o vetor de restrições em uma matriz diagonal onde os elementos da diagonal correspondem aos elementos do vetor. As restrições podem ser reescritas como

$$\text{diag}\left[(\mathbf{t} - \mathbf{y}) \otimes \mathbf{1}_{N_s} - \mathbf{X}^T\mathbf{h}\right] \geq 0, \tag{2.56}$$

ou seja, as restrições são tais que a matriz obtida deve ser semipositiva definida. Desta forma podemos reescrever o problema de otimização como

$$\begin{aligned} \min_{\mathbf{h},\mathbf{t}} \varepsilon(\mathbf{h}, \mathbf{t}) &= \text{tr}(\mathbf{t}\mathbf{t}^T) \\ \text{s.t. } &\text{diag}\left[(\mathbf{t} - \mathbf{y}) \otimes \mathbf{1}_{N_s} - \mathbf{X}^T\mathbf{h}\right] \geq 0. \end{aligned} \tag{2.57}$$

Esse problema é um problema de programação semidefinida (*semidefinite programming* – SDP) [24] que pode ser resolvido utilizando pacotes de otimização conhecidos.

2.2.4 Boost IPD

Boosting é uma técnica de combinação de classificadores com o objetivo gerar um *classificador forte* a partir de um conjunto de *classificadores fracos* [25]. Dentre as variações desta técnica, escolheu-se neste trabalho a denominada *Discrete AdaBoost* [25, 26].

O algoritmo *Discrete AdaBoost* gera um conjunto de hipóteses a partir de classificadores fracos atualizando a distribuição de pesos das amostras iterativamente. As atualizações são tais que as amostras classificadas incorretamente sejam mais prováveis no treinamento do próximo classificador (Figura 2.2). Desta forma, o conjunto de treinamento é direcionado para amostras de difícil classificação.

O primeiro passo é definir uma distribuição inicial D_1 para os pesos ao longo do conjunto de treinamento. Normalmente é escolhida uma distribuição uniforme para todas as amostras. Dada a distribuição inicial, cada hipótese $\mathbf{h}_{nk}$ é treinada segundo seu algoritmo de treinamento, $\mathbf{LearnAlgo}(\mathbf{h}_{nk})$, sobre esta distribuição. No caso dos classificadores IPD, o algoritmo de treinamento $\mathbf{LearnAlgo}(\mathbf{h}_{nk})$ é equivalente ao processo de treinamento com pesos (Seção 2.2.1), onde $w_i = D_n(i)$:

$$\mathbf{h}_{nk} = \left(\sum_{i=1}^{N_s} D_n(i) \mathbf{x}_{ki} \mathbf{x}_{ki}^T \right)^{-1} \left(\sum_{i=1}^{N_s} D_n(i) y_i \mathbf{x}_{ki} \right), \tag{2.58}$$

onde n é o índice da iteração de *boosting* e k o índice da hipótese, onde cada hipótese pode corresponder a classificadores com características distintas, como tipo, dimensão ou formato das amostras.

Para cada classificador é calculado o erro de classificação ϵ_{nk} correspondente, que é apenas a soma dos pesos das amostras classificadas incorretamente pelo classificador. Nesta etapa é escolhida a hipótese que minimiza o erro de classificação, ou seja, o classificador $\mathbf{h}_n$ tal que

$$\mathbf{h}_n = \arg\min_{\mathbf{h}_{nk}} \epsilon_{nk}(\mathbf{h}_{nk}), \quad k = \{1, \ldots, K\}$$

onde K é o número de hipóteses. É requerido que $\epsilon_n \leq 0.5$, satisfazendo a restrição de *classificadores fracos*, onde

$$\epsilon_n = \min_{\mathbf{h}_{nk}} \epsilon_{nk}(\mathbf{h}_{nk}).$$

Caso contrário, o processo é interrompido, retornando a última hipótese válida.

Com a condição satisfeita, a distribuição é atualizada tal que na nova distribuição D_2 as amostras classificadas corretamente tenham seu peso diminuído e as

classificadas incorretamente tenham seu peso aumentado, conforme

$$D_{n+1}(i) = \frac{D_n(i)}{Z_n} \times \begin{cases} \mathrm{e}^{-\alpha_n} & \text{se } \mathbf{h}_n(\mathbf{x}_i) = y_i \\ \mathrm{e}^{\alpha_n} & \text{se } \mathbf{h}_n(\mathbf{x}_i) \neq y_i, \end{cases} \tag{2.59}$$

onde $\alpha_n = \log\left(\frac{1-\epsilon_n}{\epsilon_n}\right)$ e Z_n é um fator de normalização tal que $\sum_i D_{n+1}(i) = 1$.

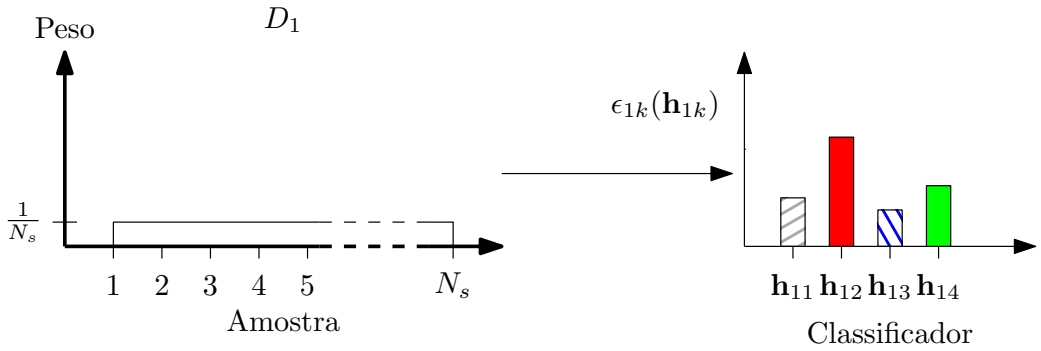
Definida a nova distribuição, o processo é repetido, como ilustrado na Figura 2.2. Ao fim do processo de treinamento, os detectores são reunidos de forma a gerar um detector forte através de combinação linear ponderada pelos respectivos α_n

$$\mathbf{h} = \sum_n \alpha_n \mathbf{h}_n \tag{2.60}$$

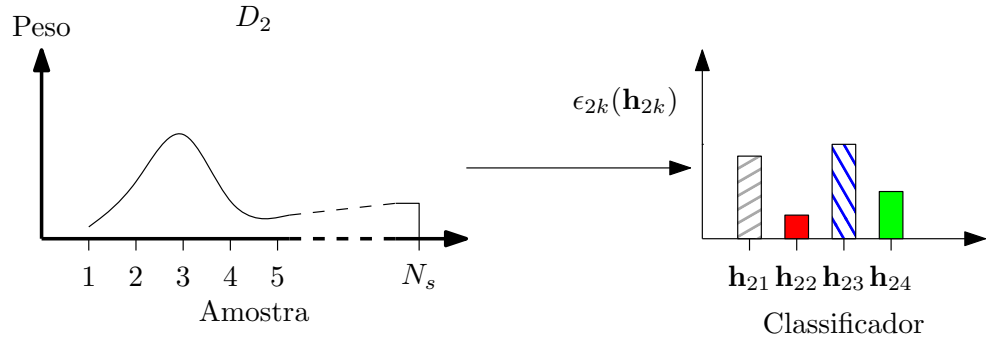
Neste trabalho, cada classificador IPD $\mathbf{h}_k$ pode possuir amostras com características distintas. O algoritmo *AdaBoost* permite selecionar dentre o conjunto de classificadores aquele que minimiza o erro de classificação ϵ_k, servindo também como processo de seleção de características. No entanto, para evitar aumento na complexidade devido a necessidade de computar conjuntos distintos de amostras, treinou-se apenas um detector IPD por iteração. Este processo é descrito no Algoritmo 2.3.

Algoritmo 2.3 AdaBoost Inner Product Detector

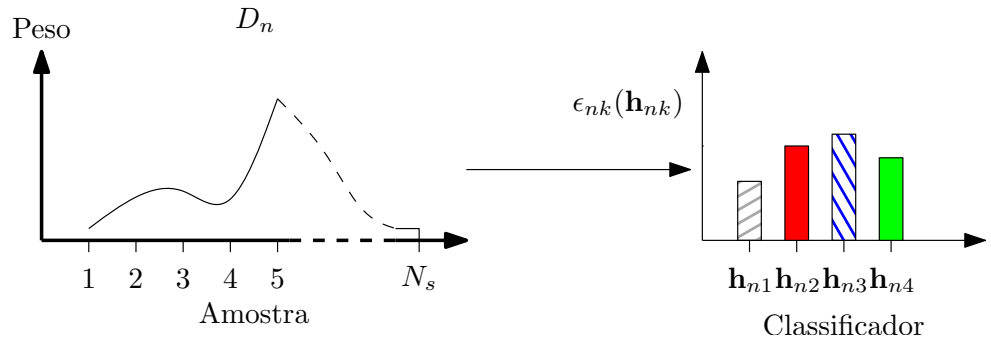
$\mathbf{h} = \mathbf{0}$
$D_1(i) = 1/N_s$ for all $i \in \{1, \ldots, N_s\}$
for n **do**
  for $k = \{1, \ldots, K\}$ **do**
    $\mathbf{h}_{nk} = \left(\sum_{i=1}^{N_s} D_n(i)\mathbf{x}_{ki}\mathbf{x}_{ki}^T\right)^{-1}\left(\sum_{i=1}^{N_s} D_n(i) y_i \mathbf{x}_{ki}\right)$
    $\epsilon_{nk} = \sum_{i:\mathbf{h}_{nk}(\mathbf{x}_{ki}) \neq y_i} D_n(i)$
  end for
  $\epsilon_n = \min \epsilon_{nk}, \quad k = \{1, \ldots, K\}$
  if $\epsilon_n \geq 1/2$ **then**
    return h
  else
    $\mathbf{h}_n = \arg\min_{\mathbf{h}_{nk}} \epsilon_{nk}, \quad k = \{1, \ldots, K\}$
    $\alpha_n = \log\left(\frac{1-\epsilon_n}{\epsilon_n}\right)$
    $D_{n+1}(i) = \frac{D_n(i)}{Z_n} \times \begin{cases} \mathrm{e}^{-\alpha_n} & \text{se } \mathbf{h}_n(\mathbf{x}_i) = y_i \\ \mathrm{e}^{\alpha_n} & \text{se } \mathbf{h}_n(\mathbf{x}_i) \neq y_i, \end{cases}$
    $\mathbf{h} = \mathbf{h} + \alpha_n \mathbf{h}_n$
  end if
end for
return h



(a) Distribuição inicial D_1



(b) Distribuição D_2



(c) Distribuição D_n

Figura 2.2: Exemplo de processo de *Boosting* com $K = 4$. Na Figura 2.2a são inicializados os pesos uniformemente, treinados os classificadores $\mathbf{h}_{11}$, $\mathbf{h}_{12}$, $\mathbf{h}_{13}$ e $\mathbf{h}_{14}$, e computados os respectivos erros de classificação. Neste caso, dado que $\mathbf{h}_{13}$ obteve o menor erro, $\mathbf{h}_1 = \mathbf{h}_{13}$. Na Figura 2.2b é apresentada a nova distribuição de pesos D_2, atualizada segundo a Equação (2.59). Nesta etapa, $\mathbf{h}_2 = \mathbf{h}_{22}$, detector com o menor erro de classificação, que é utilizado para atualizar os pesos. O processo é repetido sucessivamente (Figura 2.2c) até que $\epsilon_n \geq 1/2$ ou o número de iterações máximo seja satisfeito.

2.2.5 Bagging IPD

B**ootstrap **Agg**regat**ing, conhecido como *Bagging* [27], é um dos métodos de combinação de classificadores mais antigo e simples [28]. Essa técnica gera M conjuntos de treinamento $\mathcal{X}_m$ sorteando, com reposição, N_s amostras do conjunto de treinamento original $\mathcal{X}_T$. Cada subconjunto é utilizado para produzir classificadores distintos. O conjunto de classificadores gerados é então agregado para classificação, onde uma amostra de teste é classificada como pertencente a classe que receba a maioria dos votos do conjunto.

Dado que o sorteio é realizado com reposição, é possível que uma amostra de treinamento seja selecionada múltiplas vezes, enquanto algumas amostras não serão sorteadas. Desta forma, apesar da reposição, é provável que os conjuntos gerados sejam distintos entre si [29]. Se os modelos utilizados são *instáveis*, ou seja, diferenças no conjunto de treinamento induzem diferenças significativas na estrutura de decisão, o classificador pode ser beneficiado pelo processo de *bagging* [27].

O processo de geração de detectores IPD utilizando *bagging* é descrito no Algoritmo 2.4.

Algoritmo 2.4 Bagging Inner Product Detector

for $m = \{1, \ldots, M\}$ **do**
  $\mathcal{X}_m \sim \mathcal{X}_T$
  $\mathbf{R}_{\mathbf{xx}}(m) = \sum_{i=1}^{N_s} \mathbf{x}_i(m)\mathbf{x}_i^T(m)$
  $\mathbf{r}_{y\mathbf{x}}(m) = \sum_{i=1}^{N_s} y_i\mathbf{x}_i(m)$
  $\mathbf{h}(m) = \mathbf{R}_{\mathbf{xx}}^{-1}(m)\mathbf{r}_{y\mathbf{x}}(m)$
end for
return $\mathbf{h} = \frac{1}{M}\sum_{m=1}^{M}\mathbf{h}(m)$

2.2.6 Online IPD

Aprendizado *online* é uma modalidade de treinamento onde as instâncias são apresentadas e participam do processo de aprendizado de forma sequencial [29, 30]. Desta forma não é necessário que todo o conjunto de treinamento esteja disponível durante o processo de aprendizado, reduzindo custos computacionais de memória e até mesmo de processamento [29, 30]. Redes neurais, classificadores *Naive Bayes*, dentre outros, são exemplos de classificadores que permitem essa metodologia de treinamento [29, 31]. Nesta seção será apresentada uma proposta desenvolvida para a modalidade o treino *online* de classificadores IPD.

Observa-se que os métodos iterativos MIL apresentados na Seção 2.2.3 já permitem, com pequenas adaptações, o treinamento de classificadores *online*, dado que é possível atualizar o classificador para cada *bag* individualmente. Todavia, esses métodos não são extensíveis às outras metodologias de treinamento apresentadas.

Dada a formulação original do IPD, descrita na Equação (2.21), com k amostras, apresentada a $(k+1)$-ésima instância, $\mathbf{x}_{k+1}$, pode-se atualizar a inversa da matriz de correlação, $\hat{\mathbf{R}}_{\mathbf{xx}}^{-1}(k)$, através da identidade Sherman-Morrison [32]

$$\hat{\mathbf{R}}_{\mathbf{xx}}^{-1}(k+1) = (\hat{\mathbf{R}}_{\mathbf{xx}}(k) + \mathbf{x}_{k+1}\mathbf{x}_{k+1}^T)^{-1} = \hat{\mathbf{R}}_{\mathbf{xx}}^{-1}(k) + \frac{\hat{\mathbf{R}}_{\mathbf{xx}}^{-1}(k)\mathbf{x}_{k+1}\mathbf{x}_{k+1}^T\hat{\mathbf{R}}_{\mathbf{xx}}^{-1}(k)}{1 + \mathbf{x}_{k+1}^T\hat{\mathbf{R}}_{\mathbf{xx}}^{-1}(k)\mathbf{x}_{k+1}}, \quad (2.61)$$

A atualização da média $\hat{\mathbf{m}}_{\mathcal{X}_P}(k)$ pode ser realizada de forma simples

$$\hat{\mathbf{m}}_{\mathcal{X}_P}(k+1) = \hat{\mathbf{m}}_{\mathcal{X}_P}(k) + y_{k+1}\mathbf{x}_{k+1}. \quad (2.62)$$

Assim, o novo detector obtido por este processo é:

$$\mathbf{h}_{k+1} = \hat{\mathbf{R}}_{\mathbf{xx}}^{-1}(k+1)\hat{\mathbf{m}}_{\mathcal{X}_P}(k+1), \quad (2.63)$$

similar ao encontrada na Equação 2.21.

Este processo é facilmente aplicável às abordagens adotadas neste trabalho, permitindo o treinamento *online* dos classificadores. Contudo, podem ocorrer problemas numéricos devido ao mal-condicionamento da matriz de correlação $\hat{\mathbf{R}}_{\mathbf{xx}}(k)$. Para mitigar possíveis problemas, um termo de regularização pode ser adicionado, tal que

$$\hat{\mathbf{R}}_{\mathbf{xx}}(0) = \lambda I,$$

Onde λ é um escalar. Assim, em $k+1$, temos

$$\hat{\mathbf{R}}_{\mathbf{xx}}^{-1}(k+1) = (\hat{\mathbf{R}}_{\mathbf{xx}}(k) + \mathbf{x}_{k+1}\mathbf{x}_{k+1}^T)^{-1} = (\lambda I + \sum_{i=1}^{k+1}\mathbf{x}_i\mathbf{x}_i^T)^{-1}.$$

2.3 Filtros de Correlação

Em um problema de classificação o objetivo final é identificar o padrão (ou padrões) desejado(s) dentro de um conjunto de medidas. Uma forma muito simples e comum de realizar esta detecção é correlacionar a amostra com um exemplo ou modelo conhecido do padrão [3]. O reconhecimento é feito através da correlação cruzada entre a amostra e o sinal de referência. No caso de sinais discretos, temos:

$$c[k] = (r * x)[k], \quad (2.64)$$

onde $r[k]$ é o sinal de referência; $x[k]$ o sinal a ser avaliado; $*$ o operador convolução e $c[k]$ é o sinal resultante. Se a amostra $x[k]$ contiver uma versão deslocada e "rebatida" $r[k-k_0]$ da referência, a correlação exibirá um pico em $k = k_0$ [33].

Se a convolução for circular, esta operação também pode ser descrita na forma

matricial como:

$$\mathbf{c} = C(\mathbf{r})\mathbf{x}, \tag{2.65}$$

onde $C(\cdot)$ é o operador circulante que leva um vetor $N \times 1$ em uma matriz circulante $N \times N$:

$$C(\mathbf{r}) = \begin{bmatrix} r_0 & r_1 & r_2 & \dots & r_{N-1} \\ r_{N-1} & r_0 & r_1 & \dots & r_{N-2} \\ r_{N-2} & r_{N-1} & r_0 & \dots & r_{N-3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ r_1 & r_2 & r_3 & \dots & r_0 \end{bmatrix}.$$

Contundo, muitas vezes a amostra a ser identificada está corrompida por algum tipo de ruído ou distorção, não contém o sinal de referência, ou contém múltiplas réplicas do mesmo. Com o objetivo de evitar respostas ambíguas desenvolveu-se os filtros de correlação [34, 35].

Filtros de correlação são sinais de referência, escolhidos dentre o conjunto de treinamento ou criados a partir de algum processo, que permitem determinar o grau de semelhança entre a amostra e um padrão a ser identificado ou discriminar um padrão de interesse de outros objetos semelhantes [33].

Em particular para o problema de detecção de padrões em imagens, o uso dos filtros de correlação oferece um conjunto de vantagens [3, 15, 36]:

- retornam simultaneamente localização e classe do alvo;
- permitem a detecção e/ou rastreamento de objetos complexos;
- são robustos a condições adversas como rotações, oclusões, mudanças de iluminação, etc.; e
- chegam a atingir 20 vezes a velocidade de detectores e/ou rastreadores estado da arte.

Tendo essas qualidades em mente, nesta seção serão apresentados os classificadores baseados em filtros de correlação utilizados neste trabalho.

2.3.1 Filtragem Discriminativa por Restauração

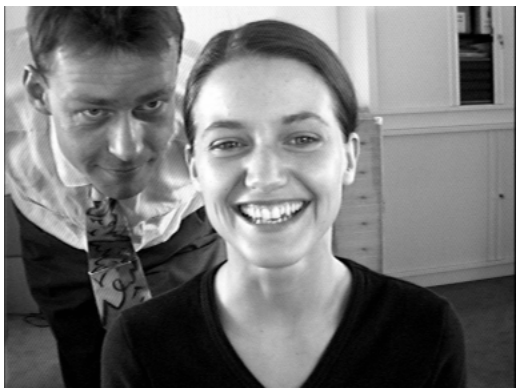
Seja um sinal $g[k]$ o sinal corrompido observado que contém o padrão desejado $x[k]$ tal que

$$g[k] = (x * v)[k] + w[k], \tag{2.66}$$

ou, assumindo convolução circular, na forma matricial

$$\mathbf{g} = C(\mathbf{x})\mathbf{v} + \mathbf{w}, \tag{2.67}$$

onde $\mathbf{w}$ é um ruído aditivo e $\mathbf{v}$ representa a resposta desejada a ser obtida pela filtro, análoga à função de dispersão pontual (*point-spread function*) ou à resposta ao impulso do canal. Um exemplo desta resposta, para um filtro detector de olho direito é apresentado na Figura 2.3. Nesta aplicação, ambos os elementos são conhecidos e, no caso mais simples, $\mathbf{v} = \boldsymbol{\delta}(k - k_0)$ e $\mathbf{w} = \mathcal{N}(0, \sigma_{\mathbf{w}})$.



(a) Imagem original

(b) Resposta desejada

(c) Imagens sobrepostas

Figura 2.3: Exemplo de resposta desejada para um detector de olho direito

O objetivo é encontrar a transformação linear $\mathbf{H}$ (filtro discriminativo) que recupere a posição do sinal $\mathbf{v}$ minimizando o erro quadrático. Dado o erro $\mathbf{e} = \mathbf{v} - \hat{\mathbf{v}}$, onde $\hat{\mathbf{v}} = \mathbf{Hg}$, e considerando amostras reais, pela formulação LMMSE, temos

$$\begin{aligned} \varepsilon(\mathbf{H}) &= \mathrm{E}\left[\mathbf{e}^T\mathbf{e}\right], \\ &= \mathrm{E}\left[\mathrm{tr}\left(\mathbf{e}\mathbf{e}^T\right)\right]. \end{aligned} \tag{2.68}$$

Sendo a função de erro convexa e diferenciável, minimizando-a em função de $\mathbf{h}$, temos

$$\nabla_{\mathbf{H}}\varepsilon = \mathbf{H}\mathrm{E}\left[\mathbf{g}\mathbf{g}^T\right] - \mathrm{E}\left[\mathbf{v}\mathbf{g}^T\right] = \mathbf{0}. \tag{2.69}$$

Solucionando para $\mathbf{H}$, temos

$$\mathbf{H} = \mathrm{E}\left[\mathbf{v}\mathbf{g}^T\right]\left(\mathrm{E}\left[\mathbf{g}\mathbf{g}^T\right]\right)^{-1}. \tag{2.70}$$

Calculando os valores esperados acima, temos

$$\begin{aligned} \mathrm{E}\left[\mathbf{v}\mathbf{g}^T\right] &= \mathrm{E}\left[\mathbf{v}\mathbf{v}^T\right] C(\mathbf{x})^T + \mathrm{E}\left[\mathbf{v}\mathbf{w}^T\right], \\ &= \mathbf{C}_\mathbf{v} C(\mathbf{x})^T, \end{aligned} \tag{2.71}$$

considerando $\mathbf{w}$ e $\mathbf{v}$ independentes. E para o segundo termo

$$\begin{aligned} \mathrm{E}\left[\mathbf{g}\mathbf{g}^T\right] &= \mathrm{E}\left[C(\mathbf{x})\mathbf{v}\mathbf{v}^T C(\mathbf{x})^T + \mathbf{w}\mathbf{w}^T + C(\mathbf{x})\mathbf{v}\mathbf{w}^T + \mathbf{w}\mathbf{v}^T C(\mathbf{x})^T\right], \\ &= C(\mathbf{x})\mathbf{C}_\mathbf{v} C(\mathbf{x})^T + \mathbf{C}_\mathbf{w}. \end{aligned} \tag{2.72}$$

Substituindo os resultados obtidos na Equação (2.70), o filtro discriminativo $\mathbf{H}$ pode ser reescrito como

$$\mathbf{H} = \left[\mathbf{C}_\mathbf{v} C(\mathbf{x})^T\right] \left[C(\mathbf{x})\mathbf{C}_\mathbf{v} C(\mathbf{x})^T + \mathbf{C}_\mathbf{w}\right]^{-1}. \tag{2.73}$$

O caso bidimensional é abordado da mesma forma utilizando vetorização, concatenando linhas ou colunas do sinal. Para o caso específico de uma imagem, podemos considerar o problema de obtenção do operador $\mathbf{H}$ como um problema de restauração da entrada impulso, onde a resposta possui máxima energia na posição do padrão.

Em [6] é demonstrada a relação existente entre o filtro discriminativo e os classificadores IPD apresentados na Seção 2.2, na qual é possível gerar filtros discriminativos concatenando classificadores IPD treinados para aceitar o padrão e rejeitar seus deslocamentos circulares. Contundo, deve-se frisar que o IPD *não é um filtro de correlação*, pois retorna apenas informação a respeito da pertinência da amostra na classe, mas nenhum informação quanto a sua posição. É possível obter informação da posição utilizando-o como uma janela móvel varrendo a imagem, mas nenhuma informação a respeito da possível posição deste padrão dentro da janela.

2.3.2 Filtragem no Domínio da Frequência

A correlação pode ser rapidamente computada no domínio da frequência através da *Fast Fourier Transform* (FFT) [15, 36]. Neste caso, a Equação (2.65) pode ser reescrita como

$$\mathbf{c} = \mathcal{F}^{-1}\left[\mathcal{F}^*(\mathbf{r}) \odot \mathcal{F}(\mathbf{x})\right], \tag{2.74}$$

onde $\odot$ é o produto de Hadamard (produto ponto-a-ponto) e $\mathcal{F}$ e $\mathcal{F}^{-1}$ são a transformada de Fourier e sua inversa. Desta forma, o gargalo deste processo está em computar a FFT direta e inversa, com complexidade $O(N \log N)$, onde N é a dimensão do sinal [15].

Nesta seção serão brevemente apresentados os filtros de correlação no domínio da frequência utilizados neste trabalho. Um estudo mais detalhado sobre o assunto

pode ser encontrado em [33].

Os filtros *Unconstrained Average Correlation Energy* (UMACE) [14] e *Minimum Output Sum of Squared Error* (MOSSE) [15] são soluções para o problema de mínimos quadrados definido da seguinte forma

$$\mathbf{h} = \arg\min_{\mathbf{h}} \sum_{i=1}^{N_s} \|\mathbf{h} \star \mathbf{g}_i - \mathbf{v}_i\|_2^2 , \tag{2.75}$$

onde $\star$ representa a operação de correlação cruzada bidimensional, $\mathbf{h}$ o filtro desejado, $\mathbf{g}_i$ a i-ésima amostra/imagem de treinamento e $\mathbf{v}_i$, assim como descrito na Seção 2.3.1, representa a resposta desejada para a respectiva amostra. No filtro MOSSE é permitido que $\mathbf{v}_i$ seja qualquer sinal com um pico bem definido no centro do alvo. Normalmente é utilizado $\mathbf{v}_i$ gaussiano, semelhante ao exemplificado na Figura 2.3. No caso do filtro UMACE, $\mathbf{v}_i$ é sempre um impulso na posição do alvo, tornando-o um caso particular do filtro MOSSE. Vale a pena observar a semelhança deste critério com o definido para o filtro discriminativo na Eq. (2.68). Utilizado o Teorema de Parseval para expressar este critério no domínio da frequência

$$\sum_{i=1}^{N_s} \|\mathbf{h} \star \mathbf{g}_i - \mathbf{v}_i\|_2^2 = \frac{1}{D} \sum_{i=1}^{N_s} \|\mathbf{G}_i \odot \mathbf{H}^* - \mathbf{V}_i\|_2^2 , \tag{2.76}$$

onde $\mathbf{H}$, $\mathbf{G}_i$ e $\mathbf{V}_i$ são, respectivamente, as transformadas 2-D discretas de $\mathbf{h}$, das amostras de treinamento $\mathbf{g}_i$, e das respectivas respostas desejadas $\mathbf{v}_i$.

Solucionando o problema para $\mathbf{H}$, encontramos

$$\mathbf{H} = \frac{\sum_{i=1}^{N_s} \mathbf{V}_i^* \odot \mathbf{G}_i}{\sum_{i=1}^{N_s} \mathbf{G}_i^* \odot \mathbf{G}_i}, \tag{2.77}$$

onde, neste caso, a divisão é realizada ponto-a-ponto. Esta solução produz o filtro MOSSE. No caso particular do UMACE esta equação assume a seguinte forma:

$$\mathbf{H} = \frac{\sum_{i=1}^{N_s} \mathbf{G}_i}{\sum_{i=1}^{N_s} \mathbf{G}_i^* \odot \mathbf{G}_i}. \tag{2.78}$$

O filtro conhecido como *Average of Synthetic Exact Filters* (ASEF) [3] assume outra abordagem para a função de otimização. Dada uma amostra, existe um filtro ideal correspondente capaz de gerar a resposta desejada

$$\mathbf{V}_i = \mathbf{H}_i^* \odot \mathbf{G}_i, \tag{2.79}$$

$$\mathbf{H}_i^* = \frac{\mathbf{V}_i}{\mathbf{G}_i} = \frac{\mathbf{V}_i \odot \mathbf{G}_i^*}{\mathbf{G}_i \odot \mathbf{G}_i^*}, \tag{2.80}$$

onde, novamente, a divisão é realizada ponto-a-ponto. Filtros inversos normalmente são instáveis e, como são treinados com apenas uma imagem, tendem à sobreajustar (*overfit*) à imagem. Calcular a média, semelhante ao processo de *bagging*, pode produzir filtros mais robustos e capazes de generalizar [27]. Desta forma, o filtro ASEF é definido como

$$\mathbf{H} = \frac{1}{N_s} \sum_{i=1}^{N_s} \frac{\mathbf{V}_i^* \odot \mathbf{G}_i}{\mathbf{G}_i^* \odot \mathbf{G}_i}. \tag{2.81}$$

Para uma única imagem, os métodos MOSSE e ASEF produzem os mesmos filtros.

Devido à divisão ponto-a-ponto, quando treinados com poucas amostras, os filtros produzidos podem vir a ser instáveis [15]. Neste caso adiciona-se um fator de regularização aos denominadores das Equações (2.77), (2.78) e (2.81).

Retornando à observação feita em relação à semelhança entre as Equações (2.68) e (2.75), existe uma forte relação entre o MOSSE e o filtro discriminativo. Reescrevendo a Equação (2.75)

$$\begin{aligned} \mathbf{h} &= \arg\min_{\mathbf{h}} \sum_{i=1}^{N_s} \|\mathbf{h} \star \mathbf{g}_i - \mathbf{v}_i\|_2^2 \\ &= \arg\min_{\mathbf{h}} \sum_{i=1}^{N_s} \|C(\mathbf{h})\, \mathbf{g}_i - \mathbf{v}_i\|_2^2 \\ &= \arg\min_{\mathbf{H}} \sum_{i=1}^{N_s} \|\mathbf{H}\mathbf{g}_i - \mathbf{v}_i\|_2^2, \end{aligned} \tag{2.82}$$

onde $\mathbf{H} = C(\mathbf{h})$. Se $\mathbf{g}_i = C(\mathbf{x})\, \mathbf{v}_i + \mathbf{w}$, o MOSSE é a aproximação mínimos quadrados do filtro discriminativo. Ou seja, o filtro discriminativo pode ser considerado um caso particular do MOSSE considerando um padrão ideal corrompido por ruído aditivo. Uma dedução mais completa desta relação é apresentada no Apêndice A.

2.3.3 Métricas de Avaliação da Correlação

Dado que filtros discriminativos e correlativos retornam uma resposta espacial, que pode ter múltiplos picos ou nenhum pico de correlação definido, faz-se necessários dispor de métricas de avaliação da correlação resultante. Tais métricas permitem descobrir se o padrão alvo está contido na amostra e realizar o descarte no caso negativo.

Em [37] é apresentado o *Discriminative Signal-to-Noise Ratio* (DSNR). Além de mensurar, de forma relativa, quão agudo é o pico, esta métrica possui a propriedade de reduzir o efeito de picos secundários sobre a classificação, considerando ruído

toda a resposta fora da região do máximo. O DSNR é definido como

$$DSNR = \log\left[\frac{\hat{v}_{\max}^2}{\sum_{k=1}^{K}\hat{v}^2[k] - \hat{v}_{\max}^2}\right], \tag{2.83}$$

onde $\hat{v}[k]$ é saída do filtro $\hat{\mathbf{v}}$ na posição k e $\hat{v}_{\max} = \max_k(\hat{v}[k])$.

Outra métrica, utilizada em [15] e descrita em [33] é conhecida como *Peak to Sidelobe Ratio* (PSR). Semelhante ao DSNR, essa métrica mensura a agudeza do pico de correlação. Existem várias definições para esta métrica [33]. A utilizada nesta dissertação é

$$PSR = \frac{\hat{v}_{\max} - \hat{\mu}_{\hat{v}}}{\hat{\sigma}_{\hat{v}}}, \tag{2.84}$$

onde $\hat{\mu}_{\hat{v}}$ e $\hat{\sigma}_{\hat{v}}$ são, respectivamente, a média e o desvio padrão calculados na região centrada no pico, excluindo uma região em torno do pico definida por uma máscara (Figura 2.4).

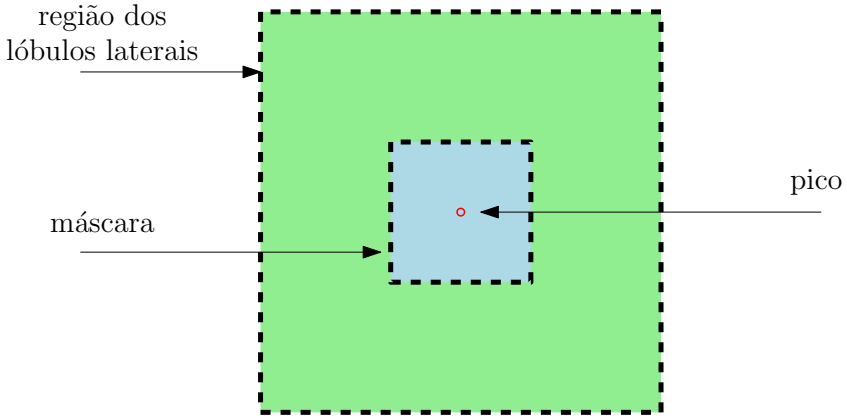


Figura 2.4: Cálculo do *Peak to Sidelobe Ratio* (PSR)

2.4 Funções de Decisão

Como descrito anteriormente, o problema de classificação é a busca por uma função que mapeie as amostras nas classes correspondentes ($f(\mathbf{x}_i) = y_i$). Em um sistema dicotômico isto é feito a partir da escolha de um limiar θ tal que

$$\text{se } \mathbf{h}^T\mathbf{x}_i \begin{cases} \geq \theta & \text{então } y_i = 1 \\ < \theta & \text{então } y_i = 0 \end{cases} \tag{2.85}$$

Porém, classificadores IPD produzem $\mathbf{h}^T\mathbf{x}_i \in \mathbb{R}$, enquanto, idealmente, deveriam ser como descrito na Equação (2.85). Para avaliar os resultados em um intervalo

limitado e escolher o limiar θ correspondente, o produto interno normalizado é utilizado, ou seja, o cosseno do ângulo entre os vetores $\mathbf{h}$ e $\mathbf{x}_i$

$$c_i = \cos(\phi_i) = \frac{\mathbf{h}^T \mathbf{x}_i}{\|\mathbf{h}\| \|\mathbf{x}_i\|}, \tag{2.86}$$

onde $c_i \in [-1, 1]$.

Como exemplo poderíamos escolher $\theta = 0$, ou, como no trabalho anterior [11], escolher um valor que classifique corretamente um conjunto de amostras. Esta abordagem é descrita na Seção 2.4.1. Uma outra abordagem é considerar a saída do classificador, no caso o produto interno, uma características extraídas de cada amostra de treinamento e gerar um outro classificador para a mesma, neste caso conhecido como função de decisão. Esta estratégia, delimitada na Seção 2.4.2, é semelhante ao processo utilizado para a escolha do limiar em [7].

2.4.1 Limiar Linear

Seja o conjunto $\mathbf{C}$ a reunião dos produtos internos c_i do detector $\mathbf{h}$ com todas as amostras do conjunto de treinamento. Dentre os elementos deste conjunto, os elementos frutos do produto interno do detector com amostras da classe positiva formam o subconjunto $\mathbf{C}_{\mathcal{X}_P}$, onde $c_i \in \mathbf{C}_{\mathcal{X}_P}$ se $\mathbf{x}_i \in \mathcal{X}_P$.

Para o conjunto $\mathbf{C}_{\mathcal{X}_P}$ existe um valor de θ tal que uma percentagem arbitrária de seus elementos seja classificada negativamente. Ou seja, dado

$$T(\theta) = \{c_i \in \mathbf{C}_{\mathcal{X}_P} : c_i < \theta\} \tag{2.87}$$

existe $\theta \in [-1, 1]$ tal que

$$\#\left[T(\theta)\right] = \left\lfloor (1 - q)\#(\mathbf{C}_{\mathcal{X}_P}) \right\rfloor, \tag{2.88}$$

onde $\#(\cdot)$ é a cardinalidade do conjunto (número de elementos) e $q \in [0, 1]$ define a percentagem desejada de elementos classificados positivamente.

O objetivo deste método é obter uma função de decisão no formato

$$c_i \begin{cases} \geq \theta & \text{se } y_i = 1 \\ < \theta & \text{se } y_i = 0 \end{cases} \tag{2.89}$$

onde o valor de θ que satisfaz a condição de cardinalidade (Equação (2.88)) pode ser encontrado através de métodos de busca em linha, como o método da bisseção.

Valores próximos de 1 para q geram valores baixos para θ, classificando uma maior parcela de pontos positivos e também uma maior quantidade de falsos positi-

vos.

2.4.2 Máxima verossimilhança

Dado os produtos internos $c_i = \mathbf{h}^T \mathbf{x}_i$ do classificador com as amostras do conjunto de treinamento, podemos definir uma função de decisão a partir das probabilidades *a posteriori* e da função *log odd ratio* [17, 31]

$$f(\mathbf{x}_i) = \ln \frac{p(y_i = 1|c_i)}{p(y_i = 0|c_i)} \begin{cases} \text{se } \geq 0 & \text{então } y_i = 1 \\ \text{se } < 0 & \text{então } y_i = 0. \end{cases} \tag{2.90}$$

Através da lei de Bayes podemos reescrever (2.90) como

$$f(\mathbf{x}_i) = \ln \frac{p(y_i = 1)p(c_i|y_i = 1)}{p(y_i = 0)p(c_i|y_i = 0)} \tag{2.91}$$

Seja a verossimilhança para cada classe $p(c_i|y_i = 0\backslash 1) \sim \mathcal{N}(\mu_{0\backslash 1}, \sigma_{0\backslash 1})$. Substituindo em 2.91:

$$f(\mathbf{x}_i) = \ln \frac{p(y_i = 1)}{p(y_i = 0)} + \frac{1}{2} \ln \frac{\sigma_0^2}{\sigma_1^2} + \frac{(c_i - \mu_0)^2}{2\sigma_0^2} - \frac{(c_i - \mu_1)^2}{2\sigma_1^2} \tag{2.92}$$

Onde μ_1 e σ_1 são definidos como

$$\mu_1 = \frac{1}{N_1} \sum_{n=1}^{N_1} c_i \tag{2.93}$$

$$\sigma_1 = \frac{1}{N_1 - 1} \sum_{n=1}^{N_1} (c_i - \mu_1)^2 , \tag{2.94}$$

para $x_n \in \mathcal{X}_P$. Obtemos μ_0 e σ_0 de maneira análoga, para $x_n \in \mathcal{X}_N$.

Considerando as distribuições *a priori* equivalentes [17], podemos definir um limiar θ como

$$\theta = -\ln \frac{\sigma_0^2}{\sigma_1^2}, \tag{2.95}$$

e reescrever a Equação (2.92) como:

$$f(\mathbf{x}_i) = \begin{cases} 1 & \text{se } \frac{(c_i - \mu_0)^2}{\sigma_0^2} - \frac{(c_i - \mu_1)^2}{\sigma_1^2} \geq \theta \\ 0 & \text{se } \frac{(c_i - \mu_0)^2}{\sigma_0^2} - \frac{(c_i - \mu_1)^2}{\sigma_1^2} < \theta \end{cases} \tag{2.96}$$

2.5 Resumo

Neste capítulo são descritos os fundamentos teóricos para classificadores utilizados neste trabalho, incluindo novas propostas de treinamento para o classificador IPD. Tais classificadores foram escolhidos por serem muito difundidos, devido à carac-

terísticas interessantes, baixo custo computacional e por serem fruto de pesquisas anteriores. Um resumo com as principais vantagens e desvantagens de cada um dos detectores é apresentado na Tabela 2.1.

Tabela 2.1: Vantagens e desvantagens dos detectores implementados

Detector	Vantagens	Desvantagens
IPD	1) Detecção rápida através de produtos internos; 2) Saída serve como medida de confiança; 3) Invariante a translação e robusto a pequenas rotações.	1) Detector local; 2) Sensível a padrões semelhantes.
Filtro discriminativo	1) Detecção rápida através de produto ponto-ponto no domínio da frequência; 2) Retorna pertinência e a posição do alvo na amostra; 3) Invariante a translação e robusto a rotações; 4) Detector global/local.	1) Sensível a padrões semelhantes; 2) Critério de detecção indireto (DSNR ou PSR).
Filtros de correlação	1) Detecção rápida através de produto ponto-ponto no domínio da frequência; 2) Retorna pertinência e a posição do alvo na amostra; 3) Invariante a translação e robusto a rotações.	1) Detector global; 2) Problemas de condicionamento; 3) Critério de detecção indireto (DSNR ou PSR).

Atualmente, muitos desses métodos têm sido objeto de estudo em diferentes áreas. Em [38] é apresentado um classificador híbrido, denominado *Maximum Margin Correlation Filter* (MMCF), que tenta aliar a capacidade dos filtros de correlação de obter simultaneamente a localização e a classe do objeto com a capacidade de generalização de classificadores SVM. Em [39] são apresentados filtros baseados nos métodos ASEF e MOSSE que fazem uso de *boosting* durante o treinamento e/ou se ajustam de forma adaptativa aos conteúdos das imagens. Tais trabalhos apontam que ainda há muito a ser investigado dentro desta área.

No próximo capítulo são apresentados os métodos de pré-processamento das e extração de características utilizados em conjunto com os detectores apresentados. No Capítulo 4 são apresentados e discutidos os resultados obtidos para os classificadores aqui no contexto de detecção de pontos fiduciais.

Capítulo 3

Extração de Características em Imagens

3.1 Introdução

Técnicas de reconhecimento de padrões funcionam melhor quando as amostras de entrada são relevantes para o problema de classificação dado [40]. Entretanto, devido à natureza e às limitações no procedimento de obtenção de amostras, as mesmas podem possuir dimensão muita alta, existir em demasia ou de forma esparsa, e conter ruídos e redundâncias que dificultem o processo classificatório. Tais qualidades podem ser nocivas ao classificador, aumentado a complexidade computacional e reduzindo a eficiência do sistema [41].

No intuito de reduzir possíveis problemas e melhorar o desempenho de classificação, as amostras de entrada são pré-processadas e transformadas para um novo espaço, denominado *espaço de características*. Espera-se que as *características* tenham propriedades que facilitem a classificação e/ou reduzam a complexidade computacional do sistema. Esse processo é denominado *extração de características* [31, 41].

Neste capítulo serão apresentados os métodos de pré-processamento e extração de características utilizados neste trabalho. Na Seção 3.2 é apresentado o método de pré-processamento utilizado quando a classificação é realizada bloco-a-bloco no domínio da imagem. Na Seção 3.3 é brevemente descrita a implementação da transformação para o domínio denominado *Histogram of Oriented Gradients* (HOG), utilizado neste trabalho. Por fim, na Seção 3.4 são apresentados os procedimentos de pré-processamento quando a imagem completa é utilizada como amostra no domínio da frequência.

3.2 Amostragem por Blocos da Imagem

Nesta seção é descrito o método de pré-processamento utilizado quando a detecção é realizada diretamente no domínio da imagem em janela deslizante, bloco-a-bloco. Ou seja, neste caso as amostras dos classificadores são blocos de dimensão $B_z \times B_z$. A primeira etapa deste método é a *normalização da iluminação*. Esta etapa tem como objetivo mitigar os efeitos da variação de iluminação sobre a imagem sem afetar características de interesse para a detecção. O sistema adotado neste caso baseia-se no descrito em [42], no qual a seguinte sequência de passos é aplicada à imagem: correção gama, filtragem por diferença de gaussianas e equalização de contraste. Propõe-se opcionalmente, em nível de bloco, uma janela de Hanning, com o intuito de reduzir artefatos no domínio da frequência.

A *correção gama* é uma transformação não-linear sobre os níveis de cinza dos pixels da imagem, com o objetivo de expandir a faixa dinâmica em regiões mais escuras e comprimi-lá nas mais iluminadas. Dada uma imagem com $M \times N$ pixels, com níveis de cinza $I(m,n)$ $(0 \leq m < M;\ 0 \leq n < N)$, a correção gama é uma transformação não linear do tipo

$$I(m,n) \leftarrow [I(m,n)]^{\gamma}, \tag{3.1}$$

onde $\gamma \in (0,1]$ é um parâmetro a ser definido pela aplicação. Segundo o recomendado em [42], utilizou-se neste trabalho o valor de $\gamma = 0.2$.

Apesar de reduzir os efeitos de sombra sobre a imagem, a correção gama tende a amplificar ruídos em áreas escuras, além de não eliminar gradientes de sombra sobre a imagem [42]. A *filtragem por diferença de gaussianas*, também conhecida como DoG (*Difference of Gaussians*), mitiga ruídos e gradientes de sombra [42]. Isso se deve ao comportamento passa-faixa da filtragem DoG, atenuando componentes de alta e baixa frequência. Seja uma gaussiana bidimensional $g(m,n)$ com desvio padrão σ:

$$g(m,n) = \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{m^2+n^2}{2\sigma^2}\right), \tag{3.2}$$

e uma imagem I, a filtragem DoG é definida como:

$$\begin{aligned} \mathrm{DoG}(m,n) &= g_{\sigma_1} * I - g_{\sigma_2} * I \\ &= (g_{\sigma_1} - g_{\sigma_2}) * I \\ &= G * I, \end{aligned} \tag{3.3}$$

sendo G:

$$G(m,n) = \frac{1}{\sqrt{2\pi}} \left[\frac{1}{\sigma_1} \exp\left(-\frac{m^2+n^2}{2\sigma_1^2}\right) - \frac{1}{\sigma_2} \exp\left(-\frac{m^2+n^2}{2\sigma_2^2}\right)\right]. \tag{3.4}$$

Como pode ser visto na Equação (3.3), a característica passa-faixa da filtragem DoG é devida ao fato que a mesma deixa passar apenas os detalhes que foram eliminados pela gaussiana de desvio padrão maior e não foram eliminados pela a que possui desvio padrão menor. Como recomendado em [42], usou-se os seguintes valores de desvio padrão: $\sigma_1 = 1.0$ e $\sigma_2 = 2.0$.

A etapa de *equalização de contraste* consiste de uma sequência de transformações nas quais os níveis de cinza da imagem são reescalados

$$I(m,n) \leftarrow \frac{I(m,n)}{(J_I)^{1/\alpha}}, \tag{3.5}$$

$$I(m,n) \leftarrow \frac{I(m,n)}{(K_I)^{1/\alpha}}, \tag{3.6}$$

$$I(m,n) \leftarrow \tau \tanh\left(\frac{I(m,n)}{\tau}\right). \tag{3.7}$$

Onde, J_I e K_I são definidos como

$$J_I = \frac{1}{M \times N} \sum_{\substack{0<m<M \\ 0<n<N}} |I(m,n)|^{\alpha}, \quad \text{e} \tag{3.8}$$

$$K_I = \frac{1}{M \times N} \sum_{\substack{0<m<M \\ 0<n<N}} \min\left[\tau, |I(m,n)|\right]^{\alpha}. \tag{3.9}$$

O parâmetro α reduz a influência de altos valores de luminância [42]. O limiar τ trunca os níveis de luminância, comprimindo e limitando $I(m,n)$ ao intervalo $(-\tau, \tau)$ após a tangente hiperbólica. Utilizou-se $\alpha = 0.1$ e $\tau = 10$, segundo proposto em [42]. As etapas deste processo são ilustradas na Figura 3.1.

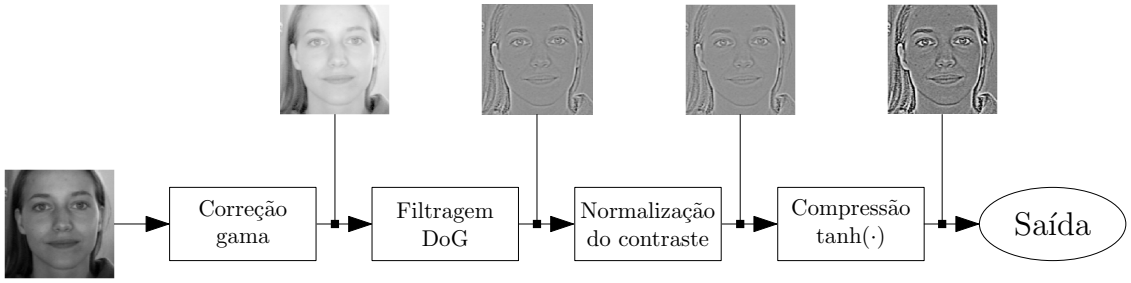


Figura 3.1: Etapas de normalização da iluminação

Este pré-processamento é aplicado sobre toda a imagem. As amostras (características) alvos extraídas são blocos da imagem centrados no ponto a ser testado com dimensões $B_z \times B_z$. Escolhido um bloco, o mesmo é vetorizado, concatenando as linhas em um único vetor. Opcionalmente pode-se aplicar uma segunda etapa de

pré-processamento. Neste segundo passo, precedendo a concatenação, uma janela de Hanning é aplicada ao bloco a ser avaliado. A janela de Hanning bidimensional é uma função discreta de janelamento definida como [43]

$$w(m,n) = \frac{1}{4}\left(1 - \cos\left(\frac{2\pi m}{M-1}\right)\right)\left(1 - \cos\left(\frac{2\pi n}{N-1}\right)\right). \tag{3.10}$$

Os coeficientes da janela são multiplicados pelos blocos da imagem. O objetivo deste procedimento é enfatizar o centro do bloco e reduzir artefatos no domínio da frequência produzidos pelas descontinuidades das bordas.

3.3 *Histogram of Oriented Gradients*

Diferentemente dos outros métodos apresentados neste capítulo, o *Histogram of Oriented Gradients* (HOG), mais que um método de processamento, é um método de extração de características. Baseado em uma das etapas de detecção do algoritmo *Scale Invariant Feature Transform* (SIFT) [44], o objetivo deste método é obter um conjunto de característica denominadas *descritores* capazes de representar informação de estrutura local como gradientes e bordas. Para isto, o primeiro passo é computar os gradientes. Esta etapa pode ser realizada de diferentes formas. Em [45], diferentes máscaras derivativas foram testadas no contexto de detecção de pedestre e os seguintes filtros foram escolhidos

$$h_x = \begin{bmatrix} -1 & 0 & 1 \end{bmatrix} \text{ e } h_y = \begin{bmatrix} -1 & 0 & 1 \end{bmatrix}^T.$$

Dados os filtros h_x e h_y, as magnitudes e as orientações dos gradientes são computadas via

$$D_{\text{mag}}(m,n) = \sqrt{D_x^2(m,n) + D_y^2(m,n)}, \tag{3.11}$$

$$D_{\text{ang}}(m,n) = \arctan\left(\frac{D_x(m,n)}{D_y(m,n)}\right), \tag{3.12}$$

onde $D_x(m,n)$ e $D_y(m,n)$ são, respectivamente, as aproximações das derivadas nas direções horizontal e vertical, computadas como $D_x = h_x * I$ e $D_y = h_y * I$. Em seguida, são definidos os histogramas de orientação. Nesta etapa será utilizada, no lugar da implementação descrita em [45], a implementação utilizando *Histograma Integral* apresentada em [46].

Histograma Integral [47], extensão da técnica imagem integral [8], é um método que permite computar de forma eficiente os histogramas de todas as possíveis regiões alvo em um espaço cartesiano. Seja um tensor $H(m,n,b)$ de dimensões $M \times N \times B$, onde B é o número de intervalos (*bins*) dos histogramas alvo. Podemos definir o

conteúdo do b-ésimo intervalo na posição (m, n) como:

$$H(m,n,b) = \sum_{\substack{m' \leq m \\ n' \leq n}} p\left(X(m',n')\right) q\left(X(m',n'),b\right), \tag{3.13}$$

onde $q(\cdot)$ é uma função indicadora

$$q\left(x,b\right) = \begin{cases} 1, & \text{se } x \in b \\ 0, & \text{caso contrário,} \end{cases} \tag{3.14}$$

e $p\left(\cdot\right)$ é uma função que retorna um escalar positivo referente ao peso da amostra. No caso mais simples, $p\left(x\right) = 1$ e $q\left(x,b\right) = u\left(t_{b-1} - x\right) - u\left(t_b - x\right)$, onde $\{t_0, \ldots, t_{B-1}\}$ são os limiares de cada intervalo. Dada uma frente de onda da esquerda para direita e de cima para baixo, o histograma na posição (m, n) é computado como:

$$\begin{aligned} H\left(m,n,b\right) = \; & H\left(m-1,n,b\right) + H\left(m,n-1,b\right) \\ & - H\left(m-1,n-1,b\right) + p\left(X(m,n)\right) q\left(X(m,n),b\right). \end{aligned} \tag{3.15}$$

Desta forma, os histogramas podem ser gerados a partir de um única passagem sobre a imagem. Este processo pode ser visualizado na Figura 3.2.

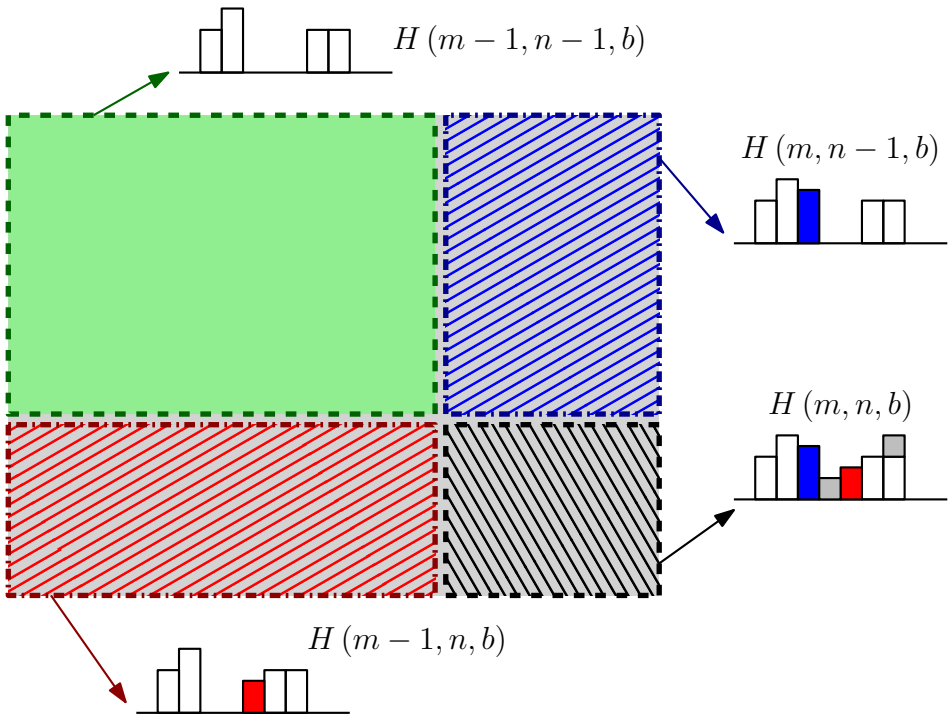


Figura 3.2: Propagação da frente de onda para o cálculo do histograma integral. A cada passo, o histograma atual é computado a partir dos valores de histogramas vizinhos acima e à esquerda.

Essa abordagem permite o rápido calculo dos HOGs. Todavia, é inferior ao método apresentado em [45] ao não realizar as etapas de pré e pós-processamento de cada conjunto de HOGs [46]. Neste trabalho, utilizou-se como parâmetros:

- $B = 18$;
- $p\left(D_{\text{mag}}(m,n)\right) = D_{\text{mag}}(m,n)$; e
- $q\left(D_{\text{ang}}(m,n), b\right) = u\left(t_{b-1} - D_{\text{ang}}(m,n)\right) - u\left(t_b - D_{\text{ang}}(m,n)\right)$, onde $t_0 = 0$ e $t_b = t_{b-1} + \frac{360^\circ}{B}$, considerando ângulos em graus pertencentes ao intervalo $[0, 360)$.

Tais parâmetros foram escolhidos por apresentarem os melhores resultados sem produzir um aumento da complexidade computacional que comprometesse o desempenho do sistema.

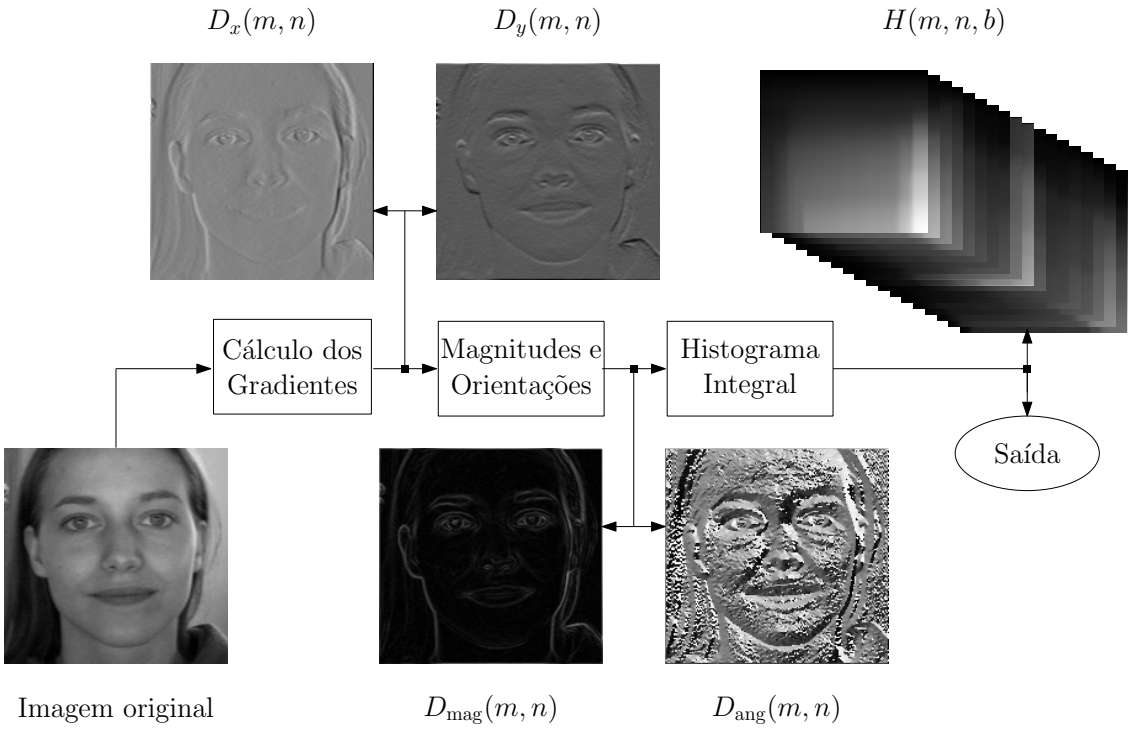


Figura 3.3: Procedimento para obtenção de HOGs utilizando histograma integral

Os procedimentos implementados são ilustrados na Figura 3.3. Ao fim dessas etapas, o sistema implementado permite dois tipos de características centradas no ponto de interesse: retornar um único HOG normalizado com dimensão B; ou retornar um descritor baseado em HOG [44, 45]. No segundo caso a região alvo, como exemplo um bloco da imagem, é subdivida em blocos com sobreposição. Subsequentemente, cada bloco é subdivido em células, sem sobreposição. Cada célula retorna um HOG correspondente a sua posição. O conjunto de HOGs é concatenado, formando um vetor de descritores. HOGs de células provenientes do mesmo bloco são

normalizados conjuntamente pela da norma L_1 dos vetores do conjunto. A norma L_1 foi escolhida por permitir rápida computação apenas contando os elementos dos histogramas. Escolheu-se empiricamente o seguinte conjunto de parâmetros para os descritores para o segundo caso

- Bloco de 2×2 células;
- Células de 6×6 pixels;
- Sobreposição entre blocos com passo de 3 pixels.

Neste caso, a dimensão dos descritores N_{HOG} é função do tamanho da janela alvo, número de *bins* dos histogramas, número de sub-blocos e tamanho das células

$$N_{\mathrm{HOG}} = (N_{BH} \times N_{CH})(N_{BV} \times N_{CV}) \times B, \tag{3.16}$$

onde N_{BH} e N_{BV} são o número de blocos nas direções horizontal e vertical, respectivamente, N_{CH} e N_{CH} o número de células em cada dimensão. O número de blocos em cada direção é computado como

$$N_{BH/V} = \frac{T_{H/V} - N_{CH/V} \times C_z}{B_{St}} + 1, \tag{3.17}$$

onde $T_{H/V}$ é a dimensão horizontal/vertical do alvo, C_z o tamanho em pixel das células e B_{St} é o passo de sobreposição (*stride*).

É importante frisar que os parâmetros escolhidos tornam a detecção subótima [46] segundo os parâmetros originalmente encontrados em [45]. Contudo, tal escolha foi necessária para garantir a eficiência do processo de extração de características, permitindo o uso em tempo real.

3.4 Imagem no Domínio da Frequência

Para os filtros de correlação no domínio da frequência é possível realizar a detecção ao longo de toda a imagem. Neste caso, as imagens são multiplicadas por uma janela senoidal, como descrito em [36]:

$$I\left(m,n\right) \leftarrow \left(I\left(m,n\right) - 0.5\right) \sin\left(\frac{\pi m}{M}\right) \sin\left(\frac{\pi n}{N}\right). \tag{3.18}$$

O objetivo deste janelamento é reduzir artefatos no domínio da frequência, reduzindo valores próximos às bordas para zero e eliminando descontinuidades.

Opcionalmente um procedimento de normalização da iluminação pode ser utilizado [3], semelhante ao descrito na Seção 3.2. Primeiramente, cada imagem é

transformada da seguinte forma

$$I(m,n) \leftarrow \log\left(I(m,n)+1\right). \tag{3.19}$$

Esta transformação tem como objetivo reduzir os efeitos de sombra e iluminação intensa. Em seguida realiza-se uma nova transformação

$$\begin{aligned} I(m,n) &\leftarrow I(m,n) - \mu_I, \\ I(m,n) &\leftarrow \frac{I(m,n)}{ss_I}, \end{aligned} \tag{3.20}$$

onde μ_I é a média dos pixels de $I(m,n)$ e ss_I a soma dos quadrados dos pixels

$$ss_I = \sum_{m,n} I(m,n)^2. \tag{3.21}$$

Caso este procedimento seja utilizado, o janelamento ocorrerá após a normalização da iluminação. Por fim, a imagem é transformada para o domínio da frequência através da DFT. Neste caso, as imagens transformadas são as amostras de entrada dos classificadores utilizados.

3.5 Resumo

Neste capítulo são apresentados os métodos de pré-processamento e/ou extração de característica utilizados ao longo deste trabalho. O método apresentado na Seção 3.2 é utilizado em diferentes trabalhos [7, 11, 42], com bons resultados no contexto de detecção de pontos fiduciais faciais. O mesmo pode ser dito para o método descrito na Seção 3.4, utilizado tanto no contexto de detecção de pontos fiduciais [3], quanto no contexto de detecção de objetos online [15, 36]. Os descritores HOG normalmente vêm sido utilizados para a detecção de objetos complexos [44–46]. Todavia, um exemplo com resultados interessantes, utilizando *boosting* e restrições geométricas é encontrado em [48].

No próximo capítulo são discutidos os resultados obtidos para os classificadores lineares descritos no Capítulo 2 em conjunto com as estratégias de extração de característica aqui apresentadas dentro do contexto de detecção de pontos fiduciais faciais em imagens estáticas.

Capítulo 4

Detecção de Pontos Fiduciais em Imagens Estáticas

4.1 Introdução

Atualmente, o problema de detecção e rastreamento de características faciais de forma robusta tem recebido considerável atenção. Isso se deve, principalmente, ao seu uso em diversas aplicações [1, 2]. Alguns exemplos de uso incluem sistemas de segurança, reconhecimento de face e realidade aumentada [1].

Características faciais são um conjunto de informações através das quais é possível definir uma face humana. Exemplos de características faciais incluem a largura da boca, espaço entre os olhos e o tamanho do nariz [6, 7].

Pontos fiduciais são pontos de controle sobre um objeto que definem regiões características com propriedades de interesse à aplicação. Normalmente estes pontos são definidos em regiões salientes ou com propriedades distintas das demais. No caso da face, as posições relativas destes pontos permitem definir as diferentes características faciais [1, 6, 7].

Neste capítulo os classificadores descritos anteriormente são avaliados no contexto de detecção de pontos fiduciais. Na Seção 4.2 é apresentado o arcabouço de detecção, do qual os detectores são módulos constituintes, e são detalhadas as etapas do processo de detecção. Na Seção 4.3 são descritos os procedimentos experimentais realizados para o treinamento e validação dos classificadores. Na Seção 4.4 são apresentados os resultados obtidos para todos os detectores e estratégias de treinamento apresentadas, com a avaliação crítica dos resultados obtidos. As conclusões obtidas são apresentadas na Seção 4.5.

4.2 Descrição do Sistema

Nesta seção é detalhado o arcabouço implementado e suas partes constituintes. Na Seção 4.2.1 é apresentado o método de pré-processamento e cada uma de suas etapas, enquanto na Seção 4.2.2 é descrito os procedimentos utilizados para a detecção.

4.2.1 Pré-Processamento

De forma a conformar a imagem de entrada ao sistema de detecção e extrair as amostras ou características interessantes à detecção, métodos de pré-processamento são aplicados aos dados de entrada. Um exemplo clássico de uso de pré-processamento é a normalização das amostras em classificadores baseados em rede neurais com objetivo de evitar a saturação dos pesos, acelerar o processo de treinamento e permitir melhor generalização da rede [49].

No universo de detecção de objetos em imagens (neste caso em particular, de pontos fiduciais faciais), algumas características são interessantes para os detectores, como: invariância à translação, robustez à rotação e a mudanças de iluminação e escala. Algumas destas propriedades são inerentes aos classificadores, devido ao processo de treinamento ou a estrutura do método de detecção. Como exemplo, classificadores IPD são naturalmente invariantes a translações devido à detecção em janela deslizante. Muitas dessas propriedades, entretanto, são adquiridas apenas após o pré-processamento. Um exemplo é a robustez a variações de iluminação, que pode ser obtida através de técnicas de normalização de iluminação.

O sistema de pré-processamento proposto, apresentado na Figura 4.1, divide-se em três partes: segmentação da face e escalamento da face, extração de características, e a etapa opcional de restrição do espaço de busca. Cada uma dessas etapas será detalhada a seguir.

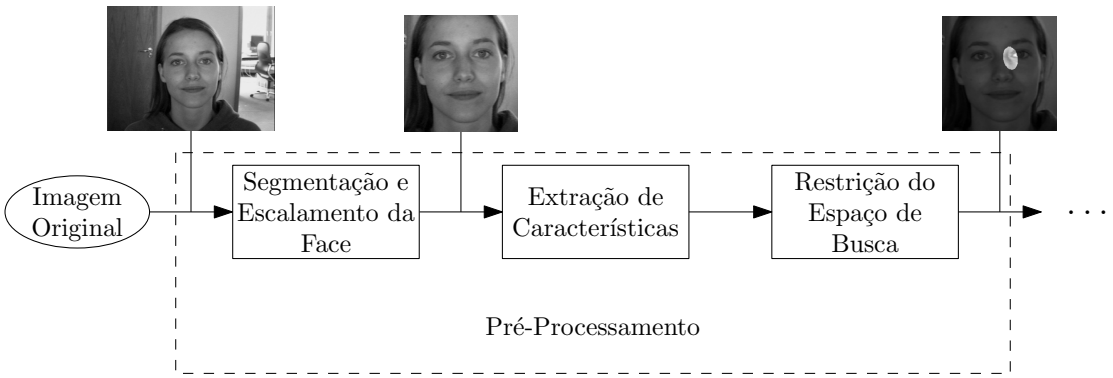


Figura 4.1: Sistema de pré-processamento proposto.

Segmentação e Escalamento da Face

Nesta etapa a face é segmentada a partir da imagem original e redimensionada para o tamanho padrão. O processo de segmentação para treinamento e validação é descrito na Figura 4.2. Primeiramente, é definido um quadrado que contém todas as marcações manuais. Dado o tamanho do bloco de detecção B_z, a região de segmentação é expandida em $B_{hz} = \lfloor {}^{B_z}/_2 \rfloor$. Este processo é descrito na Figura 4.2a. Isto é feito para que o bloco de detecção, quando centrado em um ponto de teste, não saia da região da face. Caso o quadrado não esteja completamente contido na imagem (Figura 4.2b), o mesmo é deslocado de forma até que esteja contido nos limites da imagem (Figura 4.2c).

Definida a região de segmentação, a imagem é segmentada e escalada, através de interpolação bilinear, de tal forma que a região escolhida tenha dimensão $I_z \times I_z$. A região é novamente expandida em B_{hz} e a imagem resultante é utilizada para o treinamento e validação.

Extração de Características

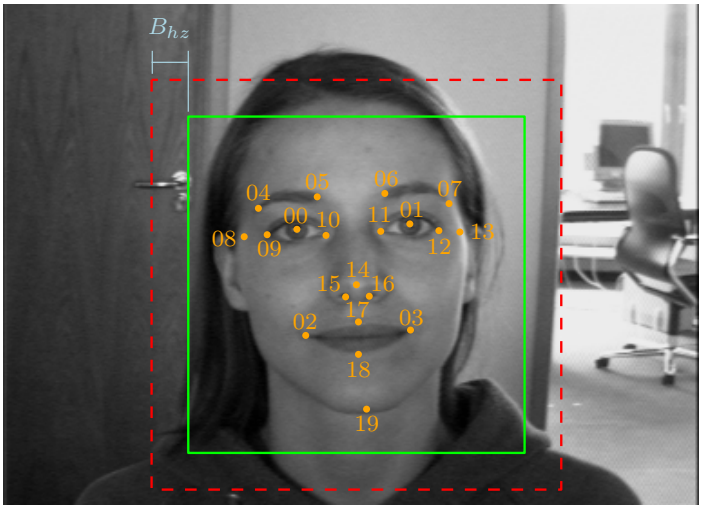
Após o processo de escalamento e segmentação, a extração de características é realizada de uma das três formas apresentadas no Capítulo 3:

- *Blocos da imagem*: A imagem segmentada e escalada para a dimensão I_z sofre o processo descrito na Seção 3.2. As amostras/características alvos extraídas são blocos da imagem centradas no ponto a ser testado com dimensões $B_z \times B_z$. No caso de classificadores que realizam a detecção e o treinamento do domínio da frequência, os blocos são pré-processados multiplicando-os por uma janela de Hanning. Os parâmetros escolhidos neste trabalho são descritos na Tabela 4.1.

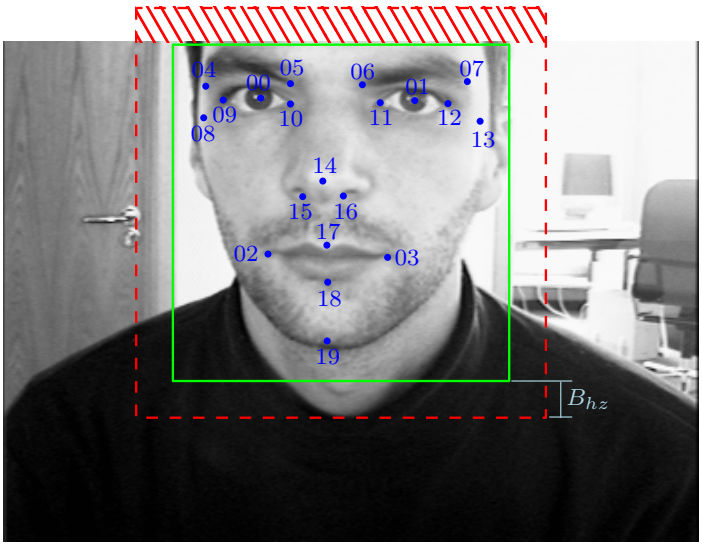
Tabela 4.1: Parâmetros utilizados no pré-processamento para extração de blocos da imagem.

Parâmetro	Valor
B_z	21
B_{hz}	10 ($\lfloor B_z \rfloor$)
I_z	95 ($75 + 2B_{hz}$)
Janela de Hanning	Não

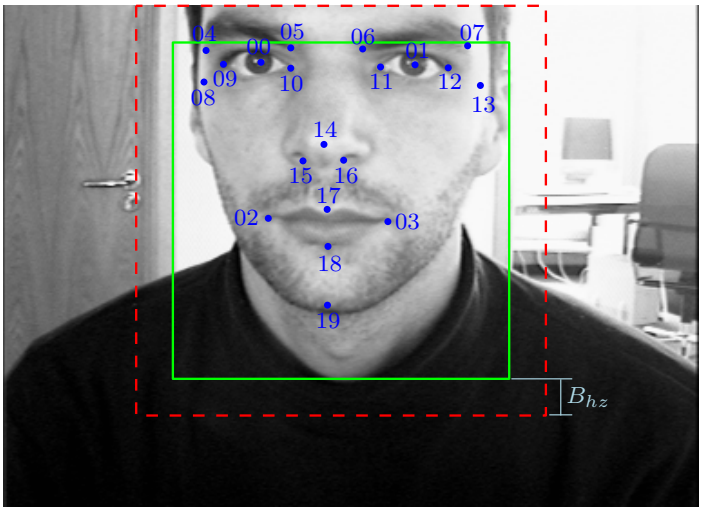
- *Integral HOG*: Os histogramas integrais são extraídos da imagem de dimensão I_z. No arcabouço implementado, duas formas de extração de características são possíveis: somente um único HOG e descritores HOG [44, 45], extraídos do bloco centrado no ponto alvo. Os dois casos são descritos na Seção 3.3. É



(a) Marcações com região de segmentação



(b) Região de segmentação fora dos limites



(c) Correção da posição do quadrado de segmentação

Figura 4.2: Região de segmentação da face. Pontos enumerados representam as marcações manuais da base de dados BioID [13]. Em verde, quadrado obtido originalmente através das marcações manuais. Em vermelho, versão expandida de B_{hz}.

possível escolher se o peso das amostras é quantitativo, nesse caso $p\left(I(m,n)\right) = 1$, ou proporcional à magnitude do gradiente, $p\left(I(m,n)\right) = D_{\text{mag}}(m,n)$. Os parâmetros utilizados são apresentados na Tabela 4.2.

Tabela 4.2: Parâmetros utilizados no método de extração de características *Integral HOG*.

Parâmetro	Valor
B_z	21
B_{hz}	10 ($\lfloor B_z \rfloor$)
I_z	95 ($75 + 2B_{hz}$)
B	18 ($0 - 360°$)
$p\left(I(m,n)\right)$	$D_{\text{mag}}(m,n)$
Descritores	Não

- *Imagem no Domínio da Frequência*: Neste caso, a imagem pré-processada é a amostra alvo, com dimensão $I_z \times I_z$, assim como descrito na Seção 3.4. Pode-se escolher entre utilizar o pré-processamento com compressão logarítmica (Equação (3.19)) ou apenas a janela senoidal. Neste utilizou-se apenas o último.

  Este método é utilizado apenas no caso de filtros de correlação no domínio da frequência (ASEF, UMACE e MOSSE, ver Seção 2.3.2). Apesar de não necessitar de blocos, manteve-se a dimensão da imagem ($I_z = 95$) apenas para fins de comparação. Os parâmetros deste método são listados Tabela 4.3.

Tabela 4.3: Parâmetros utilizados no pré-processamento para imagens no domínio da frequência.

Parâmetro	Valor
B_z	21
B_{hz}	10 ($\lfloor B_z \rfloor$)
I_z	95 ($75 + 2B_{hz}$)
Normalização	Não utilizada

Restrição do Espaço de Busca

Escalar as imagens para que todas possuam as mesmas dimensões produz regiões com grande probabilidade de encontrar um ponto fiducial alvo [7]. A área de busca para a detecção pode então restringir-se a essas regiões. Nesta seção é apresentado o método através do qual tais regiões são estimadas.

Supõe-se que a posição do ponto fiducial seja uma variável aleatória $\mathbf{P}$. Dado um conjunto de treinamento que contenha M_s realizações $\mathbf{p}_m$ de $\mathbf{P}$, a média pode

ser estimada como

$$\boldsymbol{\mu}_{\mathbf{p}} = \frac{1}{M_s} \sum_{m=1}^{M_s} \mathbf{p}_m, \tag{4.1}$$

e sua matriz de covariância como

$$\Sigma_{\mathbf{P}} = \frac{1}{M_s - 1} \sum_{m=1}^{M} \left(\mathbf{p}_m - \boldsymbol{\mu}_{\mathbf{p}}\right) \left(\mathbf{p}_m - \boldsymbol{\mu}_{\mathbf{p}}\right)^T. \tag{4.2}$$

Considerando um modelo gaussiano, a distribuição de probabilidades $p_{\mathbf{P}}(\mathbf{p})$ da posição de um ponto fiducial é dada por

$$p_{\mathbf{P}}(\mathbf{p}) = \frac{1}{\sqrt{(2\pi)^k |\Sigma_{\mathbf{P}}|}} \mathrm{e}^{\left[-\frac{1}{2}\left(\mathbf{p}_m - \boldsymbol{\mu}_{\mathbf{p}}\right)^T \Sigma_{\mathbf{P}}^{-1} \left(\mathbf{p}_m - \boldsymbol{\mu}_{\mathbf{p}}\right)\right]} \tag{4.3}$$

Através da amostra do conjunto de treinamento que maximiza a distância de Mahalanobis é possível delimitar uma região de grande probabilidade de encontrar o ponto fiducial [6, 7, 11]

$$r_{\max} = \max_{\mathbf{p}_m} \left[\sqrt{\left(\mathbf{p}_m - \boldsymbol{\mu}_{\mathbf{p}}\right)^T \Sigma_{\mathbf{P}}^{-1} \left(\mathbf{p}_m - \boldsymbol{\mu}_{\mathbf{p}}\right)} \right]. \tag{4.4}$$

Este modelo determina uma região elíptica com grande probabilidade de conter um ponto de interesse, região denominada de ROI – *Region of Interest*. Um exemplo de ROI é apresentado na Figura 4.3.

Definida a região de interesse para cada ponto (ROI - *Region of Interest*), os pontos são testados segundo a Equação (4.5):

$$\begin{aligned} \mathbf{p}_m \in \mathbf{P_d} \iff & \sqrt{(\mathbf{p}_m - \boldsymbol{\mu}_{\mathbf{p}})^t \Sigma_{\mathbf{P}}^{-1} (\mathbf{p}_m - \boldsymbol{\mu}_{\mathbf{p}})} \leq r_{\max} \\ & \wedge \max(|p_{1_m}|, |p_{2_m}|) \leq \frac{I_z + B_z}{2}, \end{aligned} \tag{4.5}$$

onde p_{1_m} e p_{2_m} são as coordenadas do ponto testado $\mathbf{p}_m$, I_z o tamanho da imagem escalada e $\mathbf{P_d}$ o conjunto de pontos pertencentes à ROI.

Desta forma são descartados todos os pontos que não pertencem à ROI ou que pertençam a borda da imagem, impossibilitando a centralização do bloco de dimensão B_z. Como as imagens das faces segmentadas são escaladas para o mesmo tamanho, o conjunto gerado de pontos é fixo e independe da imagem, podendo ser definido *offline*.

4.2.2 Detecção

O processo de detecção varia segundo o conjunto de detectores e método de pré-processamento utilizados. No caso dos detectores utilizados para as amostras em

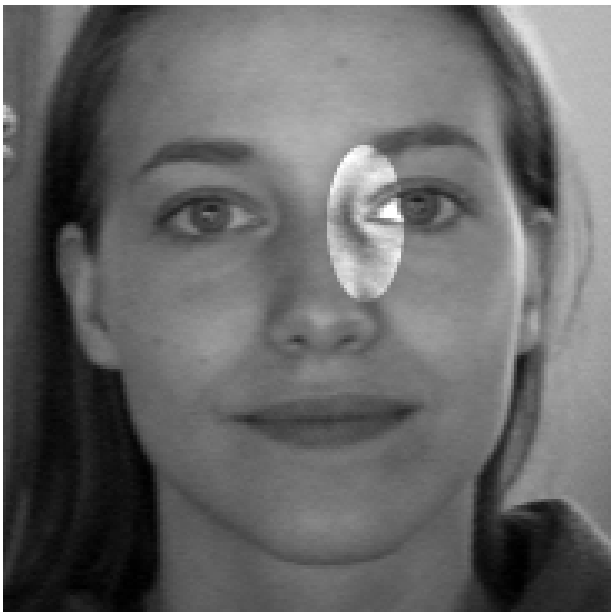


Figura 4.3: Exemplo de área de busca para o canto interno do olho esquerdo

bloco (Seção 3.2) ou com os descritores HOG (Seção 3.3), o processo de detecção ocorre da seguinte forma:

- **IPD**: Dado que os detectores retornam um valor escalar entre $[-1, 1]$, a detecção é feita através da avaliação por amostra (bloco ou descritor HOG). O descarte de amostras é realizado de acordo com um dos critérios de decisão apresentados em 2.4. A posição correspondente à amostra com maior produto interno é escolhida como saída.

- **Filtros de correlação**: Os filtros retornam como saída um vetor com dimensão igual à da amostra avaliada. É calculado o DSNR da saída e o mesmo é avaliado através dos critérios de decisão anteriores, descartando as amostras que não satisfazem o critério. A posição correspondente à amostra com maior DSNR é escolhida como saída.

O método de normalização no domínio da frequência é utilizado apenas com os filtros de correlação. Nesse caso, dado que as amostras são imagens inteiras, a avaliação de que existe um objeto alvo na imagem é dada pelo cálculo do valor de PSR sobre a saída. A posição da marcação automática corresponde ao ponto onde ocorre o máximo valor de saída.

4.3 Procedimentos Experimentais

4.3.1 Base de Dados

Para o treinamento e validação dos detectores no contexto de detecção de pontos fiduciais sobre a face, uma base de dados com imagens de faces em pose frontal e marcações manuais é necessária. Para este fim foi escolhida a base de dados BioID [13].

A BioID é uma base de dados que possui 1.521 imagens de 23 indivíduos em pose frontal em formato PGM e tamanho 384×286. As imagens possuem variação de iluminação, plano de fundo e escala das faces. Em conjunto são fornecidas anotações manuais de 20 pontos sobre a face, além de anotações em separado para a posição dos olhos. Na Figura 4.4 são apresentados exemplos de imagens desta base.

Figura 4.4: Exemplos de imagens da base BioID

4.3.2 Validação

Para medir o desempenho dos classificadores é necessário um critério de avaliação dos resultados obtidos. Definido o critério, para avaliar a capacidade de generalização dos mesmos, é feita a validação dos classificadores em conjuntos de dados não utilizados no treinamento, de forma a garantir independência estatística dos mesmos.

A distância entre a saída do classificador e a marcação manual do ponto fiducial foi escolhida para mensurar o desempenho dos classificadores neste contexto. Para fornecer uma medida padronizada, invariante à escala da imagem, a distância será expressa como um percentual relativo da distância entre as marcações das pupilas.

Originalmente definido para detecção de faces e olhos [13, 50, 51] e generalizado para outros pontos fiduciais [6, 7, 11], este critério foi escolhido por ser amplamente utilizado, permitindo futuras comparações.

Dada às marcações manuais das pupilas, $\mathbf{p}_l$ e $\mathbf{p}_r$, a marcação manual para o ponto fiducial alvo, $\mathbf{p}_f$, e a saída estimada, $\hat{\mathbf{p}}_f$, a expressão da distância relativa à distância interocular é [6, 7]

$$d_{io} = \frac{\|\mathbf{p}_f - \hat{\mathbf{p}}_f\|}{\|\mathbf{p}_l - \mathbf{p}_r\|}. \tag{4.6}$$

Os resultados do processo de detecção serão avaliados segundo a Equação (4.6), sendo considerados acertos os casos onde $d_{io} \leq 0,05$, ou seja, 5% da distância interocular, distância que corresponde ao diâmetro da pupila [51]. Para fins de comparação, $d_{io} \leq 0,25$ equivale à distância entre os cantos interno e externo de um olho e $d_{io} \leq 0,10$ corresponde ao diâmetro da íris. Algumas destas distâncias são ilustradas na Figura 4.5.

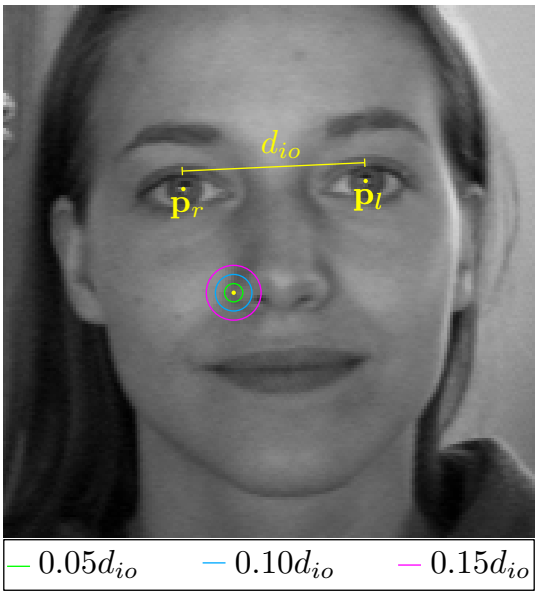


Figura 4.5: Distância interocular

Uma vez definido o critério de avaliação, é necessário definir de que forma será realizada a validação estatística dos detectores. Neste trabalho adotou-se a técnica conhecida como *validação cruzada* [31, 41]. Neste procedimento particiona-se o conjunto de dados amostrais em subconjuntos complementares. Na partição denominada *conjunto de treinamento* realiza-se o processo de aprendizado e na parcela complementar, denominada *conjunto de teste*, realiza-se a validação.

Dentre das diversas estratégias de validação cruzada, escolheu-se a conhecida como *k-fold* [31, 41], onde as amostras são divididas em k subconjuntos. Destes subconjuntos, $k-1$ compõe o conjunto de treinamento e o restante o de validação. Este processo é repetido k vezes, de forma rotativa, permitindo que todas as amos-

tras sejam utilizadas um vez para validação. O desempenho então é expresso pela média e desvio padrão dos resultados dos *folds*. Este procedimento é ilustrado pela Figura 4.6. Neste trabalho adotou-se $k = 10$.

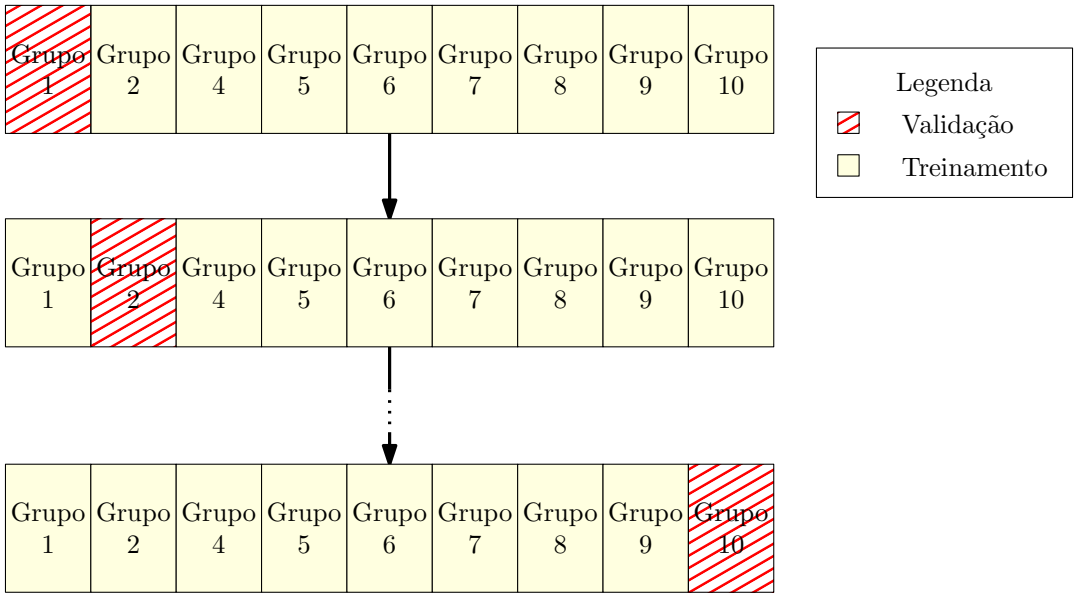


Figura 4.6: Exemplo de *k-folds* com dez partições

4.3.3 Treinamento

Apesar de cada tipo de detector requerer um treinamento distinto, algumas etapas do processo de treinamento são comuns a todos os métodos. Um diagrama com todas as etapas é apresentado na Figura 4.7. Resumidamente, considerando que as imagens já sofreram o processo de escalamento e segmentação descrito na Seção 4.2.1, computa-se a ROI utilizando as marcações manuais do conjunto de treinamento (Seção 4.2.1). Em seguida, realiza-se a extração das características (Seção 4.2.1). Em todos os casos, apenas amostras correspondentes a pontos pertencentes à ROI definida na Equação (4.4) são utilizadas durante o treinamento dos detectores. Em seguida é realizado o treinamento dos detectores. Os detectores são gerados a partir de seus respectivos processos de treinos, como descritos no Capítulo 2. No entanto, nesta seção serão brevemente apresentadas algumas questões práticas de cada método.

Os detectores IPD possuem múltiplas formas de treinamento, descritas na Seção 2.2. Excetuando o método de treinamento do IPD que o considera um regressor (ver Equação (2.27) e discussão que a acompanha), para fim de treinamento, são considerados pertencentes à classe alvo a marcação manual e seus vizinhos diretos de um pixel (vizinhança 8-conectada [52]). Em particular, no caso MIL, a marcação manual e seus vizinhos 8-conectados compõe o *bag* positivo. Cada instância negativa corresponde, individualmente, a um *bag* negativo. Já no caso de regressão, o rótulo

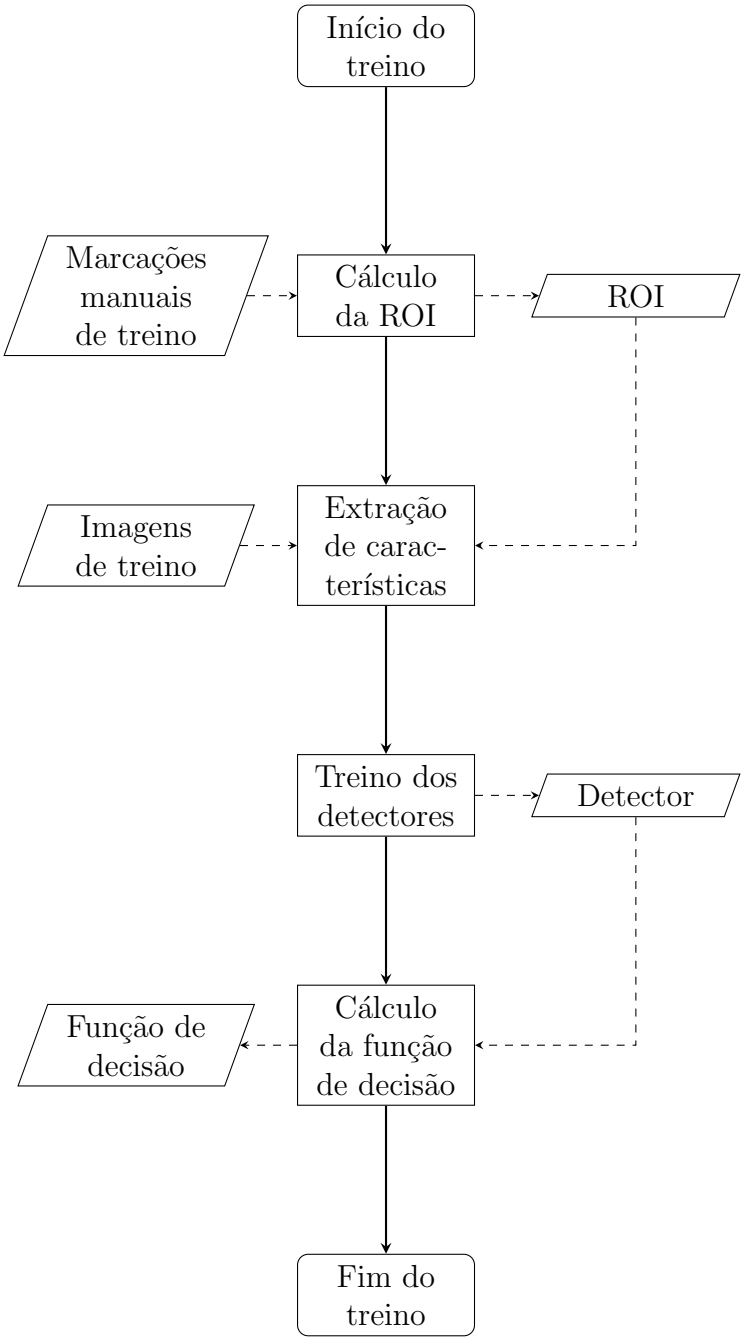


Figura 4.7: Processo de treinamento dos detectores.

y_i da i-ésima amostra de treinamento é definido a partir de uma gaussiana

$$y_i = \frac{1}{2\pi\sigma_p} \mathrm{e}^{\frac{-\|\mathbf{p}_f - \mathbf{p}_i\|^2}{2\sigma_p^2}}, \tag{4.7}$$

onde $\mathbf{p}_f$ é a posição da marcação manual, $\mathbf{p}_i$ é o i-ésimo ponto pertencente a ROI, e σ_p^2 é a variância da gaussiana, que define o espalhamento e decaimento dos pesos. Neste trabalho escolheu-se empiricamente $\sigma_p^2 = (\sigma_k I_z)^2$, onde $\sigma_k = 0.005$ e I_z é o tamanho da imagem.

O treinamento dos detectores utilizando filtros discriminativos é feito de forma semelhante, utilizando os mesmos métodos de extração de característica que o IPD, contudo realizando a operação de detecção no domínio da frequência, onde a mesma é mais rápida.

O treinamento dos filtros de correlação é feito centrando a resposta desejada do filtro $\mathbf{v}$ na marcação manual. No caso do UMACE, esta resposta é um impulso. No caso do MOSSE e ASEF, utilizou-se uma gaussiana assim como a descrita na Equação (4.7), com o mesmo formato. Além disso, adicionou-se um fator de regularização proporcional a $\epsilon = 2e10^{-8}$ e traço do denominador às equações (2.78), (2.77) e (2.81), caso em que elas se tornam, respectivamente:

$$\mathbf{H} = \frac{\sum_{i=1}^{N_s} \odot \mathbf{G}_i}{\mathbf{Q} + \mathrm{tr}\,(\mathbf{Q})\,\epsilon}, \tag{4.8}$$

$$\mathbf{H} = \frac{\sum_{i=1}^{N_s} \mathbf{V}_i^* \odot \mathbf{G}_i}{\mathbf{Q} + \mathrm{tr}\,(\mathbf{Q})\,\epsilon}, \tag{4.9}$$

e

$$\mathbf{H} = \frac{1}{N_s} \sum_{i=1}^{N_s} \frac{\mathbf{V}_i^* \odot \mathbf{G}_i}{\mathbf{U}_i + \mathrm{tr}\,(\mathbf{U}_i)\,\epsilon} \tag{4.10}$$

onde $\mathbf{U}_i = \mathbf{G}_i^* \odot \mathbf{G}_i$ e $\mathbf{Q} = \sum_{i=1}^{N_s} \mathbf{U}_i$.

Na Tabela 4.4 são apresentadas as complexidades computacionais de cada método para o treino sobre todo o conjunto de treinamento e para o teste em uma única imagem. Em relação ao treino, o IPD é o mais complexo, seguido do filtro discriminativo e os filtros de correlação. Isto é devido a natureza local do detector IPD e o cálculo da matriz de correlação (Equação (2.21)). Esses resultados corroboram com o tempo de treino mensurado empiricamente, de alguns dias, para o IPD, algumas horas, para os filtros discriminativos, e alguns minutos, para os filtros de correlação. No entanto, em relação ao teste em uma única imagem os resultados são semelhantes, dado que a dimensão das amostras para o IPD e para o filtro discriminativo N_D é proporcional ao tamanho do bloco ou ao HOG, enquanto a mesma é proporcional a dimensão da imagem para os filtros de correlação, compensando o fato que os primeiros necessitam varrer os pontos da ROI.

Tabela 4.4: Análise da complexidade computacional de cada um dos métodos para treino e teste de uma única imagem, onde N_I é o número de imagens de treino; N_P o número de pontos na ROI; e N_D a dimensão das amostras.

Método	Treino	Teste
IPD	$\approx O\left(N_I N_P N_D^2\right)$	$\approx O\left(N_P N_D\right)$
Filtro Discriminativo	$\approx O\left(N_I N_P N_D\right)$	$\approx O\left(N_P N_D\right)$
Filtros de Correlação	$\approx O\left(N_I N_D\right)$	$\approx O\left(N_D\right)$

Após o treinamento dos detectores, os mesmos são utilizados para gerar as funções de decisão correspondentes. Para este fim são computadas as respostas de cada amostra de treinamento em relação ao detector gerado. Essas respostas são usadas no cálculo das estatísticas necessárias para gerar as funções de decisão, como descrito na Seção 2.4, completando o processo de treinamento. Esse processo é repetido para cada ponto e para cada *fold*.

4.4 Resultados

Nesta seção são apresentados e discutidos os resultados obtidos para a detecção de 20 pontos fiduciais da base de dados BioID, apresentados na Figura 4.8, utilizando as técnicas de detecção implementadas.

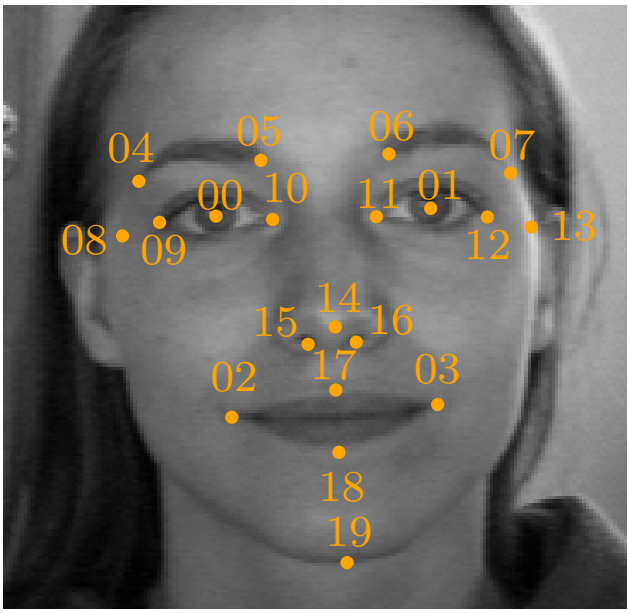


Figura 4.8: Numeração dos pontos fiduciais da base BioID

4.4.1 Resultados com IPD

Na Tabela 4.5 são apresentadas média e desvio padrão calculados a partir dos *folds* para taxas de acerto com 5% da distância interocular para pontos e algoritmos de treinamento distintos. Os rótulos denominam:

1. SMP – implementação original, ou "simples" (Seção 2.2);
2. RGR – algoritmo proposto de treinamento por regressão (Seção 2.2.2);
3. MIL – contribuição inspirada no paradigma de *Multiple Instance Learning* (Seção 2.2.3);
4. BST – algoritmo de treinamento utilizando *boosting* (Seção 2.2.4); e
5. BGG – treinamento através do método de *bagging* (Seção 2.2.5).

Excetuando o primeiro, todos os outros algoritmos são contribuições originais desta dissertação, inspirados em metodologias de treinamento conhecidas.

Os resultados apresentados foram obtidos utilizando o detector IPD tendo como amostras blocos da imagem. Examinando os resultados para os diferentes pontos, pode-se afirmar que:

- Para todos os métodos de treinamento, os melhores resultados obtidos foram para os pontos da região dos olhos (0, 1, 9, 10, 11, 12) e os pontos laterais do nariz (15, 16). Tais comportamentos eram esperados, dado que são pontos estáveis, com pequenas variações de estado e bem definidos;
- Em comparação, pontos da região da boca (2, 3, 17, 18) e o centro do nariz (14) apresentaram resultados intermediários. O que também era esperado devido à maior variedade de estados e qualidades na região da boca (fechada, aberta, pelos faciais, formato, etc...) e pela falta de características faciais salientes, no caso do centro do nariz;
- Os piores resultados encontrados foram para os pontos das sobrancelhas (4, 5, 6, 7), seguido pelos pontos externos (8, 13, 19). Pode-se inferir que as baixas taxas de acerto são decorrentes da ambiguidade inerente destas regiões, grande variabilidade de estados e qualidades, proximidade com outras características faciais mais relevantes, e ausência de características faciais salientes, tornando a detecção nestas regiões mais difícil. A pior taxa de acerto foi obtida para o ponto 19 sobre à ponta do queixo, região que dificilmente poderia ser considerada uma característica facial relevante, dado que sua definição é dúbia até mesmo para um observador humano.

- Enquanto quase todos os pontos que apresentam algum grau de simetria obtiveram taxas de acertos semelhantes, dois pontos externos, 8 e 13, apesar da simetria, possuem uma significativa diferença. Possíveis causas são um viés para um dos lados nas imagens da base, ROIs distintas ou algum objeto que produza ambiguidade em um dos lados.

Tabela 4.5: Taxa de acerto de detectores IPD por ponto fiducial para base BioID utilizando blocos da imagem. Em vermelho itálico, resultados onde ocorreu diferença estatística significativa em relação ao algoritmo IPD original com degradação no desempenho. Em negrito, onde ocorreu diferença estatística significativa com melhora.

Ponto	Método de treinamento				
	SMP	RGR	MIL	BST	BGG
00	88.6 ± 2.0	89.2 ± 2.1	88.1 ± 1.9	89.0 ± 2.3	88.6 ± 2.2
01	88.4 ± 1.6	88.8 ± 1.8	*87.2 ± 1.8*	*85.0 ± 2.5*	88.3 ± 1.7
02	65.9 ± 3.4	66.3 ± 4.0	65.2 ± 2.2	65.3 ± 3.2	65.5 ± 3.3
03	60.4 ± 2.9	*58.2 ± 4.0*	59.7 ± 3.4	*58.9 ± 2.4*	60.7 ± 3.3
04	48.3 ± 4.7	45.4 ± 5.2	44.8 ± 5.5	52.5 ± 4.9	47.7 ± 4.4
05	51.8 ± 3.8	51.2 ± 2.8	51.4 ± 3.2	52.0 ± 3.7	51.7 ± 3.5
06	52.9 ± 2.4	52.0 ± 3.5	52.2 ± 1.9	52.6 ± 3.1	52.3 ± 2.3
07	46.4 ± 2.9	45.8 ± 2.7	*45.0 ± 2.2*	47.8 ± 3.3	46.4 ± 2.9
08	40.6 ± 4.5	38.1 ± 5.5	*34.5 ± 5.2*	**45.9 ± 3.2**	40.8 ± 4.1
09	81.3 ± 1.9	81.8 ± 2.0	*80.1 ± 2.2*	**83.6 ± 1.6**	81.3 ± 1.9
10	80.5 ± 2.8	80.0 ± 3.3	80.1 ± 2.9	81.3 ± 3.0	80.5 ± 2.9
11	78.8 ± 4.0	78.6 ± 4.3	78.3 ± 4.2	78.4 ± 3.3	78.6 ± 4.1
12	80.7 ± 2.7	80.5 ± 1.7	79.8 ± 3.0	**83.6 ± 2.7**	80.5 ± 2.8
13	21.0 ± 3.1	21.2 ± 3.0	20.4 ± 3.7	21.5 ± 2.9	21.2 ± 2.8
14	56.0 ± 3.0	*53.2 ± 2.4*	55.0 ± 3.0	55.6 ± 3.2	56.3 ± 2.7
15	75.9 ± 2.6	**79.6 ± 2.8**	74.7 ± 2.9	*72.7 ± 2.9*	75.5 ± 2.5
16	76.8 ± 2.4	**80.0 ± 2.3**	76.3 ± 2.1	*74.4 ± 3.0*	77.1 ± 2.5
17	63.7 ± 4.8	60.4 ± 4.7	61.0 ± 3.9	**67.7 ± 3.8**	63.3 ± 4.3
18	56.1 ± 3.9	56.7 ± 3.6	55.5 ± 3.9	57.6 ± 3.4	55.8 ± 4.1
19	0.2 ± 0.3	*0.0 ± 0.0*	0.2 ± 0.3	*0.1 ± 0.2*	0.1 ± 0.3

Em relação às taxas de acertos para os diferentes métodos de treinamento, observa-se que, em geral, foram obtidos resultados semelhantes, mostrando que existe certa consistência entre os mesmos. Para avaliar se existe uma real diferença entre os resultados, os mesmos foram tratados utilizando uma variação do teste t de *Student* [53] para variâncias distintas, conhecida como *teste t de Welch* [53].

Um teste t é um teste de hipótese estatística no qual a estatística de teste segue uma distribuição t de Student caso a hipótese nula seja válida. O teste de Welch é uma adaptação que contempla o caso onde as duas amostras a serem comparadas podem possuir variâncias distintas.

O teste t de Welch define a estatística t a partir da seguinte equação

$$t = \frac{\mu_1 - \mu_2}{\sqrt{\frac{\sigma_1^2}{N_1} + \frac{\sigma_2^2}{N_2}}}, \tag{4.11}$$

onde μ_i, σ_i^2 e N_i são, respectivamente, a média, variância e número de amostras da i-ésima variável. O número de graus de liberdade v associado com a estimativa da variação é aproximado a partir da equação de Welch–Satterthwaite [53]

$$v \approx \frac{\left(\frac{\sigma_1^2}{N_1} + \frac{\sigma_1^2}{N_2}\right)^2}{\frac{\sigma_1^4}{N_1^2(N_1-1)} + \frac{\sigma_1^4}{N_2^2(N_2-1)}}. \tag{4.12}$$

Em seguida, os parâmetros t e v são utilizados para computar o coeficiente p relacionado as hipóteses. Neste caso, o interesse está em avaliar se existe diferença estatística real entre as médias das metodologias de treinamento, ou seja

$$\begin{cases} H_0, & \text{se } \mu_1 = \mu_2, \\ H_1, & \text{se } \mu_1 \neq \mu_2, \end{cases}$$

onde H_i são as duas hipóteses contempladas. Neste caso o teste é realizado extraindo o coeficiente p da distribuição t de Student com cauda dupla. A distribuição cumulativa t de Student é definida como [53]

$$\text{tcdf}(x, v) = F(x|v) = \int_{-\infty}^{x} \frac{\Gamma\left(\frac{v+1}{2}\right)}{\Gamma\left(\frac{v}{2}\right)} \frac{1}{\sqrt{v\pi}} \frac{1}{\left(1 + \frac{y^2}{v}\right)^{\frac{v+1}{2}}} \mathrm{d}y. \tag{4.13}$$

No caso do teste com cauda dupla, define-se o coeficiente p como

$$p = 2\left[1 - \text{tcdf}\left(|t|, v\right)\right]. \tag{4.14}$$

O coeficiente $p \in [0, 1]$ é uma medida de probabilidade da hipótese H_0. Ou seja, $p = 1$ indica 100% de confiança que $\mu_1 = \mu_2$. Na Tabela 4.6 são apresentados os coeficientes p comparando os algoritmos novos em relação à implementação original do IPD, primeira coluna da Tabela 4.5. Em negrito, estão os casos onde o coeficiente está abaixo de 0.5. Na Tabela 4.5 estes casos estão em negrito quando ocorre diferença estatística em relação ao original, e em vermelho quando essa diferença é resultante de uma piora.

Tendo tais informações, e ignorando os resultados do ponto 19 para fins de comparação, devido à baixa taxa de acerto, pode-se afirmar que:

- O método de *bagging* praticamente não produz diferenças em relação ao ori-

Tabela 4.6: Coeficiente p do teste t de Welch para detectores IPD por ponto fiducial para base BioID utilizando blocos da imagem.

Ponto	Método de treinamento			
	RGR	MIL	BST	BGG
00	0.599	0.665	0.736	0.953
01	0.660	**0.232**	**0.085**	0.876
02	0.873	0.669	0.779	0.828
03	**0.477**	0.718	**0.411**	0.889
04	0.612	0.594	0.515	0.863
05	0.765	0.861	0.936	0.977
06	0.667	0.587	0.871	0.695
07	0.754	**0.430**	0.517	0.969
08	0.658	**0.467**	**0.363**	0.949
09	0.657	**0.338**	**0.071**	0.946
10	0.805	0.815	0.665	0.967
11	0.946	0.873	0.871	0.962
12	0.873	0.631	**0.221**	0.933
13	0.914	0.769	0.799	0.909
14	**0.255**	0.618	0.836	0.817
15	**0.170**	0.501	**0.218**	0.783
16	**0.121**	0.672	**0.281**	0.845
17	0.555	0.556	**0.466**	0.906
18	0.798	0.826	0.594	0.923
19	**0.066**	1.000	**0.301**	0.641

ginal, dado que em nenhum dos pontos o coeficiente p ficou abaixo de 0.69. Assim, é possível afirmar que, neste caso, o detector IPD é um detector estável, pouco se beneficiando do *bagging*;

- Em contrapartida, o treinamento utilizando *boosting* foi o que produziu o maior número de diferenças estatísticas. Ocorreu piora para os pontos 1, 3, 15 e 16, e melhora para os pontos 8, 9 e 12;

- Empatados, com quatro casos de diferenças estatística, seguem o treinamento utilizando de regressão e a versão utilizando MIL. Enquanto o método MIL degradou a taxa de acerto em todos os casos onde ocorreu diferença (pontos 1, 7, 8, 9), o uso da regressão permitiu melhora em dois casos (pontos 15, 16), e piora no restantes (pontos 3, 14). É interessante notar que as duas abordagens tentam modelar, de forma distinta, a ambiguidade inerente das amostras;

- Curiosamente, existe uma ligeira relação entre variações no método de *boosting* e nos métodos de regressão e MIL, onde a taxa de acerto do primeiro degrada nos pontos em que ocorre melhora nos últimos. Dado que a regressão pode ser utilizada em conjunto com *boosting*, talvez essa relação possa ser estudada no futuro.

- A respeito do método MIL, neste trabalho foi implementada a aproximação *softmax*. A implementação da solução heurística iterativa apresentou problemas de convergência que tornavam a decisão de parada difícil, enquanto para a solução de programação semidefinida faltou encontrar uma biblioteca ou *toolbox* adaptável para o arcabouço proposto.

Os resultados utilizando HOG em conjunto com os detectores IPD, entretanto, foram muito aquém do esperado. Possíveis causas são problemas na implementação, descritores não representativos da amostra (devido à dimensão da região ou do descritor) ou simplesmente a falta de sinergia entre as características procuradas e a técnica de detecção. Lembrando que originalmente os descritores HOG são utilizados para detecção de objetos maiores, como seres humanos [45, 46], ou para registro em imagens, no caso do SIFT [44]. Uma característica desses objetos são bordas e quinas bem definidas, o que já não ocorre na dimensão dos pontos fiduciais.

Porém é interessante notar que existe uma consistente melhora na taxa de acerto utilizando as estratégias de *bagging* e regressão. Provavelmente esta melhora está relacionada à instabilidade dos detectores ao utilizar HOG, permitindo que os mesmo se beneficiem do aumento do número de amostras obtido por tais métodos. Entretanto, os resultados ainda são muitos ruins, bem inferiores aos obtidos utilizando blocos das imagens (Tabela 4.5).

Tabela 4.7: Taxa de acerto de detectores IPD por ponto fiducial para base BioID utilizando HOG.

Ponto	Método de treinamento				
	SMP	RGR	MIL	BST	BGG
00	5.1 ± 4.6	15.0 ± 4.0	5.1 ± 4.6	2.5 ± 2.5	21.5 ± 4.2
01	4.4 ± 5.1	18.9 ± 3.4	4.4 ± 5.1	4.4 ± 5.7	19.3 ± 3.2
02	2.4 ± 2.6	4.5 ± 2.0	2.4 ± 2.6	1.5 ± 1.4	5.0 ± 1.5
03	2.6 ± 2.9	6.0 ± 1.9	2.4 ± 2.8	1.5 ± 1.3	5.8 ± 2.1
04	2.4 ± 2.4	3.6 ± 1.5	2.4 ± 2.2	1.8 ± 2.0	6.2 ± 3.1
05	1.8 ± 1.7	3.6 ± 1.4	1.8 ± 1.7	0.7 ± 0.8	9.3 ± 1.8
06	1.1 ± 0.9	1.1 ± 0.7	1.1 ± 0.8	1.1 ± 1.3	7.1 ± 2.3
07	1.5 ± 1.1	5.4 ± 1.8	1.6 ± 1.0	1.5 ± 0.9	6.3 ± 2.0
08	1.8 ± 1.5	2.6 ± 0.8	1.9 ± 1.5	1.4 ± 1.6	4.3 ± 1.5
09	3.5 ± 3.8	22.2 ± 2.5	3.4 ± 3.6	0.7 ± 0.9	20.4 ± 2.8
10	2.7 ± 1.3	10.2 ± 2.0	2.8 ± 1.4	3.6 ± 2.9	12.2 ± 2.5
11	3.4 ± 2.6	17.7 ± 3.4	3.2 ± 2.8	2.5 ± 2.7	19.1 ± 2.8
12	2.0 ± 2.5	16.5 ± 4.4	2.1 ± 2.5	2.4 ± 2.8	15.8 ± 2.4
13	0.5 ± 0.7	5.5 ± 1.4	0.5 ± 0.7	0.5 ± 0.5	6.8 ± 1.3
14	1.2 ± 0.7	1.8 ± 0.8	1.3 ± 0.8	1.2 ± 1.4	5.4 ± 2.2
15	3.9 ± 3.1	7.7 ± 1.4	3.9 ± 3.1	3.4 ± 2.8	7.7 ± 2.3
16	3.0 ± 2.4	7.0 ± 1.4	3.0 ± 2.5	1.7 ± 1.9	7.9 ± 1.8
17	1.4 ± 1.0	3.7 ± 1.6	1.4 ± 1.0	1.1 ± 0.8	4.7 ± 1.8
18	1.6 ± 1.4	4.6 ± 1.8	1.6 ± 1.5	1.4 ± 2.1	5.7 ± 1.5
19	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0	0.0 ± 0.0

4.4.2 Resultados com Filtro Discriminativo

Na Tabela 4.8 são exibidas média e desvio padrão das taxas de acerto à 5% da distância interocular para os filtros discriminativos, computadas a partir dos *folds*. São apresentados resultados para os dois casos de pré-processamento: IMGP – blocos das imagens; e – IHOG – utilizando HOG integral. Os valores obtidos foram abaixo do esperado, com taxas de acerto para os pontos dos olhos, considerados pontos fáceis, em torno de 50%, no melhor caso, e muito inferiores para os pontos restantes, em ambos os casos.

Como este detector é rápido para treinar, mesmo para dimensões altas, utilizaram-se descritores HOG completos, de dimensão 1.152, na expectativa de melhoria nos resultados. Ocorreu melhora, dado que as taxas de acerto originais estavam na faixa de 1% acerto. Ainda assim, eles estão aquém do que seria aceitável, corroborando com a afirmação anterior a respeito do desempenho dos descritores HOG neste problema.

Dentre as possíveis causas pode-se apontar a dimensão dos blocos e dos descritores, que escolhida para obter os melhores resultados para os classificadores IPD, não necessariamente a melhor para os filtros discriminativos; e a incompatibilidade com o problema proposto. Apesar da taxa de acerto baixa, estes resultados são coerentes com os encontrados em [6] utilizando a mesma técnica em um subconjunto de imagens da mesma base.

4.4.3 Resultados com Filtros de Correlação

Na Tabela 4.9 são apresentadas médias e desvios padrão calculados a partir dos *folds* para a taxas de acerto para 5% da distância interocular para os pontos da BioID utilizando os filtros de correlação para a detecção. Além disso, dado que esses detectores realizam a detecção em toda a imagem, decidiu-se comparar os resultados com e sem o uso ROI para restringir a área de busca. Analisando os resultados, observa-se que:

- Os filtros de correlação obtiveram bons resultados, principalmente o MOSSE e o UMACE, superando o IPD em muitos casos;
- Os melhores resultados foram encontrados para os pontos da região dos olhos (0, 1, 9 10, 11, 12), nariz (15, 16) e, algo surpreendente, da boca (17), todos com taxas de acerto acima de 75%. Este comportamento é semelhante ao encontrado com o IPD, exceto pelo ponto da boca, de dificuldade intermediária;
- Nota-se que, enquanto em alguns casos a ROI produzia um pequena melhora, em outros casos, como os pontos 13 e 19, o uso da ROI degradou gravemente

Tabela 4.8: Taxa de acerto para os filtros discriminativos para base BioID.

Ponto	Pré-processamento	
	IMGP	IHOG
00	59.4 ± 4.6	6.0 ± 1.8
01	48.1 ± 1.5	4.8 ± 2.3
02	4.5 ± 1.1	3.9 ± 2.4
03	2.8 ± 1.0	2.6 ± 1.4
04	20.2 ± 3.9	3.5 ± 2.3
05	3.1 ± 1.5	4.5 ± 1.6
06	1.8 ± 1.0	5.1 ± 1.7
07	13.3 ± 2.0	3.7 ± 1.6
08	3.0 ± 1.6	2.8 ± 1.2
09	8.6 ± 3.8	5.4 ± 2.6
10	0.5 ± 0.5	3.4 ± 1.2
11	1.8 ± 1.3	3.2 ± 1.5
12	16.2 ± 3.4	4.3 ± 1.4
13	6.5 ± 2.1	0.7 ± 0.7
14	14.2 ± 3.0	4.1 ± 1.5
15	20.0 ± 3.9	3.6 ± 1.0
16	17.0 ± 4.4	4.0 ± 1.5
17	13.9 ± 2.3	3.6 ± 1.9
18	0.5 ± 0.6	4.3 ± 2.3
19	0.0 ± 0.0	0.1 ± 0.2

o processo de detecção. Tais taxas de acerto são consistentes com as baixas taxas de acerto obtidas para os mesmos pontos utilizando detectores IPD (Tabela 4.7) e apontam que, futuramente, tal estratégia precisa ser revista, dado que a ROI do conjunto de treino não é representativa durante a validação;

- Para o restante dos pontos o comportamento foi similar ao encontrado para o IPD, com taxas de acerto equivalentes ou melhores.

Tabela 4.9: Taxa de acerto para os métodos de filtros de correlação por ponto fiducial para base BioID

Ponto	Detector					
	UMACE		ASEF		MOSSE	
	Sem ROI	Com ROI	Sem ROI	Com ROI	Sem ROI	Com ROI
00	91.8 ± 1.9	92.8 ± 1.7	91.7 ± 2.1	91.8 ± 2.1	92.6 ± 2.3	93.0 ± 2.1
01	90.4 ± 1.3	91.8 ± 1.2	90.8 ± 1.7	91.4 ± 1.7	91.1 ± 1.2	92.0 ± 1.2
02	66.2 ± 5.1	67.1 ± 5.3	65.7 ± 6.3	65.8 ± 6.1	67.0 ± 5.4	67.5 ± 5.3
03	67.2 ± 2.8	67.5 ± 3.0	63.8 ± 2.7	63.8 ± 2.7	67.1 ± 2.7	67.3 ± 3.0
04	46.2 ± 3.8	46.2 ± 3.8	44.8 ± 4.3	44.8 ± 4.3	46.1 ± 3.2	46.1 ± 3.2
05	50.2 ± 4.7	50.2 ± 4.7	49.7 ± 2.7	49.7 ± 2.7	49.6 ± 4.0	49.6 ± 4.0
06	46.3 ± 2.4	46.3 ± 2.4	46.7 ± 2.0	46.7 ± 2.0	46.0 ± 2.3	46.1 ± 2.3
07	51.9 ± 3.9	45.3 ± 4.7	49.6 ± 4.1	43.9 ± 3.9	51.9 ± 4.3	46.1 ± 4.9
08	49.4 ± 3.5	49.4 ± 3.5	47.6 ± 4.0	47.6 ± 4.0	50.4 ± 3.7	50.5 ± 3.8
09	84.7 ± 2.2	86.1 ± 2.2	84.1 ± 2.0	84.2 ± 2.1	85.3 ± 2.2	85.8 ± 2.3
10	86.8 ± 2.6	87.5 ± 2.5	86.8 ± 2.7	86.9 ± 2.8	87.1 ± 2.6	87.3 ± 2.8
11	88.2 ± 1.4	88.4 ± 1.3	88.0 ± 0.9	88.2 ± 1.0	88.4 ± 1.6	88.5 ± 1.6
12	83.6 ± 2.7	84.2 ± 2.4	83.0 ± 3.9	83.1 ± 3.8	84.3 ± 2.5	84.6 ± 2.5
13	55.2 ± 3.3	23.7 ± 3.3	52.1 ± 2.8	23.8 ± 4.1	55.3 ± 3.4	24.6 ± 3.8
14	61.4 ± 4.0	61.4 ± 4.0	61.1 ± 3.7	61.1 ± 3.7	61.1 ± 4.3	61.1 ± 4.3
15	88.9 ± 3.4	90.9 ± 3.1	90.3 ± 3.3	90.5 ± 3.1	89.7 ± 3.6	90.9 ± 3.4
16	89.3 ± 2.3	89.9 ± 2.4	91.3 ± 1.5	91.5 ± 1.6	89.2 ± 2.5	89.5 ± 2.6
17	76.7 ± 3.5	77.3 ± 3.4	75.1 ± 2.8	75.3 ± 2.7	77.1 ± 3.3	77.3 ± 3.3
18	56.4 ± 4.0	56.4 ± 4.1	55.4 ± 4.5	55.4 ± 4.5	57.5 ± 3.6	57.4 ± 3.7
19	51.9 ± 5.0	0.3 ± 0.5	49.6 ± 3.2	0.5 ± 0.6	52.7 ± 4.5	0.4 ± 0.5

Para comparar os diferentes métodos, na Figura 4.9 são apresentados as taxas de acerto para 5% da distância interocular com os respectivos desvios padrão para cada método e para cada ponto. A ilustração corrobora com as afirmações prévias, onde observa-se que o comportamento dos métodos é semelhante para cada ponto, com uma pequena vantagem para os filtros de correlação na maioria dos casos, exceto nos pontos 5 e 6, pontos internos das sobrancelhas direita e esquerda.

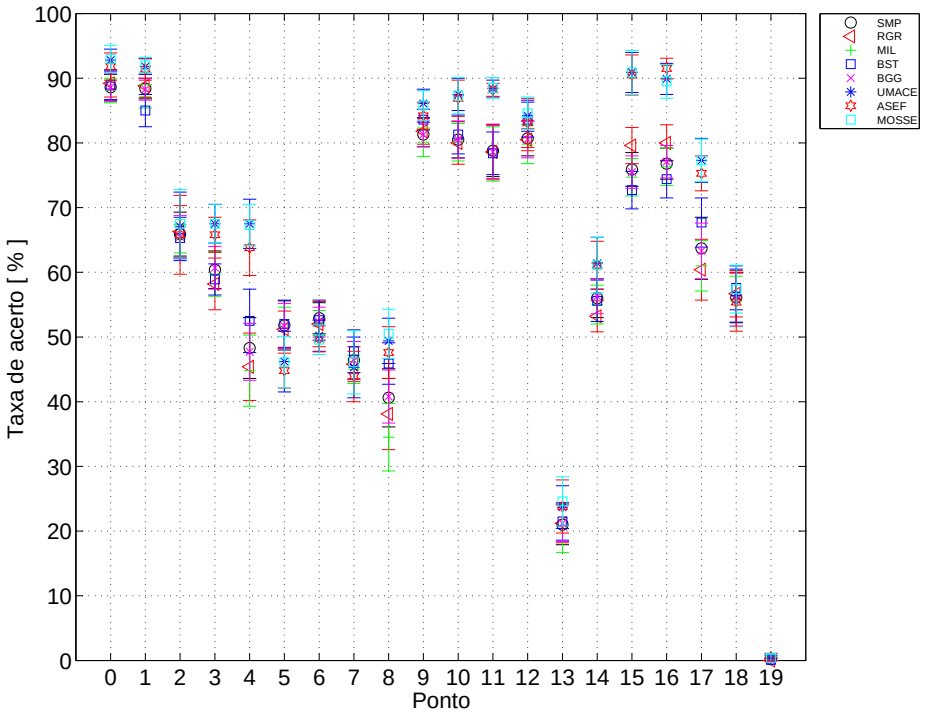


Figura 4.9: Comparação entre os métodos IPD e filtros de correlação

4.5 Conclusões

Neste capítulo são apresentados os resultados obtidos utilizando o conjunto de classificadores no contexto de detecção de pontos fiduciais em faces humanos para imagens estatísticas. Descreveu-se o experimento proposto e discutiu-se os resultados obtidos.

A respeito dos resultados obtidos utilizando os detectores IPD, pode-se afirmar que as modificações propostas no algoritmo de treinamento, apesar de não produzirem mudanças significativas em todos os casos, permitiram explorar características interessantes do problema e apontam para novas possíveis modificações no futuro, como a integração das técnicas de regressão e *boosting*.

Os filtros discriminativos apresentaram desempenho muito inferior ao das outras técnicas estudadas. Em [6], para contornar as taxas de acerto baixas, propôs-se uma alteração na qual o detector era treinado sobre característica extraídas a partir da decomposição das amostras em componentes principais através do método PCA [31]. Esta abordagem ofereceu uma grande melhora se comparada com a apresentada neste trabalho.

Em relação aos resultados para filtros de correlação, a análise dos resultados permitiu não só observar a viabilidade da aplicação dos mesmos no contexto proposto,

como demonstrou uma deficiência no sistema de restrição da área de busca que, em alguns casos, pode provocar degradações no desempenho global do sistema.

Na Tabela 4.10 é realizada uma comparação entre três métodos atuais [4, 54, 55] com os resultados obtidos neste trabalho para o classificador IPD tradicional e para os filtros UMACE, ASEF e MOSSE, apenas para o pior dentre os pontos da pupila. Os métodos estão ordenados a partir da taxa de acerto, a 5% da distância interocular, de forma descendente. Através dessa comparação podemos afirmar que tanto os detectores IPD quanto os filtros de correlação demonstram boas taxas de acerto com complexidade inferior à encontrada em outros métodos, e com tempo de treinamento reduzido (algumas horas, para o IPD, e alguns minutos para os filtros de correlação). Os outros pontos não foram comparados devido a falta de resultados quantitativos disponíveis na literatura, pelo uso de diferentes métricas ou métodos que impossibilitam a comparação.

Tabela 4.10: Comparação de desempenho entre diferentes métodos para os pontos das pupila na base de dados BioID à 5% da distância interocular.

Método	Taxa de acerto ($d_{io} \leq 0.05$)
MOSSE	92.0
UMACE	91.8
ASEF	91.4
IPD	88.4
Valenti e Gevers [54]	86.1
Timm e Barth [55]	82.5
Ren et al. [4]	77.1

Capítulo 5

Conclusões

Neste trabalho avaliou-se um conjunto de filtros no contexto de detecção de pontos fiduciais em faces. Analisou-se técnicas distintas, novas abordagens de treinamento foram propostas e observou-se o desempenho dos mesmos ao classificar conjuntos distintos de características extraídos das mesmas amostras.

Além disso, implementou-se um arcabouço que é capaz de treinar e utilizar as diferentes técnicas aqui apresentadas em inúmeros contextos. O sistema permite, de forma flexível, selecionar as partes constituintes do sistema de classificação.

O sistema foi completamente implementado em linguagem *C++* de programação, fazendo uso da biblioteca de visão computacional *OpenCV*. Tal implementação permite o uso do sistema em tempo real, e em alguns casos, o treinamento completo dos classificadores ocorre em questão de minutos.

Pode-se afirmar, em relação aos resultados obtidos, que dentre o conjunto de detectores aqui estudados, os filtros de correlação no domínio da frequência, UMACE, ASEF e MOSSE, apresentaram os melhores resultados. Entretanto, o IPD também obteve boas taxas de acerto, em alguns pontos até mesmo melhores, porém possui um algoritmo de treinamento mais lento e sofre com os problemas da ROI, que possivelmente deverá ser modificada futuramente. Infelizmente, o uso de HOG e ou os filtros discriminativos, em relação a implementação utilizada, não produziu resultados aceitáveis. Porém, provavelmente podem ter melhores resultados em outros contextos, como detecção de pedestre, de objetos, etc...

5.1 Trabalhos Futuros

A seguir, alguns tópicos de investigação futura:

- Detecção em vídeo: Este é um projeto já em andamento, onde os detectores, aliados com métodos de rastreamento que exploram a consistência temporal e espacial entre quadros, são utilizados para realizar a detecção em vídeo. Em

alguns casos, os mesmos foram adaptados para o treinamento *online*, permitindo a adaptação dos detectores às variações de características do alvo ao longo do tempo.

Os resultados obtidos são promissores, com boas taxas de acerto mesmo em situações de oclusão e movimentos bruscos do alvo.

- Ajuste automático dos parâmetros dos classificadores: Muitos dos classificadores aqui apresentados possuem um conjunto de parâmetros que necessitam de ajuste. Atualmente, a otimização destes parâmetros ocorre de forma heurística, através do conhecimento a priori do problema; ou através de métodos de busca em *grid*; ou, em alguns casos, através do ajuste manual.

  Inicialmente, tais abordagens foram escolhidas devido às limitações de tempo e número de amostras na base de dados, que tornavam inviável o uso de técnicas mais complexas ou validação através da divisão da base em três grupos: treino, validação e teste. Futuramente, o sistema deverá ser modificado para contemplar este novo formato, e novos algoritmos podem ser utilizados, como por exemplo, técnicas de otimização natural.

- Detecção hierárquica: Dada que a maioria das técnicas aqui apresentadas são rápidas o suficiente para realizar múltiplas detecções na mesma imagem em um tempo curto, a detecção poderia ser realizada de forma multi-escala, permitindo identificar a posição inicialmente de forma grosseira e progressivamente aprimorando a estimativa através de escalas mais finas.

- Abordagem baseada em modelos: No caso específico da detecção de pontos fiduciais em faces humanas, a geometria da face é uma informação que pode ser adicionada ao problema no intuito de tornar mais precisa a detecção da posição das características faciais, semelhante ao apresentado em [1].

Referências Bibliográficas

[1] BELHUMEUR, P. N., JACOBS, D. W., KRIEGMAN, D. J., et al. "Localizing Parts of Faces Using a Consensus of Exemplars". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'2011)*, Junho 2011.

[2] DANTONE, M., GALL, J., FANELLI, G., et al. "Real-time Facial Feature Detection Using Conditional Regression Forests", *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'2012)*, pp. 2578–2585, Junho 2012.

[3] BOLME, D. S., DRAPER, B. A., BEVERIDGE, J. R. "Average of Synthetic Exact Filters". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'2009)*, Junho 2009.

[4] REN, Y., WANG, S., HOU, B., et al. "A Novel Eye Localization Method With Rotation Invariance", *IEEE Transactions on Image Processing*, v. 23, n. 1, pp. 226–239, Janeiro 2014.

[5] ARAUJO, G. M., DA SILVA, E. A. B., CIÂNCIO, A. G., et al. "Integration of eye detection and tracking in videoconference sequences using temporal consistency and geometrical constraints". In: *Proceeding of 19th IEEE International Conference on Image Processing (ICIP)*, pp. 421–424, Setembro 2012.

[6] DA SILVA JÚNIOR, W. S. *Reconhecimento de Padrões Utilizando Filtros de Correlação com Análise de Componentes Principais.* Tese de D. Sc., COPPE/UFRJ, Rio de Janeiro, RJ, Brasil, 2010.

[7] ARAUJO, G. M. *Algoritmo para reconhecimento de características faciais baseado em filtros de correlação.* Dissertação de M. Sc., COPPE/UFRJ, Rio de Janeiro, RJ, Brasil, 2010.

[8] VIOLA, P., JONES, M. "Rapid Object Detection using a Boosted Cascade of Simple Features", *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'2001)*, v. 1, pp. 511–518, Dezembro 2001.

[9] LUCAS, B. D., KANADE, T. "An iterative image registration technique with an application to stereo vision". In: *Proceedings of International Joint Conference on Artificial Intelligence*, pp. 674–679, 1981.

[10] ARAUJO, G. M., DA SILVA, E. A. B., CIÂNCIO, A. G., et al. "Rastreamento Robusto de Olhos Usando Consistência Temporal e Restrições Geométricas". In: *XXXI Simpósio Brasileiro de Telecomunicações (SBrT'2013)*, Setembro 2013.

[11] RIBEIRO, F. M. L., ARAUJO, G. M., DA SILVA, E. A. B., et al. "Detecção de Pontos Fiduciais sobre a Face em Tempo Real". In: *XXX Simpósio Brasileiro de Telecomunicações (SBrT'2012)*, Setembro 2012.

[12] OPENCV. "OpenCV: Open Computer Vision Library". `http://opencv.org/`, 2014. último acesso em Fevereiro de 2014.

[13] JESORSKY, O., KIRCHBERG, K. J., FRISCHHOLZ, R. W. "Robust Face Detection Using the Hausdorff Distance". In: *AVBPA '01: Proceedings of the Third International Conference on Audio and Video-Based Biometric Person Authentication*, pp. 90–95. Springer, 2001.

[14] MAHALANOBIS, A., KUMAR, B. V. K. V., SONG, S., et al. "Unconstrained correlation filters", *Appl. Opt.*, v. 33, n. 17, pp. 3751–3759, Junho 1994.

[15] BOLME, D. S., BEVERIDGE, J. R., DRAPER, B. A., et al. "Visual object tracking using adaptive correlation filters". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'2010)*, Junho 2010.

[16] ANDREWS, S. J. D. *Learning from ambiguous examples.* PhD thesis, Brown University, Providence, RI, USA, 2007.

[17] BABENKO, B., DOLLÁR, P., TU, Z., et al. "Simultaneous Learning and Alignment: Multi-Instance and Multi-Pose Learning". In: *Faces in Real-Life Images*, Outubro 2008.

[18] SCHÖLKOPF, B., HERBRICH, R., SMOLA, A. J. "A Generalized Representer Theorem". In: *Proceedings of the 14th Annual Conference on Computational Learning Theory and and 5th European Conference on Computational Learning Theory*, pp. 416–426. Springer-Verlag, 2001.

[19] RIFKIN, R., YEO, G., POGGIO, T. "Regularized Least Squares Classification". In: Suykens, Horvath, Basu, et al. (Eds.), *Advances in Learning Theory: Methods, Model and Applications*, v. 190, *NATO Science Series*

III: Computer and Systems Sciences, VIOS Press, cap. 7, pp. 131–154, 2003.

[20] VAPNIK, V. N. *Statistical Learning Theory*. 1 ed. New York, NY, USA, Wiley, 1998.

[21] GIROSI, F. *An Equivalence Between Sparse Approximation and Support Vector Machines*. Relatório técnico, Massachusetts Institute of Technology, 1997.

[22] HASTIE, T. J., TIBSHIRANI, R. J., FRIEDMAN, J. H. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. Springer Series in Statistics. 2 ed. New York, NY, USA, Springer, 2009.

[23] BOYD, S., VANDENBERGHE, L. *Convex Optimization*. New York, NY, USA, Cambridge University Press, 2004.

[24] ANTONIOU, A., LU, W. *Practical Optimization: Algorithms and Engineering Applications*. 1 ed. New York, NY, USA, Springer Publishing Company, Incorporated, 2007.

[25] FRIEDMAN, J., HASTIE, T., TIBSHIRANI, R. "Additive logistic regression: a statistical view of boosting", *The Annals of Statistics*, v. 28, n. 2, pp. 337–374, Abril 2000.

[26] FREUND, Y., SCHAPIRE, R. E. "Experiments with a new boosting algorithm". In: *Thirteenth International Conference on Machine Learning*, v. 0, pp. 148–156, Bary, Italy, 1996.

[27] BREIMAN, L. "Bagging predictors", *Machine Learning*, v. 24, n. 2, pp. 123–140, Agosto 1996.

[28] POLIKAR, R. "Ensemble Based Systems in Decision Making", *IEEE Circuits and Systems Magazine*, v. 6, n. 3, pp. 21–45, 2012.

[29] OZA, N. C. *Online Ensemble Learning*. PhD thesis, The University of California, Berkeley, CA, USA, Setembro 2001.

[30] SAFFARI, A., LEISTNER, C., SANTNER, J., et al. "On-line Random Forests". In: *3rd IEEE ICCV Workshop on On-line Computer Vision*, 2009.

[31] BISHOP, C. M. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. 1 ed. Secaucus, NJ, USA, Springer-Verlag New York, Inc., 2006.

[32] BARTLETT, M. S. "An Inverse Matrix Adjustment Arising in Discriminant Analysis", *The Annals of Mathematical Statistics*, v. 22, n. 1, pp. 107–111, Março 1951.

[33] KUMAR, B. V. K. V., MAHALANOBIS, A., JUDAY, R. D. *Correlation Pattern Recognition*. Cambridge, UK, Cambridge University Press, 2005.

[34] LUGT, A. V. "Signal detection by complex spatial filtering", *IEEE Transactions on Information Theory*, v. 10, n. 2, pp. 139–145, Abril 1964.

[35] HESTER, C. F., CASASENT, D. "Multivariant technique for multiclass pattern recognition", *Applied Optics*, v. 19, n. 11, pp. 1758–1761, Junho 1980.

[36] HENRIQUES, J. F., CASEIRO, R., MARTINS, P., et al. "Exploiting the Circulant Structure of Tracking-by-detection with Kernels". In: *Proceedings of the 12th European conference on Computer Vision (ECCV'12)*, Florence, Italy, 2012.

[37] BEN-ARIE, J., RAO, K. R. "A novel approach for template matching by nonorthogonal image expansion", *IEEE Transactions on Circuits and Systems for Video Technology*, v. 3, n. 1, pp. 71–84, Fevereiro 1993.

[38] RODRIGUEZ, A., BODDETI, V. N., KUMAR, B. V. K. V., et al. "Maximum Margin Correlation Filter: A New Approach for Localization and Classification", *IEEE Transactions on Image Processing*, v. 22, n. 2, pp. 631–643, Fevereiro 2013.

[39] ZHOU, L., WANG, H. "Facial Landmark Localization via Boosted and Adaptive Filters". In: *Proceeding of 20th IEEE International Conference on Image Processing (ICIP)*, pp. –, Setembro 2013.

[40] DUDA, R. O., HART, P. E., STORK, D. G. *Pattern Classification*. 2 ed. New York, NY, USA, Wiley-Interscience, 2000.

[41] VAN DER HEIJDEN, F., DUIN, R. P. W., DE RIDDER, D., et al. *Classification, Parameter Estimation and State Estimation: An Engineering Approach Using MATLAB*. Chichester, UK, Wiley, 2004.

[42] TAN, X., TRIGGS, B. "Enhanced Local Texture Feature Sets for Face Recognition Under Difficult Lighting Conditions". In: *AMFG*, pp. 168–182, 2007.

[43] OPPENHEIM, A. V., SCHAFER, R. W. *Discrete-time signal processing*. 3 ed. Englewood Cliffs, NJ, USA, Prentice Hall, 2010.

[44] LOWE, D. G. "Object recognition from local scale-invariant features". In: *International Conference on Computer Vision (ICCV'1999)*, v. 2, pp. 1150–1157, Setembro 1999.

[45] DALAL, N., TRIGGS, B. "Histograms of Oriented Gradients for Human Detection". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'2005)*, v. 1, pp. 886–893, Junho 2005.

[46] ZHU, Q., AVIDAN, S., YEH, M., et al. "Fast Human Detection Using a Cascade of Histograms of Oriented Gradients". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'2006)*, v. 2, pp. 1491–1498, Junho 2006.

[47] PORIKLI, F. "Integral histogram: a fast way to extract histograms in Cartesian spaces". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'2005)*, v. 1, pp. 829–836, Junho 2005.

[48] NONES, M. R. Q., MASIP, D., VITRIÀ, J. "Automatic Detection of Facial Feature Points via HOGs and Geometric Prior Models". In: *Proceedings of the 5th Iberian Conference on Pattern Recognition and Image Analysis*, Junho 2011.

[49] HAYKIN, S. *Neural Networks: A Comprehensive Foundation*. Upper Saddle River, NJ, Prentice Hall, 1999.

[50] MAIA, J. G. R., DE CARVALHO GOMES, F., DE SOUZA, O. "An Extended Set of Haar-Like Features for Rapid Object Detection". In: *SIBGRAPI 2007: XX Brazilian Symposium on Computer Graphics and Image Processing*, pp. 195–204, Outubro 2007.

[51] VALENTI, R., GEVERS, T. "Accurate Eye Center Location and Tracking Using Isophote Curvature". In: *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'2008)*, v. 0, pp. 1–8, Los Alamitos, CA, USA, 2008.

[52] SOILLE, P. *Morphological Image Analysis: Principles and Applications*. 2 ed. Secaucus, NJ, USA, Springer-Verlag New York, Inc., 2003.

[53] MCDONALD, J. H. *Handbook of Biological Statistics*. 2 ed. Baltimore, Maryland, USA, Sparky House Publishing, 2009.

[54] VALENTI, R., GEVERS, T. "Accurate Eye Center Location through Invariant Isocentric Patterns", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, v. 34, n. 9, pp. 1785–1798, 2012.

[55] TIMM, F., BARTH, E. “Accurate eye centre localisation by means of gradients”. In: *International Conference on Computer Theory and Applications*, pp. 125–130, Algarve, Portugal, 2011.

Apêndice A

Relação entre o MOSSE e o Filtro Discriminativo

Neste apêndice é feita a demonstração completa da relação entre o filtro discriminativo e o MOSSE, onde constata-se assim que o primeiro é um caso particular do segundo.

Retornando ao problema proposto na Seção 2.3.1: encontrar a transformação linear $\mathbf{H}$ (o filtro discriminativo) que aproxime o sinal $\mathbf{v}$, que representa a resposta ideal do sistema, minimizando o erro quadrático.

Dado o erro $\mathbf{e} = \mathbf{v} - \hat{\mathbf{v}}$, onde $\hat{\mathbf{v}} = \mathbf{H}\mathbf{g}$ e $\mathbf{g}$ a amostra a ser avaliada. Considerando o caso de amostras pertencentes a $\mathbb{C}^D$, pela formulação LMSSE, temos

$$\begin{aligned} \varepsilon(\mathbf{H}) &= \mathrm{E}\left[\mathbf{e}^H\mathbf{e}\right], \\ &= \mathrm{E}\left[\mathrm{tr}\left(\mathbf{e}\mathbf{e}^H\right)\right]. \end{aligned} \tag{A.1}$$

onde H é o operador conjugado transposto, $\mathbf{A}^H = (\mathbf{A}^*)^T$. Considerando que a transformação linear pode ser escrita como $\mathbf{H} = C(\mathbf{h})$, o problema pode ser escrito como

$$\varepsilon(\mathbf{h}) = \mathrm{E}\left[\|\mathbf{v} - \mathbf{h} \star \mathbf{g}\|_2^2\right], \tag{A.2}$$

Através do Teorema de Parseval, transformando para o domínio da frequência

$$\varepsilon\left(\tilde{\mathbf{h}}\right) = \mathrm{E}\left[\left\|\tilde{\mathbf{v}} - \tilde{\mathbf{g}} \odot \tilde{\mathbf{h}}^*\right\|_2^2\right], \tag{A.3}$$

onde $\tilde{\mathbf{h}}$, $\tilde{\mathbf{g}}$ e $\tilde{\mathbf{v}}$ são respectivamente, as transformadas 2-D discretas de $\mathbf{h}$, $\mathbf{g}$ e $\mathbf{v}$.

Reescrevendo a Equação (A.3):

$$\begin{aligned}\varepsilon\left(\tilde{\mathbf{h}}\right) &= \mathrm{E}\left[\left\|\tilde{\mathbf{v}} - \tilde{\mathbf{g}} \odot \tilde{\mathbf{h}}^{*}\right\|_{2}^{2}\right], \\ &= \mathrm{E}\left\{\left[\tilde{\mathbf{v}} - \tilde{\mathbf{g}} \odot \tilde{\mathbf{h}}^{*}\right]^{H}\left[\tilde{\mathbf{v}} - \tilde{\mathbf{g}} \odot \tilde{\mathbf{h}}^{*}\right]\right\}, \\ &= \mathrm{E}\left\{\tilde{\mathbf{v}}^{H}\tilde{\mathbf{v}} + \left(\tilde{\mathbf{h}}^{T}\tilde{\mathbf{G}}^{H}\right)\left(\tilde{\mathbf{G}}\tilde{\mathbf{h}}^{*}\right) - \left(\tilde{\mathbf{h}}^{T}\tilde{\mathbf{G}}^{H}\right)\tilde{\mathbf{v}} - \tilde{\mathbf{v}}^{H}\left(\tilde{\mathbf{G}}\tilde{\mathbf{h}}^{*}\right)\right\}.\end{aligned} \tag{A.4}$$

Onde $\tilde{\mathbf{G}}$ é uma matriz diagonal na qual os elementos da diagonal são os elementos de $\tilde{\mathbf{g}}$. Minimizando em função de $\tilde{\mathbf{h}}$

$$\frac{\partial\varepsilon\left(\tilde{\mathbf{h}}\right)}{\partial\tilde{\mathbf{h}}^{T}} = \mathrm{E}\left\{\tilde{\mathbf{G}}^{H}\tilde{\mathbf{G}}\tilde{\mathbf{h}}^{*} - \tilde{\mathbf{G}}^{H}\tilde{\mathbf{v}}\right\}. \tag{A.5}$$

Solucionando para $\tilde{\mathbf{h}}$, obtém-se

$$\tilde{\mathbf{h}}^{*} = \frac{\mathrm{E}\left[\tilde{\mathbf{G}}^{H}\tilde{\mathbf{v}}\right]}{\mathrm{E}\left[\tilde{\mathbf{G}}^{H}\tilde{\mathbf{G}}\right]} = \frac{\mathrm{E}\left[\tilde{\mathbf{g}}^{*} \odot \tilde{\mathbf{v}}\right]}{\mathrm{E}\left[\tilde{\mathbf{g}}^{*} \odot \tilde{\mathbf{g}}\right]}. \tag{A.6}$$

É possível observar que a Equação (A.6) é equivalente a Equação (2.77) quando o operador de valor esperado é substituído pela sua aproximação LSE. Considerando o caso do filtro discriminativo, onde o sinal $\mathbf{g}$ é definido na Equação 2.67 como:

$$\mathbf{g} = C(\mathbf{x})\mathbf{v} + \mathbf{w},$$

pode-se considerar que o filtro discriminativo é um caso particular do MOSSE onde supõe-se que o sinal corresponde ao padrão alvo $\mathbf{x}$ corrompido por ruído aditivo e pela resposta desejada.