

# Bagging de Detectores por Produto Interno para Detecção de Olhos

Wesley L. Passos<sup>1,2</sup>, Gabriel M. Araujo<sup>1</sup>, Amaro A. Lima<sup>1</sup>, Felipe M. L. Ribeiro<sup>2</sup>, Eduardo A. B. da Silva<sup>2</sup>.

**Resumo**—Este artigo apresenta um sistema de detecção de olhos utilizando *ensemble* de classificadores baseados em filtros de correlação, conhecidos como detectores por produto interno (IPD – *Inner Product Detectors*). Este trabalho tem três contribuições principais: i) um classificador *ensemble* com precisão maior que o IPD original, utilizando o algoritmo *bagging*; ii) novas funções discriminantes baseadas na informação da *ensemble*; e iii) o estudo da influência do *bagging* sobre o desempenho do sistema. O método proposto foi avaliado na base de dados BioID, atingindo taxas de acerto de 98,02% e 95,71% para os olhos direito e esquerdo, respectivamente, onde um desvio de até 10% da distância interocular é considerado um acerto.

**Palavras-Chave**—Visão computacional, Detecção de olhos, Características faciais, Métodos *ensemble*.

**Abstract**—This article presents a system for eye detection using an ensemble of correlation-based filters known as Inner Product Detector (IPD). This work has three main contributions: i) an ensemble classifier with higher accuracy than the original IPD detector, using the bagging algorithm; ii) new discriminant functions based on the ensemble output; and iii) the study of the influence of bagging on the system performance. The proposed method was evaluated on the BioID dataset, achieving an average accuracy of 98.02% and 95.71% for right and left eyes, respectively, where a deviation of up to 10% of the interocular distance is considered a hit.

**Keywords**—Computer vision, Eye detection, Facial features, Ensemble methods.

## I. INTRODUÇÃO

Muitas aplicações como biometria, reconhecimento de expressão, alinhamento facial, estimativa de pose, modelagem de face 3D, rastreamento facial, e tecnologias assistivas como estimativa de atenção e olhar de pessoas com deficiência, utilizam rastreamento ocular. Existem diversas formas de realizar essa tarefa, porém as mais baratas e menos intrusivas são as baseadas em técnicas de visão computacional [1]. Atualmente, as abordagens mais comuns para detecção de características faciais são baseadas em aprendizado de máquina. Estas podem ser divididas em métodos locais ou globais [2]. Métodos globais são capazes de localizar múltiplas características simultaneamente e, em sua maioria, incluem restrições geométricas. O desempenho dos métodos globais é dependente de suas inicializações e, geralmente, apresentam altos custos computacionais quando comparados aos métodos locais. Dada a sua dependência de um ponto de inicialização, a maioria dos métodos globais depende de aproximações locais para produzir pontos de partida confiáveis. Um exemplo é apresentado em [3], onde uma cascata de redes neurais convolucionais é empregada

para a detecção de características faciais, partindo de uma estimativa grosseira na primeira camada que é sucessivamente refinada nas camadas seguintes. Outra abordagem, que utiliza modelos de olhos para produzir características de entrada para uma rede neural de retropropagação que realiza detecção de olhos, é apresentada em [4].

Métodos locais são capazes de localizar uma única característica facial com melhor precisão e baixo custo computacional. Entretanto, em contextos complexos ou de oclusão, estes métodos tendem a sofrer com a ambiguidade. Um exemplo de abordagem local para detecção de pontos faciais usando uma combinação de *support vector regression* e *Markov random fields* para reduzir o tempo de busca e aumentar a precisão é apresentado em [5].

O *Inner Product Detector* (IPD) [6], [7] tem sido utilizado, com sucesso, para a detecção de olhos de humanos. O IPD é um classificador linear, mas não há evidências de que os objetos de interesse sejam linearmente separáveis no espaço de características. Por essa razão são necessárias modificações na função discriminante que acarretam maior custo computacional. Esse artigo é uma extensão dos trabalhos em [6], [7], simplificando a função discriminante e utilizando um *ensemble* de classificadores com o intuito de tratar a natureza não-linear do problema. Para tal, a técnica *ensemble bootstrap aggregating* (*bagging*) [8], [9], [10], [11] é utilizada para gerar múltiplos conjuntos de treinamento. Cada conjunto é empregado para treinar uma instância do IPD. Assim sendo, esse trabalho possui as seguintes contribuições: i) um classificador *ensemble* com precisão maior que o IPD original, utilizando o algoritmo *bagging*, no qual são consideradas positivas as amostras que recebem o maior número de votos ou, em caso de mais de uma amostra classificada como positiva, por meio de uma função discriminante mais simples; ii) novas funções discriminantes baseadas na média ou norma das saídas do conjunto de classificadores, ambas calculadas a partir das amostras classificadas como positivas; e iii) o estudo da influência do *bagging* sobre o desempenho do sistema. O método proposto foi avaliado na base de dados BioID, atingindo uma taxa de acerto média de 98,02% e 95,71% para os olhos direito e esquerdo, respectivamente, para uma precisão de 10% da distância interocular.

## II. INNER PRODUCT DETECTOR

Neste trabalho, será utilizado um detector baseado em filtros de correlação para detecção dos pontos fiduciais, o IPD, onde a saída é obtida através do produto interno entre o detector e um sinal de entrada, justamente como nos filtros de correlação.

Seja uma variável aleatória  $X_{d \times 1}$  cujas realizações  $x_i$  devem ser classificadas em uma das duas classes  $\mathcal{A}_1$  ou  $\mathcal{A}_0$ .

<sup>1</sup>Centro Federal de Educação Tecnológica, Nova Iguaçu, RJ, 26041-271, Brasil. <sup>2</sup>PEE/COPPE/DEL/POLI, Universidade Federal do Rio de Janeiro, Cx. P. 68504, Rio de Janeiro, RJ, 21945-970, Brasil. E-mails: wesley.passos@smt.ufrj.br, {gabriel.araujo, amaro.lima}@cefet-rj.br, {felipe.ribeiro, eduardo}@smt.ufrj.br.

O objetivo é encontrar um detector  $\mathbf{h}_{\mathcal{A}_1}$ , ótimo no sentido dos mínimos quadrados, capaz de detectar um objeto que pertença a classe  $\mathcal{A}_1$  [6].

$$\mathbf{h}_{\mathcal{A}_1}^T \mathbf{x}_i = \begin{cases} 1, & \text{se } \mathbf{x}_i \in \mathcal{A}_1 \\ 0, & \text{caso contrário.} \end{cases} \quad (1)$$

O operador  $[\cdot]^T$  denota a operação de transposição. O detector  $\mathbf{h}_{\mathcal{A}_1}$  pode ser reescrito em termos de momentos da variável aleatória  $X$  da seguinte forma [6]:

$$\mathbf{h}_{\mathcal{A}_1} = \mathbf{R}_{XX}^{-1} P(\mathcal{A}_1) \boldsymbol{\mu}_{\mathcal{A}_1} \quad (2)$$

onde  $\mathbf{R}_{XX}$  é a matriz de correlação global dos dados de treinamento,  $P(\mathcal{A}_1)$  é a probabilidade de uma amostra pertencer à classe  $\mathcal{A}_1$ , e  $\boldsymbol{\mu}_{\mathcal{A}_1}$  é a média das amostras da classe  $\mathcal{A}_1$ . Note que o termo  $\mathbf{R}_{XX}$  deve ser uma matriz inversível. Isto significa que a dimensão dos vetores deve ser menor que o número de amostras estatisticamente independentes. Ainda, é interessante notar que na expressão  $P(\mathcal{A}_1) \boldsymbol{\mu}_{\mathcal{A}_1}$  a média da classe desejada é ponderada por sua probabilidade [6].

### III. O BOOTSTRAP AGGREGATING

Proposto inicialmente em [8], o método *bootstrap aggregating*, também conhecido como *bagging*, é um tipo de algoritmo *ensemble learning* [10], [11], que tem por objetivo gerar um conjunto de modelos preditivos. Cada um destes modelos é gerado a partir de um subconjunto formado através de sorteios com reposição do conjunto de treinamento original. É esperado que a decisão tomada usando a coleção (*ensemble*) destes modelos seja melhor que a tomada por cada modelo individualmente [9]. Isso é possível pois a decisão tomada usando a coleção apresenta menor viés *bias* e variância do que a de cada modelo treinado nos dados completos, devido a suavização das fronteiras de decisão.

O processo de *bagging* consiste em gerar  $B$  novos conjuntos  $\mathcal{X}_b$ , a partir de um conjunto de treinamento original  $\mathcal{X}_L$ , sorteando amostras uniformemente e com reposição. Este processo é ilustrado na Fig. 1. Então, cada conjunto  $\mathcal{X}_b$  é utilizado para gerar uma instância do classificador. Por fim, é construído um modelo de predição combinando uma coleção (*ensemble*) dos  $B$  classificadores individuais.

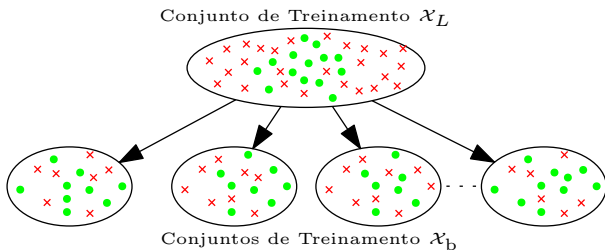


Fig. 1. Amostragem bootstrap.

### IV. MÉTODO PROPOSTO

As técnicas IPD e *bagging* foram implementadas e agregadas com o intuito de elaborar um algoritmo de detecção de olhos em imagens. Implementou-se o sistema proposto em [12], apresentado na Fig. 2, mas substituindo o IPD pelo

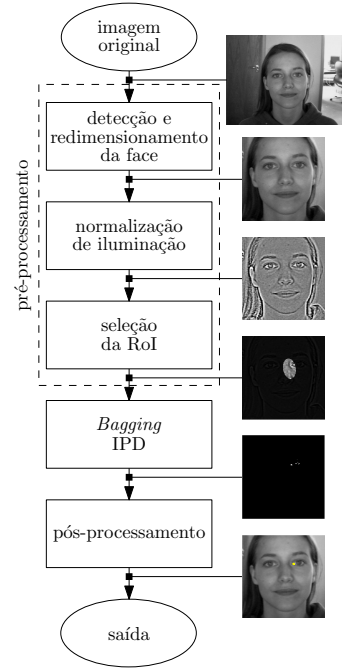


Fig. 2. Diagrama de blocos do sistema.

IPD com *bagging*. Para fins de treinamento, as amostras positivas são blocos de dimensão  $27 \times 27$  pixels centrados nas coordenadas da anotação manual e todos seus vizinhos 8-conectados, fornecendo 9 amostras positivas por imagem. A classe negativa consiste nos blocos restantes, também de dimensão  $27 \times 27$  pixels, cujo os pontos centrais estão dentro da região de interesse (ROI – *Region of Interest*, descrita na Seção IV-B). Note que existem muito mais amostras negativas que positivas por imagem. Sendo assim, para manter o equilíbrio entre as classes e reduzir o custo computacional, foram selecionados aleatoriamente 9 exemplos negativos de cada imagem de treinamento. Ou seja, cada imagem de treinamento contribuirá com 18 amostras, sendo 9 positivas e 9 negativas.

#### A. Base de dados

O banco de dados utilizado foi a BioID [13], que consiste em 1521 imagens em escala de cinza com resolução de  $384 \times 286$  pixels. Cada imagem mostra a visão aproximadamente frontal da face de uma pessoa de um grupo de 23 indivíduos. Nesse banco de dados há indivíduos com pelos faciais, olhos fechados, usando óculos, variações de iluminação e fundo, diferentes escalas faciais, rotações e expressões, o que faz com que a BioID seja uma base de dados desafiadora. Para propósitos de comparação, esse banco de dados contém dois arquivos com anotações. O primeiro possui as posições das coordenadas dos centros dos olhos, e o segundo as coordenadas de 20 pontos fiduciais em cada imagem.

#### B. Pré-processamento

Todas as imagens são pré-processadas antes dos procedimentos de treino e teste. Primeiramente, a face é localizada utilizando uma implementação do algoritmo de detecção

Viola-Jones [14] da biblioteca OpenCV [15]. Em seguida, as faces são escaladas para  $100 \times 100$  pixels para reduzir o efeito das diferenças de escala. Com o objetivo de conferir robustez à iluminação ao sistema, foram empregadas técnicas de normalização de iluminação seguindo a abordagem feita em [16].

A área de busca é restringida a uma região denominada de Região de Interesse (RoI), tanto para melhorar a acurácia do detector quanto para reduzir a complexidade computacional. A RoI é definida pelo sistema por meio das anotações manuais do conjunto de treinamento, assumindo que a localização das características é uma variável aleatória  $P$  com distribuição Gaussiana, média  $\mu_P$  e covariância  $\Sigma_P$ . Para cada posição de pixel  $\mathbf{p}$  nas  $N$  imagens do conjunto de treinamento, calculamos a distância de Mahalanobis como

$$d = \sqrt{(\mathbf{p} - \mu_P)^T \Sigma_P^{-1} (\mathbf{p} - \mu_P)}. \quad (3)$$

A RoI é a região elíptica limitada pela amostra de treinamento que maximiza a distância de Mahalanobis  $d_{max}$  com uma tolerância de 5%. A Eq. (4) é usada para verificar se um ponto candidato  $\mathbf{p}_c$  pertence ou não à RoI. Um candidato é considerado pertencente à RoI quando a condição da Eq. (4) é satisfeita:

$$(\mathbf{p}_c - \mu_P)^T \Sigma_P^{-1} (\mathbf{p}_c - \mu_P) \leq (1,05d_{max})^2. \quad (4)$$

### C. Treinamento e Teste

Na fase de treinamento são gerados  $B$  conjuntos de treinamento  $\mathcal{X}_b$  seguindo o procedimento de *bagging* descrito na Seção III. Cada um desses  $\mathcal{X}_b$  conjuntos é utilizado para gerar um único classificador, produzindo  $B$  classificadores. Com o fim de avaliar o efeito do tamanho do *ensemble* no desempenho, variou-se  $B$  no intervalo  $[1, 60]$  durante o processo de validação cruzada.

Com o objetivo de aumentar a variância dos dados de treinamento, a fim de analisar a eficácia do *bagging*, experimentos descartando amostras, antes de gerar os conjuntos *bootstrap*, foram realizados. Este procedimento é mostrado na Fig. 3. Foram executados experimentos descartando 0%, 50%, 75%, 80%, 85%, 90% e 95% das amostras do conjunto de treinamento original  $\mathcal{X}_L$ .

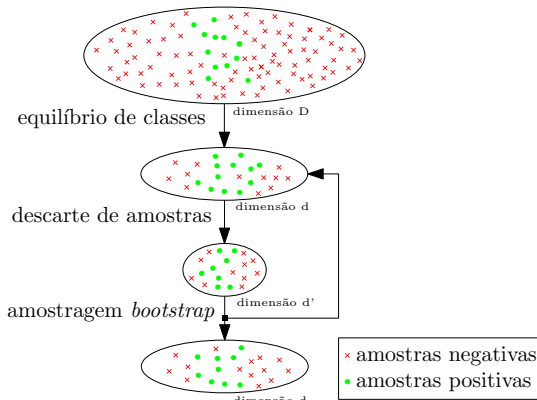


Fig. 3. Método de geração dos conjuntos de treinamento com descarte de amostras.

Tendo treinado os classificadores, estes são então agregados (*ensemble classifiers*) [10] para a fase de teste. Cada amostra é classificada por cada classificador treinado, e é considerada pertencente à classe que recebeu o maior número de votos.

Antes das fases de treinamento e teste, a média  $\mu_L$ , de todo o conjunto de treino  $\mathcal{X}_L$ , é subtraída das amostras. Ainda, é importante destacar que os parâmetros  $P(\mathcal{A}_1)$  e  $\mu_{\mathcal{A}_1}$ , utilizados para gerar cada classificador, foram calculados a partir do conjunto de treinamento original  $\mathcal{X}_L$ , com a média  $\mu_L$  subtraída.

O processo de treinamento e teste está esquematizado na Fig. 4.

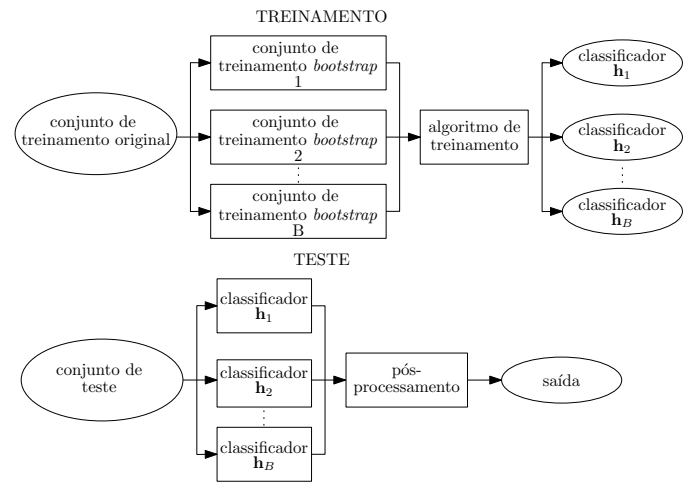


Fig. 4. Diagrama de blocos da metodologia de treinamento e teste.

### D. IPD com Função Discriminante Modificada

A restrição imposta pela função discriminante na Eq. (1) é válida apenas para classes ortonormais. Entretanto, tal característica é improvável em dados reais. Então, para as amostras não normalizadas, a saída pode ser qualquer valor real. Como resultado, o IPD tem uma alta taxa de falsos positivos na sua formulação original, tendo fraca correlação com a classe desejada. Neste trabalho, os valores de IPD foram limitados ao conjunto  $\{-1, 0, 1\}$ , por meio da função sinal (Eq. (5))

$$\text{sign}(\mathbf{h}_b^T \mathbf{x}_i) = \begin{cases} 1, & \text{se } (\mathbf{h}_b^T \mathbf{x}_i) > 0 \\ 0, & \text{se } (\mathbf{h}_b^T \mathbf{x}_i) = 0 \\ -1, & \text{se } (\mathbf{h}_b^T \mathbf{x}_i) < 0. \end{cases} \quad (5)$$

Em outras palavras, cada classificador  $\mathbf{h}_b$  fornece um voto, que pode assumir valores  $\{-1, 0, 1\}$ , a um ponto  $\mathbf{x}_i$ . Ao final, todos os votos são somados e somente o candidato que recebe a maioria dos votos permanece. No caso de empates, os pontos que empataram são pós-processados de acordo com o descrito na seção a seguir, onde um único ponto é selecionado.

### E. Pós-processamento

Em alguns casos, em cada imagem de teste, dois ou mais pontos podem ser reconhecidos pela maioria dos classificadores. Estes casos podem ser úteis quando é desejável manter os candidatos mais prováveis ou quando o *bagging* IPD for utilizado como etapa anterior de outro método [7]. Entretanto, para este trabalho, um único ponto é desejado. Para obter

apenas um ponto de saída, dois métodos de pós-processamento foram avaliados, e são descritos a seguir.

Seja  $\mathcal{X}_R$  o conjunto de pontos  $\mathbf{x}_i$  classificados como pertencente à classe de interesse por meio da votação de  $B$  classificadores. Uma solução para estimar o ponto de saída do sistema  $\hat{\mathbf{p}}_f$  é escolher o ponto que apresentar o conjunto de produtos internos de maior norma, conforme a Eq. (6).

$$\hat{\mathbf{p}}_f = \arg \max_{\mathbf{x}_i \in \mathcal{X}_R} \left( \sum_{b=1}^B (\mathbf{h}_b^T \mathbf{x}_i)^2 \right)^{1/2} \quad (6)$$

Uma segunda solução é escolher o ponto que apresentar o maior valor médio de IPD, conforme a Eq. (7).

$$\hat{\mathbf{p}}_f = \arg \max_{\mathbf{x}_i \in \mathcal{X}_R} \left( \frac{1}{B} \sum_{b=1}^B \mathbf{h}_b^T \mathbf{x}_i \right) \quad (7)$$

#### F. Validação

O critério adotado para avaliar o desempenho do sistema foi a distância entre a anotação manual e o ponto detectado. Para evitar que a medida de desempenho não dependa de variações de escala da imagem, essa distância é expressa em relação à distância entre as marcações das pupilas obtidas a partir da anotação manual, também conhecida como distância interocular  $d_{io}$ . Sendo  $\mathbf{p}_l$  e  $\mathbf{p}_r$  as marcações manuais das pupilas esquerda e direita, respectivamente, o erro  $e$  entre a anotação manual do ponto fiducial alvo  $\mathbf{p}_f$  e a saída estimada  $\hat{\mathbf{p}}_f$ , em relação à distância interocular é:

$$e = \frac{\|\hat{\mathbf{p}}_f - \mathbf{p}_f\|}{\|\mathbf{p}_r - \mathbf{p}_l\|} \quad (8)$$

Nesta medida, para fins de comparação,  $e \leq 0,25$ , 25% da distância interocular corresponde a distância entre o centro e o canto do olho, e  $e \leq 0,1$  corresponde ao diâmetro da íris. Serão considerados acertos os casos em que  $e \leq 0,1$  [17].

A técnica de validação cruzada, com  $k = 10$ , foi utilizada para avaliação estatística dos resultados [18]. Então, todos os resultados reportados são média e desvio-padrão dos *folds*.

## V. RESULTADOS E DISCUSSÕES

A TABELA I apresenta o erro médio para 10% da distância interocular, juntamente com desvio-padrão, para cada método pós-processamento e porcentagem de descarte de amostras de treino. Estes resultados consideram a melhor quantidade de classificadores obtida durante a validação cruzada para cada conjunto de parâmetros de método de pós-processamento e porcentagem de descarte.

Observando-se os resultados obtidos, pode-se afirmar que: i) o uso da *ensemble* adiciona uma melhoria consistente ao desempenho do IPD original [12]; ii) a escolha do fator de descarte é mais crítica que o uso de diferentes funções discriminantes; e iii) o *ensemble* consegue manter o desempenho mesmo com mais de 90% das amostras descartadas, comprovando a eficácia do método de *bagging*.

O efeito do tamanho do *ensemble* no desempenho do classificador é apresentado na TABELA II. Nota-se, como esperado, que para um maior fator de descarte de amostras, maior é o número de classificadores necessários para manter

TABELA I

ERRO PARA  $e \leq 0,10$  PARA CADA MÉTODO DE PÓS-PROCESSAMENTO E PORCENTAGEM DE AMOSTRAS DESCARTADAS.

Descarte de Amostras	Olho direito		Olho esquerdo	
	Maior norma IPD	Maior média IPD	Maior norma IPD	Maior média IPD
0%	2,18±1,35	2,11±0,82	5,04±1,92	5,18±1,08
50%	2,11±1,34	2,11±1,13	<b>4,84±2,12</b>	5,24±2,38
75%	2,32±1,41	2,16±1,31	<b>4,70±0,88</b>	5,25±1,71
80%	2,39±1,70	<b>1,98±1,72</b>	5,11±1,33	4,84±1,74
85%	2,25±1,07	2,04±1,24	5,04±1,25	4,97±0,28
90%	<b>1,97±1,88</b>	<b>1,98±1,03</b>	5,18±1,80	<b>4,29±1,73</b>
95%	2,32±1,41	2,25±0,91	5,45±1,92	5,04±2,47
<b>Original [12]</b>	2,4±1,1		6,1±1,6	

TABELA II

ERRO PARA  $e \leq 0,10$ , VARIANDO O NÚMERO DE CLASSIFICADORES E UTILIZANDO PÓS-PROCESSAMENTO MAIOR MÉDIA IPD.

#Classificadores	Olho direito		Olho esquerdo	
	Descarte 0%	Descarte 95%	Descarte 0%	Descarte 95%
01	<b>2,45±1,33</b>	88,5±3,19	5,71±1,25	96,5±2,21
05	2,52±0,91	38,6±6,74	<b>5,65±1,20</b>	51,4±8,08
10	2,79±1,18	7,21±1,21	5,79±0,29	11,0±3,97
25	3,34±1,30	3,48±0,35	<b>5,66±1,97</b>	5,86±1,98
50	2,73±0,85	<b>2,39±1,80</b>	<b>5,65±0,28</b>	7,49±2,91
60	3,00±1,33	2,59±1,63	5,79±1,21	8,79±2,83
<b>Original [12]</b>	<b>2,4±1,1</b>		6,1±1,6	

o mesmo desempenho. Entretanto, cerca que 25 classificadores são o suficiente para manter o desempenho do sistema satisfatório. Os resultados apresentados na Fig. 5 corroboram com essa afirmação. Nela são apresentados as taxas de erro para a detecção do olho direito, descartando 95% de amostras e utilizando a melhor média IPD como pós-processamento. Pode-se observar o efeito dos classificadores IPD em *ensemble* para diferentes tolerâncias de distância relativa à distância interocular. As taxas de erro começam a estabilizar a partir de 25 classificadores, como observado anteriormente.

Por fim, a TABELA III mostra a comparação com os métodos em [3], [4], [5], [12]. Os resultados de [3], [5] foram obtidos a partir de [4]. Nestes experimentos, os olhos foram considerados corretamente detectados se  $e \leq 0,10$ . Para fins de comparação, foram escolhidos os resultados dos experimentos sem descarte de amostras, de modo a considerar todo o conjunto de treinamento original; com descarte de 90% das amostras, a configuração que obteve os melhores resultados; e em ambos os casos, maior média IPD como método de pós-processamento, já que este apresentou os melhores resultados entre os dois métodos de pós-processamento. Estes resultados foram atingidos com 41 e 57 classificadores para os olhos direito e esquerdo, respectivamente, sem utilizar o descarte durante o *bagging*, e 24 e 15 classificadores, utilizando o descarte. Mesmo que os procedimentos de treinamento e teste sejam diferentes entre si, as comparações podem ser consideradas justas tomando cada método como uma “caixa preta”. No caso sem descarte houve um redução no desvio-padrão, indicando um aumento da consistência no teste ao longo dos *folds*. No entanto, a diminuição do erro de classificação foi pouco significativa. Já nos caso em que foram descartadas 90% das amostras, o desvio padrão se manteve aproximadamente o mesmo do IPD original, mas neste caso houve uma queda

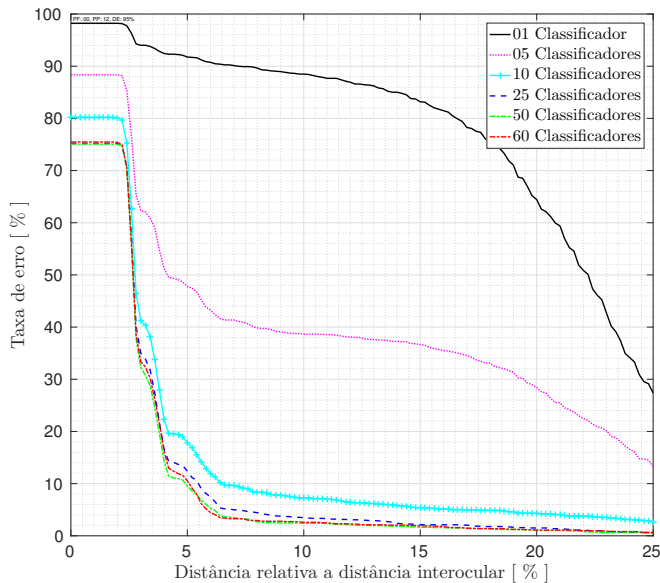


Fig. 5. Comparação entre número de classificadores para detecção do olho direito, descartando 95% das amostras e utilizando maior média como método de pós-processamento.

maior na taxa média de erro. Apesar da redução do erro ser relativamente pequena, o uso de validação cruzada por  $k$ -fold indica que esses resultados são estatisticamente relevantes. Comparando quantitativamente os erros obtidos, o método proposto neste trabalho é o terceiro melhor para detecção de ambos os olhos dentre os trabalhos comparados. Os melhores resultados, apresentados em [3], e os obtidos neste trabalho são mostrados em negrito. Os resultados demonstram que a metodologia proposta produziu melhorias quando comparado com IPD original e atinge taxas de erro competitivas quando comparado a outros métodos, apesar do incremento em custo computacional quando em relação ao original. Porém, é importante ressaltar que, assim como os métodos baseados em redes neurais convolucionais apresentados em [3] e [4], o *bagging* IPD pode ser paralelizado e executado em GPU, reduzindo assim o tempo de execução, e sendo menos complexo que sistemas comumente encontrados na literatura.

TABELA III

COMPARAÇÃO DE DESEMPENHO ENTRE DIFERENTES MÉTODOS UTILIZANDO A BIOID PARA  $e \leq 0,10$ . OS MELHORES E OS NOSSOS RESULTADOS ENCONTRAM-SE EM NEGRITO. (§) VALORES FORNECIDOS EM [4].

Método	Olho direito	Olho esquerdo
Valstar et al., 2010 <sup>§</sup> [5]	20,9	24,2
Sun et al., 2013 <sup>§</sup> [3]	<b>0,01</b>	<b>0,00</b>
Li et al., 2015 [4]	1,90	1,80
Araujo et al., 2015 [12]	2,40±1,1	6,10±1,6
Bagging IPD 0%	<b>2,11±0,82</b>	<b>5,18±1,08</b>
Bagging IPD 90%	<b>1,98±1,03</b>	<b>4,29±1,73</b>

## VI. CONCLUSÕES

Neste trabalho estudou-se a influência do uso de *bagging* em um algoritmo para detecção de olhos em imagens de faces humanas denominado *inner product detector* IPD. Os

classificadores foram treinados por meio de *bagging*, gerando uma *ensemble* para realizar a classificação de novas amostras. Foram propostas novas funções discriminante que consideram o efeito do *ensemble*. Por fim, realizou-se um estudo sobre o efeito do *bagging* no desempenho do sistema. O método proposto foi comparado com outros trabalhos que utilizaram a BioID [13]. Dado que cada classificador é independente, podem ser treinados de forma paralela, fazendo uso de arquiteturas paralelas como GPUs ou máquinas que possuem multiprocessadores, tornando-o ainda mais rápido. Futuramente deseja-se empregar esta técnica para rastreamento em sequência de vídeos e testar o uso de *kernels* junto aos classificadores.

## REFERÊNCIAS

- [1] R. Valenti and T. Gevers, "Accurate eye center location through invariant isocentric patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 9, pp. 1785–1798, 2012.
- [2] C. Du, Q. Wu, J. Yang, and Z. Wu, "SVM based ASM for facial landmarks location," in *Proc. of the IEEE Int. Conf. on Computer and Information Technology*, 2008, pp. 321–326.
- [3] Y. Sun, X. Wang, and X. Tang, "Deep convolutional network cascade for facial point detection," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'2013)*, vol. 0, Portland, OR, USA, 2013, pp. 3476–3483.
- [4] X. Li, Y. Dou, X. Niu, J. Xu, and R. Xiao, "An efficient robust eye localization by learning the convolution distribution using eye template," *Computational intelligence and neuroscience*, vol. 2015, pp. 709 072:1–709 072:10, 2015.
- [5] M. Valstar, B. Martinez, X. Binefa, and M. Pantic, "Facial point detection using boosted regression and graph models," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'2010)*, vol. 0, San Francisco, CA, USA, 2010, pp. 2729–2736.
- [6] G. M. Araujo, F. M. L. Ribeiro, E. A. B. Da Silva, and S. K. Goldenstein, "Fast eye localization without a face model using inner product detectors," in *IEEE International Conference on Image Processing (ICIP)*, 2014, pp. 1366–1370.
- [7] G. M. Araujo, E. A. Silva, A. G. Ciancio, J. F. de Oliveira, F. M. Ribeiro, and A. Said, "Integration of eye detection and tracking in videoconference sequences using temporal consistency and geometrical constraints," in *IEEE International Conference on Image Processing (ICIP)*, 2012, pp. 421–424.
- [8] L. Breiman, "Bagging predictors," *Machine Learning*, vol. 24, no. 2, pp. 123–140, 1996.
- [9] I. H. Witten and E. Frank, *Data Mining: Practical machine learning tools and techniques*, 2nd ed. San Francisco, California, USA: Morgan Kaufmann, 2005.
- [10] T. Hastie, R. Tibshirani, and J. H. Friedman, *The elements of statistical learning: Data mining, inference, and prediction*, 2nd ed. Springer, 2009.
- [11] T. Dietterich, "An experimental comparison of three methods of constructing ensembles of decision trees: Bagging, boosting and randomization," *Machine Learning*, pp. 139–157, Apr. 2000.
- [12] G. M. Araujo, F. M. L. Ribeiro, W. S. Junior, E. A. B. da Silva, and S. K. Goldenstein, "Weak classifier for density estimation in eye localization and tracking," *Transactions on Image Processing*, vol. PP, no. 99, pp. 1–1, 2017.
- [13] BioID Technology Research, "The bioid face database," <http://www.bioid.com>, 2011.
- [14] P. Viola and M. J. Jones, "Robust real-time face detection," *International journal of computer vision*, vol. 57, no. 2, pp. 137–154, 2004.
- [15] OPENCV, "Opencv: Open computer vision library," <http://opencv.org>, 2014.
- [16] X. Tan and B. Triggs, "Enhanced local texture feature sets for face recognition under difficult lighting conditions," *IEEE Transactions on Image Processing*, vol. 19, no. 6, pp. 1635–1650, 2010.
- [17] R. Valenti and T. Gevers, "Accurate eye center location and tracking using isophote curvature," in *IEEE Conference on Computer Vision and Pattern Recognition (CVPR'2008)*, vol. 0, Los Alamitos, CA, USA, 2008, pp. 1–8.
- [18] R. Kohavi, "A study of cross-validation and bootstrap for accuracy estimation and model selection," in *Proc. of the International Joint Conference on Artificial Intelligence (IJCAI)*, vol. 14, no. 2, Montreal, Quebec, CA, 1995, pp. 1137–1145.