

Universidade De São Paulo  
Instituto de Matemática e Estatística  
Departamento de Ciência da Computação

Fabio Mazete Reaes

## **Reconhecimento de Faces em Imagens: Projeto Beholder**

*MAC0499 - Trabalho de formatura Supervisionado do Departamento de Ciência da Computação da Universidade De São Paulo para obtenção do grau de Bacharel em Ciência da Computação.*

Orientadora: *Nina S. T. Hirata*  
Co-orientador: *Roberto Hirata Jr.*

São Paulo  
4 de dezembro de 2006

# Resumo

O *Projeto Beholder* visa implementar um sistema de cadastro e busca de pessoas com módulo para reconhecimento de faces. Como parte deste projeto, pretendíamos implementar tal módulo para um conjunto de fotos 3x4 comuns. No entanto, a segmentação de faces dessas fotos, que supúnhamos ser um processo relativamente fácil, mostrou-se um grande desafio devido à diversidade das fotos consideradas (diferentes qualidades de nitidez, fundo não uniforme, coloridas e em preto e branco, diferentes tamanhos, presença de ruídos tais como manchas e rasuras). Portanto, neste trabalho basicamente estudamos e implementamos técnicas de segmentação de faces frontais nessas fotos 3x4. Foram implementadas duas técnicas: uma baseada em bordas, e outra baseada em cor de pele. Foi também proposta e testada uma técnica híbrida que combina as duas anteriores. Os resultados obtidos são satisfatórios.

**Palavras-chave:** Reconhecimento de Faces, Visão Computacional, Análise de Formas

# Sumário

<b>1</b>	<b>Introdução</b>	<b>1</b>
<b>2</b>	<b>Objetivos</b>	<b>4</b>
2.1	O Projeto	4
2.2	O Trabalho	4
<b>3</b>	<b>Desenvolvimento</b>	<b>6</b>
3.1	Aquisição das Imagens	6
3.2	Separação das Fotos	6
3.3	Redimensionamento de Imagens	7
3.4	Segmentação de Faces	10
3.4.1	Segmentação Baseada em Bordas	10
3.4.2	Segmentação Baseada em Cor da Pele	12
3.4.3	Segmentação Híbrida	15
<b>4</b>	<b>Resultados</b>	<b>17</b>
<b>5</b>	<b>Parte Subjetiva</b>	<b>20</b>
5.1	Considerações sobre o Curso	20
5.2	Considerações sobre a Equipe	21
5.3	Dificuldades	21
5.4	Próximos Passos	22
5.5	Agradecimentos	22

# Introdução

O problema de reconhecimento de faces é um ramo da Visão Computacional que tem atraído muitos pesquisadores durante os últimos anos. Isso se deve, sobretudo, à busca do mercado por essas aplicações e à acessibilidade da tecnologia necessária para o seu desenvolvimento. Essas aplicações são principalmente voltadas para a área de segurança.

Os métodos existentes de identificação pessoal - crachá, carteira de motorista, leitura de impressão digital e íris dos olhos, etc - dependem da cooperação do usuário, ao passo que a análise de uma imagem frontal da face de um indivíduo tirada sem a percepção do mesmo é passiva de cooperação. Isso atrai discussões polêmicas sobre o grande potencial de invasão de privacidade que possuem os sistemas de reconhecimento de faces.

Com os ataques de 11 de setembro de 2001, o governo dos Estados Unidos conseguiu o apoio da amedrontada população americana para que sistemas de reconhecimento de faces fossem instalados em locais públicos. Aproveitando a situação, o governo americano vem investindo pesado em pesquisas para esse fim. Porém, a eficiência de tais sistemas não tem correspondido às expectativas.

Essa ineficiência é principalmente causada pela dificuldade que um sistema de reconhecimento de faces tem para reconhecer pessoas que ocultam suas características faciais, como o uso de óculos escuros ou barba. Além disso, com o aumento da base de dados as pessoas tendem a ser cada vez mais parecidas e, como consequência, a diferenciação entre elas tende a ser mais complicada.

Como exemplo, há caso do sistema de reconhecimento de faces instalado no aeroporto de Logan, em Boston. A meta era evitar terroristas em aviões, portanto, no teste de hipóteses, o sistema deveria ter baixo erro do tipo I (aceitar como inocente um suspeito), o que consequentemente aumentava o erro do tipo II (aceitar como suspeito uma pessoa inocente). Esse sistema frequentemente acusava viajantes inocentes, e até mesmo deixava de acusar terroristas usando disfarces. Um fracasso total.

O reconhecimento de faces é muito importante para o ser humano em sua rotina diária, tanto para identificar pessoas ao redor quanto para analisar suas emoções. Sem essa capacidade de percepção, o relacionamento social seria bem mais difícil.

Tal capacidade é dominada desde cedo. Apenas com alguns meses de vida bebês já reco-

nhecem as pessoas que lhes são mais chegadas. Ao longo dos anos, essa capacidade é desenvolvida pelo cérebro humano, culminando numa poderosa ferramenta de reconhecimento. Mesmo que passem muitos anos ou a face apresente alterações significativas, tais como a presença de óculos, barba ou um corte de cabelo diferente, o reconhecimento acontece sem grandes dificuldades.

Apesar de ser trivial para o sistema visual humano, reconhecer faces é uma tarefa extremamente complexa para máquinas. Por esse sistema humano de percepção ser tão eficiente, pesquisadores desenvolvem técnicas inspiradas nesse sistema, que é estudado amplamente por diversas áreas, tais como psicologia, neurociência, engenharia e visão computacional.

Atualmente não existem sistemas de reconhecimento de formas tão poderosos quanto o sistema humano de percepção (veja a figura 1.1). Porém, o sucesso de alguns desses sistemas artificiais deve-se pelas restrições feitas acerca do universo de imagens a serem analisadas. Quanto mais restrições, mais eficiente o algoritmo pode ser; porém, menor é a abrangência de casos onde o algoritmo funciona. Um sistema que identifica parafusos pretos numa esteira branca, por exemplo, precisa apenas distinguir 'parafuso (preto)' de 'esteira (branca)'. É interessante fazer restrições, desde que o sistema não deixe de abranger todos os casos possíveis.



**Figura 1.1** Imagine o quão difícil é para uma máquina identificar o dalmata presente nesta figura. Para o ser humano, bastam apenas alguns segundos. Imagem extraída de *P. Lindsay and D. Norman (1977)*

O texto apresenta-se organizado da seguinte forma:

- No capítulo 2 são apresentados os objetivos e estrutura do projeto, e também os objetivos deste trabalho.
- No capítulo 3 são dados detalhes sobre o desenvolvimento do que já foi implementado, em ordem cronológica.
- Os resultados obtidos pela segmentação das faces nas imagens digitalizadas das fotos 3x4 são mostrados e os algoritmos analisados no capítulo 4.
- Por fim, no capítulo 5, está a parte subjetiva, onde são relatadas minhas experiências ganhas durante o desenvolvimento do projeto.

# Objetivos

### 2.1 O Projeto

O intuito do *Projeto Beholder* é implementar um sistema visual de cadastro e busca de pessoas, com um grande diferencial: O sistema conterà um módulo de reconhecimento de faces, que é capaz de extrair características faciais a partir de análise automática de imagens de rosto.

Isso significa que, no cadastro de uma pessoa, quando uma imagem de rosto for inserida, os campos onde constarem características que o sistema for capaz de detectar a partir da análise automática dessa imagem serão preenchidos automaticamente. Ex: cor dos olhos, cor de pele, gênero, largura do queixo, distância entre os olhos. Os campos preenchidos automaticamente, juntamente com os campos que o usuário obrigatoriamente deve digitar - nome, CPF, etc. - serão armazenados em XML.

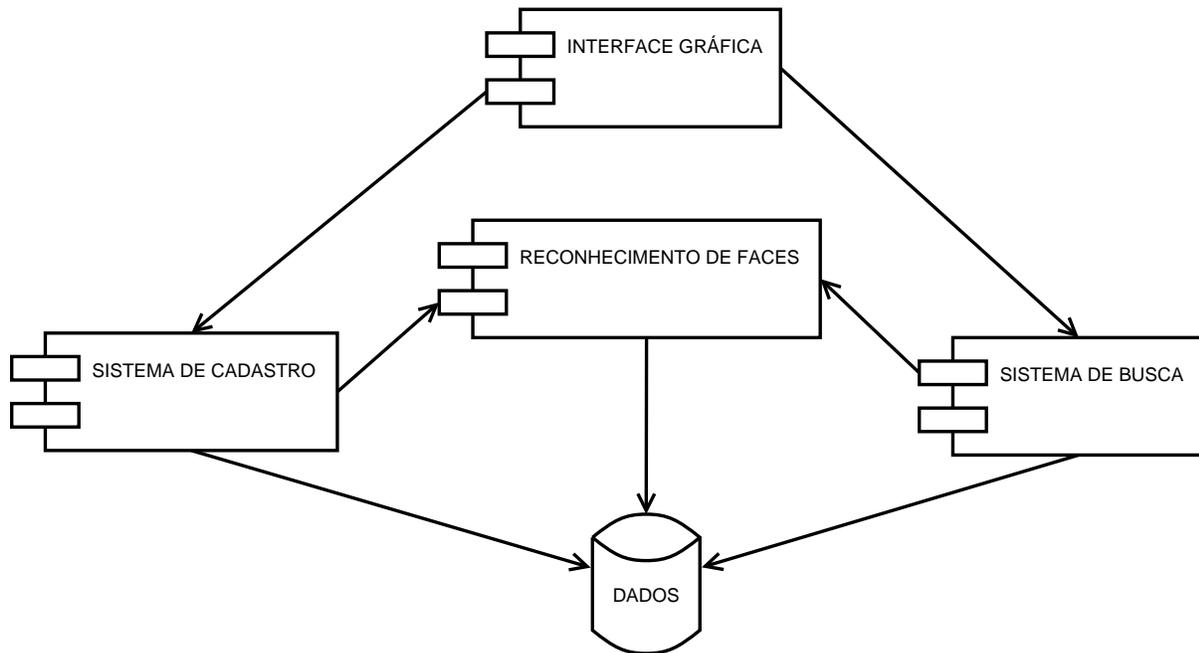
A operação de busca poderá ser realizada a partir de preenchimento de campos e/ou com imagem de rosto como entrada. Cada cadastrado terá uma probabilidade de ser a pessoa procurada, e o ajuste de tolerância filtrará os resultados.

Esse sistema poderá ser usado em delegacias, para cadastro e busca de criminosos. Isso poupará o tempo gasto na identificação de suspeitos. Atualmente, o método usado é lamentável: folhear álbuns e álbuns de fotografias de criminosos até encontrar o suspeito ou não. Usando o *Projeto Beholder*, basta a testemunha informar características da pessoa procurada para obter a listagem de todas que sigam o padrão. Essas características não precisam ser necessariamente faciais, já que no cadastro preenche-se manualmente outras informações como: nome, CPF, altura e massa corpórea.

O *Projeto Beholder* está sendo implementado na linguagem *Java 1.5* (<http://www.java.com>), e sua interface gráfica faz uso de funções do pacote *org.eclipse.swt* (<http://www.eclipse.org/swt/>).

### 2.2 O Trabalho

A solução para o problema de reconhecimento automatizado de faces envolve três tarefas: (1) Segmentação e normalização bruta, (2) Extração de características e normalização precisa,



**Figura 2.1** Diagrama dos módulos do projeto

(3) Identificação e/ou verificação [7].

A segmentação é o ponto crítico para algoritmos de análise de formas. Uma boa segmentação é fundamental para que as fases posteriores possam obter bons resultados.

Devido à importância e complexidade da fase de segmentação para todo o módulo de reconhecimento de faces, neste trabalho é realizado, sobretudo, o estudo e implementação de técnicas para segmentação de faces num conjunto heterogêneo de fotos 3x4 de rostos digitalizadas.

# Desenvolvimento

## 3.1 Aquisição das Imagens

Imagens no computador são representadas por matrizes tais que  $g(i,j)$  representa a cor de um pixel na linha  $i$  e coluna  $j$ , e geralmente  $g(i,j) \in [0, 255]$ . Cada matriz representa uma banda na escala de cores. Por exemplo, na escala RGB são necessárias 3 matrizes - (R) vermelho, (G) verde e (B) azul - para representar a imagem. Já para níveis de cinza, apenas uma matriz é necessária.

O conjunto de imagens é composto de 740 fotos 3x4 comuns de rosto, tiradas por diferentes câmeras, em diferentes épocas e em diferentes estados de conservação.

Ao invés de digitalizar as fotos uma a uma, estas foram coladas em folhas de caderno ou sulfite, para posterior digitalização por folhas. Diversas pessoas se encarregaram de fazer as colagens. Sem um padrão bem definido, o número de fotos variou de 25 a 26 por colagem.

Essas colagens foram digitalizadas a 600dpi utilizando-se um scanner de mesa, resultando em 26 imagens no formato PNG.

Dois padrões importantes a respeito dessas colagens podem ser observados: Há espaçamento entre as fotos, e as fotos estão dispostas em fileiras horizontais. O algoritmo para segmentar as fotos, descrito na próxima seção, faz uso dessas informações.

## 3.2 Separação das Fotos

No *Projeto Beholder*, o cadastramento é individual. Portanto, foi necessário desenvolver um separador de fotos automático para as colagens digitalizadas.

Para segmentar as fotos nas imagens digitalizadas foi desenvolvido um algoritmo cuja idéia é analisar a variância entre os pixels nas linhas e colunas da matriz da imagem transformada em níveis de cinza. Para níveis baixos de variância pode-se assumir como região de espaçamento entre fotos, já que a cor do papel onde as fotos foram coladas é uniforme. Para níveis altos, assume-se que a região contém fotos, já que a presença destas aumenta a variância entre os valores de coloração dos pixels em questão.

Como as fotos são horizontalmente alinhadas, mas não necessariamente no sentido vertical, primeiramente são recortadas as fileiras horizontais de fotos, e para cada fileira, aplica-se a mesma idéia para encontrar as fileiras verticais de cada fileira horizontal. Com isso, todas as fotos são encontradas. Veja a figura 3.1 e o algoritmo 1.

---

**Algorithm 1** Encontra as fotos contidas na tabela *tab*

---

```

1:  $g \leftarrow tab$  em níveis de cinza
2:  $r[ ] \leftarrow$  linhas de  $g$ 
3: for each  $r[i]$  do
4:    $variance[i] \leftarrow$  variância entre os pixels na linha  $r[i]$ 
5: end for
6:  $h[ ] \leftarrow$  intervalos  $[i,j]$  onde  $variance[x] > K$ , para todo  $x \in [i,j]$ 
7: for each  $h[i]$  do
8:    $c[ ] \leftarrow$  colunas de  $g$  no intervalo horizontal  $h[i]$ 
9:   for each  $c[i]$  do
10:     $variance[i] \leftarrow$  variância entre os pixels na coluna  $c[i]$ 
11:   end for
12:    $v[ ] \leftarrow$  intervalos  $[i,j]$  onde  $variance[x] > K$ , para todo  $x \in [i,j]$ 
13:   for each  $v[j]$  do
14:      $pics \leftarrow pics \cup (h[i], v[j])$ 
        $\{(h[i],v[j]) \text{ é o retângulo } (\Delta x, \Delta y) \text{ que contém uma foto}\}$ 
15:   end for
16: end for
17: return  $pics$ 

```

---

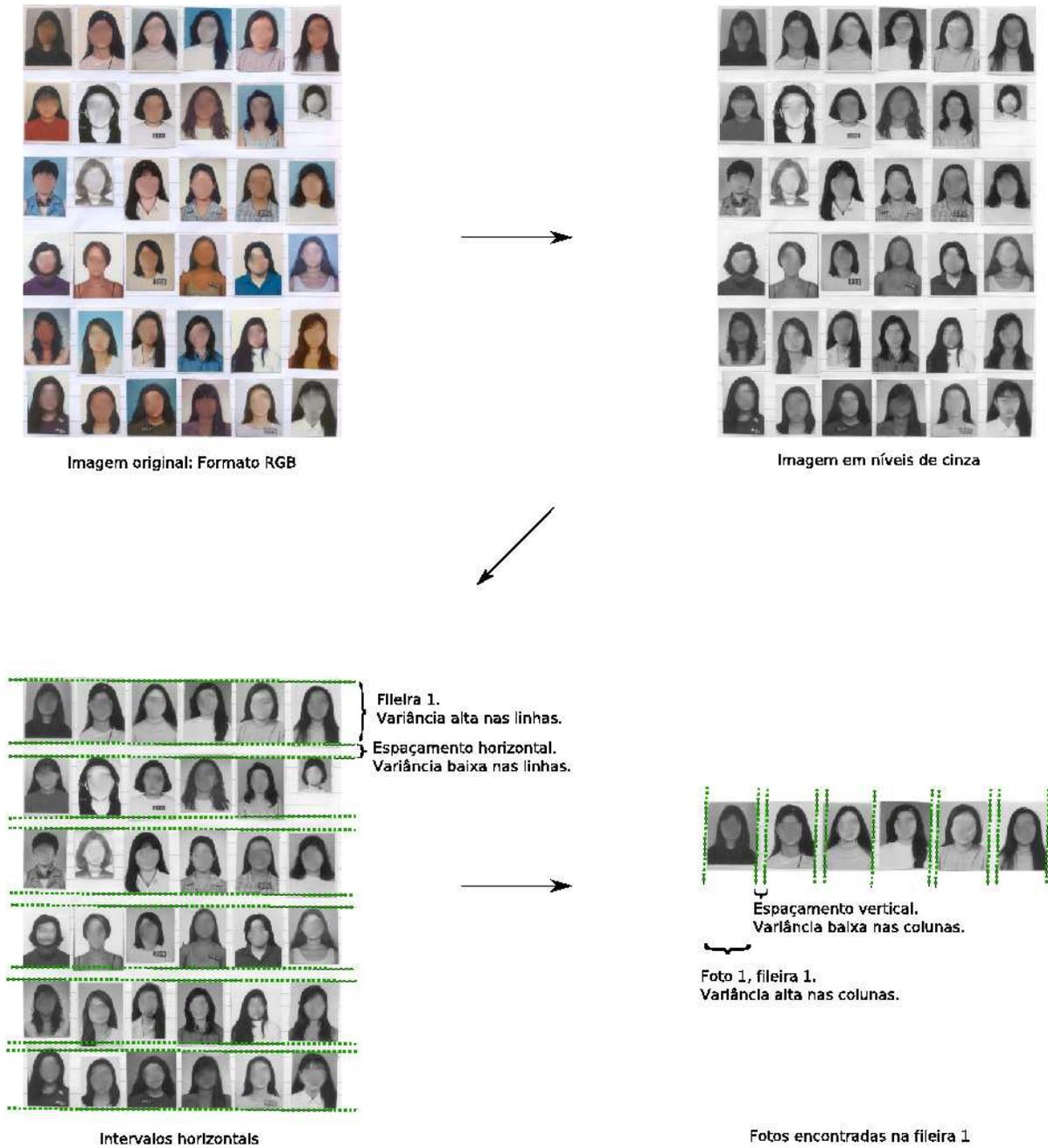
Para o algoritmo 1, o limiar  $K$  que funcionou bem para todas as imagens foi  $K = 192$ .

Para transformar a imagem RGB para níveis de cinza foi usado o padrão NTSC, descrito na equação 3.1.

$$gray^* = \begin{pmatrix} 0.299 & 0.587 & 0.114 \end{pmatrix} \begin{pmatrix} red^* \\ green^* \\ blue^* \end{pmatrix} \quad (3.1)$$

### 3.3 Redimensionamento de Imagens

Algoritmos para o reconhecimento de formas exigem grande custo computacional. Portanto é comum reduzir a dimensão de uma imagem para diminuir o tempo de retorno. Como as imagens no conjunto de testes foram digitalizadas a 600dpi, a dimensão destas imagens pode ser significativamente reduzida sem que haja perda na eficiência de detecção. Foi implementado então um redimensionador de imagens.



**Figura 3.1** Ilustração do algoritmo 1, usado para a separação das fotos

Existem diversos algoritmos para redimensionamento de imagens. Dentre eles, foram escolhidos dois para implementação: (1) intercalação, (2) médias.

1. O algoritmo de intercalação é o mais simples. Consiste em substituir um conjunto de pixels pelo primeiro pixel do conjunto.

Seja  $g$  uma imagem de dimensão  $width \times height$ , a taxa de redução  $t$ , e a imagem gerada  $g'$  de dimensão  $width' \times height'$ . Temos que  $width \cdot height \cdot t = width' \cdot height'$ .

Para o redimensionamento ser justo, o redimensionamento em relação às linhas e às colunas deve ser proporcional. Portanto, é necessário que:

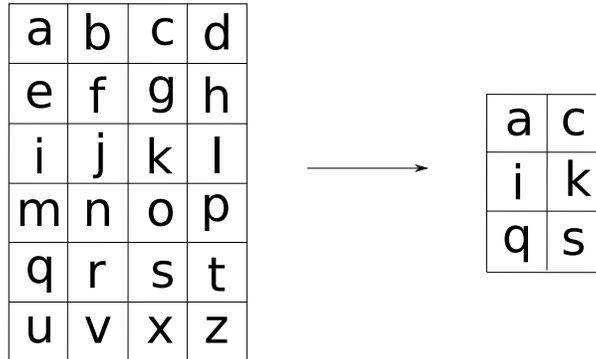
$$(a) \quad width' = width \cdot \sqrt{t}$$

$$(b) \quad height' = height \cdot \sqrt{t}$$

Portanto,  $\frac{width'}{width} = \frac{height'}{height} = \sqrt{t^{-1}}$  é o período de intercalação dos pixels. A nova imagem  $g'$  é dada pela equação:

$$g'(i, j) = g(i\sqrt{t^{-1}}, j\sqrt{t^{-1}}). \quad (3.2)$$

A figura 3.2 ilustra a aplicação desse algoritmo para uma imagem 4x6, usando 25% como taxa de redução.

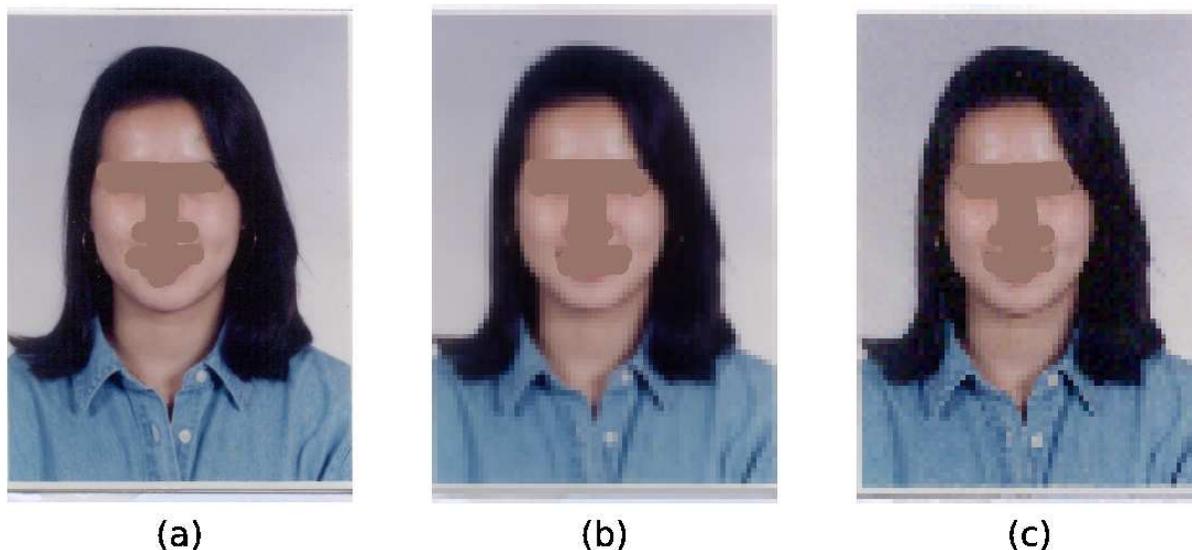


**Figura 3.2** Redimensionamento de uma imagem 4x6 a 25% pelo algoritmo de intercalação. Neste caso  $\sqrt{t^{-1}} = 2$ . A imagem resultante tem dimensão 2x3.

2. O algoritmo de médias faz uso de todos os pixels da imagem original para gerar a imagem reduzida. Um conjunto de pixels é substituído pela média entre os valores dos pixels do conjunto. Para a taxa de redução  $t$  de entrada é usado o filtro quadrático  $q$  de lado  $m = \sqrt{t^{-1}}$  onde  $q(i, j) = 1$ . A imagem  $g'$  é definida por:

$$g'(i, j) = \frac{1}{t} \sum_{k=0}^{m-1} \sum_{l=0}^{m-1} g(i+k, j+l). \quad (3.3)$$

A figura 3.3 mostra os resultados desses dois algoritmos para a mesma imagem. Como pode ser notado, o resultado do redimensionamento pelo algoritmo de médias ganha em suavidade (menos ruído); enquanto que o resultado pelo algoritmo de intercalação, em nitidez.



**Figura 3.3** (a) imagem original, (b) redimensionada pelo algoritmo de médias, (c) redimensionada pelo algoritmo de intercalação.

### 3.4 Segmentação de Faces

A segmentação é o ponto chave para a eficiência de qualquer algoritmo de reconhecimento de faces, já que as fases posteriores são rigidamente dependentes às suposições feitas sobre os resultados dessa etapa.

Nas subseções seguintes são descritos os algoritmos usados para a segmentação de faces.

#### 3.4.1 Segmentação Baseada em Bordas

Na segmentação baseada em bordas, uma imagem na escala RGB é transformada em níveis de cinza, e em seguida informações sobre o gradiente morfológico [2] dessa imagem são usadas para gerar a imagem binária de bordas. Veja os algoritmos 2 e 3.

A idéia inicial para gerar a imagem de silhueta a partir da imagem de bordas foi simplesmente aplicar varreduras dos pixels acesos nos sentidos esquerda-direita, direita-esquerda e cima-baixo, e em seguida aplicar intersecção dessas imagens. A figura 3.4 ilustra esse algoritmo.

---

**Algorithm 2** Calcula gradiente morfológico  $grad$  da imagem  $g$

---

```

1: for each  $g(x,y) \in img$  do
2:    $max \leftarrow$  intensidade máxima entre  $g(x,y)$  e seus 8 vizinhos
3:    $min \leftarrow$  intensidade mínima entre  $g(x,y)$  e seus 8 vizinhos
4:    $grad(x,y) \leftarrow max - min$ 
5: end for
6: return  $grad$ 

```

---



---

**Algorithm 3** Encontra imagem de bordas  $edge$  de  $g$

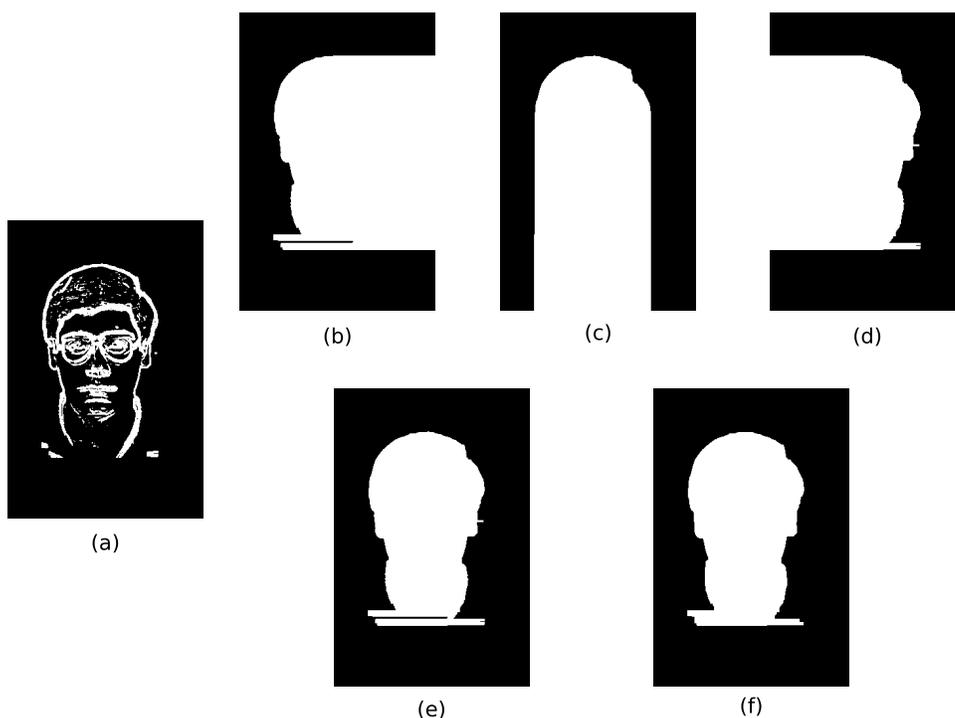
---

```

1:  $grad \leftarrow$  gradiente de  $img$ 
2:  $média \leftarrow$  média entre todos os valores de  $grad$ 
3: for each  $grad(i,j)$  do
4:   if  $grad(i,j) - média \geq média/2$  then
5:      $edge(i,j) \leftarrow 1$ 
6:   else
7:      $edge(i,j) \leftarrow 0$ 
8:   end if
9: end for
10: return  $edge$ 

```

---

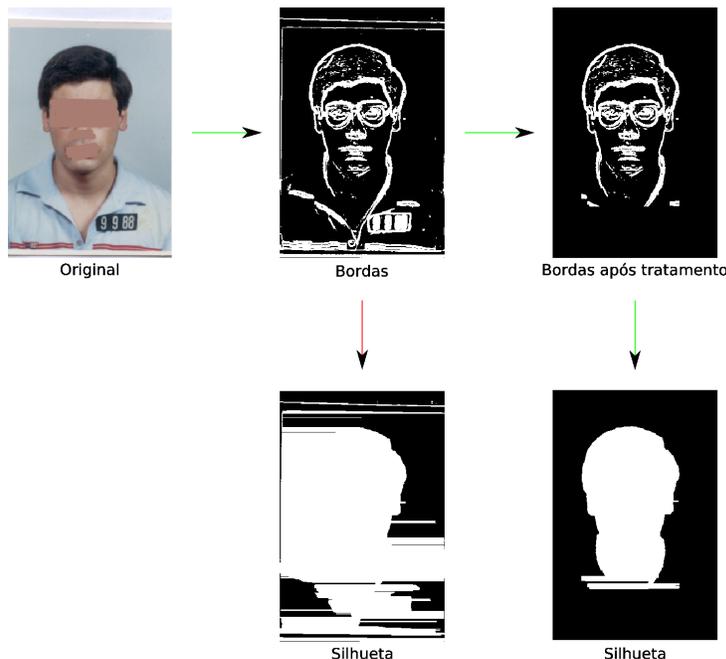


**Figura 3.4** (a): imagem original. (b)(c)(d): varreduras dos pixels acesos de (a). (e): intersecção entre (b)(c)(d). (f): aplicação de abertura seguido de fechamento em (e) com elemento estruturante quadrado 2x2 para tirar ruídos e possíveis pequenos "buracos" na imagem.

Porém, como as imagens são recortadas automaticamente pelo algoritmo 1, os cantos geralmente contêm resíduos da tabela. Portanto, para grande parte das imagens, os resultados iniciais foram insatisfatórios.

Para resolver esse problema, assume-se que a face do indivíduo está razoavelmente centrada na imagem. A imagem de bordas é analisada do centro para os cantos, e quando a soma dos pixels acesos é baixa na linha (para o corte superior e inferior) ou na coluna (para os cortes laterais), a região é o limite da expansão, que é aceita como região que contém a cabeça da pessoa.

Esse tratamento melhora significativamente os resultados, como mostra a figura 3.5.



**Figura 3.5** O tratamento da imagem de bordas traz melhores resultados na detecção da silhueta da pessoa.

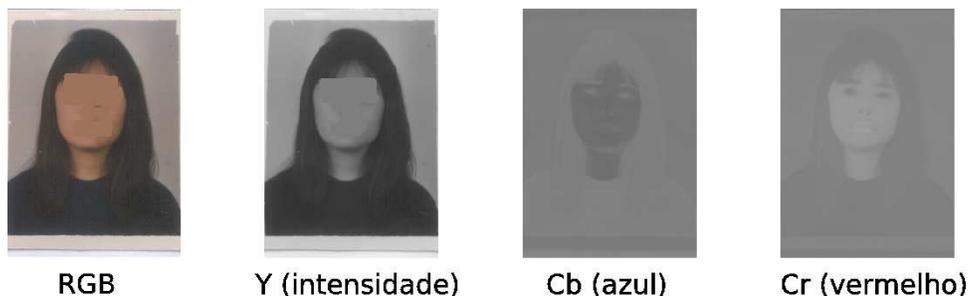
### 3.4.2 Segmentação Baseada em Cor da Pele

O algoritmo baseado em cor da pele estima a probabilidade de determinado pixel pertencer a uma região de pele, a partir de aprendizado por amostragem.

Apesar da cor de pele em imagens no formato RGB serem muito variantes, a cor da pele das pessoas é semelhante, variando apenas em intensidade [4]. A fim de eliminar os efeitos da luminosidade sobre as imagens, o modelo representativo da pele é transformado para a escala YCbCb (luminance-chrominance), onde Y representa a intensidade, Cb a componente azul e Cr a componente vermelha. A Equação 3.4, extraída de <http://en.wikipedia.org/wiki/YCbCr>,

mostra como gerar YCbCr a partir da escala RGB.

$$\begin{pmatrix} Y^* \\ Cb^* \\ Cr^* \end{pmatrix} = \begin{pmatrix} 0.299 & 0.587 & 0.114 \\ -0.168736 & -0.331264 & 0.5 \\ 0.5 & -0.418688 & -0.081312 \end{pmatrix} \begin{pmatrix} R^* \\ G^* \\ B^* \end{pmatrix} \quad (3.4)$$



**Figura 3.6** Transformação de RGB para a escala YCbCr

A função de distribuição da cor da pele pode ser aproximada para o modelo gaussiano  $N(\mu, C)$ , onde  $\mu$  é a média das componentes  $Cb$  e  $Cr$ , e  $C$  a covariância entre elas [1].

Segundo Nariman Habili, Cheng-Chew Lim e Alireza Moini (2001), a probabilidade de um pixel pertencer a uma região de pele é:

$$p(x) = \frac{1}{2\pi\|C\|^{\frac{1}{2}}} e^{-\frac{1}{2}(x-\mu)C^{-1}(x-\mu)} \quad (3.5)$$

A quantidade  $d$  em

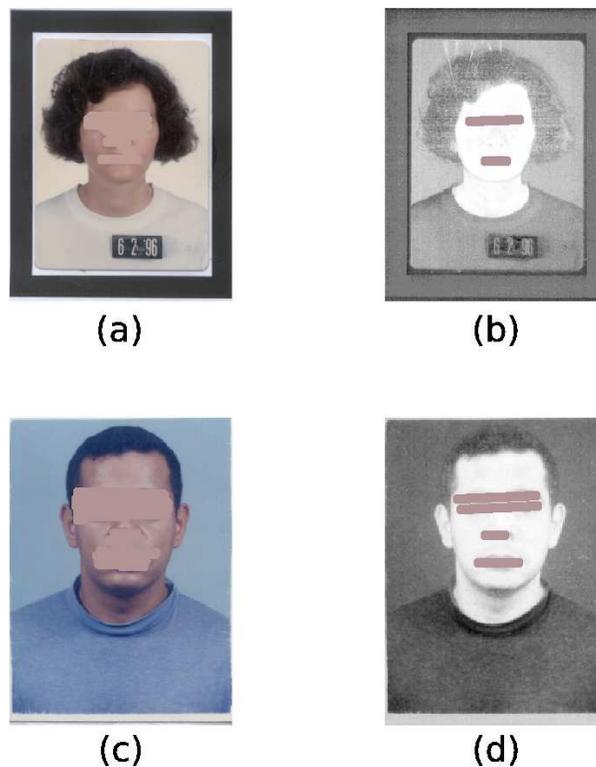
$$d^2 = (x - \mu)C^{-1}(x - \mu) \quad (3.6)$$

é conhecida como distância de Mahalanobis de  $x$  a  $\mu$ . O valor de  $d$  é inversamente proporcional à probabilidade de dado pixel pertencer a região de pele [3].

Os parâmetros  $\mu$  e  $C$  foram estimados a partir de 81 amostras de dimensão 25x25 pixels de regiões de pele das imagens, recortadas manualmente, e os resultados foram:

$$\mu^* = ( 148.875 \quad 114.872 ) \quad C^{-1} = \begin{pmatrix} 0.044 & 0.0318 \\ 0.0318 & 0.042 \end{pmatrix}$$

A imagem de probabilidades em escala de cinza é gerada a partir da normalização de  $d$  no intervalo  $[0, 255]$ . Quanto mais claro o pixel, maior a probabilidade dele pertencer à uma região de pele. Veja a figura 3.7.



**Figura 3.7** (b) e (d) são as imagens de probabilidades respectivamente de (a) e (c)

Binarizando-se a imagem de probabilidades por limiarização chega-se à máscara da face segmentada possivelmente acompanhada de alguns ruídos.

Para identificar apenas a face nessa imagem binária assume-se que a face contém apenas uma componente conexa. As caixas que envolvem cada uma das componentes são avaliadas, e a componente conexa cujas propriedades da caixa envolvente (relação largura/comprimento, centralização, área) são mais aceitáveis, é escolhida como componente da face.

Porém, um limiar constante para todas as imagens mostrou-se ineficiente, já que o universo de testes é muito heterogêneo. Portanto, mostrou-se necessária a adaptação do fator limiar para cada imagem. Veja o algoritmo 4.

Mesmo com tal adaptação do fator limiar, os resultados ainda não foram satisfatórios. A amostragem é tão heterogênea que, para alguns resultados, a probabilidade dos pixels de não-face são a mesma dos pixels da face. Se a amostragem fosse reduzida, algumas imagens obviamente teriam resultados problemáticos na segmentação. Surgiu então a idéia de extrair uma amostra da própria imagem para aplicar a segmentação baseada em cor de pele.

Um simples teste de segmentação com extração de alguns pixels (quadrado 20x20) centrais da própria imagem para amostragem mostrou-se mais eficaz que o mesmo algoritmo com

**Algorithm 4** Segmenta *img* com limiar adaptativo

---

```

1: prob  $\leftarrow$  imagem de probabilidades de img
2: limiar  $\leftarrow$  LIMIARpADRAO
3: repeat
4:   bin  $\leftarrow$  imagem binaria de prob usando limiar como threshold
5:   qualidade  $\leftarrow$  melhor resultado de caixa envolvente entre as componentes conexas de bin
6:   if qualidade = PEQUENA_DEMAIS & (limiar + MAX)/2 >  $\epsilon$  then
7:     limiar  $\leftarrow$  (limiar + MAX)/2
8:   else if qualidade = GRANDE_DEMAIS & (limiar + MIN)/2 >  $\epsilon$  then
9:     limiar  $\leftarrow$  (limiar + MIN)/2
10:  else
11:    desiste
12:  end if
13: until qualidade = OK
14: return componente conexa de bin envolvida pela melhor caixa

```

---

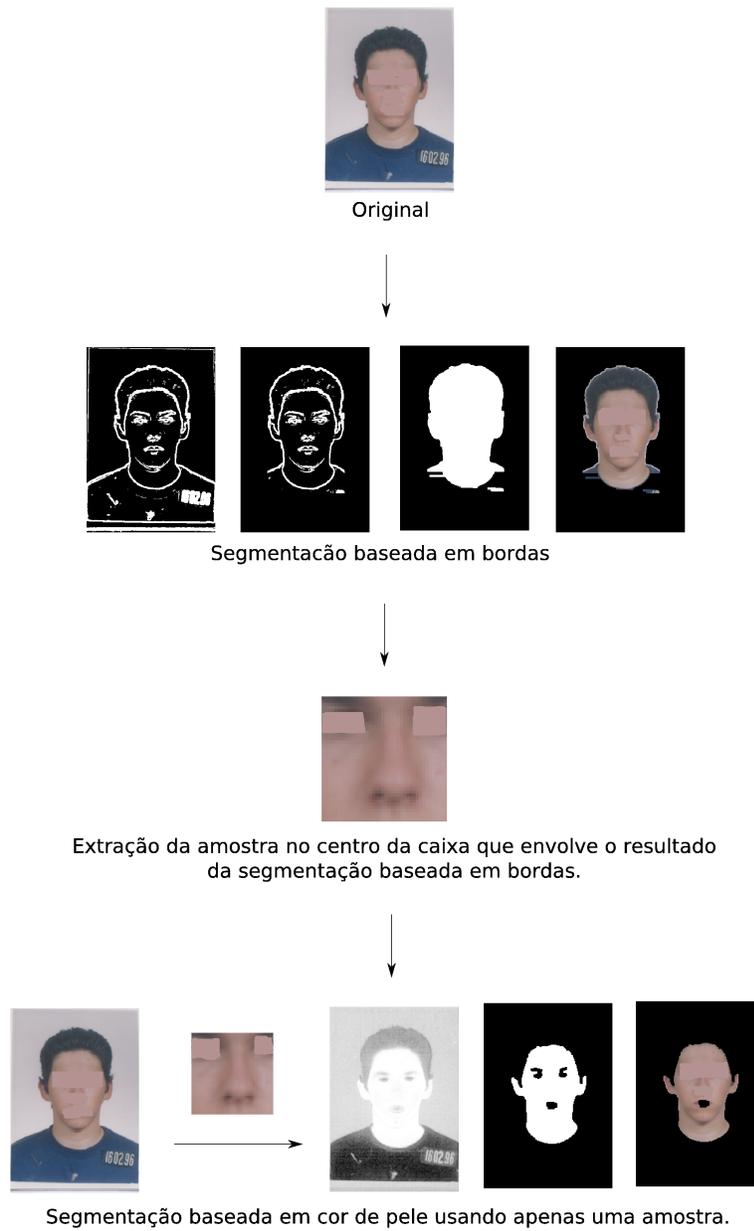
as 81 amostras 25x25 de diversas imagens, recortadas manualmente. O resultado desse teste não foi ruim porque o rosto das pessoas, na maioria dos casos, está centrado na imagem, o que garante muitas vezes uma boa coleta amostral para a pele.

O sucesso no teste anterior sugeriu a abordagem que é discutida na próxima seção.

### 3.4.3 Segmentação Híbrida

Seguindo a idéia de coletar amostras de pele da própria face a ser segmentada, discutida na seção anterior, nessa abordagem a amostra é coletada a partir de informações do resultado da segmentação baseada em bordas para a mesma imagem.

A segmentação híbrida ainda está em fase de desenvolvimento. Nos testes aplicados, é simplesmente extraída uma amostra 20x20 pixels do centro da caixa que envolve o resultado da segmentação baseada em bordas. Veja a figura 3.8, que ilustra o algoritmo.



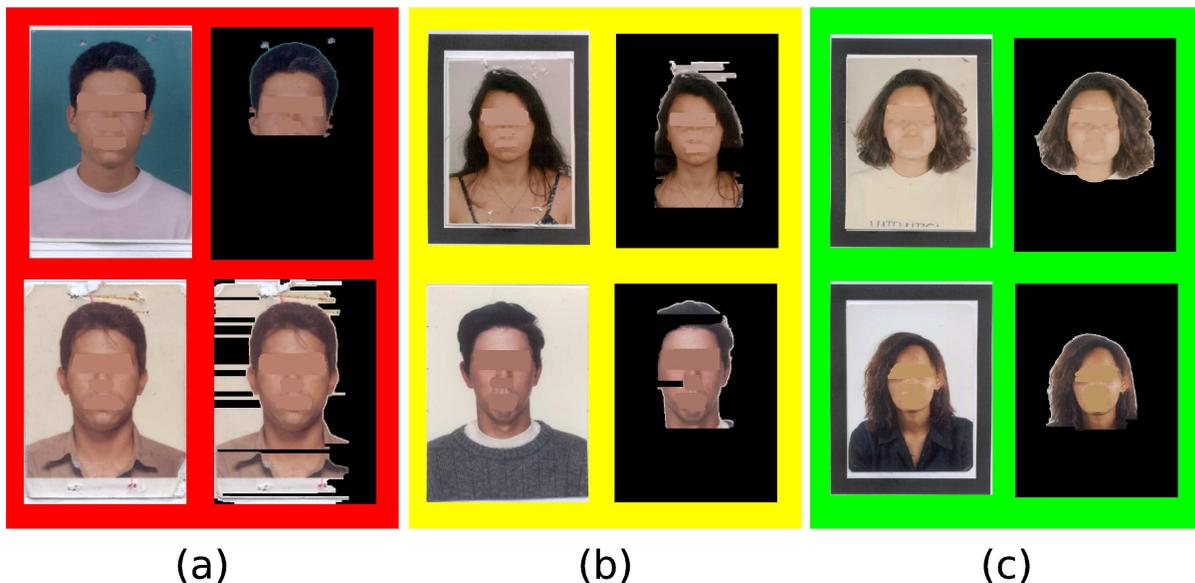
**Figura 3.8** Exemplo de segmentação híbrida

# Resultados

Os resultados da segmentação facial são classificados em 3 grupos: (1) RUIM, (2) REGULAR e (3) BOM. As figuras 4.1 e 4.2 mostram exemplos dessas classes respectivamente para os algoritmos de segmentação baseado em bordas e baseado em cor de pele.

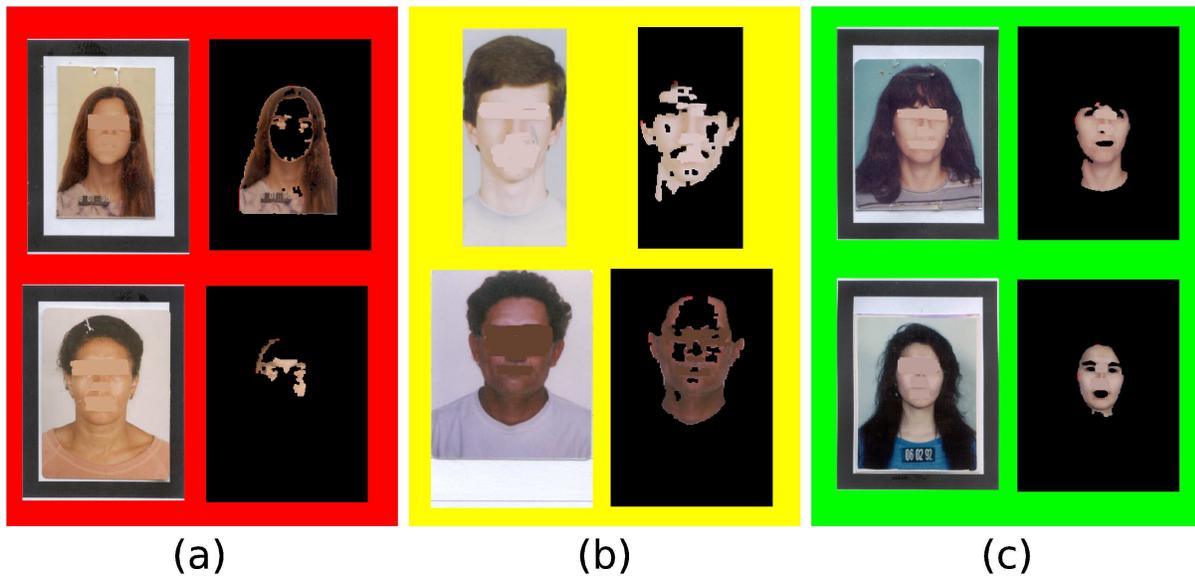
1. RUIM: A região segmentada não corresponde ao objetivo.
2. REGULAR: A região segmentada corresponde parcialmente ao objetivo.
3. BOM: A região segmentada corresponde exatamente ao objetivo.

O objetivo (resultado esperado) da segmentação baseada em bordas é a segmentação da cabeça (face + cabelo), enquanto que o objetivo da segmentação baseada em cor de pele é a face. Por isso, uma comparação entre os resultados obtidos por esses algoritmos talvez não possam ser comparados entre si.



**Figura 4.1** Classificação dos resultados da segmentação baseada em bordas em 3 níveis: (a) RUIM, (b) REGULAR, (c) BOM

A eficiência da segmentação baseada em cor de pele foi baixa. A diversidade das fotos consideradas fez com que, mesmo na escala YCbCr, a cor (Cb e Cr) de pele das pessoas ainda



**Figura 4.2** Classificação dos resultados da segmentação baseada em cor de pele em 3 níveis: (a) RUIM, (b) REGULAR, (c) BOM

fosse muito variante. Com uma amostragem abrangente, o desvio padrão aumenta muito, e para uma amostragem pequena, as cores de pele de algumas pessoas não são reconhecidas pelo algoritmo.

Porém, a modelagem estatística para cor de pele mostrou ser eficiente nos casos onde o desvio padrão da amostragem é pequeno, como no caso da segmentação híbrida.

Como pode ser observado na tabela 4, o método mais eficiente para segmentação das faces foi o hibridismo da técnica baseada em bordas com a técnica baseada em cor de pele. Apesar da simplicidade do método, os resultados são surpreendentes.

Segue a tabela de resultados obtidos pelas 3 diferentes técnicas de segmentação para uma amostra de 40% das fotos.

	<b>Baseada em Bordas</b>	<b>Baseada em Cor de Pele</b>	<b>Segmentação Híbrida</b>
<b>RUIM</b>	16,6%	23,3%	6,9%
<b>REGULAR</b>	9,7%	13,2%	9,5%
<b>BOM</b>	73,7%	60,5%	83,6%

**Tabela 4.1** Classes de resultados para as 3 abordagens descritas

*obs: As fotos em preto e branco (que na verdade deveriam ser chamadas de "fotos em escala de cinza") não foram consideradas nos testes para a segmentação baseada em cor de pele e segmentação híbrida.*

## Parte Subjetiva

### 5.1 Considerações sobre o Curso

O curso de Bacharelado em *Ciências da Computação* do IME é bem abrangente. Cobre praticamente todos os tópicos relacionados ao assunto. A abordagem genérica dos conceitos lecionados nas disciplinas permite que o aluno desenvolva a capacidade de aprender tecnologias facilmente.

Porém, eu imaginava, antes de me matricular, que o curso fosse de cunho mais prático. Acredito que algumas matérias são teóricas demais, e acabam reprovando alunos que não estão muito interessados no assunto, como é o caso de MAC300 e MAC315. Sei que teoria é muito importante, mas é mais importante satisfazer as necessidades da maioria do corpo discente. Se essas disciplinas fossem mais superficiais, os alunos interessados poderiam cursar uma suposta disciplina optativa que aprofundasse mais o assunto. Assim, os alunos que acreditam precisar apenas do básico dessas disciplinas ficariam mais satisfeitos.

De forma geral, o curso exige bastante do aluno. Isso é muito importante, pois este pode mostrar seu verdadeiro potencial.

Sem desmerecer as demais disciplinas, separei algumas que foram e serão as mais relevantes para o desenvolvimento do projeto:

- **Introdução à Computação**

Foi nessa disciplina que tive os primeiros contatos com programação de computadores.

- **Princípios de Desenvolvimento de Algoritmos**

Fornece conhecimentos fundamentais para criação e manipulação de estruturas de dados básicas (pilha, fila, lista ligada, árvore, etc), além de conceitos básicos em análise de algoritmos. Até mesmo o problema de menor caminho em grafos (Dijkstra) foi abordado.

- **Introdução à Probabilidade e à Estatística I e II**

Na segmentação baseada em cor de pele, é analisada a probabilidade de pertencer a uma região de pele para cada pixel. Foi necessário o uso de conceitos como *estimação da*

*variância, matriz de covariância e modelo gaussiano bidimensional.*

- **Cálculo I, II e III**

Para desenvolver algoritmos de visão computacional, bons conhecimentos em cálculo são de extrema importância.

- **Programação Orientada a Objetos**

O projeto está sendo desenvolvido em *Java*, e o conhecimento sobre a filosofia de orientação a objetos é importante para o desenvolvimento de projetos grandes nessa linguagem.

- **Análise e Reconhecimento de Formas: Teoria e Prática**

Apesar de um pouco tarde para ter me ajudado no desenvolvimento do projeto até a data de entrega neste trabalho, diversos conceitos relacionados ao projeto foram lecionados. Com esses novos conhecimentos pretendo melhorar a eficiência na segmentação das faces no projeto usando outras abordagens.

## 5.2 Considerações sobre a Equipe

Quando optei por desenvolver tal sistema, no mês de abril deste ano, por recomendação dos orientadores procurei por alguns matriculados na disciplina para formar uma equipe. Porém, todos já tinham projetos para o trabalho de formatura em desenvolvimento, ou iriam apresentar o projeto em que estavam estagiando.

Desenvolver um projeto individualmente e sem experiência no assunto é bem complicado. Por isso, comunicação constante com os orientadores foi fundamental para discussão sobre os caminhos que seriam seguidos.

O relacionamento com os orientadores não poderia ter sido melhor. O ambiente amigável e a constante discussão de idéias permitiram que, mesmo com as dificuldades encontradas durante desenvolvimento, eu não desanimasse.

## 5.3 Dificuldades

O maior problema encontrado foi a total inexperiência em visão computacional. Além disso, os algoritmos usados atualmente para reconhecimento de formas são muito complexos.

Apesar da diversidade das imagens, não era esperada tamanha complexidade na segmentação das faces. Por isso, o resultado final foi menor do que o proposto inicialmente.

## **5.4 Próximos Passos**

Gostei muito de ter trabalhado no projeto, e quero continuar desenvolvendo-o. E antes de seguir para a extração de características, quero conseguir eficiência próxima dos 100% para a segmentação de faces, usando técnicas mais avançadas.

## **5.5 Agradecimentos**

Gostaria de agradecer aos meus orientadores no projeto graduação: À Nina pela sua disposição e paciência, e ao Roberto pelas imagens cedidas.

# Referências Bibliográficas

- [1] Maria João Medeiros de Vasconcelos. Modelos pontuais de distribuição em visão computacional: Estudo, desenvolvimento e aplicação. Master's thesis, Faculdade de Engenharia da Universidade do Porto, 2005.
- [2] R. A. Lotufo E. R. Dougherty. *Hands-on Morphological Image Processing*. SPIE Press, 2003.
- [3] Nariman Habili, Cheng-Chew Lim, and Alireza Moini. Hand and face segmentation using motion and color cues in digital image sequences. IEEE International Conference on Multimedia & Expo 2001, Tokyo, Japan.
- [4] M. J. Jones and J. M. Rehg. Statistical color models with application to skin detection. IEEE Conference on Computer Vision and Pattern Recognition, Ft. Collins, CO, USA, 1999.
- [5] Peter H. Lindsay and Donald Norman. *Human Information Processing: An Introduction to Psychology*. Academic Press. New York, USA, 1977.
- [6] Wikipedia, The Free Encyclopedia. <http://www.wikipedia.com>.
- [7] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld. Face recognition: A literature survey. *ACM Press*, 2003.

