

ALGORITMOS PARA SEGMENTAÇÃO DA PELE UTILIZANDO MODELOS DE CORES RGB EM AMBIENTE MATLAB/SIMULINK

Rodrigo Fernandes Freitas⁽²⁾

Graduando em Mecatrônica Industrial (CEFETCE).

Rodrigo Carvalho Souza Costa⁽²⁾

Engenheiro Eletricista pela Universidade Federal do Ceará. Mestrando Engenharia de Teleinformática pela Universidade Federal do Ceará (UFC).

Antonio Carlos da Silva Barros⁽²⁾

Tecnólogo em Mecatrônica pelo Centro Federal de Educação Tecnológica do Ceará. Mestrando Engenharia de Teleinformática pela Universidade Federal do Ceará (UFC).

Robson da Silva Siqueira⁽¹⁾

Engenheiro Eletricista pela Universidade Federal do Ceará. Mestre em Engenharia de Teleinformática pela Universidade Federal do Ceará (UFC). Professor do curso de Telemática UNED Maracanaú do CEFET-CE.

Paulo César Cortez⁽²⁾

Doutor em Engenharia Elétrica pela Universidade Federal da Paraíba (UFPB), Pós Doutor pelo Institut National Dès Télécommunications. Professor Adjunto da Universidade Federal do Ceará.

José Marques Soares⁽¹⁾

Bacharel em Informática pela Universidade de Fortaleza (UNIFOR). Doutor pelo Institut National Dès Télécommunications. Professor do curso de Telemática UNED Maracanaú do CEFET-CE.

Endereço⁽¹⁾: Centro Federal de Educação Tecnológica – CEFETCE, Av. 13 de Maio, 2081 - Benfica - Fortaleza/CE CEP:60040-531 Fone +55 (85) 3307-3666 Fax (85) 3307-3711. E-mail: {siqueira, marques}@cefetce.br

Endereço⁽²⁾: Universidade Federal do Ceará, Laboratório de Engenharia de Sistemas de Computação (LESC), Campus do Pici S/N, Bloco 723, CEP 60455-970. Tel.: (85) 3366-9608. E-mail: {rodrigo, rodcosta, carlos, cortez}@lesc.ufc.br

RESUMO

Este documento propõe uma técnica de segmentação de imagens digitais capturadas por *webcam* pela cor da pele utilizando o sistema de cores RGB. A técnica visa aplicações em novas formas de interface homem-máquina, consistindo de um algoritmo executado em tempo real com aplicação de um processo de limiarização nos três canais do modelo RGB. Um protótipo é desenvolvido em *Matlab/Simulink*. Testes são realizados e os resultados são apresentados neste artigo, demonstrando a eficiência da técnica proposta em ambientes com diferentes condições de iluminação e cenários de complexidades variadas.

PALAVRAS-CHAVE: Visão Artificial, *Matlab* e *Simulink*, Processamento de Imagens, Segmentação da Pele, Limiarização, RGB.

INTRODUÇÃO

Os primeiros dispositivos periféricos usados para entrada de dados necessitavam de conhecimentos específicos dos usuários, como os cartões perfurados. Com o desenvolvimento de novos dispositivos de entrada, algumas formas de interação com o computador têm migrado do teclado para o *mouse*, restringindo o teclado a funções de digitação de texto [1].

Pode-se dizer que o *mouse* se tornou um dispositivo indispensável ao usuário, oferecendo melhor ergonomia na interação com objetos gráficos como botões, janelas, menus e barras de rolagem, tornando o uso deste dispositivo mais intuitivo do que o do teclado [1,2].

Em decorrência da diminuição dos custos de *hardware*, a câmera digital tem se tornado um periférico cada vez mais presente nas configurações usuais dos computadores pessoais. Paralelamente, a capacidade dos processadores tem aumentado sobremaneira e, conjugados às câmeras digitais, estimulam o emprego de técnicas de visão computacional e processamento de imagens para o desenvolvimento de novas aplicações de interface entre o homem e o computador [3].

Assim como algumas funções do teclado estão sendo substituídas pelo uso do *mouse*, a Visão Computacional, possibilita novas formas de interação mais intuitivas, por vezes permitindo o uso do próprio corpo para a execução de comandos em substituição a dispositivos físicos [1]. Neste contexto, a identificação de regiões de pele em imagens digitais é de fundamental importância para esta nova forma de interação homem-máquina.

A segmentação de pele compreende o processo de selecionar os *pixels* em uma imagem correspondentes à pele humana. Este processo é útil, por exemplo, para detecção e rastreamento de face para aplicações em segurança, ou para o reconhecimento de gestos e comunicação à distância [4,5]. A maior parte dos métodos desenvolvidos para a segmentação da pele é baseada em atributos de cor. As regiões de pele são detectadas através de *pixels* que tem tonalidades próximas a da pele [4].

Este trabalho apresenta um algoritmo de segmentação da pele em imagens digitais para aplicações em tempo real, o qual pode ser usado no desenvolvimento de interfaces baseadas em comunicação por gestos. O processo constitui-se basicamente da aquisição da imagem através de uma *webcam* e da aplicação de um processo de limiarização sobre cada *frame* a fim de se separar o que é pele do restante da imagem. Através de um protótipo implementado em ambiente *Simulink®*, a presente metodologia possibilita a avaliação da qualidade da técnica de segmentação proposta, bem como o esforço computacional necessário, gerando indicativos para possíveis aplicações em tempo real.

O restante do trabalho está organizado da seguinte forma: a seção de materiais e métodos descreve os conceitos básicos, as ferramentas e técnicas necessárias ao desenvolvimento do trabalho. Em seguida, são apresentados os resultados, algumas discussões e avaliações sobre os mesmos na seção resultados. Finalmente, na última seção são apresentadas algumas conclusões sobre o algoritmo proposto.

CONCEITOS BÁSICOS

Alguns conceitos básicos são importantes para a compreensão deste trabalho, tais como: espaço de cores, imagem digital, vídeo digital, segmentação de imagens digitais.

Espaço de Cores

Espaço de cores ou sistemas de cores são especificações de sistemas de coordenadas em que cada ponto deste corresponde a uma cor [6]. O espaço de cores RGB se caracteriza por um cubo de lado R, comprimento G e altura B, tendo sido idealizado por Thomas Young (1773-1829). Neste cubo, três dos vértices são ocupados pelas cores Vermelha (R), Verde (G) e Azul (B). O modelo de espaço de cores RGB é provavelmente o mais usado entre os modelos de cores, especialmente se os dados são representados por 8 bits, neste caso, tem-se um número total de cores de $(2^8)^3$, ou seja, mais de 16 milhões de cores.

Imagem Digital

Uma imagem digital é caracterizada por uma função em duas dimensões $f(x,y)$, em que x e y são as coordenadas de um determinado ponto, denominado de *pixel* (*picture element*), e a amplitude de f de qualquer par ordenado (x,y) é chamado de intensidade ou nível de cinza. Esta função pode ser representada matematicamente por uma matriz, em que cada elemento desta matriz é a amplitude de f e as coordenadas x e y correspondem às colunas e linhas, respectivamente [6].

As imagens digitais podem ser coloridas ou em níveis de cinza. No sistema RGB, por exemplo, as imagens coloridas assumem três canais que são os canais R (*Red*), G (*Green*) e B (*Blue*), sendo representada por uma matriz para cada canal [6]. Pode-se denominar de vídeo digital uma seqüência de imagens digitais e que estas são denominadas *frames*.

Histograma de uma Imagem Digital

A distribuição pela freqüência de ocorrência dos níveis de cinza de uma imagem digital é denominada de histograma desta imagem. Assim, uma imagem possui uma faixa de níveis de cinza no intervalo $[0, \dots, L]$, seu histograma $h(r_k) = n_k$ é uma função discreta, em que n_k é o número total de ocorrência dos níveis de cinza r_k dentro daquele intervalo [6].

Uma operação usual realizada sobre um histograma, com o objetivo de realce, é a limiarização. Esta consiste num mapeamento de *pixels* de uma imagem original $f(x,y)$ em outra imagem $g(x,y)$, por exemplo, dado pela equação

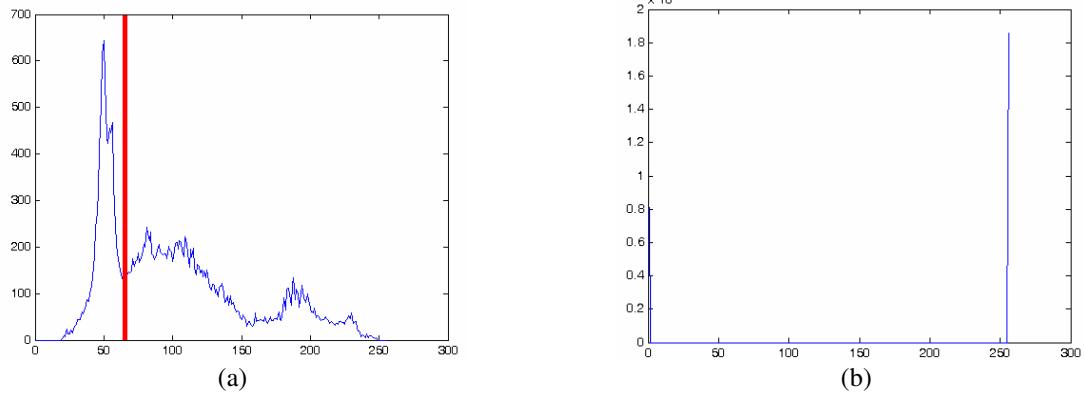
$$g(x, y) = \begin{cases} a, & \text{se } f(x, y) \leq L, \\ b, & \text{caso contrário,} \end{cases} \quad (1)$$

em que a , b e L são valores fixos, sendo que L é denominado de limiar, ou seja, é um valor limite que altera as propriedades dos *pixels* da imagem resultante $g(x,y)$, a partir da imagem original [6].

Figura 1: Resultado de um exemplo de limiarização aplicado a um histograma.(a) Imagem original.(b) Imagem limiarizada.



Figura 2: Histograma da imagem antes (a) e depois (b) da limiarização.



Segmentação de Imagens Digitais

O processo de segmentação de uma imagem digital consiste em subdividir uma dada imagem digital em regiões ou objetos com base em características destas regiões. Estas regiões são compostas por um conjunto de *pixels* com propriedades semelhantes. O nível de subdivisão é determinado pela aplicação específica [6].

SEGMENTAÇÃO DE PELE POR LIMIARIZAÇÃO

Segmentação é um processo que divide uma imagem em regiões. Apesar do modelo de cores RGB não ser o modelo de cores mais intuitivo, os melhores resultados na área de segmentação de imagens coloridas são geralmente obtidos através deste formato de dados. A partir de um conjunto de amostra representativo das cores de interesse, pode-se obter uma estimativa da cor média que se deseja segmentar na imagem. O processo de segmentação consiste em classificar cada *pixel* da imagem dentro da faixa de interesse especificada para a cor. Para realizar esta classificação, utiliza-se a limiarização, um dos algoritmos de segmentação mais intuitivos e rápidos empregados em processamento de imagens [6].

A limiarização funciona através do estabelecimento de um limiar T que separa a imagem em duas regiões. Os *pixels* da imagem cujos valores são maiores do que T são chamados de pontos do objeto e os pontos da imagem cujos valores sejam inferiores a T são chamados de fundo. Dessa forma, pode-se criar uma imagem de saída em que, por exemplo, os *pixels* referentes aos pontos do objeto recebem um valor 1 (branco) e os pixels referentes ao fundo recebem o valor 0 (preto) [6].

ALGORITMO PROPOSTO

A especificação de limiares consiste em delimitar as regiões no espaço de cor que consistem no objeto desejado [7]. Um conjunto de regras para realizar a segmentação da imagem pela cor da pele utilizando o formato RGB é descrito por `q`q [8], conforme apresentado nas equações que se seguem.

$$R > 95 \text{ e } G > 40 \text{ e } B > 20 \quad (2)$$

$$\max(R, G, B) - \min(R, G, B) > 15 \quad (3)$$

$$|R - G| > 15 \quad (4)$$

$$R > G \text{ e } R > B \quad (5)$$

A primeira regra é responsável por definir a tonalidade da cor de pele, a segunda é responsável por diminuir a influência de tons de cinza devido a proximidade dos valores de R, G e B. A terceira é responsável por manter a complexidade dos tons de pele, mantendo os valores dos tons vermelho (R) e verde (G) distantes. Por fim, a quarta regra é responsável por limitar as regiões de pele, pois o valor do pigmento vermelho tem de ser o maior componente [8].

A maior vantagem deste método é sua simplicidade, possibilitando o desenvolvimento de algoritmos com baixo esforço computacional [7]. Para a realização da segmentação da pele foi utilizada a metodologia ilustrada na Figura 3.

Figura 3: Fluxograma do processo de segmentação da pele.



METODOLOGIA (SIMULAÇÃO)

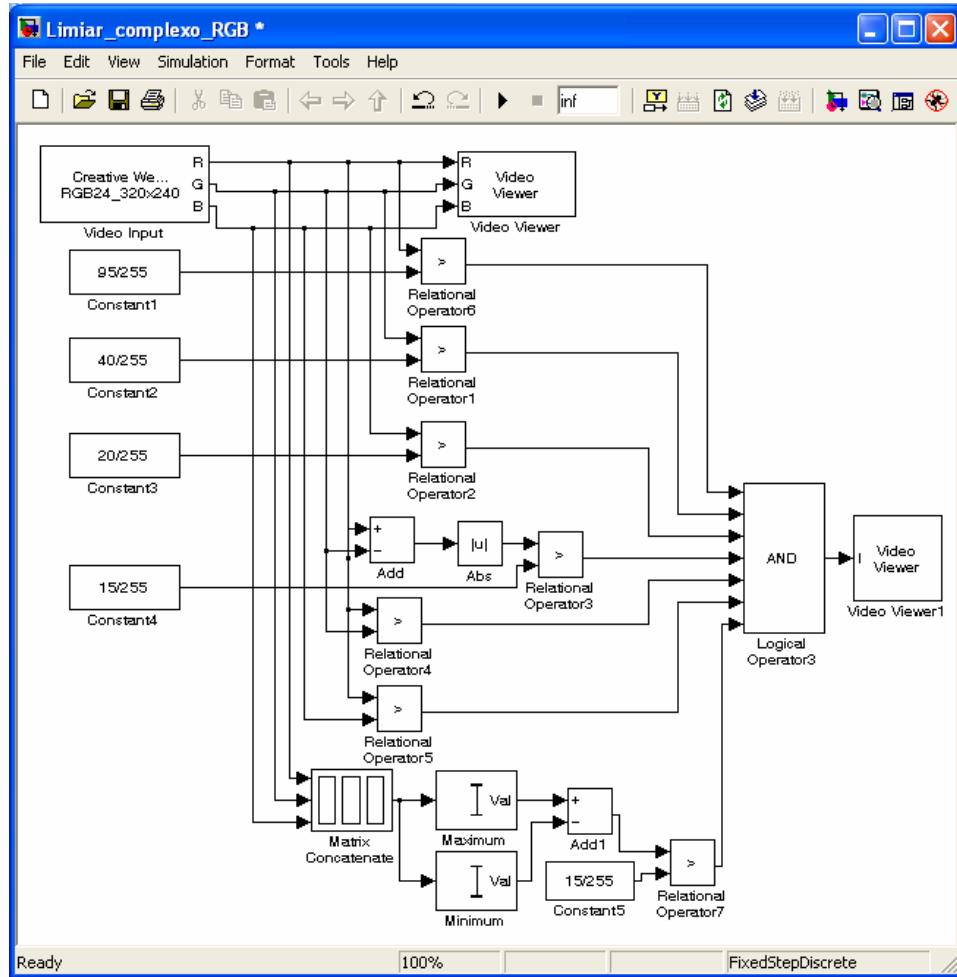
Como ambiente de simulação foram utilizados os pacotes de Aquisição de Imagens, *Simulink* e o *blockset* de processamento de imagens e vídeo do *Matlab®* versão 2006b em um computador Pentium D 2.8 GHz com 1Gb de RAM. O *Matlab* é uma ferramenta que dispõe de vários recursos para tratamento de imagens, bem como de vídeo.

O *Simulink* é uma plataforma para simulação em multidomínios e projeto de sistemas dinâmicos baseados em modelos. Ele fornece um ambiente gráfico interativo e um conjunto de blocos de bibliotecas personalizáveis, podendo ser extendido para aplicações especializadas [9].

Ele também permite realizar chamadas de funções desenvolvidas no *Matlab*, além de fornecer acesso a dispositivos, como por exemplo, a câmera. As funcionalidades do *Simulink* propiciam o desenvolvimento de

um protótipo em pouco tempo, pois realiza a investigação do tempo computacional, além de disponibilizar o poder de exibição disponível pelo *Matlab®*. Na Figura 4 é ilustrada a implementação desenvolvida neste ambiente.

Figura 4: Diagrama do *Simulink* para o algoritmo proposto.



Abaixo segue uma breve descrição dos blocos utilizados na implementação.

Vídeo Input: adquire os frames capturados pela webcam selecionada.

Vídeo Viewer: mostra o vídeo de entrada.

Constant: coloca na saída um valor constante definido.

Relational Operator: aplica uma operação relacional à entrada do bloco (maior que, menor que, igual a) e gera uma saída adequada (verdadeiro ou falso).

Add: soma ou subtrai algebricamente os valores das duas entradas.

Abs: gera na saída o valor absoluto da entrada.

Matrix Concatenate: concatena os sinais de entrada.

Maximum: gera na saída o valor máximo da entrada.

Minimum: gera na saída o valor mínimo da entrada.

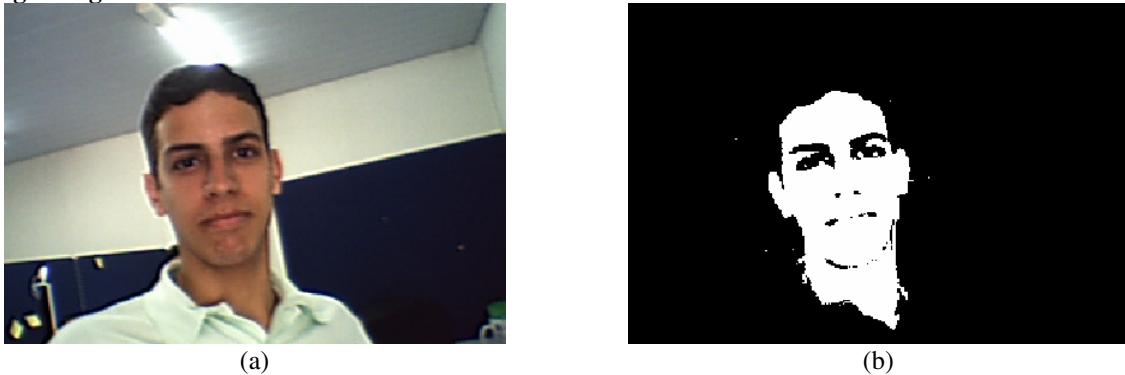
Na técnica proposta, o bloco *Video Viewer* é responsável pela visualização da imagem original, depois temos os blocos dos sete limiares, o bloco *Logical Operator* do tipo AND para verificar se todos os limiares estão sendo seguidos e por último a visualização da imagem processada.

RESULTADOS

Na obtenção dos resultados, são considerados ambientes do tipo simples e do tipo complexo. Esta definição de simples ou complexo para os ambientes, baseia-se na menor ou maior presença destes de objetos com cores semelhantes ao da cor da pele humana, respectivamente. Os resultados são apresentados em forma de imagens binárias, em que a cor branca está associada à pele e a cor preta ao fundo do ambiente.

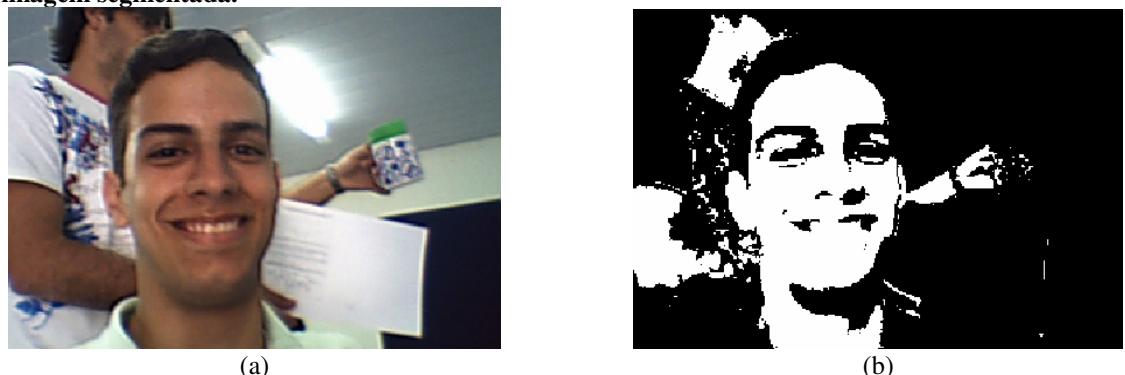
Na Figura 5, é mostrado o resultado da segmentação do rosto numa imagem com um fundo simples. A segmentação obteve resultados satisfatórios, tendo conseguido separar em torno de 95% o rosto presente na imagem.

Figura 5 – Resultado da segmentação do rosto numa imagem com fundo simples: a) imagem original e b) imagem segmentada.



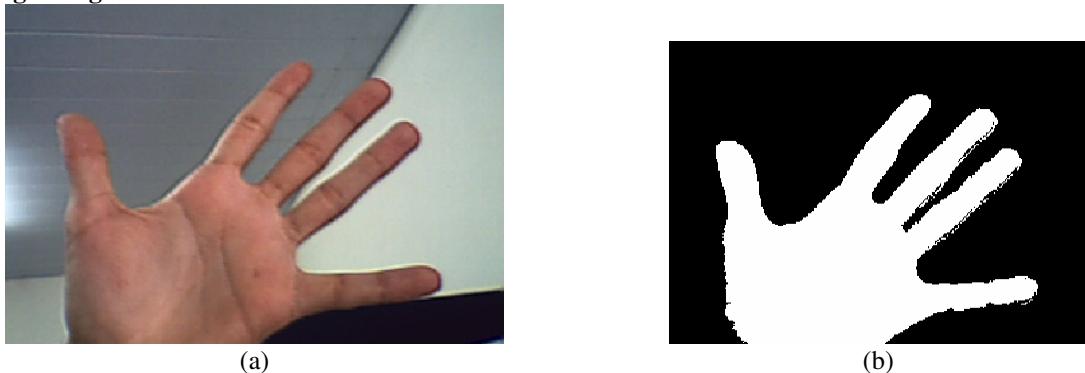
Na Figura 6 temos o resultado da segmentação do rosto numa imagem com fundo complexo. Neste caso não foram obtidos resultados tão satisfatórios quanto na situação anterior, pois algumas regiões do fundo estão sendo segmentadas como pele, igual na Figura 6.

Figura 6 – Resultado da segmentação do rosto numa imagem com fundo complexo: a) imagem original e b) imagem segmentada.



Na Figura 7 temos o resultado da segmentação da mão numa imagem com fundo simples. A segmentação apresentou resultados satisfatórios, tendo conseguido separar bem quase que toda a região da mão do resto da imagem.

Figura 7 – Resultado da segmentação da mão numa imagem com fundo simples: a) imagem original e b) imagem segmentada.



Na Figura 8 temos o resultado da segmentação da mão numa imagem com fundo complexo. Neste caso também não se obteve resultados tão satisfatórios, pois algumas regiões do fundo estão sendo segmentadas como pele, igual na Figura 6.

Figura 8 – Resultado da segmentação da mão numa imagem com fundo complexo.



Nas Figuras 9 e 10 temos o resultado da segmentação em imagens somente com fundo complexo, mostrando a pouca robustez do algoritmo proposto quando usado em imagens com fundo complexo.

Figura 9 – Resultado da segmentação num ambiente com fundo complexo: a) imagem original e b) imagem segmentada.

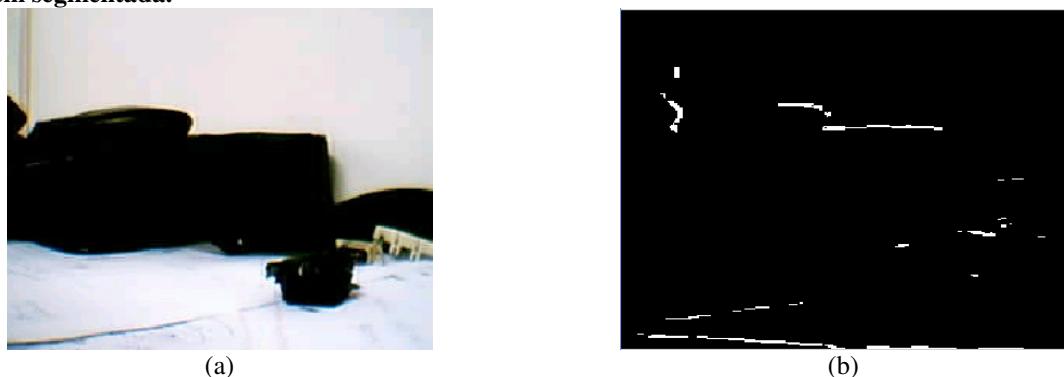
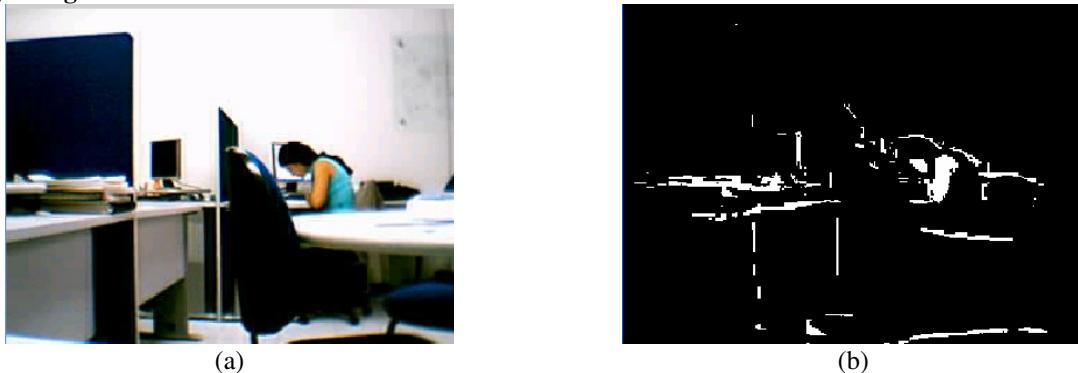


Figura 10 – Resultado da segmentação em outro ambiente com fundo complexo: a) imagem original e b) imagem segmentada.



Nas Figuras 11 e 12 temos o resultado da segmentação em imagens com objetos de cores parecidas com o tom da pele, que acabam sendo erroneamente segmentadas como regiões de pele.

Figura 11 – Resultado da segmentação em vários tipos de cores: a) imagem original e b) imagem segmentada.

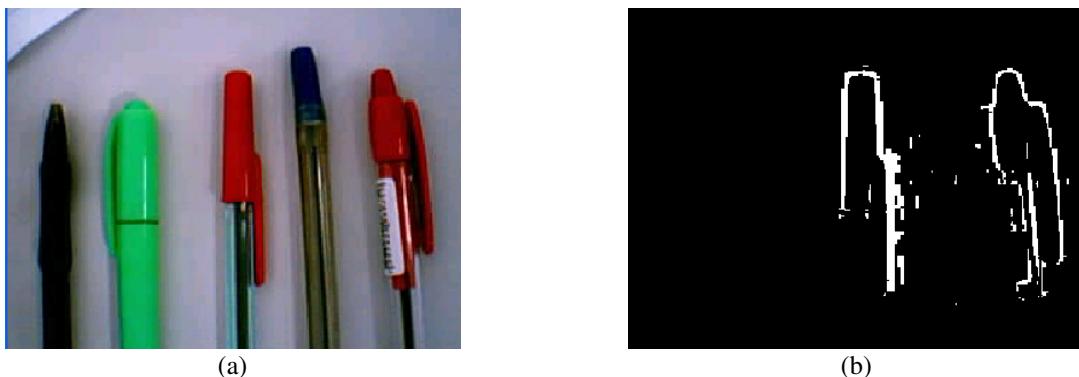


Figura 12 – Resultado da segmentação em cores parecidas com as cores da pele humana: a) imagem original e b) imagem segmentada.



CONSIDERAÇÕES

A técnica de segmentação apresenta resultados promissores, tendo limiarizado bem as regiões de pele tanto na imagem do rosto quanto na imagem da mão, quando numa imagem com fundo homogêneo e simples, como mostrado nas Figuras 5 e 7, respectivamente. Porém, esta técnica não funciona adequadamente quando aplicada numa imagem cujo fundo seja complexo, como mostrado nas Figuras 6, 8, 9 e 10, e segmenta erroneamente as cores semelhantes ao tom da pele (vermelho, amarelo e laranja), como mostrado nas Figuras 11 e 12.

CONCLUSÕES

A técnica de segmentação da pele proposta mostra ter boa aplicabilidade em ambientes de fundo simples, tendo conseguido fazer uma boa separação da pele do restante da imagem. Porém ela apresenta algumas limitações, como a exigência de um fundo uniforme para funcionar adequadamente e a deficiência de segmentar objetos das cores vermelha, amarela e laranja como pele.

Como trabalhos futuros, propõem-se o desenvolvimento de técnicas que eliminem o ruído presente nas imagens com fundo complexo e eliminem a falsa segmentação das cores vermelha, amarela e laranja.

REFERÊNCIAS BIBLIOGRÁFICAS

- [1] Truyenque, M. A. Q. *Uma Aplicação de Visão Computacional que Utiliza Gestos da Mão para Interagir com o Computador*, Mestrado em Informática, PUC, Rio de Janeiro, 2005.
- [2] Ribeiro, H. L. Reconhecimento de Gestos Usando Segmentação de Imagens Dinâmicas de Mão Baseada no Modelo de Mistura de Gaussianas e Cor de Pele, Dissertação, Universidade de São Paulo, São Paulo, 2006.
- [3] Grunnet—Jepsen, A.; Salsman, K.; Sweetser, J., Handheld Device For Handheld Vision Based Absolute Pointing System, Patente US 2006/0152487, 13. Jul 2006.
- [4] Albiol, A; Torres; Delp, E.J. 2001 Optimum colors spaces for skin detection. In Proceedings of the International Conference on Image Processing, vol 1, pág 122-124.
- [5] Han, J. et al. Automatic Skin Segmentation for Gesture Recognition Combining Region and Support Vector Machine Active Learning, Proceedings of the 7th International Conference on Automatic Face and Gesture Recognition (FGR'06).
- [6] Gonzalez, R.C.; Woods, R.E. *Image Processing*. Boston, Addison-Wesley Publishing.
- [7] Vezhnevets, V.; Sazonov, V.; Andreeva, A. A Survey on Pixel-Based Skin Color Detection Techniques, Graphics and Media Laboratory. Faculty of Computational Mathematics and Cybernetics, Moscow State University, Moscow, Russia.
- [8] Peer, P.; Kovak, J.; Solina, F. Human skin colour clustering for face detection. In submitted to EUROCON 2003 – International Conference on Computer as a Tool., 2003.
- [9] <http://www.mathworks.com/products/simulink/>