

# Mouse Ocular para Pseudocoma de Baixo Custo

Francisco de Assis Zampiroli<sup>1</sup>

Centro de Matemática, Computação e Cognição  
Universidade Federal do ABC  
Santo André, Brasil  
fzampiroli@ufabc.edu.br

Rafael Iamamoto Cuzziol

Centro de Matemática, Computação e Cognição  
Universidade Federal do ABC  
Santo André, Brasil  
Rafael.cuzziol@aluno.ufabc.edu.br

**Resumo** — Este artigo tem como objetivo desenvolver uma metodologia nova e de baixo custo para facilitar a interação de pacientes que não possuem qualquer forma de comunicação com o seu meio, seja ela gesticular ou verbal, vítimas da síndrome *Locked-In* sem a presença de oftalmologia. Com ele o paciente poderá interagir através de movimentos dos olhos com um computador que por sua vez intermediará a comunicação com o ambiente. O software desenvolvido é de código aberto, facilitando o acesso de grande parte da população e que possa ser implantado em larga escala em hospitais tanto particulares quanto públicos<sup>1</sup>.

**Palavras-chave** - mouse, pseudocoma, interfa, visão computacional.

## I. INTRODUÇÃO

Como uma reação ao sistema hegemônico de assistência à saúde passou-se a discutir-se amplamente o tema “humanização” principalmente nos serviços de saúde e hospitais [1].

Houve então o desenvolvimento de várias políticas e programas que auxiliam e melhoram o atendimento nestes locais como o Programa Nacional de Humanização da Assistência Hospitalar (PNHAH) onde alguns dos seus objetivos são: “Melhorar a qualidade e a eficácia da atenção dispensada aos usuários dos hospitais públicos no Brasil” e “Conceber e implantar novas iniciativas de humanização dos hospitais que venham a beneficiar os usuários e os profissionais de saúde” [2].

Há porém um grupo de pacientes onde estes programas têm muitas dificuldades em atuar pois os enfermos não têm nenhuma comunicação com o ambiente. Eles são principalmente vítimas de lesões cerebrais como o Acidente Vascular Cerebral (AVC), o Traumatismo Crânio Encefálico (TCE) ou algum tipo de Lesão Medular que danifica o tecido cerebral e medular podendo impossibilitar a movimentação e fala.

Desta forma o paciente permanece com um nível de consciência tido como normal, porém fica totalmente paralisado podendo em certos casos ter somente o movimento voluntário dos olhos, caracterizando a Síndrome *Locked-In* [3].

Estes pacientes se tornam restritos ao leito, necessitando de cuidados integrais da equipe médica e de enfermagem tornando fundamental a comunicação com o enfermo para proporcionar uma assistência mais eficiente e humanizada a

fim de diminuir o impacto dessa síndrome. Existem ainda alguns pacientes que foram vítimas de acidentes ocorridos fora de sua residência ou trabalho, sem que nenhuma pessoa conhecida soubesse do ocorrido. Esses pacientes quando chegam ao hospital são atendidos pelo serviço de assistência social, porém devido à falta de documentos e principalmente comunicação eles perdem sua identificação [4] e nunca mais conseguirão entrar em contato com qualquer familiar.

Este trabalho tem como objetivo sanar estes problemas de comunicação podendo facilitar o trabalho das equipes médicas e principalmente facilitar a vida destes pacientes dando a eles uma nova forma de comunicação e interação com o meio.

O ‘MOP’ (Mouse Ocular para Pseudocoma) tem como objetivo ser um software leve e possuir um hardware simples, de baixo custo, sendo capaz de ser executado em computadores baratos a fim de atender o maior número de pessoas podendo assim ser implantado em larga escala no sistema de saúde brasileiro.

A função que o Mouse Ocular para Pseudocoma se propõe a exercer já existe em alguns programas e dispositivos que estão em desenvolvimento ou já estão no mercado, porém muitos estão voltados para substituir o mouse comum ignorando o usuário final, ou então têm sua função voltada para a área de marketing analisando o olhar das pessoas frente a um anúncio [5][6], ou até mesmo necessitam de algum dispositivo diferenciado normalmente com um custo elevado em comparação com a capacidade financeira da maior parte da população, como é o caso do Iriscom® que custa 7.800,00 euros [7] e os sistemas da Tobii® que custam dezenas de milhares de dólares [8][9].

## II. PRELIMINARES

### A. Processo de Segmentação de Vídeos

Em visão computacional a segmentação é o processo de divisão de uma imagem em segmentos permitindo assim uma análise computacional e matemática a fim de se extrair dados [11]. O resultado desse processo é uma imagem binária (preto – nível de cinza com valor zero, e branco – nível de cinza com valor 1) que acaba por segmentar o objeto desejado.

Neste trabalho, para realizar esta segmentação utilizou-se a biblioteca OpenCV (*Open Source Computer Vision Library*) da Intel® que é uma biblioteca multiplataforma e livre para uso tanto não-comercial como comercial. Esta biblioteca foi utilizada com uma programação na linguagem

<sup>1</sup> O autor agradece o apoio da FAPESP, processo 2009/14430–1.

C++ que por ser de baixo nível permite desenvolver um software leve e versátil sem a necessidade de plug-ins especiais. Esta escolha deve-se ao fato de esta biblioteca já ter funções e classes prontas para visão computacional em tempo real o que a torna ideal para o desenvolvimento do projeto [12].

No início do módulo de programação e de validação de conceitos comparou-se a programação gerada através do MatLab® e a gerada através do Visual C++/OpenCV, com isso observou-se que as funções e principalmente os resultados gerados através da biblioteca OpenCV são diferentes dos gerados através do software MatLab®. Desta forma, a validação de conceitos foi realizada através da biblioteca OpenCV a fim de evitar esforços desnecessários e poupar tempo.

### B. Materiais

Por ter como um de seus objetivos atingir o maior público possível, o Mouse Ocular para Pseudocoma não precisa de materiais diferentes dos encontrados nos computadores simples e comuns. Basta uma CPU básica, do tipo doméstica, uma webcam e um boné. No estudo apresentado utilizou-se o modelo 10029 da marca Clone® que possui uma fonte de iluminação própria e cujo preço atualmente varia de R\$30,00 à R\$90,00. Porém há outras marcas que vendem modelos parecidos como a Neox® NXW 036 ou a Multilaser® WC040 cujos preços oscilam entre R\$24,00 e R\$80,00. Estes tipos de câmera têm 1.3 Mega Pixels e uma taxa de 30 fps (frames por segundo) e os modelos mencionados têm um clipe para prendê-las a notebooks o que as permite um melhor acoplamento à cabeça do paciente através do boné.

## III. RESULTADOS

### A. Estrutura do MOP

A estrutura do MOP é composta pelo computador e a câmera ajustada na aba do boné como mostra a Figura 2. Escolheu-se um boné para a estrutura por ser um apoio pronto para a câmera, que não necessita de reparos técnicos nem de ajustes complexos além de ser de fácil acesso. Também foi levado em consideração o fato de que o boné, assim como qualquer peça de roupa, é naturalmente feito para entrar em contato direto com a pele, e portanto o risco de uma alergia é menor e mesmo quando ocorrida poderá ser sanada apenas trocando-o por um com um tecido que o paciente não seja sensível.

Vale ressaltar que para um melhor funcionamento do sistema a aba do boné deve ser o mais plana possível como mostra a Figura 3(a), para que a câmera tenha um alinhamento próximo do ideal, como ilustra a Figura 3(b) onde o alinhamento do olho está representado pela linha verde e o alinhamento da câmera pela linha vermelha.

### B. Processo de Ajuste

O ajuste começa pela escolha da câmera a ser utilizada. Caso o usuário não altere as configurações iniciais é selecionada a câmera padrão. Essa informação é relevante visto que os netbooks e notebooks estão difundidos no

mercado, e eles já possuem uma câmera integrada cujo uso para este fim não é recomendado tendo em vista a distância que ela ficará do paciente. Nestes casos é necessária a escolha de uma segunda câmera que possa ser posicionada próximo ao olho do paciente como supracitado.

Após a escolha da câmera ela é posicionada conforme Figura 2 e sua direção e foco é ajustado para o olho do paciente.

Tendo feito isso o programa pergunta se a imagem da câmera será invertida ou não, pois uma vez acoplada ao boné a câmera ficará de ponta cabeça, caso o usuário não faça nada dentro de alguns instantes o programa carrega o valor padrão que é a câmera invertida.

Para evitar um consumo desnecessário de processamento, é solicitada uma ROI (*Region Of Interest*– Região de Interesse<sup>2</sup>) que é a região da imagem em que está o olho do paciente. A fim de que este ajuste não seja manual, isto é, não seja necessário que alguém tenha que selecionar manualmente a área desejada, o próprio programa reconhece a região dos olhos bastando que o usuário pisque algumas vezes. Este reconhecimento é feito através da diferença entre dois quadros [14] o que se demonstrou eficiente mesmo com o uso de óculos tendo reflexo como mostra a Figura 4.

No caso, cada vez que o usuário pisca o programa verifica se a região segmentada possui dimensões características de um olho a fim de se evitar ruídos. Quando o sistema identifica cinco ROI (após o usuário piscar algumas vezes) ele tira uma média dos valores para obter uma ROI definitiva. Uma vez definida a região ela recebe uma correção a partir de uma tolerância para que a ROI não fique rente ao contorno do olho. Isto se torna necessário para evitar que pequenos desajustes na orientação da câmera retire o olho do paciente de dentro da ROI. A Figura 5 ilustra uma ROI identificada dentro do frame representada pelo retângulo vermelho

Inicia-se então o processo de calibração para correlacionar a posição do olho com uma posição na tela do computador.

Pode-se assumir que a relação entre os movimentos dos olhos e a posição no monitor seja linear para as distâncias comumente utilizadas entre o usuário e a tela [14].

Sabendo-se disso, no canto superior esquerdo e logo após no canto inferior direito aparecerá um ícone por cinco segundos. Durante estes 5 segundos o usuário olha para o ícone e são registrados as posições da sua íris. É feita então uma média da posição para cada canto obtendo-se assim uma região que seja proporcional às posições na tela do computador.

### C. Extração dos Dados

Após a determinação da ROI dentro dos frames durante o processo de ajuste, é necessária a segmentação da íris do usuário tanto para a calibração, quanto para a utilização final do programa que é movimentar o mouse. Esta segmentação é realizada conforme Figura 6:

---

<sup>2</sup> ROI – Region of Interest: uma região dentro de uma imagem ou frame onde o processamento será realizado, toda a área fora desta região será ignorada.

Para que as funções aqui mencionadas possam ser aplicadas o quadro deve ser convertido para escala de cinza fazendo com que cada pixel da imagem seja apenas um valor representando uma intensidade variando de 0 a 255.

Aplica-se então a região de interesse obtida durante o processo de ajuste a fim de que o programa não gaste processamento em áreas fora da região de movimentação do olho.

Na Figura 7 é demonstrado o resultado da equalização do histograma para duas iluminações diferentes onde a área retangular destacada dentro da imagem é onde o histograma foi equalizado.

Neste contexto a função *Threshold* converte a imagem de valores que variam de 0 até 255 para valores binários (0-preto ou 255-branco). Para definir o que será considerado como preto e o que será considerado como branco é estabelecido um valor em que valores abaixo deste número são considerados como preto e acima como branco. Um exemplo da utilização desta função está na Figura 8.

Uma vez aplicada a função *Threshold*, utiliza-se em seguida as funções *Canny* que acaba por definir as bordas da imagem, e *Smooth* que suaviza os contornos. Estas duas últimas funções servem para preparar a imagem para a função *Hough Circles* que identifica os círculos na imagem.

Para identificar se um círculo é válido ou é um ruído, somente são selecionados os maiores que 10% da largura do quadro, e menores que 50% da largura do quadro pois entende-se que o olho estará próximo da câmera, por isso a utilização destas dimensões em relação ao frame.

Sendo um círculo válido, sua posição é então utilizada, veja Figura 9.

Para tornar o controle do mouse mais estável e suave, evitando trepidações devido à ruídos na posição obtida da íris, a posição repassada para o mouse é a média das últimas vinte posições obtidas da íris. Desta forma, mesmo que o olho do usuário trema, ou eventualmente focalize em pontos diversos da tela, ou simplesmente a segmentação e processamento da imagem cause algum tipo de ruído nas coordenadas obtidas do olho, por existir uma média dos últimos valores o mouse tende a permanecer próximo das últimas posições causando um movimento mais suave e fácil de controlar.

A etapa de segmentação do olho etapa de segmentação foi a mais trabalhosa por necessitar a elaboração de processos que tornam a sua utilização mais robusta sendo possível a sua utilização em ambientes com variações de luminosidade. Também demandou tempo a especificação dos ajustes das funções, principalmente a *Threshold* para que ela pudesse segmentar da melhor maneira a íris do usuário.

A configuração do sistema é otimizada, onde a única coisa que se deve fazer manualmente é executar o programa e posicionar a câmera corretamente apontando para o olho do paciente. Todas as demais configurações são realizadas automaticamente. Definições como qual câmera será utilizada ou se a imagem será invertida assumem valores padrão caso o usuário não se manifeste durante alguns segundos. A definição da ROI (Região de Interesse no quadro) também é identificada automaticamente, bastando o usuário piscar algumas vezes. Assim como a definição da

ROI, os extremos alcançados com a íris também são identificados de maneira simples, bastando o usuário olhar para dois pontos na tela.

A forma como o sistema é fixado no usuário também já pode ser definida pois a utilização do boné se mostrou eficiente e versátil, e as câmeras com cliques para fixação são muito comuns e estão entre os modelos mais baratos disponíveis atualmente.

A identificação da íris está eficaz, com poucos erros. E mesmo quando ocorre algum erro na identificação da posição da íris, como o sistema utiliza a média das últimas vinte posições, o erro é “dissolvido” nestes valores diminuindo seu impacto no funcionamento do sistema.

A princípio as coordenadas do mouse eram obtidas através da média das últimas dez posições identificadas da íris, mas ainda assim o mouse tremia muito, para solucionar este problema utilizou-se o dobro de coordenadas para gerar a média, totalizando vinte a quantidade de posições necessárias. Desta forma a velocidade da movimentação do mouse diminuiu, porém ganhou-se consideravelmente em precisão.

Com esta maior precisão foi possível utilizar este software em conjunto com o Teclado Virtual do Windows®. Este teclado faz parte do pacote do sistema operacional Windows® como acessório de acessibilidade e pode ser configurado para que suas teclas sejam ‘pressionadas’ bastando posicionar o mouse sobre elas por alguns segundos. Como neste teste as coordenadas da íris não foram tratadas proporcionalmente de forma a traduzir os pequenos movimentos do olho em movimentos proporcionais na tela, os movimentos do mouse foram tão pequenos quanto o movimento capturado do olho, porém já foi o suficiente para ‘pressionar’ algumas teclas no Teclado Virtual.

#### IV. TRABALHOS CORRELATOS E CONCLUSÕES

##### A. Trabalhos correlatos

Como apresentado na introdução, há outros trabalhos com objetivos parecido. Porém diferem em alguns aspectos básicos, que podem ser demonstrados através de três outros exemplos de sistemas similares como mostrado na Tabela 1.

O projeto HeadMouse® foi concebido pelo Grupo de Robótica da Universidade de Lleida na Espanha [10] que utiliza uma câmera simples de computador e através dos movimentos da cabeça e expressões faciais controla o mouse na tela.

Já o Iriscom® [7] e os equipamentos da Tobii® [8], são sistemas muito eficientes em controlar o mouse através apenas dos olhos, porém eles possuem a necessidade de equipamentos que vão além de uma simples câmera USB. Estes sistemas precisam de componentes como *leds* infravermelho ou até mesmo câmeras especiais de alta velocidade e toda uma aparelhagem às vezes com um computador dedicado somente a isso, o que torna estes projetos caros e inviáveis para uma popularização.

##### B. Atividades Futuras

O controle do mouse foi implementado com sucesso. Espera-se poder implementar a filtragem dos círculos válidos

através da análise dos pixels em seu interior, caso sejam olhos claros provavelmente não será uma íris. A resolução da tela do computador também deverá ser reajustada para valores menores a fim de que assim os botões, texto, ícones, etc. fiquem maiores e mais fáceis de serem escolhidos. Como uma próxima versão, pode-se criar um segundo software com alguns botões para facilitar a utilização e comunicação através do mouse para mensagens frequentes e importantes como por exemplo o índice de dor, humor e sim/não ilustrado na Figura 10.

#### REFERÊNCIAS

- [1] GALLO, A. M. - Atendimento Humanizado em Unidades de Urgência e Emergência - Revista F@pciência, Apucarana-PR, v.5, n. 1, p. 1 – 11, 2009.
- [2] PNHAH Programa Nacional de Humanização da Assistência Hospitalar, Ministério da Saúde - Secretaria de Assistência à Saúde, Série C. Projetos, Programas e Relatórios, n. 20 – Brasília 2001
- [3] Cranford, Ronald E., “The Persistent Vegetable State: The Medical Reality (Getting the Facts Straight)”, 1988, The Hastings Center Report, v.18, nº1, p30.
- [4] Secretaria da Saúde - <http://www.saude.sp.gov.br/ses/perfil/cidadao/homepage/acesso-rapido/pacientes-nao-identificados>. Acessado em 29 de maio de 2012.
- [5] i-Track da El-mar inc. - <http://elmar-inc.com/apps1TV.html>. Acessado em 29 de maio de 2012.
- [6] Gaze Analytic System Dikablis da Ergoneers - <http://www.ergoneers.com/en/products/dlab-dikablis/overview.html>. Acessado em 29 de maio de 2012.
- [7] Iriscom - <http://www.iriscom.org> . Acessado em 29 de maio de 2012.
- [8] Produtos Tobii – <http://www.tobii.com/en/assistive-technology/global/products>. Acessado em 29 de maio de 2012.
- [9] Expresso Portugal – matéria: “Vai ser possível controlar um portátil só com os olhos”, publicada dia 02 de março de 2011. Disponível em <http://aeiou.expresso.pt/vai-ser-possivel-controlar-um-portatil-so-com-os-olhos=f635372>. Acessado em 29 de maio de 2012.
- [10] Grupo de Robótica da Escola Politecnica Superior da Universidade de Lleida - <http://robotica.udl.cat>. Acessado em 29 de maio de 2012.
- [11] R. C. Gonzales and R.E. Woods. “Digital Image Processing”. Addison-Wesley Pub. Company, 3nd, pg. 976, 2007.
- [12] OpenCV Open Source Computer Vision - <http://opencv.willowgarage.com/wiki/>. Acessado em 29 de maio de 2012.
- [13] Michael Chau, Margrit Betke, “Real Time Eye Tracking and Blink Detection with USB Cameras”, Boston University - USA
- [14] Hammal Z., Massot C., Bedoya G., Caplier A., “Eyes Segmentation Applied to Gaze Direction and Vigilance Estimation”, Laboratory of Images and Signals, Grenoble, France.

Tabela 1 - Comparativo de vantagens entre o Mouse Ocular e outros sistemas similares

	<i>Baixo custo</i>	<i>Utiliza somente o olho como fonte de informação</i>	<i>Utiliza equipamentos simples</i>
<b>Mouse Ocular</b>	✓	✓	✓
<b>HeadMouse®</b>	✓	✗	✓
<b>Iriscom®</b>	✗	✓	✗
<b>Tobii®</b>	✗	✓	✗



Figure 1. Modelo de câmera utilizada.

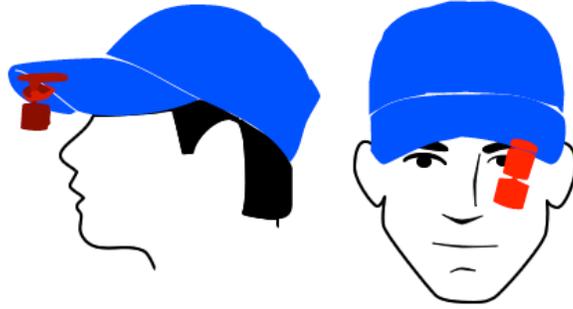


Figure 2. Acoplamento da câmera ao boné.

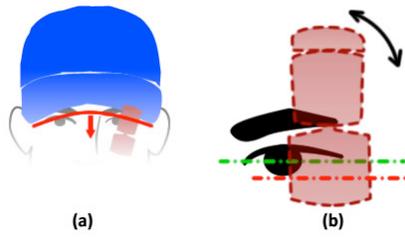


Figure 3. Alinhamento da câmera com o olho.

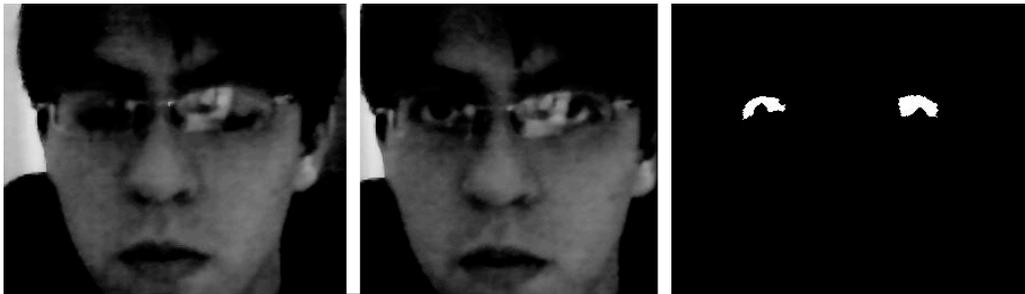


Figure 4. Diferença entre quadros para identificação da região dos olhos.

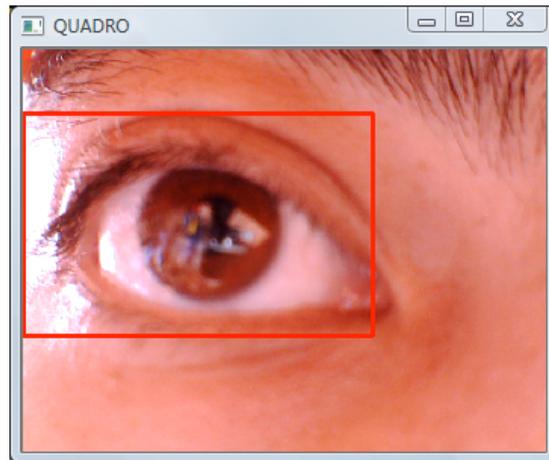


Figure 5. ROI selecionada dentro do frame.

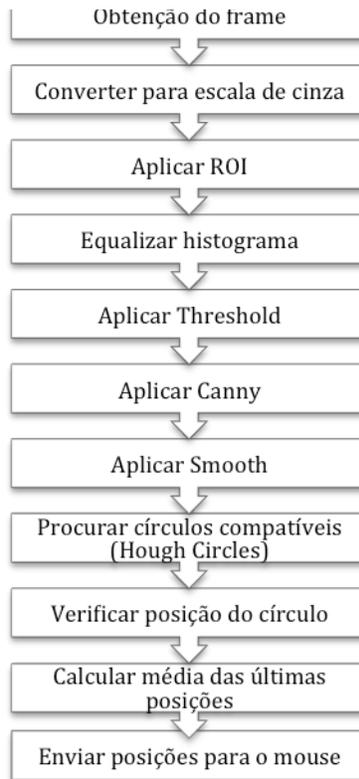


Figure 6. Sequência de etapas para a segmentação da íris.

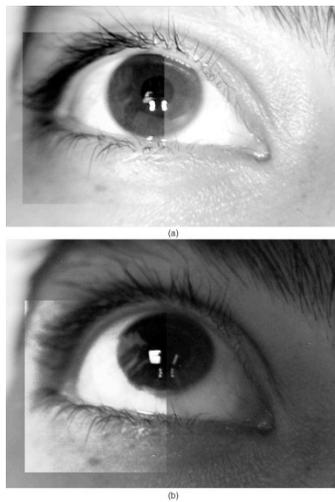


Figure 7. Demonstração do resultado da Equalização do Histograma. Na figura superior com muita iluminação e na figura abaixo com pouca iluminação.

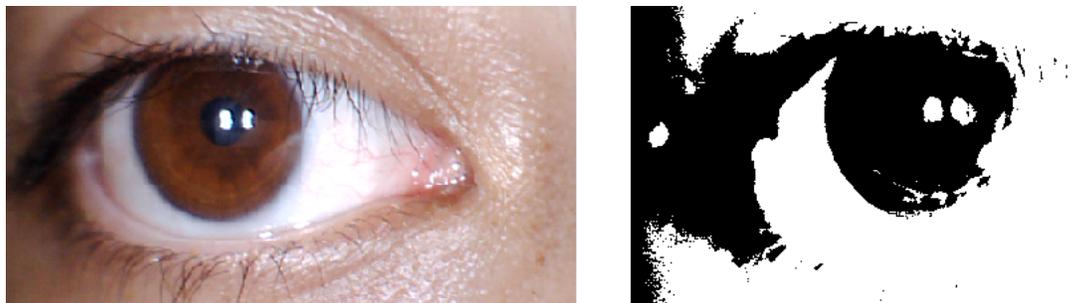


Figure 8. Resultado da função *Threshold*.

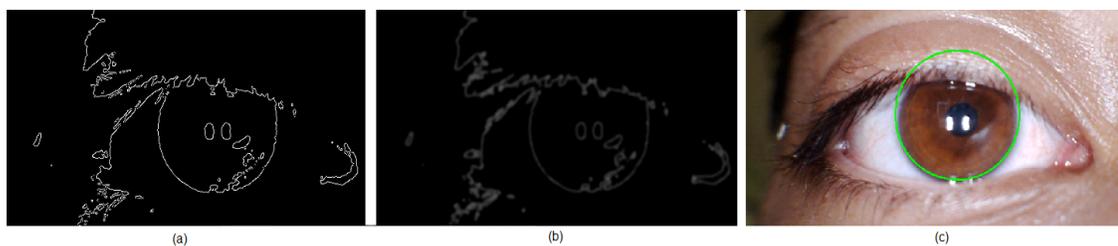


Figure 9. a) resultado da função Canny b) resultado da função Smooth c) imagem original com o círculo identificado em verde através da função Hough Circles.



Figure 10. Exemplos de botões a serem disponibilizados para o usuário.