

Um Sistema de Visão Computacional para Comando de Orientação

Jose Roberto A. Junior¹, Paulo S. Do Carmo¹, Wanderson S. Da Silva¹,
José Lucas C. Magalhães¹, Prof. Dr. Wener B. Sampaio², Prof. Dr. Anselmo C. Paiva³

¹Bacharelado em Ciência e Tecnologia – Universidade Federal do Maranhão (UFMA)

²Engenharia da Computação – Universidade Federal do Maranhão (UFMA)

³Núcleo de Computação Aplicada – Universidade Federal do Maranhão (UFMA)

robertoaranhajr@gmail.com, paulonayron@hotmail.com, wankapef.silva@gmail.com,

lucasmagalhaesjc@gmail.com, wener.sampaio@ufma.br, paiva@deinf.ufma.br

Abstract. *This article describes an algorithm that captures commands of eye and face movements in computational query and Digital Image Processing applied to the orientation of an intelligent wheelchair. The system acts when the user's face is completely facing a camera. In this way, it is considered that it is concentrated in the operation. To validate the program, a shipment experience is performed.*

Resumo. *Este artigo descreve um algoritmo que captura comandos dos movimentos dos olhos e face baseado em visão computacional e Processamento Digital de Imagens, aplicado à orientação de uma cadeira de rodas inteligente. O sistema age quando o usuário está com o rosto completamente virado para a câmera. Dessa maneira, considera-se que o mesmo está concentrado na operação. Para validação do programa é realizada uma experimentação em-barcada.*

1. Introdução

O avanço tecnológico deve ter como objetivo a melhoria da qualidade de vida daqueles que o utilizam. Tendo como público alvo portadores de tetraplegia e idosos, o projeto consiste em desenvolver um sistema de baixo custo adaptável às cadeiras de rodas simples, guiadas por comandos faciais do usuário. Dessa forma, busca-se propiciar uma estrutura acessível que gera autonomia para deficientes em atividades rotineiras.

O objetivo desse artigo é apresentar um sistema responsável por capturar os gestos faciais e transformá-los em comandos de início/término de ação e direção. Para fins experimentais, é utilizado um protótipo, onde é possível simular na prática os resultados das decisões tomadas.

Na seção 2, são apresentados de forma teórica o algoritmo de Viola-Jones e o Filtro de Kalman. Na Seção 3, é descrito o desenvolvimento do sistema. Na Seção 4, é apresentada as escolhas para as interpretações dos comandos. Por fim, na Seção 5, são analisados os resultados obtidos e a confiança.

2. Referencial Teórico

2.1. Viola-Jones para Captura de Face

O algoritmo desenvolvido por [Viola Jones 2001] é um dos mais populares algoritmos para detecção de padrões de face. O mesmo apresenta um baixo custo computacional, aliado a uma boa velocidade de resposta, tornando-o extremamente eficiente.

O método consiste em duas partes principais: os classificadores em cascata, responsáveis pela velocidade do processamento e um treinador de classificadores (de nome *Adaboost*), que combina classificadores fracos, com o objetivo de obter um forte.

A unidade básica do algoritmo são os chamados *features* retangulares, que possuem quatro padrões específicos, conforme visto na imagem abaixo:

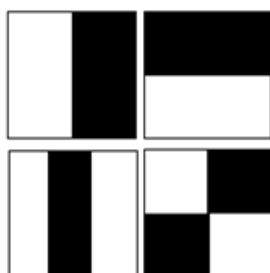


Figura 1. Features

Assumiremos agora que “*w*” seja uma janela que contém os *features*. Assim podemos calcular o valor de cada um deles da seguinte forma:

$$f(w) = \sum^w P_{preto} - \sum^w P_{branco}$$

Onde:

$$\begin{aligned} f(w): & \text{ valor do feature na janela } w; \\ \sum^w P_{preto}: & \text{ somatório dos pixels na região preta;} \\ \sum^w P_{branco}: & \text{ somatório dos pixels na região branca.} \end{aligned}$$

Além disso, para acelerar ainda mais o processo, utiliza-se a técnica de imagem integral. Este método consiste em pegar uma imagem $N \times M$, e somar, a cada pixel dela, o valor de cada pixel a esquerda e acima do pixel dado. Dessa forma, executam-se cálculos com maior liberdade, em subregiões desejadas.

A popularidade do mesmo é consequência da forma de detecção. Os *features*, inicialmente, percorrem toda a imagem e verificam quais partes dela são úteis aos classificadores. Como visto na figura acima, tais objetos podem ter formas diferentes.

Desse modo, é possível medir a diferença de intensidade entre a região dos olhos e as bochechas (ou mesmo a região da testa), pois geralmente a região dos olhos apresenta tons mais escuros que as demais partes do rosto; e a região dos olhos com a ponta do nariz. Com a junção de todas essas características, responsáveis pela “alimentação” dos classificadores, montam-se as cascatas, que identificam a região da face como um todo. Trabalhar dessa maneira, em vez de utilizar as informações contidas em cada pixel individualmente, faz com que o processo seja acelerado, além de reduzir o esforço da máquina.

2.2. Filtro de Kalman

Para estimar os pontos adquiridos, pelo processo de detecção de faces de Viola Jones com maior fidelidade a realidade, foi utilizada a técnica de suavização através da aplicação do método desenvolvido por Kalman (1960), que consiste em utilizar medidas de grandezas realizadas ao longo do tempo, contaminadas com ruídos e incertezas e produzir resultados mais próximos da realidade. O algoritmo discreto de filtro de Kalman é um conjunto de equações matemáticas que, de maneira recursiva, consegue estimar de forma eficiente estados passados, presentes e até mesmo futuros. [Welch e Bishop 1997].

Esse filtro é importante para a detecção do centro dos olhos de maneira mais suave, uma vez que essa região é extremamente sensível à variações.

3. Sistema de Reconhecimento

O método aqui desenvolvido se utiliza da biblioteca OpenCV, referência nas áreas de visão computacional, desenvolvida inicialmente pela Intel (2000) e atualmente de código aberto sob licença BSD.

A captura da imagem é feita através de uma *webcam*, pois trata-se de um artefato de baixo-custo, disponível para a maior parte do público alvo. A mesma deve ser mantida fixa através de um suporte, de modo que, fique sempre de frente pro usuário, para melhor captura da imagem de entrada.

Inicialmente, utiliza-se o algoritmo de Viola-Jones, responsável pela detecção do rosto do usuário a partir da imagem capturada pela câmera e delimitação da ROI (*Region of Interest*) na forma de um quadrado que delimita toda a região correspondente ao mesmo, que será utilizada nas etapas subsequentes. [Gonzalez 2008]

Com a ROI encontrada, elimina-se grande parte dos ruídos indesejáveis e há garantia de que a etapa de detecção dos olhos será executada apenas onde é possível encontrar os mesmos. Essa estratégia busca eficiência no processamento, a fim de evitar falsos-positivos, visto que o processo de detecção de olhos adotado é mais robusto e necessita de uma quantidade maior de parâmetros, uma vez que são analisados detalhes mais sutis.

O primeiro passo é converter a imagem segmentada para escala de cinza, uma vez que há olhos de cores variadas. A utilização da imagem em dois canais, diminui a possibilidade de erro com relação a esse tipo de variável. Após isso, e de maneira análoga ao que foi feito no rosto, aplica-se novamente o *Adaboost* para obtenção dos classificadores fortes, que conforme a intensidade dos pixels e na posição relativa ao rosto, fazem a identificação dos olhos. Um retângulo é traçado na região encontrada, para cada olho, individualmente. De acordo com as características principais do mesmo (topo, largura e altura) estima-se a região da pupila, utilizada como referência para desenhar mais dois retângulos menores e dinâmicos. De forma recursiva é tomada a largura total da ROI menos a largura até o centro do olho para obter o quadrado à direita. O quadrado à esquerda é tomado como a distância da origem até o centro do olho. Tal procedimento é análogo aos dois olhos. Como feito por [Asi 2014].

Por último, basta alterar o sistema de coordenadas das pupilas para as coordenadas faciais e traçar círculos nas regiões encontradas.

Devido importância que foi atribuída ao movimento da pupila, observou-se a necessidade de uma suavização do mesmo. Dessa maneira, foi lançado mão do Filtro de Kalman, implementado através da classe “*KalmanFilter*” da biblioteca OpenCV. Na etapa de inicialização declara-se ao construtor desta classe a utilização de uma matriz de estado

4x4 (além das coordenadas dos pontos iremos considerar suas velocidades em x e y), o vetor de medidas contendo 2 variáveis (coordenadas x e y), além de indicar que não temos parâmetros de controle para estas variáveis, ou seja, o estado da variável não se modifica ao longo do tempo de forma bem definida. Após a construção do objeto desta classe, podemos definir uma matriz de transição adequada (A) e definir o estado previsto (X_k), inicialmente igual as coordenadas medidas, ao final desta etapa inicializam-se as matrizes identidade escalonadas, da matriz de observação (H), matriz de covariância de ruído do processo (Q), matriz de covariância de ruído da medição (R) e a matriz de covariância de erro estimado.

Na segunda etapa, em cada iteração é calculado o estado atual do sistema (X_{est}), efetuado nova aquisição de medidas (X_{med}) e calculado o estado estimado do sistema (X_k), ao final as coordenadas estimadas são extraídas e utilizadas para interpretação dos comandos e uma nova iteração é realizada.

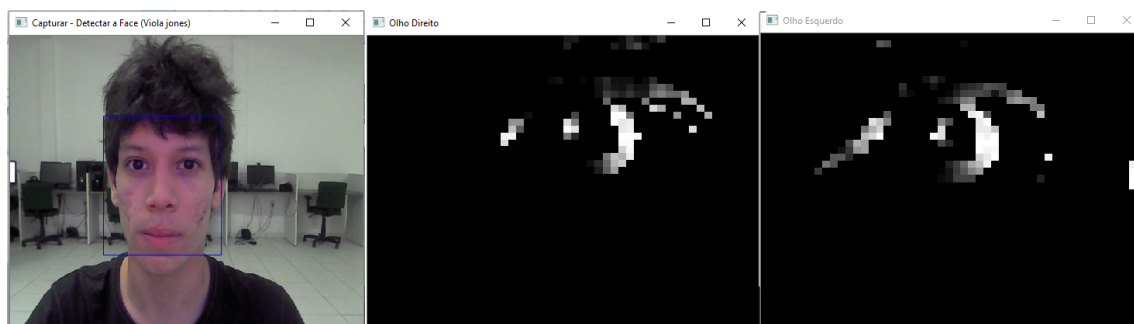


Figura 2. Detecção da face e cantos dos olhos

4. Comandos para Operar a Cadeira

Para os comandos relacionados aos sentidos frente/trás foi adotado o critério de proximidade dos centro dos olhos em relação as linhas horizontais inferior e superior subsequentes.

Para o sistema de interpretação da decisão do Controle de Direção, os parâmetros de comandos esquerda/direita da cadeira, foi escolhido que o rosto permaneça em direção à câmera, pois é considerado que o movimento de rotação do rosto possa ter um caráter involuntário ou que não corresponda necessariamente ao real desejo do usuário de tomar determinada orientação. Em teste primário, foi possível concluir que a associação do rosto estático com apenas o movimento dos olhos, cria um padrão confortável e que evita erros simples de interpretação, já que é mais fácil de estimar se o usuário está concentrado na operação.

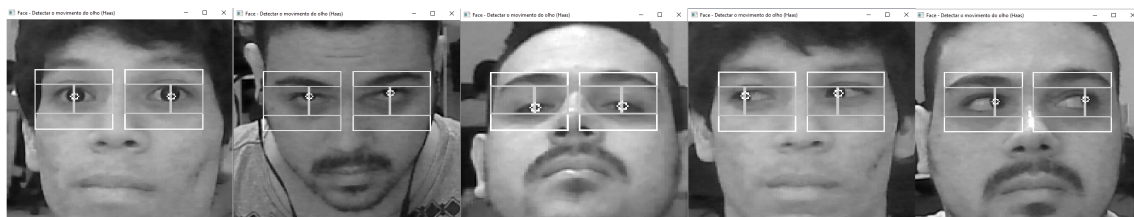


Figura 3. Mecanismo para controle de direção

5. Resultados Experimentais

5.1. Verificação de Resposta

Os comandos capturados foram testados via display, onde foram impressas as interpretações de orientação do sistema. Nessa experimentação cada comando (esquerda, direita, frente e trás) foi testado continuamente durante 30 segundos, a fim de obter a quantidade de erros - três comandos em sequência diferentes do desejado - nesse intervalo de tempo.

Os resultados obtidos em tais testes estão dispostos nas tabelas 1 e 2 apresentadas abaixo.

Tabela 1. Teste: esquerda/direita

TESTE:	POSITIVO	NEGATIVO	TOTAL
Esquerda	28 (a)	2 (b)	30
Direita	26 (c)	4 (d)	30

Sensibilidade = 0,87

Especificidade = 0,93

Acurácia = 0,90

Tabela 2. Teste: frente/trás

TESTE:	POSITIVO	NEGATIVO	TOTAL
Frente	27 (a)	3 (b)	30
Trás	23 (c)	7 (d)	30

Sensibilidade = 0,79

Especificidade = 0,88

Acurácia = 0,83

Além disso, foram realizadas medições individuais para cada comando, com o objetivo de obter o tempo de resposta, ou seja, o tempo de comunicação entre software e hardware. Os resultados de média e desvio padrão obtidos nesse teste estão descritos para direita: $(2,80 \pm 0,65)s$; para esquerda: $(1,76 \pm 0,19)s$; comando frente: $(1,73 \pm 0,38)s$; e para trás: $(1,82 \pm 0,13)s$.

Dessa maneira, é possível considerar que se obteve um bom resultado na interpretação dos comandos, apesar da necessidade de melhorias da interpretação do comando para trás.

5.2. Experimento Embarcado

Para realizar o teste embarcado são utilizados dois sistemas.

O primeiro trata-se do “Transmissor”, responsável por receber os comandos de orientação capturados pelo Programa de Reconhecimento - visto nos capítulos anteriores - e enviá-las para o segundo; o “Receptor-Atuador”, responsável por traduzir as instruções recebidas, acionando os atuadores, que são dispositivos que produzem movimento atendendo a comandos que podem ser manuais, elétricos ou mecânicos.

O Sistema Transmissor é composto por um Arduino Nano e um Módulo RF Transmissor. O Sistema Receptor-Atuador é composto por 1 Arduino Uno, 1 Módulo RF Receptor, 1 Shield L293D, 2 motores DC e duas Rodas Motriz.

6. Conclusão

A escolha pelo Método desenvolvido por Viola-Jones, ajudou para que, naturalmente, o programa pare quando o rosto não estiver voltado à câmera. O Filtro de Kalman, utilizado para suavização da posição da pupila, ajudou com que o movimento se tornasse mais

fluido, mas teve um aumento significativo no custo computacional. Observou-se, através dos experimentos, ainda ser necessário um aprimoramento dos limiares de interpretação dos comandos gerados pelo olhar do usuário.

7. Trabalhos Futuros

Desenvolvimento do sistema embarcado fazendo uso de um hoverboard - equipamento que consiste em um conjunto de motores elétricos ligados a um par rodas guiadas por giroscópios e unidas através de um eixo - adaptado à cadeiras de rodas comuns através de um sistema mecânico que envolve uma estrutura simples; bem como, o aprimoramento do sistema já desenvolvido, através de novos métodos de captura de movimentos.

8. Agradecimentos

À Fundação de Amparo à Pesquisa e Desenvolvimento Científico do Maranhão (FAPEMA) pelo apoio financeiro, que tem sido de grande importância para que este projeto possa lograr êxito.

Referências

- Viola, P. Jones, M. (2001). Robust Real-Time Face Detection. International Journal of Computer Vision 57, pages 137 - 154.
- Gonzalez, R. and Woods, R. (2008). Digital Image Processing. Pearson, 3th edition.
- OpenCV: Open Source Computer Vision. Acesso em Junho de 2017. Disponível em: <http://opencv.org>.
- Asi et al. (2014). Eye Corners Detection Using Haar Cascade Classifiers in Controlled Environment. Journal of Engineering and Technology, pages 69 – 75.
- Welch, G. and Bishop, G. (1997). An Introduction to the Kalman Filter. University of North Carolina at Chapel Hill.
- Taylor PB, Nguyen HT. (2003). Performance of a Head-Movement Interface for Wheel-chair Control. Proceedings of the 25th Annual International Conference of the IEEE EMBS, pages 17 – 21.
- Miranda, J. (2009). Sistema de Visão para Controlo de Cadeira de Rodas Inteligente. Universidade do Porto.