

“Weeping Angel”: Biblioteca cross plataforma de baixo custo para detecção de atenção em dispositivos móveis**“Weeping Angel”: Low-cost cross platform library for detecting attention on mobile devices**

DOI:10.34117/bjdv6n4-236

Recebimento dos originais: 10/03/2020

Aceitação para publicação: 16/04/2020

Alano Martins Pinto

Mestrando em Ciências da Computação pela Universidade Estadual do Ceará

Instituição: Instituto Atlântico

Endereço: Avenida Washington Soares, 909 - Loja 42 43 44 45 - Edson Queiroz, Ceará – CE,
Brasil

E-mail: alano_pinto@atlantico.com.br

Yuri Lenon Barbosa Nogueira

Doutor em Ciências da Computação pela Universidade Federal do Ceará

Instituição: Universidade Federal do Ceará

Endereço: Departamento de Computação - UFC, Campus do Pici - Centro de Ciências - Bloco 910,
CEP 60440-900 Fortaleza - CE

E-mail: yurilenon@gmail.com

RESUMO

Objetivo: Propor e demonstrar um sistema que utiliza imagens de câmera de um dispositivo móvel capaz de identificar e analisar a atenção do usuário sem a necessidade da análise do conteúdo exibido. Método: O projeto utiliza algoritmos consolidados de visão computacional para reconhecer padrões geométricos, detecção facial e filtros de imagem para, em seguida, extrair dados de piscar dos olhos a serem analisados. Resultados: Os dados de execução do algoritmo demonstra ser capaz de rodar em quase todos os dispositivos móveis de hoje, no entanto, a execução em tempo real (30 frames por segundos) só foi possível em desktop ou dispositivos móveis comparáveis. Conclusões: A detecção de atenção tem aplicabilidade em muitos ambientes práticos. O algoritmo proposto visa inferir dados para aplicativos de terceiros, podendo ser usado como uma ajuda à tomada de decisão, encapsulando uma análise complexa em uma interface simples e intuitiva.

Palavras-chave: OpenCV, Visão Computacional. Processamento de Imagens. Fadiga. Rastreamento de olhar.

ABSTRACT

Objective: To propose and demonstrate a system that uses camera images from a mobile device capable of identifying and analyzing the user's attention without the need to analyze the displayed content. Method: The project uses consolidated computer vision algorithms to recognize geometric patterns, facial detection and image filters to then extract blink data to be analyzed. Results: The execution data of the algorithm demonstrates being able to run on almost all mobile devices today, however, real-time execution (30 frames per second) was only possible on desktop or comparable mobile devices. Conclusions: Attention detection has applicability in many practical environments. The proposed algorithm aims to infer data for third party applications, which can be used as an aid to decision making, encapsulating a complex analysis in a simple and intuitive interface.

Keywords: OpenCV, Computer Vision. Image processing. Fatigue. Look tracking.

1 INTRODUÇÃO

Em 2015, a ABS – Associação Brasileira do Sono, indicou o cansaço e a sonolência como segunda maior causa de acidentes no trânsito, sendo o motivo de 30% de acidentes de trânsito fatais. Já nos Estados Unidos, os acidentes automobilísticos são considerados a terceira principal causa de morte e feridos, tendo entre 40 e 50 mil vítimas (ROSEN, 2005). Assim como o álcool, a sonolência e a fadiga agem em funções básicas do corpo, como redução de atenção, coordenação motora, auditiva e visual (ROSEN, 2005). Segundo Rosen (2005), dirigir sob privação de sono deveria ser considerado crime; a autora fundamenta sua pesquisa a partir de respostas cognitivas dadas nessas condições relacionadas a acidentes de trânsito nos EUA.

A Polícia Rodoviária Federal brasileira realiza campanhas de saúde direcionadas para motoristas profissionais, já que estes são considerados os principais envolvidos em acidentes causados por dormir ao volante. Para Antunes et al. (2008), a privação de sono está diretamente ligada à diminuição do desempenho físico. Criando um mecanismo de mascaramento dos efeitos dessa privação, o organismo aumenta a percepção subjetiva (ANTUNES et al., 2008), causando mudança da concentração e riscos durante o exercício de atividades que requerem foco constante.

Em muitas situações, o estado de sonolência pode ser identificado através da simples observação de outro indivíduo. A partir disso, nosso projeto visa a criação de um algoritmo que analisa imagens da câmera de um dispositivo móvel de uso popular para inferir o nível de sonolência e atenção em tempo real, servindo como um meio de observação individual constante e consequente preservação da vida de muitos motoristas. Com a disseminação do uso de dispositivos móveis, não há necessidade de adquirir qualquer equipamento especializado para monitorar um motorista profissional no atributo de sonolência além de um celular de uso pessoal. Isso facilita a adesão a uma possível monitoria.

O processo de mensurar a atenção do usuário que utiliza passos de análise do conteúdo exibido é visto como uma intrusão à privacidade e, atualmente, proibido para publicação de acesso público. Muitos dispositivos móveis possuem programas que detectam faces para controlar a rotação da tela, mas eles não definem o grau de atenção. Apesar de muitas pesquisas serem realizadas nessa área, não existe uma aplicação de referência ou largamente utilizada para isso.

Estudos sobre fenômenos neurológicos dos olhos (LEE; ZEIGH, 1991) mostram a sua ligação com a variação de alerta do indivíduo, dessa forma é proposta a implementação desses estudos através de algoritmos de visão computacional e processamento de imagens. Assim, nossa pesquisa objetivou implementar uma biblioteca de reconhecimento de atenção de usuários para dispositivos como celulares, *tablets* e *notebooks*, de forma não intrusiva e sem interação com o usuário. Utilizando-se de câmeras providas por esses dispositivos, o projeto visou analisar imagens com a finalidade de

coletar e inferir informações através de comportamentos dos olhos sem interferir na usabilidade e privacidade.

2 METODOLOGIA

A implementação dessa pesquisa resulta em uma criação de uma biblioteca genérica para diversas plataformas. Essa biblioteca utiliza algoritmos de visão computacional e processamento de imagens e deve ser executada em dispositivos móveis. Para contornar as limitações de hardware desses dispositivos, foi escolhida a diminuição na taxa de acerto a fim de não consumir memória, bateria e processamento.

A análise da imagem inicia com uma conversão do quadro original para outro com apenas um canal de cor (tonalidade de cinza) com o intuito de utilizar o algoritmo de Viola e Jones (2003) para detecção de face em tempo real, reduzindo a área de busca da região da pupila. Para detecção da íris foi utilizado o algoritmo de Timm (2012). Usando cálculo de gradiente, o autor descreve um algoritmo que determina o centro da pupila e realiza um rastreamento com uma performance superior a outros trabalhos, demonstrado na sua publicação.

Após a detecção e rastreamento da íris, o evento de piscada pôde ser identificado através da implementação de um algoritmo de *flood* vertical para contagem de pixels. Assim, foi entendido como uma piscada quando a íris não pôde ser identificada e uma detecção das duas pálpebras fechadas formaram uma linha ou semelhante. Através da identificação de piscada foi possível determinar o tempo e frequência entre o evento de abertura e fechamento das pálpebras, resultando nos indicadores PERCLES e AECS. Após as análises desses indicadores, a biblioteca foi capaz de inferir o nível de atenção.

Todos os algoritmos descritos anteriormente estão implementados na biblioteca do OpenCV, com exceção da proposição de Timm (2012). Dessa maneira, a versão em C++ do OpenCV foi utilizada para a implementação, com o objetivo de ser facilmente portátil para qualquer ambiente móvel.

3 RESULTADOS E DISCUSSÃO

Em nosso trabalho, focamos nos seguintes indicadores relativos ao movimento das pálpebras: Average Eye Closure/Open Speed (AECS) e Percentage of Eye Closure (PERCLOS) (FABIAN; YANG, 2010). Experimentos realizados por JI e YANG (2001) demonstram uma média desses indicadores, onde valores de PERCLOS abaixo de 30% é considerado estado de atenção máxima e acima de 40% um alto grau de sonolência ou fadiga (JI; YANG, 2002). Já AECS no intervalo entre 150.000 ms e 200.000 ms demonstra o crescimento da fadiga, sendo no valor superior e acima dado

como sonolência e abaixo como atenção máxima (JI; YANG, 2002). Nosso projeto visou relacionar esses dois indicadores através da captura e mensuração de piscadas.

Durante a execução de tarefas no experimento com efeito Stroop (interferência no tempo de reação para uma dada tarefa) por Oh et al. (2012) houve uma média de piscadas de 30.7 ps/min (piscadas por minuto) com uma faixa de erro de 13,0 ps/min para mais ou menos e, no estado de repouso, 20,7 ps/min, com uma faixa de erro de 11,2 ps/min. Foi admitido que o estado de olhar fixo é manter-se em um ponto fixo por 2 minutos; nesse estado, o indivíduo, apesar de direcionado para a tela, não possui nenhuma atenção de fato.

A frequência de piscada (ERB) no pior caso é em torno de 7,6 ps/min. Porém, para um ganho de performance e otimização de recursos, o projeto adotará a frequência sugerida por Han et al. (2012) de 13,5 ps/min. Com esse resultado, pode-se inferir uma piscada a cada 4,44 s. Considerando uma margem de erro referente a uma piscada, o intervalo passa para 8,88 s, utilizando-se, portanto, de uma aproximação de 10s para um intervalo onde deve haver uma piscada. Para captura do AECS em um máximo de 50.000 ms, precisamos de 8 fps, daremos uma margem de 50% para garantir uma frequência mínima de quadros em 12 fps.

A implementação padrão e instrumentação foram realizadas em um Macbook Air, com processador Intel i5, 4GB de memória RAM, utilizando a câmera integrada Facetime 480p, para obtenção das imagens analisadas. A instrumentação da biblioteca foi feita usando o módulo *Instrumentation* do Xcode.

A linguagem de programação de implementação é o C++, com as bibliotecas do OpenCV 2.4.9; para realizar a pré-compilação a ferramenta CMake foi escolhida, esta permite através de um script personalizado, gerar projetos que são compilados para os formatos .dylib, .so, .dll e assim consumidos nas seguintes plataformas: MACOS, Android e Windows. Para o ambiente Android foi usado a tecnologia JNI (Java Native Interface) para conectar os arquivos fontes em C++ a aplicação que os consomem. Durante a instrumentação foram gravados vídeos a fim de manter a coesão entre todos os testes, porém a biblioteca também possui uma implementação em tempo real para dispositivos de maior poder de processamento, mas em dispositivos móveis medianos (dual core e 1 gb de RAM) esse comportamento sofre atraso de processamento. Isso diminui a taxa de quadros por segundos para abaixo do limite de 12 fps, definido anteriormente. Nessa situação não se deve usar tempo real, mas um conjunto de quadros previamente armazenados, o que garante que todos sejam avaliados.

Para uma avaliação coesa, o projeto foi instrumentado com os seguintes objetivos: uso de memória e processamento. Esses são dados generalistas que podem ser expandidos independente de plataforma ou hardware utilizado, já que a memória é calculada através da quantidade utilizada e o

processamento por quantidade de acessos ao processador, assim a frequência de acesso ao *clock* e quantidades de núcleos influenciam na performance.

Uma especificação essencial do projeto é a necessidade de execução em um dispositivo móvel com configurações moderadas. Logo, o desempenho pode ser um gargalo para viabilidade, pois a manutenção de uma taxa relativamente alta de quadros é necessária segundo estudos já descritos. Visando isso, os algoritmos foram escolhidos levando em consideração o custo de processamento e memória em detrimento à taxa de acerto. Baseando-se nisso, o classificador usado no algoritmo de Viola-Jones foi o “*lpdcascade_frontalface.xml*”, que possui cerca de 20% menos de acerto que o “*haarcascade_frotalface.xml*”. Porém, como o dispositivo fica próximo ao rosto e é utilizado por apenas um usuário por vez, a taxa de erro é desprezível e esses quadros são ignorados pelo algoritmo.

Uso de memória

Segundo a análise de alocação de memória agrupado por hierarquia de execução, pudemos observar dois trechos de maior uso de recursos de memória, cerca de 75% do uso total da aplicação. Esses trechos são internos da biblioteca do OpenCV, utilizados nos algoritmos de conversão de canais de cores da imagem (*cvtColor*) e na identificação de faces (*CascadeClassifier::detectMultiScale*). A aplicação mantém uso constante de memória, não identificando vazamento de alocação. É observado um aumento de recursos para a consolidação de dados coletados durante a execução. Ao término do vídeo, o total de memória consumida foi de 62,95 MB, dentre os quais 47,61 MB são referentes ao consumo citado acima, 8,17 MB (12,9%) para armazenagem temporária do vídeo e 7,07MB (11,1%) para leitura do vídeo utilizado pela aplicação de referência, resultando em cerca de 100 KB para os outros algoritmos usados dentro da biblioteca.

Uso da CPU

O uso de CPU foi calculado por um contador de ciclos do processador, sendo esses divididos entre cada núcleo nas seguintes proporções, 35%, 26,9%, 24,1%, 13,8%. Durante a execução, pode haver diminuição de processamento, pois existem rotinas na biblioteca que evitam cálculos desnecessários. Por exemplo, ao detectar que os olhos estão fechados, não existe necessidade de realizar o rastreamento das pupilas. Com isso, observa-se eventuais quedas no uso do processador. É observado um consumo mais intenso no início da execução para inicializar estruturas básicas para os algoritmos.

Taxa de acerto

O processo mais crítico para a taxa de acerto da biblioteca foi referente à identificação de piscada, pois esse evento define os parâmetros AECS e PERCLOS, usados para o cálculo do resultado da atenção. Com isso, para demonstrar a taxa de acerto da biblioteca, foram criados testes unitários que recebem como entrada a quantidade de piscadas previamente observadas no vídeo analisado. Após a análise de todos os quadros do vídeo, o teste unitário comparou a quantidade de piscadas com o parâmetro recebido. Mesmo com as quantidades de piscadas calculadas, ainda foi necessária uma avaliação manual de cada vídeo durante a detecção, mostrando o exato momento do vídeo em que houve uma detecção de piscada, pois deve ser avaliado a ocorrência de falsos positivos (detecção falsa) durante o teste. A partir daí, foram gravados 5 vídeos para realizar tais testes; cada um teve os seguintes resultados:

	Piscadas reais	Piscadas detectadas	Falso positivos
Vídeo 1	6	4	0
Vídeo 2	8	7	2
Vídeo 3	7	5	1
Vídeo 4	3	3	0
Vídeo 5	10	7	3

De acordo com os testes realizados, é calculado uma média de 78,8% de acerto e 22,95% de falsos positivos.

4 CONCLUSÃO

Utilizando tal solução em dispositivos móveis, é possível analisar o estado de sonolência em motoristas que dirigem por longos percursos, cabendo à aplicação que a utiliza gerar uma ação que previna acidentes a partir dessa informação, sendo possível notificar o motorista ou responsável. Toda a solução não necessita de componentes específicos, como câmera estéreo, infravermelho ou de alta performance – que aumentariam o custo –, mas apenas um celular comum fixado em direção ao motorista durante seu trajeto. Portanto, o impacto social desse estudo seria tornar a referida tecnologia acessível para muitos motoristas que ainda não possuem esse tipo de detector.

Assim, este trabalho apresentou uma técnica eficiente para a detecção de atenção de usuários a partir da captura de imagens por uma câmera. O projeto possui baixas exigências de hardware, permitindo sua aplicação em dispositivos móveis que possuem poder computacional relativamente baixo. Futuros testes devem envolver dispositivos móveis reais para garantir as funcionalidades demonstradas e realizar novos testes relativos à taxa de acerto, usando vídeos gravados por esses dispositivos. Além disso, a quantidade de testes deverá ser aumentada para garantir uma maior precisão sobre a taxa de acerto.

Estudos futuros podem testar algoritmos para outros fatores relevantes na identificação de atenção. Entre eles, pode-se citar o reconhecimento de expressões faciais, como o *Active Appearance Model* (AMM), para inferir nível de fadiga ou movimentos da cabeça para considerar momentos de distração ou fuga do foco. A pesquisa pode ser ainda estendida para análise do ambiente inserido, visando observar e aprender ocasiões externas onde a atenção foi diminuída, como sons e imagens invasivas.

REFERÊNCIAS

ANTUNES, Hanna Karen M. et al . Privação de sono e exercício físico. Rev Bras Med Esporte, Niterói, v. 14, n. 1, p. 51-56, Feb. 2008. Disponível em <http://www.scielo.br/scielo.php?script=sci_arttext&pid=S1517-86922008000100010&lng=en&nrm=iso>. Acessado em 16 de outubro de 2016. <http://dx.doi.org/10.1590/S1517-86922008000100010>.

FABIAN, F.; YANG, B., Camera- based Drowsiness Reference for Driver State Classification under Real Driving Conditions, 2010 IEEE Intelligent Vehicles Symposium, Junho, 22, 2010.

TIMM, F.; BARTH, E., Accurate eye center localization by means of gradients, Institute for Neuro- and Bioinformatics, University of Lubeck, Ratzeburger Allee 160, D-23538 Lubeck, Germany " Pattern Recognition Company GmbH, Innovations Campus Lubeck, Maria-Goeppert-Strasse 1, D-23562 L

JI, Q.; YANG X.; Real Time Visual Cues Extraction for Monitoring Driver Vigilance, Department of Electrical, Computer, and System Engineering Rensselaer Polytechnic Institute, New York, 2001.

JI, Q.; YANG X.; Real-Time eye, gaze, and face pose tracking for monitoring driver vigilance, Elsevier Science, Department of Computer Science, University of Nevada, New York, 2002.

LEE, J.; ZEIGH, D.; The neurology of eye movements. Philadelphia: F.A. Davies, 1991.

OH, J.; HAN, M.; PETERSON, BS.; Spontaneous Eyeblinks Are Correlated with Responses during the Stroop Task, PLoS ONE 7(4): e34871. doi:10.1371/journal.pone.003487, Abril, 2012.

ROSEN, Ilene. Driving While Sleepy Should Be A Criminal Offense. Journal of Clinical Sleep Medicine, Philadelphia, v. 1, n. 4, 2005. Disponível em http://www.cadastro.abneuro.org/site/conteudo.asp?id_secao=78&id_conteudo=85&ds_secao=%C3%83%C5%A1ltimos%20Artigos&ds_grupo=Departamento%20de%20Sono Acessado em 15 de outubro de 2016.

VIOLA, P.; JONES, M.; Rapid object detection using a boosted cascade of simple features, Conference on computer vision and pattern recognition, 2001.

YAN, Z.; HU, L.; CHEN, H.; Computer vision syndrome: A widely spreading but largely unknown epidemic among computer users. Comput. Hum. Behav. 24, 2026{2042, Setembro, 2008.