

MÉTODO DE SUBTRAÇÃO DE FUNDO PARA DETECÇÃO VEICULAR EM CONDIÇÕES DE TRÁFEGO INTERROMPIDO

Lucas Sousa Ferreira

Flávio José Craveiro Cunto

Departamento de Engenharia de Transportes

Universidade Federal do Ceará

RESUMO

Um dos grandes problemas no processamento de imagens para o estudo de tráfego urbano ocorre na etapa de subtração de fundo devido a ocorrência de interrupções no fluxo. As funções de segmentação, em geral, apresentam dificuldades de detecção de objetos em situações inerciais, no caso, veículos. O presente trabalho consiste na proposição de um método capaz de fazer a diferenciação dos elementos do tráfego quando este é interrompido, a partir de um método de subtração de fundo por limiarização, que consiste basicamente em uma atualização da função de misturas gaussianas MOG2 do OpenCV. A análise comparativa com o MOG2 mostrou que o método proposto obteve os melhores resultados, com taxas de exatidão e precisão de 91,5% e 89,0%, respectivamente, contra taxas de 79,2% e 85,1% da função do OpenCV. Faz-se também a apresentação das limitações do método, bem como proposições para estudos posteriores.

ABSTRACT

One of the major problems in image processing for the study of urban traffic occurs in the background subtraction stage due to the occurrence of interruptions in the flow. The segmentation functions, generally, present difficulties in detecting objects in inertial situations, in this case, vehicles. The present paper consists in the proposition of a method able to differentiate the traffic elements when it is interrupted, from a method of background subtraction by thresholding, which basically consists of an update of the function of Gaussian mixtures MOG2 of the OpenCV. The comparative analysis with MOG2 showed that the proposed method obtained the best results, with accuracy and precision rates of 91.5% and 89.0%, respectively, against rates of 79.2% and 85.1% of the OpenCV function. It also presents the limitations of the method, as well as propositions for further studies.

1. INTRODUÇÃO

A aplicação de técnicas de processamento de imagens e visão computacional para a análise de sequências de vídeos de fluxo de tráfego é capaz de fornecer dados como de volume, velocidade, coordenadas e classificação veicular, podendo ter relativo sucesso nessas determinações em aplicações e ambientes diversos (Martin et al., 2003; Mimbela e Klein, 2000; Kanagamalliga et al., 2014; Gupte et al., 2002; Luvizon et al., 2014). Além disso, as técnicas de detecção por vídeo apresentam benefícios em relação aos outros métodos, como a simplicidade de instalação e seu baixo nível de interrupção no tráfego (Cunha, 2013; Kastrinaki, 2003).

Dentre as estratégias mais comumente encontradas na literatura para a identificação e rastreamento de objetos aplicadas ao fluxo veicular destaca-se a estratégia baseada em características (*feature based approach*) que é capaz de efetuar o rastreamento mesmo com oclusões parciais, além de poder ser usada em condições diversas de iluminação (McLauchlan et al., 1997; Coifman et al., 1998; Hsieh et al., 2006). Essa estratégia apresenta várias etapas, porém, como a maioria dos demais métodos, geralmente baseia-se na subtração de fundo, para posterior detecção e agrupamento de *features* (Dallalzadeh e Guru, 2010; Kim, 2008).

A subtração de fundo (*background subtraction*) é uma etapa que visa a identificação das regiões da imagem que apresentam objetos que se movem, separando-as das áreas estáticas (*background*). Em aplicações com foco no tráfego é comum a utilização de técnicas de modelagem que permitam a atualização do *background* para posterior subtração entre as imagens do vídeo e de fundo ao nível matricial. Essa atualização garante que os valores de pixel

do fundo (padrão RGB, por exemplo) cheguem mais próximo dos valores de pixel “reais” no momento da subtração, já que as mudanças de luminosidade vão alterando esses valores com o tempo, interferindo assim na subtração.

Apesar dessa vantagem, existem vários desafios associados à modelagem do *background* na aplicação de tráfego e, um deles, seria o fato de que se um veículo parar de se mover por um longo período ele passará a ser aceito como parte do plano de fundo (Lee e Park, 2012). Quando se lida com tráfego em áreas urbanas têm-se a condição de tráfego interrompido, como nas proximidades dos semáforos, onde os veículos podem parar de ser detectados sendo “aderidos” ao plano de fundo.

Uma estratégia de detecção veicular com potencial para aplicação no regime de tráfego ininterrupto necessita, portanto, lidar de forma adequada com situações que envolvam a parada sistemática do fluxo veicular. O presente trabalho tem por objetivo apresentar um método de subtração de fundo com foco na diferenciação dos objetos para análise de tráfego mesmo em condição de fluxo interrompido, de forma a manter a sua detecção.

2. REVISÃO DA LITERATURA

Segundo Albuquerque et al. (2000) segmentar seria nada mais que a divisão de uma imagem em diferentes regiões, as quais serão processadas por algoritmos na busca de informações. Por ser responsável pela separação do objeto de estudo do seu entorno, esta etapa é de fundamental importância para o processo de detecção e rastreamento veicular, sendo uma das mais difíceis de serem feitas de forma automática em processamento de imagens (Gonzalez e Woods, 2007).

Há na literatura diversos métodos de segmentação, dos mais simples que fazem operações básicas, pixel-a-pixel, aos mais complexos que buscam ter o entendimento da imagem a partir de modelos matemáticos (Cunha, 2013). Esses inúmeros métodos buscam sanar a necessidade de tratamento dos diferentes tipos de imagem (noturnas, ou de baixa iluminação, com focos de luz, com sombras, entre outras) de forma que não existe um método único que possa ser considerado bom para todas as situações (Pal e Pal, 1993).

Um dos métodos de segmentação mais utilizados é chamado subtração de fundo (*background subtraction*), o qual é considerado uma operação fundamental para processamento de imagens de vídeos (Gonzalez e Woods, 2007). Nesse método, faz-se a diferenciação entre uma imagem que represente o fundo (Figura 1) da sequência de imagens do vídeo (Figura 2), de forma que os objetos que não pertencem ao fundo ficam em evidência na forma de imagem binária (Figura 3). Para aplicações em transportes, os objetos de interesse seriam os veículos e pedestres (*foreground*, em branco) e o fundo seriam os demais elementos (*background*, em preto).

Quando considerada a sua característica de modelagem do *background*, pode-se dividir as técnicas de segmentação em dois grandes grupos: as técnicas não recursivas e recursivas (Cheung et al, 2003). As técnicas não recursivas estimam o fundo a partir de uma amostragem de imagens armazenadas em um buffer. Como exemplo delas têm-se a subtração frame a frame (que usa o frame t-1 na subtração do frame atual) e o filtro da mediana, o qual faz uma amostragem para uma quantidade x de frames anteriores e calcula a mediana dos valores de cada pixel de forma a gerar uma representação do plano de fundo.



Figura 1: Plano de fundo (Background) do vídeo



Figura 2: Frame do vídeo utilizado na análise



Figura 3: Imagem binária após subtração de fundo

As técnicas recursivas, por sua vez, atualizam a modelagem de fundo a cada novo frame, utilizando de modelos matemáticos. Entre essas técnicas destacam-se as que utilizam Misturas Gaussianas (*Mixture Of Gaussians Model* - MOG, Stauffer & Grimson, 1999; Zivkovic & Van der Hijden, 2006), apontada como tendo os melhores resultados em ambientes de tráfego urbano (Cheung et al, 2003).

A *Open Source Computer Vision Library* (OpenCV) é uma biblioteca de visão computacional sob licença da BSD. De uso livre, a biblioteca possui interfaces para C ++, Python e Java, sendo suportada em Windows, Linux, Mac OS, iOS e Android. Esta biblioteca possui duas funções de subtração de fundo utilizando misturas gaussianas, as funções MOG e MOG2.

A função MOG, `< cv2.createBackgroundSubtractorMOG() >`, é um algoritmo introduzido através do artigo "*An improved adaptive background mixture model for real-time tracking with shadow detection*" por P. KaewTraKulPong e R. Bowden em 2001. Neste método é necessário dar entrada com um número k de misturas gaussianas a serem utilizadas na modelagem de cada pixel do *background*.

Por sua vez, a função MOG2, `<cv2.createBackgroundSubtractorMOG2()>`, é baseada em dois trabalhos de Z. Zivkovic, "*Improved adaptive Gaussian mixture model for background subtraction*" de 2004 e "*Efficient Adaptive Density Estimation per Image Pixel for the Task of Background Subtraction*" de 2006. Diferentemente da função MOG, esta função seleciona automaticamente o número de distribuições gaussianas apropriadas para modelar cada pixel do *background*. Isso a torna mais adaptada as variações de luminosidade do ambiente, sendo uma alternativa melhorada do MOG, principalmente quando busca-se trabalhar em condições de ambiente urbano.

A função MOG2 possui três parâmetros principais, sendo então chamada da seguinte forma: `<cv2.createBackgroundSubtractorMOG2(history,varThreshold,bShadowDetection)>`, em que *history* é o número *n* de frames considerados na modelagem; *varThreshold* é o limiar dado em distância Mahalanobis do espaço de cores dos pixels (pode-se entender como um tipo de “desvio padrão”) para diferenciar os valores de pixel do fundo e do *foreground*; e *bShadowDetection* é o valor que define se a detecção de sombra estará ativa.

Um problema encontrado nos métodos recursivos, como o MOG, é que o acréscimo de cada novo frame com objetos que se movem lentamente, ou parados, poderá gerar uma modelagem de fundo errônea ao aderir o objeto ao fundo (Stauffer e Grimson, 1999). A Figura 4 apresenta o *background* gerado por uma aplicação da função MOG2 em um *frame* onde o fluxo veicular encontra-se interrompido, enquanto a Figura 5 apresenta o *background* dezoito segundos depois. Observa-se o início incorporação do carro ao *background* no canto superior direito da imagem, o que pode gerar a perda do objeto como *foreground*.



Figura 4: *Background* gerado pela função MOG2 do *Frame* 1440 do vídeo utilizado na análise

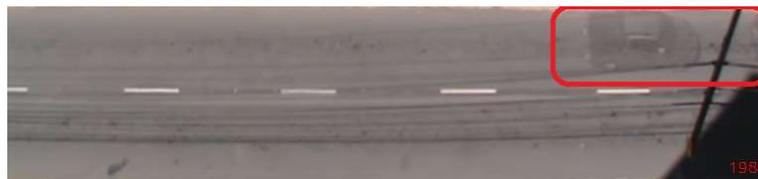


Figura 5: *Background* gerado pela função MOG2 do *Frame* 1980 do vídeo utilizado na análise

3. MÉTODO PROPOSTO

O método proposto neste trabalho consiste das seguintes etapas: (i) Definição da sequência de vídeo para análise; (ii) Processamento da segmentação de referência (*ground truth*); (iii) Apresentação do método de subtração proposto; (iv) Calibração e execução dos algoritmos de segmentação e; (v) Análise comparativa dos resultados encontrados.

3.1. Definição da sequência de vídeo para análise

Na etapa de coleta de dados (definição da sequência de frames), utilizou-se um vídeo em tráfego urbano, com 7209 quadros de 766x179 pixels, na Avenida Pontes Vieira, em Fortaleza, nas proximidades da interseção semaforizada (Av. Pontes Vieira x Rua Barbosa de Freitas). Desse vídeo selecionaram-se os períodos onde o semáforo fechava, e, conseqüentemente, os veículos paravam, de forma a avaliar o desempenho das rotinas perante o fluxo interrompido. Dentre esses períodos de interrupção do fluxo, foram analisados 30 quadros escolhidos aleatoriamente em intervalos de 1 segundo de vídeo, compreendendo três períodos de interrupção no fluxo, com amostragem de 10 *frames* em cada um. No momento de aquisição do vídeo havia condições de luminosidade variável e uma sombra no canto inferior direito do plano de filmagem (Figura 1).

3.2. Processamento da segmentação de referência (*ground truth*)

Para fazer a análise comparativa entre as diferentes técnicas de segmentação, é comum observar na literatura um procedimento de análise de acurácia que utiliza a segmentação ideal (*ground truth*, Figura 6), feita manualmente. O *ground truth* utilizado neste trabalho foi feito através do aplicativo Adobe Photoshop®, onde separaram-se os objetos de interesse, no caso os veículos, do plano de fundo, produzindo as imagens binárias, atribuindo a cor branca aos veículos e a cor preta ao plano de fundo. Cada imagem levou aproximadamente 20 minutos para ser feita, totalizando 10 horas de processamento manual.

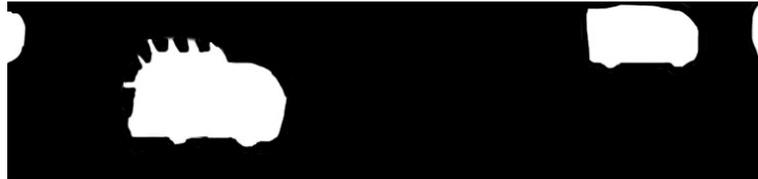


Figura 6: Ground truth do frame apresentado na Figura 1

3.3. Apresentação do método de subtração proposto

Quanto ao algoritmo proposto, o mesmo consiste em uma rotina escrita em linguagem Python, versão 3.5, com o uso da biblioteca OpenCV, versão 3.4, que pode ser vista como um uso combinado da limiarização com o MOG2 para a adaptação de um método de subtração de fundo ao tráfego urbano. A concepção do algoritmo parte das etapas utilizadas nos métodos de subtração de fundo em geral: criação e atualização de fundo, e a subtração propriamente dita. A estrutura geral do algoritmo proposto é apresentada na Figura 7.

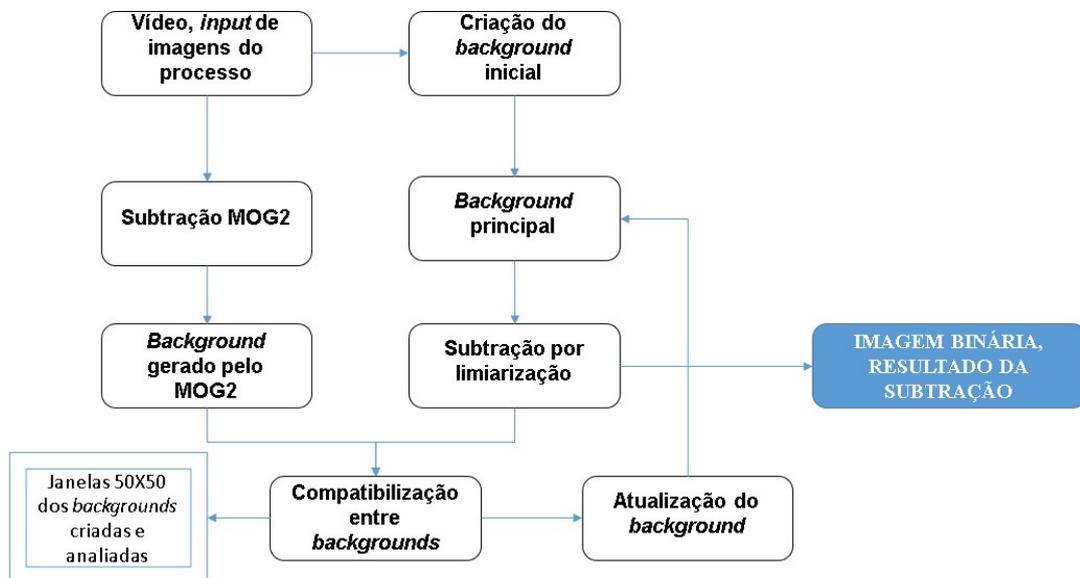


Figura 7: Fluxograma do método de subtração de fundo proposto

A primeira etapa compreende a criação de um plano de fundo inicial anterior a subtração em si, colocado em escala de cinza para posterior atualização deste durante o processo de subtração.

Uma prática corrente é iniciar o algoritmo utilizando um quadro “vazio” (Mandellos et al, 2011), o que gera certas dificuldades na aplicação ao tráfego, já que se faria necessária uma imagem com a pista vazia em horários de uso, o que nem sempre é possível. O algoritmo proposto põe de lado esta prática fazendo uma modelagem prévia do *background*, através da função MOG2, utilizando um número n de quadros iniciais, de modo a dar o ponto de partida para o posterior processo de subtração. A amostragem desses quadros deve ocorrer em período de fluxo livre, para que a função seja capaz de fazer uma boa modelagem do fundo.

A atualização do fundo é feita baseando-se em um princípio de atualização seletiva (Elgammal, 1999) passando por uma dupla filtragem, que se baseia em critérios relativos a uma segmentação em paralelo feita pela função MOG2 e por uma estratégia de subtração de fundo por limiarização simples (*gray level thresholding*), onde se adota um valor limite na subtração de fundo, que separa o *foreground* do *background* (valores maiores que esse limite na subtração são considerados *foreground*). A atualização do fundo no *frame* atual irá gerar o *background* usado na subtração usando limiarização do próximo *frame*.

A compatibilização entre os dois *backgrounds* gerados é feita através de uma comparação feita por regiões das imagens binárias, tanto do MOG2 quanto da limiarização, em um dado *frame*. As dimensões das regiões são dados de entrada no algoritmo proposto. O modelo padrão adotado para esta análise foi de seções quadradas de 50x50 pixels. Isso quer dizer que o algoritmo irá seccionar cada *frame* das imagens binárias, dos dois métodos, em quadrados de 50 pixels de lado para analisar.

Em cada seção (quadrados) é verificado se a função MOG2, ou a subtração feita por limiarização, detecta a presença de *foreground* (veículos ou ruído, pixels brancos). Caso isso aconteça, é feita uma análise para constatar se as taxas de *foreground* presentes nas duas imagens binárias testadas (MOG2 e limiarização) estão dentro dos valores considerados aceitáveis para cada uma (ruído), sendo esse valor também um parâmetro de entrada no algoritmo. Se na área analisada houver a presença de *foreground* acima desses valores, em qualquer uma das estratégias de subtração, o fundo não é atualizado, caso contrário, o *background* na respectiva área é atualizado, recebendo a seção do *frame* do vídeo referente a área das imagens binárias analisadas.

A atualização baseada nos dois métodos de subtração procura também resolver o problema relativo à mudança gradual da iluminação, que ocorre devido à mudança da luz solar com o tempo (Lee e Park, 2012). Dessa forma o fundo pode adaptar-se a leves variações de luminosidade do ambiente e a até mesmo pequenas movimentações de sombra, considerando também que não se deixará de segmentar veículos parados, já que a subtração é feita por um fundo que onde a modelagem não depende diretamente da movimentação do veículo.

Após a compatibilização entre os dois *backgrounds* gera-se um plano de fundo atualizado, o qual será utilizado na continuidade do processo de subtração por limiarização. Com esse *background* atualizado em mãos, gera-se o resultado do processo que é a diferenciação entre o fundo e os objetos de interesse.

Quanto aos parâmetros do algoritmo proposto, destacam-se os relativos ao MOG2 (*history*, *varThreshold* e *bShadowDetection*), ao processo de subtração de fundo por limiarização simples (*thresh*) e os parâmetros de compatibilização entre os *backgrounds* (tamanho das áreas

de atualização e taxas limite de aceitação de *foreground*). Para os testes, utilizaram-se uma taxa de aceitação para o MOG2 de 0% e para a limiarização uma taxa de 8%, devido à sombra na pista. Irá então se analisar a variação apenas do parâmetro mais diretamente ligado a distinção entre o que é fundo e o que é objeto na subtração, ou seja o *thresh*.

3.4. Calibração e execução dos algoritmos de segmentação

De forma a permitir uma análise entre os algoritmos de segmentação, realizou-se uma etapa de calibração dos parâmetros do algoritmo MOG2 e do algoritmo proposto (MOG2+Limiarização). As medidas de desempenho utilizadas para o processo de calibração estão relacionadas aos valores médios de detecção correta de pixels de *background* e de *foreground* considerando os 30 quadros amostrais, que dirão quão próximo a configuração de parâmetros de cada método chegou da subtração ideal representada pelo *ground truth*.

A Tabela 1 apresenta os indicadores de acurácia utilizados, além de outros parâmetros que serão observados, comuns na literatura (Metz, 1978; Fawcett, 2005; Cunha, 2013; Cheung et al, 2003). Como critério para a definição dos valores ótimos (função de otimização) dos parâmetros foi utilizada uma função soma englobando as métricas exatidão e precisão. A Equação 1 representa a função de otimização usada.

$$f_{otimização} = Máx[Exatidão(p) + Precisão(p)] \quad (1)$$

em que p: parâmetro calibrado.

Na estratégia baseada no MOG2, aplicou-se o valor padrão de *varThreshold=16*, disposto na documentação do OpenCV. Fez-se, porém, a avaliação da variação do parâmetro *history*, em um intervalo de 15 a 4000 frames, de forma a procurar o melhor desempenho do método perante as métricas utilizadas para a situação de fluxo veicular interrompido (função no código: `<cv2.createBackgroundSubtractorMOG2(15-4000,16, False)>`).

No algoritmo proposto (MOG2 + limiarização) foram testados valores do parâmetro *thresh* entre 20 e 70, de forma a possibilitar a escolha de um valor que maximize a qualidade da subtração com as mesmas métricas propostas. A função de subtração absoluta por limiarização simples é chamada com os seguintes valores: `<s, imagem=cv2.threshold(sub,thresh,maxval,cv2.THRESH_BINARY)>` em que *sub*: subtração entre o fundo modelado e o frame do vídeo; *imagem*: subtração com *threshold* aplicado, transformando em imagem binária pela função `cv2.THRESH_BINARY`; *thresh*: variando entre 20 e 70; *maxval* = 255 (branco). Já a função MOG2 utilizada para filtragem e atualização do fundo tem os valores a seguir: *history: 30, varThreshold: 16, detectShadows: False*.

Tabela 1: Métricas de desempenho

Taxa de verdadeiro positivo (%)=VP/P	Taxa de falso negativo(%)=FN/N
Taxa de falso positivo (%)=FP/N	Taxa de verdadeiro negativo(%)=VN/N
Exatidão(%)=(VP+VN)/(P+N)	Precisão(%)=VP/(VP+FP)
VP= Verdadeiro positivo, pixels classificados corretamente como <i>foreground</i>	VN= Verdadeiro Negativo, pixels classificados corretamente como <i>background</i>
FP= Falso Positivo, pixels classificados incorretamente como <i>foreground</i>	FN= Falso negativo, pixels classificados incorretamente como <i>background</i>
P = número de pixels considerados <i>foreground</i> no <i>ground truth</i>	N = número de pixels considerados <i>background</i> no <i>ground truth</i>

A combinação da quantidade de testes do parâmetro *history*, do MOG2, com a do parâmetro *thresh*, da estratégia baseada na limiarização, gerou um total de 68 cenários de 7209 frames (da sequência de vídeo) durante o processo de calibração. A comparação entre os pixels detectados pelos métodos e o *ground truth* para cada cenário foi obtida a partir de um algoritmo escrito em *python*, o qual foi aplicado aos 30 *frames* amostrais. Cada configuração de parâmetros (cenário) levou aproximadamente sete minutos para o completo processamento, ou seja, subtração de fundo e análise de acurácia.

3.5. Análise comparativa dos resultados encontrados

A análise comparativa dos resultados foi realizada para a configuração ótima dos dois algoritmos de subtração de fundo. Foram observadas as médias das métricas e os desvios padrões de desempenho apresentadas na Tabela 1, complementadas por testes de estatística t entre os resultados dos dois métodos, onde se assumiu um $\alpha=0,05$ e a hipótese nula de que a diferença entre as médias das amostras dos dois métodos é zero, consideraram-se ainda diferentes as variâncias das duas amostras.

4. RESULTADOS

Nesta seção serão apresentados os principais resultados das etapas de calibração e da análise comparativa entre o método aplicando MOG2 e o método proposto (MOG2+Limiarização).

4.1. Calibração dos algoritmos de subtração de fundo

Na calibração da variável *history* para a função MOG2, observou-se que tanto a taxa de verdadeiro positivo quanto a de falso positivo crescem com o aumento do parâmetro (Figura 8A). Quando se observa a variação da exatidão e da precisão (critério de otimização), nota-se que a primeira é continuamente crescente, porém a segunda começa a decair em determinado ponto (Figura 8B). Então, para o caso em estudo, optou-se pelo *history=3000*, que maximizou a soma dos valores de exatidão e precisão, com taxas de 79,20% e 85,09%, respectivamente. Observa-se que esse *history* representa mais de 40% dos frames do vídeo analisado.

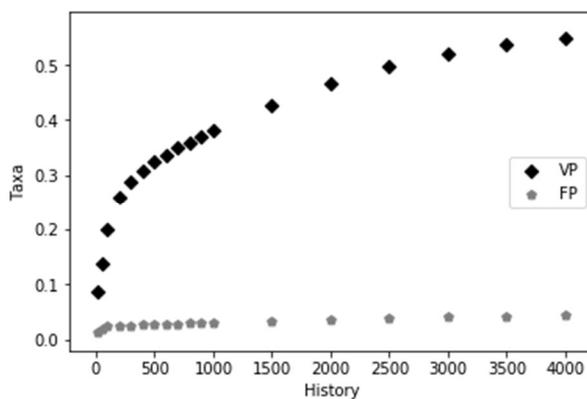


Figura 8A: *History* x Taxas Positivas

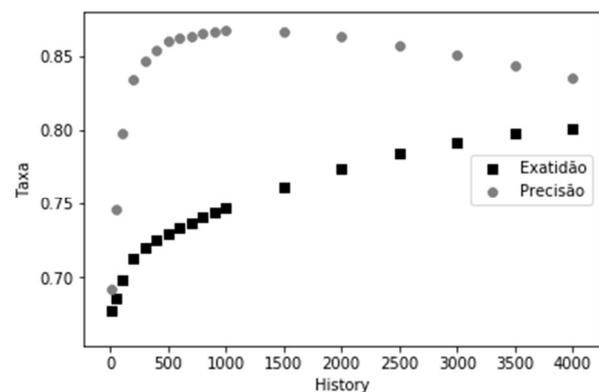


Figura 8B: *History* x Métricas de acurácia

Quanto a calibração do parâmetro *thresh* da estratégia de limiarização do algoritmo proposto, percebeu-se que há uma tendência de diminuição, tanto da taxa de falso positivo quanto de verdadeiro positivo, com o aumento do *thresh* (Figura 9A). Isso se dá devido ao fato de que com o aumento desse parâmetro o número de pixels fora do limiar para ser considerado *background* será menor, ou seja, o aumento do *thresh* diminui o número de pixels considerados

foreground (lembrando que os valores P e N são constantes, denominadores das taxas de verdadeiro positivo e falso positivo, respectivamente).

Além disso, percebeu-se que há uma tendência de aumento da precisão com o aumento do *thresh*, até um valor praticamente constante de 90% (Figura 9B). Isso pode ser explicado pelo fato observado nos resultados que a taxa de decaimento dos valores de falso positivo ser cada vez menor, fazendo com que, em determinado ponto, essa diminuição pare de influenciar substancialmente no valor da precisão.

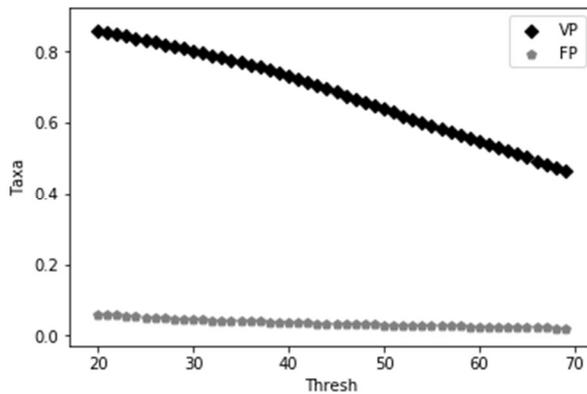


Figura 9A: Thresh x Taxas Positivas

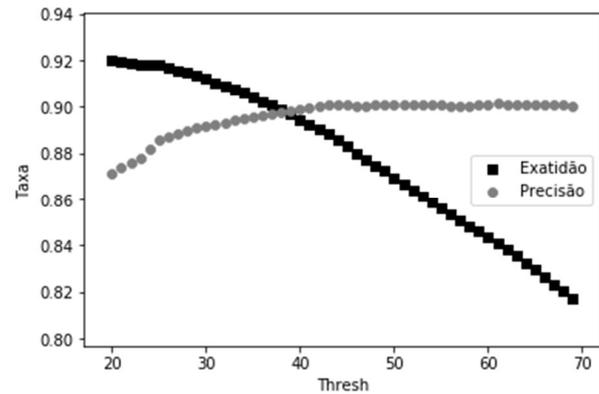


Figura 9B: Thresh x Métricas de acurácia

A partir da Figura 9B pode-se observar que há uma tendência contrária das taxas de exatidão em relação ao *thresh* quando comparada com as taxas de precisão. Isso ocorre por que o número de verdadeiros positivos decresce rapidamente com o aumento do *thresh* e, apesar do número de verdadeiros negativos aumentar devido ao aumento do limiar, ele não tem o crescimento proporcional ao decrescimento do número de VP's.

Como os valores de *thresh* mais baixos tendem a maximizar as taxas de exatidão e esta tem uma taxa de decrescimento maior que a taxa de crescimento da precisão com o *thresh*, percebe-se que menores valores tendem a maximizar uma função soma entre as duas variáveis. Então, para as condições observadas, adotou-se um valor de 28 para o *thresh*, com taxa de exatidão de 91,45% e de precisão de 88,96%, sendo os maiores valores encontrados considerando as duas variáveis.

4.2 Análise Comparativa entre as estratégias

Após a determinação do parâmetro *thresh* do algoritmo e do *history* da função MOG2, pode-se fazer a análise comparativa de desempenho entre os dois métodos. A Tabela 2 apresenta a discriminação dos resultados médios dos dois métodos, enquanto a Tabela 3 apresenta os desvios padrões de cada indicador e a Tabela 4 apresenta os resultados da análise estatística que analisa a diferença das médias dos dois valores, considerando os resultados amostrais com *thresh* = 28 para o algoritmo proposto e com *history* = 3000 para o MOG2.

Observa-se pela Tabela 4 que não há evidências para crer que taxa de falso positivo (valor-P=0,1578 > $\alpha=0,05$) ou a taxa de verdadeiro negativo (valor-P=0,1578 << $\alpha=0,05$) foram maiores no algoritmo proposto, apesar do uso da limiarização simples utilizando *thresh*, que poderia acarretar em problemas de falso positivo. Quanto ao valor da taxa de falso negativo, há evidências para crer que o valor obtido no algoritmo proposto foi bem menor que o observado no MOG2 (valor-P=0,0004 << $\alpha=0,05$; estatística t= -3,763), enquanto o verdadeiro positivo foi

bem maior (valor- $P=0,0001 \ll \alpha=0,05$). Isso ocorre pela tendência dos métodos recursivos de aderirem ao *background* os objetos parados depois de determinado tempo, como já citado anteriormente, o que não é o caso do método apresentado aqui.

Tabela 2: Médias das métricas de desempenho dos métodos

Algoritmo	Taxa FP	Taxa FN	Taxa VP	Taxa VN	Taxa Exatidão	Taxa Precisão
Proposto	4,6%	8,5%	81,6%	95,4%	91,5%	89,0%
MOG2	4,0%	30,2%	52,1%	96,0%	79,2%	85,1%
(Proposto-MOG2)	0,6%	-21,6%	29,5%	-0,6%	12,3%	3,9%

Tabela 3: Desvios padrões das métricas de desempenho

Algoritmo	Taxa FP	Taxa FN	Taxa VP	Taxa VN	Taxa Exatidão	Taxa Precisão
Proposto	1,1%	1,9%	6,6%	1,1%	1,0%	5,1%
MOG2	2,9%	31,5%	38,0%	2,9%	15,6%	8,2%

Tabela 4: Análise estatística das métricas de desempenho

	Taxa FP	Taxa FN	Taxa VP	Taxa VN	Taxa Exatidão	Taxa Precisão
Valor-P	0,1578	0,0004	0,0001	0,1578	0,0001	0,0164
Estatística - t	1,017	-3,763	4,183	-1,017	4,289	2,197

Observa-se também que tanto a taxa de exatidão quanto a de precisão, que são os indicadores diretamente observados nesta análise, apresentam resultados melhores no algoritmo proposto, considerando ainda que os dois apresentaram valores- P menores que 2% ($\ll \alpha=5\%$), levando a crer que este apresenta um desempenho melhor que o método de misturas Gaussianas, representados pela função MOG2, nas condições de fluxo interrompido observadas.

6. CONSIDERAÇÕES FINAIS

O presente trabalho discorreu sobre os conceitos de segmentação e a dificuldade que é observada nos métodos em geral quando deparados com condições de tráfego interrompido. Diante dessa dificuldade apresentada, este trabalho propôs um método capaz de manter a segmentação mesmo quando ocorre a parada do fluxo veicular.

Para analisar a eficácia do método proposto, utilizou-se da associação de uma segmentação ideal, chamada de *ground truth*, com métricas de desempenho utilizadas na literatura. Obtidos os resultados dessas métricas comparou-se com os obtidos pela aplicação do método de misturas gaussianas, que é apontado na literatura como tendo bom desempenho, mesmo em tráfego urbano, tomando como exemplo do método a função MOG2 do OpenCV.

As análises comparativas entre os dois métodos mostraram que o método adaptado (MOG2+Limiarização) resultou em taxas de exatidão e precisão médias de 91,5% e 89,0% respectivamente, em comparação às taxas de 79,2% e 85,1% para o MOG2. A análise estatística ainda leva a crer que o algoritmo proposto se mostra superior ($\alpha=0,05$) na estimação de falsos negativos e verdadeiros positivos, além de ter maior taxa de exatidão e precisão.

Apesar do desempenho superior do método proposto, deve-se levar em consideração algumas limitações. Observou-se durante os testes que a movimentação de sombras pode interferir na atualização, necessitando de valores de *thresh* maiores para sanar esse problema. Como visto,

porém, valores de *thresh* maiores acarretam em uma subtração de fundo menos exata, logo situações onde há sombra na pista podem diminuir a qualidade do processo.

Uma proposição que pode ser feita para o método, tema de trabalhos futuros, seria a troca da limiarização por outro método de filtragem. A adoção de um filtro que permita que o fundo seja comparado com a imagem de vídeo de forma probabilística provavelmente melhoraria o desempenho do mesmo, além de melhorar sua qualidade para condições de iluminação menos ideais.

Por fim, avalia-se que, devido os resultados obtidos pelas métricas de desempenho nas condições observadas, o método proposto obteve êxito como alternativa mais eficaz de subtração de fundo para condições inerciais de tráfego no comparativo com o método que utiliza misturas gaussianas.

Agradecimentos

Os autores agradecem o apoio do Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) para o desenvolvimento dessa pesquisa.

REFERÊNCIAS BIBLIOGRÁFICAS

- Albuquerque, M. P. de e Albuquerque, M. P. de (2000) *Processamento de Imagens: Métodos e Análises*. Rio de Janeiro: Centro Brasileiro de Pesquisas Físicas. 12 p.
- Beymer, D.; Mclauchlan, P.; Coifman, B. e J. Malik (1997) A Real-time Computer Vision System for Measuring Traffic Parameters. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, p. 495-501, doi: 10.1109/CVPR.1997.609371.
- Cheung, S. e C. Kamath (2004) Robust Techniques for Background Subtraction in Urban Traffic Video. *Proceedings of SPIE - Visual Communications And Image Processing 2004*. 5308. 10.1117/12.526886.
- Cunha, A. L. B. N. da (2013) Sistema automático para obtenção de parâmetros do tráfego veicular a partir de imagens de vídeo usando OpenCV. São Carlos, SP.
- Dallalzadeh, E. e D. S. Guru (2010) Feature-based tracking approach for detection of moving vehicle in traffic videos. *Proceedings Of The First International Conference On Intelligent Interactive Technologies And Multimedia* – v. 10, n.7, p.254-260. ACM Press. <http://dx.doi.org/10.1145/1963564.1963609>.
- Elgammal, A., Harwood, D., e L.S. Davis (2000) Nonparametric Model for Background Subtraction. *Proceedings of ECCV '00 Proceedings of the 6th European Conference on Computer Vision-Part II*, p.751-767. ISBN:3-540-67686-4.
- Gonzalez, R. C. e R. E. Woods (2007) *Processamento de Imagens Digitais*. Editora Blucher: São Paulo. 509 p. ISBN: 85-212-0264-44.
- Gupte, S.; Masoud, O.; Martin, R. F. K. e N. P. Papanikolopoulos (2002) Detection and classification of vehicles, *IEEE Transactions on Intelligent Transportation Systems*, v.3, n.1, p. 37-47. doi: 10.1109/6979.994794.
- Hsieh, J.-W.; Yu S.-H.; Chen, Y.-S. e W.-F. Hu (2006) Automatic traffic surveillance system for vehicle tracking and classification, *Intelligent Transportation Systems*, IEEE Transactions on, v. 7, p. 175-187.
- Zivkovic, Z. (2004) Improved adaptive Gaussian mixture model for background subtraction. *Proceedings of the 17th International Conference on Pattern Recognition*. doi:10.1109/ICPR.2004.1333992
- Kanagamalliga, S.; Vasuki, S.; Kanimozhidevi, A.; Priyadharshni, S. e S. Rajeswari (2014) Tracking And Counting The Vehicles In Night Scenes. *International Journal of Information Sciences and Techniques*, v. 4, p. 165-172. 10.5121/ijist.2014.4320.
- Kastrinaki, V.; Zervakis, M. e K. Kalaitzakis (2003) A survey of video processing techniques for traffic applications. *Image and Vision Computing*, v. 21, p. 359-381. 10.1016/S0262-8856(03)00004-0.
- Kim, Z. (2008) Real time object tracking based on dynamic feature grouping with background subtraction. *Ieee Conference On Computer Vision And Pattern Recognition*, [s.l.], p.1-8, IEEE. doi: 10.1109/CVPR.2008.4587551
- Lee, J. e M. Park (2012) An Adaptive Background Subtraction Method Based on Kernel Density Estimation. *Sensors* (Basel, Switzerland) [s.l.], v.12, n.9, p.12279-12300. <http://dx.doi.org/10.3390/s120912279>.

- Luvizon, D. C.; Nassu, B. T. e R. Minetto (2014) Vehicle speed estimation by license plate detection and tracking," *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, Florence, p. 6563-6567. doi: 10.1109/ICASSP.2014.6854869
- Mandellos, N.; Keramitsoglou, I. e C. T. Kiranoudis (2011) A Background Subtraction Algorithm for Detecting and Tracking Vehicles. *Expert Systems with Applications*, v.38, n. 3, p. 1619–31, doi:10.1016/J.ESWA.2010.07.083.
- Martin, P.T.; Feng, Y. e X. Wang (2003) Detector Technology Evaluation, Technical Report, Utah Transportation Center.
- Mimbela, L. E. Y. e Klein, L. A. (2000) A Summary of Vehicle Detection and Surveillance Technologies Used in Intelligent Transportation Systems, the Vehicle Detector Clearinghouse, New Mexico State University, Fall 2000.
- Pal, N. R. e S. K. Pal (1993) A Review on Image Segmentation Techniques. *Pattern Recognition*, v. 26, n. 9, p.1277-1294, doi: 10.1016/0031-3203(93)90135-J.
- Stauffer C. e W. E. L. Grimson (1999) Adaptive background mixture models for real-time tracking, *Proceedings. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No PR00149)*, Fort Collins, CO, v. 2, p. 252, doi: 10.1109/CVPR.1999.784637
- Zivkovic, Z. (2004) Improved adaptive Gaussian mixture model for background subtraction. *Proceedings of the 17th International Conference on Pattern Recognition*. doi:10.1109/ICPR.2004.1333992
- Zivkovic, Z., e Van der Hijden, F. (2006). Efficient adaptive density estimation per image pixel for the task of background subtraction. *Pattern Recognition Letters*, 55(5), 773–780.
-

Flávio Jose Craveiro Cunto (flaviocunto@det.ufc.br)

Lucas Sousa Ferreira (lucsouferreira@gmail.com)