

Segmentação e Identificação de Automóveis pelo método Self-adaptive Detection

Pedro Augusto Ramalho Duarte
Estudante de Engenharia de Computação
Universidade de Brasília
Email: pedro_aduarte@aluno.unb.br

Waliff Cordeiro Bandeira
Estudante de Engenharia de Computação
Universidade de Brasília
Email: WaliffCordeiro@aluno.unb.br

Abstract—Com base na dificuldade de obtenção e a grande imprecisão do background que podemos obter inicialmente, este artigo propõe um método baseado na subtração de imagens no qual extraímos o background e o foreground detectando suas bordas para poder, baseado na movimentação dos objetos (automóveis) entre os frames, identificar o que está ou não em movimento.

Ao nos depararmos com a dificuldade da atualização dos frames em tempo real implementamos a solução de utilizar o cython como auxiliar para melhor o processamento pixel a pixel, deixando assim muito mais veloz e eficaz o algoritmo apresentado. Após separar background e foreground, fazendo a diferença de frames, podemos finalmente obter o objeto em movimento. Resultados experimentais mostram que este método ...

I. INTRODUCTION

Com o aumento no tráfego de veículos torna-se inviável um humano gerenciar, por exemplo, rodovias tornando fundamental um sistema ad-hoc para auxiliar nesse tipo de gerenciamento. As informações desse sistema não podem facilmente serem atualizadas em tempo hábil, tornando-se necessário a buscar outras alternativas. Esse sistema torna-se complexo devido as condições de tráfego que são bastante adversas, tendo em vista que o background não pode ser obtido de forma trivial.

Novembro 27, 2018

II. BACKGROUND AND RELATED WORK

Com base no artigo Self-adaptive Detection of Moving Vehicles in Traffic Video

A. First Segmentation of the Current Frame

A cada frame do vídeo o passamos inicialmente para a escala cinza.

A primeira segmentação consiste na obtenção da diferença do frame atual em relação ao frame anterior que é dada pela seguinte equação:

$$D_k(x, y) = F_k(x, y) - F_{k-1}(x, y)$$

Após obter a diferença de imagens deve ser feita uma binarização da imagem, no qual atribui 0 para $D(i, j) \geq T$ e 1 para $D(i, j) < T$, sendo T o threshold que equivale a 10% do valor do pixel mais intenso.

Pela diferença de imagens pode-se perceber a movimentação dos automóveis, utilizou-se então o Canny edge detection

method que ressalta as bordas do veículo de maneira ainda muito ineficiente, gerando grande quantidade de falso-positivo. Pensou-se então em um método capaz de melhorar esse contorno gerado pelo Canny.



Fig. 1. Diferença de Imagens.

Visando aperfeiçoar e preencher as bordas geradas, foram utilizadas as seguintes funções morfológicas: Closing que consiste em uma dilatação seguida de uma erosão e preenche de forma considerável as imagens contornadas pelo Canny e posteriormente foi utilizado a Opening que é o inverso da Closing e executa uma erosão seguida de uma dilatação diminuindo a quantidade de ruídos da imagem. Tendo como resultado do Canny melhorado pelas operações morfológicas uma máscara de qualidade razoável, é possível separar as duas regiões desejadas através das seguintes equações:

$$background = frame * (1 - mascara)$$

$$foreground = mascara * frame$$

B. Self-adaptive Background Updating

Após a segmentação do foreground, deve-se utilizar o frame atual para obter uma atualização do background:

$$B_k = (1 - \alpha)B_{k-1} + \alpha F_k, \quad se D_k = 0$$

$$B_k = B_{k-1}, \quad se D_k = 1$$

Visto que a operação pixel a pixel é muito custosa utilizamos uma biblioteca chamada Cython, no qual funciona como um código mesclado de python com declarações de variáveis em

C com intuito de aumentar o desempenho, no qual é essencial para ter uma segmentação em tempo real.



Fig. 2. Self-adaptative Background Updating.

III. PROPOSED SOLUTION

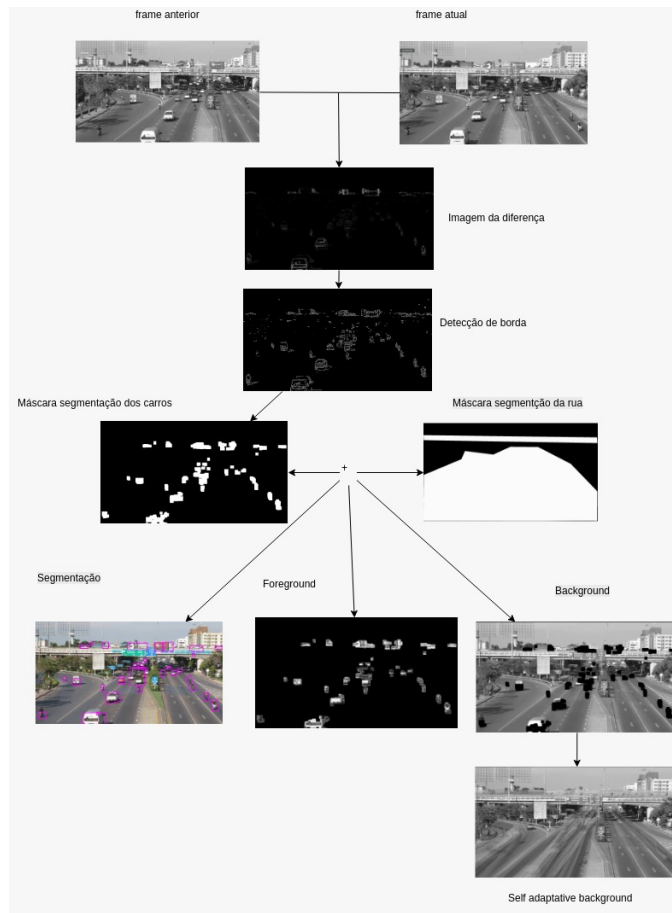


Fig. 3. Fluxo de Resolução.

O paper que serviu de inspiração para o nosso, consegue fazer um bom trabalho na parte de segmentar os automóveis, porém ainda é visto uma série de ruídos causados pelo vento ou passáros. Tendo em vista que os vídeos são filmados por câmeras paradas nas estradas, utilizamos de uma máscara binária que segmenta a estrada já moldada anteriormente. A operação de segmentar a estrada automaticamente é muito

custosa e isso retiraria a segmentação em tempo real, além disso usar uma máscara moldada anteriormente não prejudica o algoritmo por esse motivo.

Outra estratégia utilizada é analisar a área do retângulo que segmenta o automóvel, se ela for muito menor que a média, provavelmente a segmentação é de um ruído.

Incluimos também um classificador simples no qual por meio de uma análise de área e proporção dos retângulos, classifica como um carro ou conjunto de carros.

IV. EXPERIMENTAL RESULTS

O primeiro experimento a ser avaliado é o suprimento de ruído fora da rodovia por meio de uma primeira segmentação da pista e depois um processamento para formar uma máscara binária que é utilizada antes de marcar os carros com os retângulo. A primeira imagem é o algoritmo sem a máscara binária da rodovia, já segunda imagem é o algoritmo com a máscara binária da rodovia

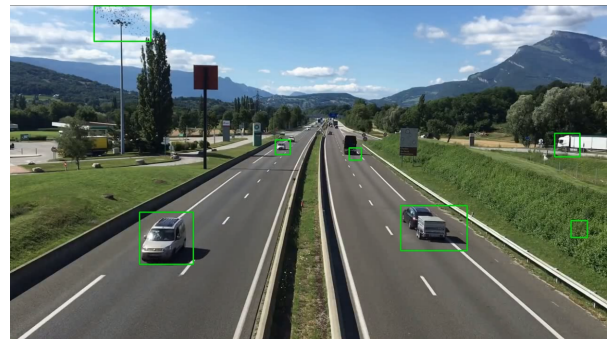


Fig. 4. Algoritmo detectando coisas indevidas.

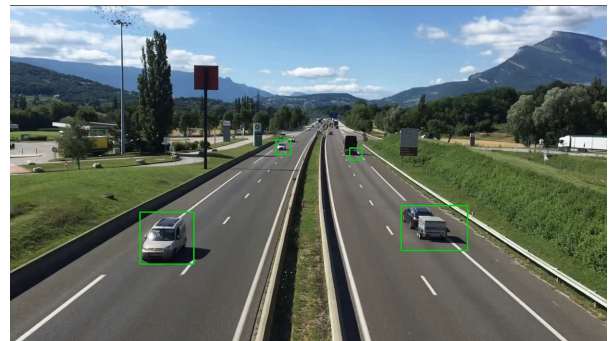


Fig. 5. Algoritmo melhorado.

É possível ver o que o resultado é bem satisfatório, suprimindo os passáros na esquerda da imagem e as plantas que se moviam por conta do vento, porém um caminhão a direita que não estava na rodovia também foi suprimido

O Segundo experimento foi para testar a acurácia da detecção dos carros. Nele contamos a quantidade de carros presentes na área de detecção e quantos carros eram detectados pelo algoritmo, e o resultado foram os seguintes

TABLE I
PERFORMACE DE DETECÇÃO

False detections 3.0%	Right detection 92.8%	Lost detection 1.2%
--------------------------	--------------------------	------------------------

Tendo em vista a área de segmentação, o algoritmo apresenta poucas perdas, porém o false detections é alto já que as sombras dos carros nos vídeos utilizados são grandes

V. ACKNOWLEDGMENT

Este projeto foi impulsionado pela Universidade de Brasília e apoiado pelo professor Alexandre Zaghetto durante a disciplina Introdução ao Processamento de Imagens.

VI. CONCLUSION

A identificação de automóveis é algo que demanda muito custo de processamento e existe alta dificuldade em se obter o background com precisão eliminando da melhor forma possível os ruídos. A detecção de bordas é outro fator dificultante no processo, pois os veículos fazem sombra e ao tentar detectar a diferença dos frames torna-se difícil separar o que se procura do que se encontra.

Ao utilizar a biblioteca Cython conseguimos eliminar parte dos problemas de velocidade no processamento, juntando essa técnica às operações morfológicas conseguimos obter bons resultados ao separar o background do foreground, sendo que ao implementar as melhorias, conseguimos em muitos casos separar o que é um veículo do que é um conjunto de veículos.

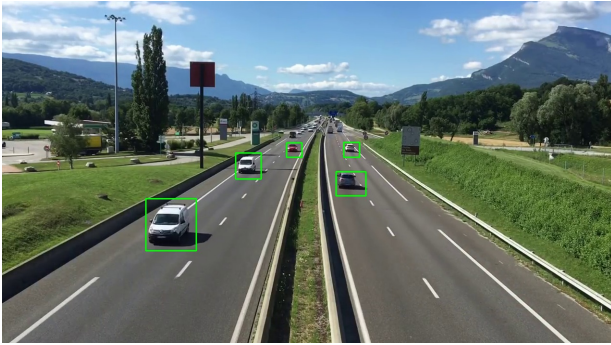


Fig. 6. Identificação dos Carros.

REFERENCES

- [1] Z. Hai-tao, W. Jian, X. Jie and C. Zhi-ming "Self-adaptive Detection of Moving Vehicles in Traffic Video" Proceedings of the 2009 International Symposium on Web Information Systems and Applications (WISA'09) Nanchang, P. R. China, May 22-24, 2009, pp. 449-453
- [2] IEEE Xplore. General Road Detection From a Single Image. Disponível em: <https://ieeexplore.ieee.org/document/5443611>. Acesso em: 22 de novembro de 2018.
- [3] OpenCV. Morphological Transformations. Disponível em: https://docs.opencv.org/3.0-beta/doc/py_tutorials/py_imgproc/py_morphological_ops/py_morphological_ops.html. Acesso em: 23 de novembro de 2018.
- [4] OpenCV. Canny Edge Detection. Disponível em: https://docs.opencv.org/3.1.0/da/d22/tutorial_py_canny.html. Acesso em: 23 de novembro de 2018.