

Técnicas de processamento digital de imagens aplicadas na contagem volumétrica de veículos

Hugo Cabral Tannús¹, Vitor Castro Veloso Soares², Ronaldo Martins da Costa³

Instituto de Informática – Universidade Federal de Goiás

¹hugotannus@inf.ufg.br, ²vitorsoares@inf.ufg.br, ³ronaldocosta@inf.ufg.br

Resumo

Uma das etapas cruciais para as soluções da engenharia de tráfego é a contagem volumétrica de veículos. Neste trabalho, em uma abordagem simplificada, nós apresentamos um método computacional baseado em técnicas de processamento digital de imagens, com o propósito de automatizar a contagem volumétrica a partir de arquivos de vídeo previamente armazenados. O método foi validado em alguns pequenos casos de teste que demonstraram resultados satisfatórios dentro das nossas expectativas com o projeto.

1. Introdução

O estado caótico do trânsito é hoje uma realidade na maioria das grandes cidades do mundo. Este problema pode ser observado principalmente nos engarrafamentos, que trazem uma série de prejuízos, tais como a dificuldade de deslocamento dos usuários comuns e de serviços, o aumento da poluição e o aumento de acidentes, que por sua vez geram vários outros problemas que acabam causando um grande impacto na economia, saúde e na qualidade de vida da população. [2][16][19][23]

Devemos então buscar soluções que resolvam ou ao menos minimizem os problemas do trânsito. Uma destas soluções é o uso da engenharia de tráfego, que aplica princípios da engenharia para resolver problemas de transporte, usando também conhecimentos de psicologia e hábitos dos usuários dos sistemas de transporte. [1]

O principal objetivo da engenharia de tráfego é proporcionar um uso eficiente e seguro da malha viária. Para conseguir atingir este objetivo ela utiliza sistemas de gerenciamento de transportes, que realizam simulações com diferentes focos, para resolver diferentes problemas potenciais ou ainda prever o impacto de possíveis modificações. [17]

Alguns desses problemas são a otimização da velocidade, da fluidez, da capacidade, do acesso, da segurança, do conforto e das informações a um grupo de usuários ou ao sistema, ou buscam diminuir os custos de expansão e manutenção da malha, minimizar a poluição

gerada pelos usuários, criar uma hierarquia viária, entre outras.

De maneira geral, uma etapa crucial para a obtenção de informações importantes sobre o sistema é a contagem volumétrica de usuários, mais especificamente, a contagem volumétrica de veículos. Esta contagem consiste na quantificação de veículos que trafegam por um trecho da malha viária em determinado intervalo de tempo. [6] É a contagem volumétrica que permite, por exemplo, estabelecer um diagnóstico da situação de tráfego atual, para a formulação de alternativas que melhorem o trânsito. [21]

O objetivo deste trabalho é oferecer um apoio às soluções da engenharia de tráfego que são baseadas em simulações, necessitando constantemente da contagem volumétrica de veículos para a construção de modelos mais próximos da realidade.

2. Materiais e métodos

Por se tratar de uma área com inúmeras aplicações, o monitoramento de tráfego vem sendo amplamente estudado. Existem diferentes abordagens na literatura que exploram a detecção e o reconhecimento de veículos, com resultados promissores na área de visão computacional, utilizando vetores de características em conjunto com redes de aprendizado [25], ou ainda realizando a comparação entre objetos processados e objetos de uma base de dados inicial [7]. O método apresentado em [4] propõe melhorias na obtenção de características para uso geral, por meio da otimização da subtração de cenário e uso da diferença simétrica entre frames consecutivos.

Também existem abordagens híbridas, que além da visão computacional buscam outros meios de otimizar o reconhecimento e a detecção, tais como o uso de sensores baseados em laser [10]. Além disso, [20] traz uma densa revisão do campo da detecção de veículos, com o foco em sistemas onde a câmera está localizada no próprio veículo, exibindo e discutindo a direção de diferentes abordagens.

Outra revisão [24], traz um estudo sobre diferentes técnicas de segmentação e seleção de regiões de interesse, técnicas para detecção e reconhecimento de veículos e remoção de sombras, com o foco em abordagens que utilizam câmeras estáticas, voltadas para o monitoramento

ativo do trânsito. Nosso trabalho foi baseado neste tipo de abordagem.

Entre os métodos para a contagem volumétrica, podemos destacar [5], [22], [3] e [14], que trazem abordagens utilizando câmeras de monitoramento estáticas.

Em [5] é feito um rastreamento dos veículos por todo o frame, utilizando características espaciais e temporais para o reconhecimento, com uma contagem simples de cada novo objeto válido encontrado.

Em [22] é feita uma abordagem simplificada com uso específico em rodovias, ainda assim apresentando um módulo proposto para o reconhecimento de veículos a partir de um conjunto de treinamento inicial. O rastreamento de objetos é realizado em todo o frame e os veículos novos são contabilizados, com taxas de acerto entre 94.89% e 96.10%.

Em [3] é feito o rastreamento apenas nos pontos de interesse das vias observadas. Os veículos encontrados são representados com a utilização de um Filtro de Kalman, levando em consideração a velocidade do modelo e a distância estimada do veículo até a câmera. A partir daí é feita uma redução de ruídos e tratamento de oclusões, para então contabilizar corretamente os novos veículos. Os resultados apresentados indicam taxas de acerto superiores a 93% em ambientes um pouco mais complexos.

Em [14] são apresentados grandes avanços no tratamento e solução de oclusões envolvendo vários veículos. O método utiliza modelos tridimensionais deformáveis dos contornos dos veículos para a localização e tratamento das oclusões. Este tratamento é realizado por um módulo individual após a identificação das oclusões, que analisa os segmentos dos contornos e tenta estimar as dimensões dos veículos oclusos, associando a modelos que resolvam a oclusão. A taxa de acerto obtida foi de 100% ao analisar 267 conjuntos de teste com 3074 veículos em ambientes complexos com grande quantidade de oclusões.

Encontramos também algumas propostas de sistemas práticos que realizam a contagem de veículos. Em [15] é apresentado e implementado um sistema de contagem por frames individuais, onde a detecção e contagem são apoiadas pelo usuário. Em [26] é projetado e implementado um sistema de contagem automatizada em tempo real, que monitora também a velocidade dos veículos e demanda uma calibração para cada cenário de uso.

O sistema apresentado em [12] possui muitas características similares ao sistema que estamos desenvolvendo, como o posicionamento da câmera e a atual não verificação dos objetos. Esse sistema utiliza uma estimativa adaptativa de cenário, gerando uma imagem com objetos móveis através da subtração do cenário estimado e da remoção de sombras. O processo de

contagem se dá por meio de “detectores virtuais”, que devem ser calibrados e posicionados sobre cada uma das faixas da via observada. O sistema funciona em tempo real a uma resolução de 320x240 pixels. Os resultados obtidos com essa abordagem indicam taxas de acerto entre 70.31% e 98.39%, com testes realizados em diferentes horários e condições de tempo.

2.1. Escopo do projeto

A ideia inicial do projeto era desenvolver uma ferramenta para realizar unicamente a contagem volumétrica de veículos em tempo real. Porém, devido ao curto espaço de tempo disponível para a concretização do projeto e à grande complexidade de desenvolver um projeto de contagem de propósito geral, decidimos realizar algumas simplificações com o objetivo de viabilizar a implementação e o teste prático de um primeiro algoritmo de contagem. Essas simplificações dizem respeito à aquisição de imagens, ao cenário e aos objetos.

2.1.1. A aquisição de imagens. Deve ser realizada através da leitura de arquivos de vídeo previamente armazenados, pois o algoritmo não possui uma eficiência computacional suficiente para lidar com vídeos em tempo real.

2.1.2. O cenário. Deve ser preferencialmente estático, apesar de o sistema ser capaz lidar razoavelmente bem com movimentações suaves, mas é fundamental que a câmera esteja posicionada de modo que ofereça uma visão superior da via, como será demonstrado na seção de resultados. Esta escolha de cenário tem o propósito de evitar a ocorrência de oclusões e simplificar o processo de segmentação da imagem e tratamento dos objetos.

2.1.3. Os objetos. Foi feita uma generalização dos objetos, ou seja, todos os objetos são considerados veículos. Como estamos utilizando um cenário e vídeos específicos, esta é uma generalização válida, que possibilita um processo de reconhecimento mais simples, sendo necessário apenas rastrear os objetos.

Com essas simplificações, o projeto fica resumido a duas principais atividades, que são a segmentação do vídeo e a contagem dos objetos móveis, que envolve a detecção e o rastreamento dos mesmos. A figura 2.1 mostra o fluxograma do algoritmo que foi desenvolvido.

Utilizamos a biblioteca OpenCV [13] para a manipulação e segmentação da imagem. Para a manipulação, utilizamos as funções básicas da biblioteca que permitem a leitura, o acesso, a escrita e a apresentação das imagens – a escrita e a apresentação

foram utilizadas apenas para auxiliar na realização de testes. A seguir apresentaremos a segmentação e a contagem em maiores detalhes.

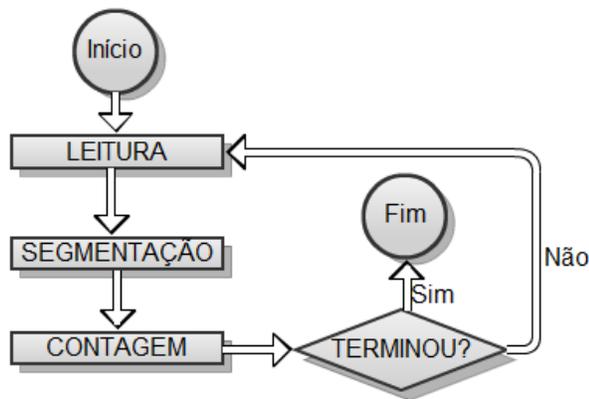


Figura 2.1. Fluxograma do algoritmo

2.2. Segmentação

A segmentação consiste em utilizar técnicas que exploram algumas propriedades da imagem, a fim de subdividir a imagem em regiões ou objetos que a compõem, dependendo do problema a ser resolvido. [9] No nosso caso, queremos subdividir cada frame do vídeo em dois tipos de regiões: cenário e objetos móveis. Uma abordagem simples para a detecção destes objetos é a subtração de uma imagem de cenário do frame atual. O modelo de misturas gaussianas realiza essa detecção de uma maneira mais refinada, mas similar a essa abordagem.

O modelo de misturas gaussianas que utilizamos foi baseado nos trabalhos [11] e [18]. Nesse modelo, cada pixel do cenário é mapeado por uma mistura de k distribuições gaussianas com pesos específicos. Essas distribuições indicam a probabilidade de determinado pixel possuir um valor v em um frame que corresponde à unidade de tempo t . As k distribuições são ordenadas segundo um valor de adaptação e aquelas que possuem maior adaptação são utilizadas para compor um cenário atual, ao mesmo tempo em que indicam os pixels referentes a objetos móveis. Assim, é possível obter um cenário dinamicamente, com uma relativa adaptação a cada frame do vídeo e obter uma segmentação da imagem em cenários e objetos móveis com resultados mais próximos da realidade.

Para a nossa implementação, utilizamos o modelo de misturas gaussianas da biblioteca OpenCV [13] com os parâmetros padrões oferecidos pela biblioteca, sendo a função `cvCreateGaussianBGModel()` responsável por criar o modelo e a função `cvUpdateBGStatModel()` por atualizá-lo a cada novo frame. Ao atualizar o modelo, é gerado um frame de foreground, que diferencia o cenário

dos objetos móveis presentes no frame correspondente. Trata-se de uma imagem em preto e branco, onde preto (0) indica um elemento do cenário e branco (255) indica a presença de um objeto móvel.

2.3. Contagem

Após a segmentação, utilizamos o frame de foreground que foi gerado para auxiliar na contagem. Além disso, utilizamos também uma lista de veículos visíveis e um vetor de pixels que auxiliam também na detecção e rastreamento dos veículos. A quantidade total de veículos encontrados é mantida em uma variável que é atualizada à medida que novos veículos são encontrados.

Dentro do campo da contagem de veículos, existem dispositivos mecânicos intrusivos, que são instalados sob a via e detectam os veículos, os tipos de veículos, a velocidade, entre outras propriedades dependendo do seu modo de funcionamento. Estes dispositivos, em geral, são instalados na forma de linhas que possuem sensores capazes de detectar a presença de um ou mais veículos. [8]

A ideia central do algoritmo de contagem que desenvolvemos é simular virtualmente um desses dispositivos. Para realizar essa simulação, criamos uma linha – vetor de pixels - posicionada na metade do frame, *meio*, onde acontece a detecção e rastreamento dos veículos. O algoritmo 2.1. realiza a contagem de veículos, sendo executado a cada frame do vídeo.

O método `pegaVeiculo(foreground, i)` é responsável por analisar os veículos detectados. Esta etapa consiste em plantar uma semente no pixel `foreground[meio][i]` e realizar o crescimento de região, enquanto os pixels forem correspondentes a um objeto móvel, ou seja, possuírem valor igual a 255. Ao final da execução, o método retorna o menor e maior valor de posições encontradas correspondentes a objetos móveis, que serão utilizadas para representar um veículo.

O método `processaVeiculo(veiculo)` é responsável por realizar o rastreamento dos veículos, para garantir que cada veículo seja contabilizado apenas uma vez. Para isso, percorre-se a lista de veículos visíveis, verificando se existe alguma interseção entre as dimensões do veículo encontrado e as dimensões de algum dos veículos visíveis. Se existir, a lista é atualizada, deixando apenas o veículo com maior dimensão. Senão, o novo veículo é adicionado na lista e a variável com o número total de veículos encontrados é incrementada. Ao final da execução, retorna o veículo encontrado, ou o correspondente de maiores dimensões, caso exista.

Algoritmo 2.1. Contagem de veículos

entrada: Frame *foreground* atual obtido na segmentação do vídeo, a quantidade total de veículos encontrados qv e a lista de veículos visíveis *veics*.

saída: A quantidade total de veículos encontrados qv e a lista de veículos visíveis *veics* atualizados.

variáveis:

view[] – vetor que mapeia o estado visível no frame atual
altura – altura da imagem
meio – posição correspondente à metade do frame
veículo – estrutura que contém as dimensões do veículo, representadas pelos limites *mínimo* e *máximo*
saiu – variável booleana que controla a saída de um veículo da faixa de visão

início

```
para  $i \leftarrow 1$  até altura faça
  se foreground[meio][ $i$ ] = 255 então
    view[ $i$ ]  $\leftarrow$  255
    veículo  $\leftarrow$  pegaVeículo(foreground,  $i$ )
    veículo  $\leftarrow$  processaVeículo(veículo)
    enquanto  $i \leq$  veículo.máximo faça
      view[ $i$ ]  $\leftarrow$  255
       $i \leftarrow i + 1$ 
    fimenquanto
  fimse
  senão
    view[ $i$ ]  $\leftarrow$  0
  fimsenão
fimpara
para veics.inicio até veics.fim faça
  veículo  $\leftarrow$  veics.atual
  saiu  $\leftarrow$  verdadeiro
  para  $i \leftarrow$  veículo.mínimo até veículo.máximo faça
    se view[ $i$ ] = 255 então
      saiu  $\leftarrow$  falso
    fimse
  fimpara
  se saiu  $\leftarrow$  verdadeiro então
    veics.remove(atual)
  fimse
fimpara
fim
```

3. Resultados

O método foi validado utilizando pequenos arquivos de vídeo que foram previamente armazenados. Os vídeos foram gravados com resolução de 1280x720 pixels, com taxa de 30 frames por segundo e duração variando entre 20 segundos e 5 minutos. A figura 3.1 mostra um exemplo

de um frame, ilustrando o cenário onde o algoritmo é aplicável.



Figura 3.1. Exemplo de frame utilizado nos testes

Utilizamos uma interface de visualização para testar o resultado da segmentação e o processamento parcial do vídeo. A figura 3.2 mostra um frame de foreground, demonstrando o resultado da segmentação, e a figura 3.3 mostra o mesmo frame correspondente processado, com os contornos desenhados na cor verde ao redor dos objetos encontrados.

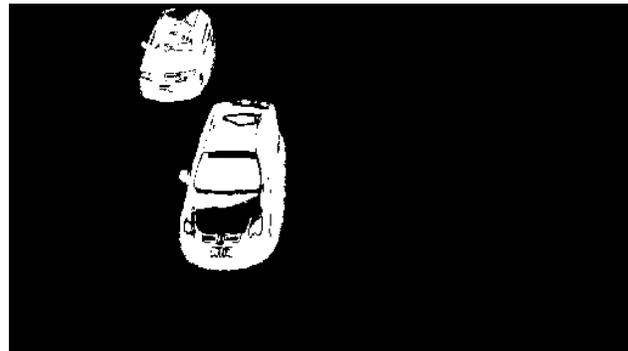


Figura 3.2. Exemplo de frame de foreground

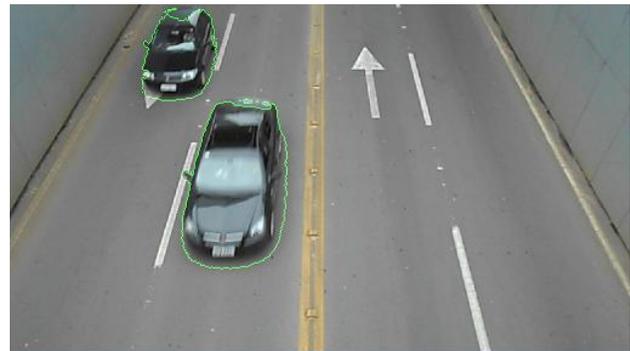


Figura 3.3. Exemplo de frame processado

Os resultados obtidos com os testes podem ser observados na tabela 3.1. A coluna *Caso* apresenta o nome do arquivo utilizado; a coluna *Qm* apresenta a quantidade de veículos obtida pela contagem manual; a

coluna Qa apresenta a quantidade de veículos retornada pelo algoritmo; a coluna $D(s)$ apresenta a duração do vídeo, em segundos e a coluna $P(s)$ apresenta o tempo gasto no processamento do vídeo, em segundos.

Caso	Qm	Qa	D(s)	P(s)
t001.avi	32	32	148	300
t002.avi	16	16	56	114
t003.avi	75	75	300	600
t004.avi	19	19	81	164
t005.avi	5	5	20	44

Tabela 3.1. **Resultados obtidos**

4. Conclusão

Apresentamos um novo método para a contagem de veículos utilizando uma abordagem simplificada. Os resultados obtidos foram promissores, possibilitando uma utilização prática do método dentro do contexto proposto. Algumas experiências com vídeos de menor resolução mostraram ganhos significativos em desempenho de processamento, sem perdas de eficiência na contagem, e podem ser mais exploradas futuramente.

Porém, para o desenvolvimento de um método de contagem de propósito mais geral, devemos realizar otimizações mais complexas baseadas nos trabalhos que revisamos, a fim de cobrir as simplificações que foram feitas inicialmente. Uma etapa fundamental nessa direção é o desenvolvimento do reconhecimento e validação dos veículos, classificando os objetos móveis encontrados. Outras possíveis otimizações são o tratamento de oclusões e o aperfeiçoamento das técnicas de segmentação, rastreamento e contagem que foram apresentadas.

Esperamos continuar o trabalho, com o foco inicial na otimização dos códigos já desenvolvidos e em estudos direcionados a técnicas de detecção e reconhecimento de veículos e técnicas de segmentação mais sofisticadas.

5. Referências

[1] ADOT: Arizona Department Of Transportation, I.T. D. *Traffic Engineering - What? Why? How?*, Arizona Highway Department, 1970.

[2] Arnott, R. e Small, K. A., *The Economics of Traffic Congestion*, American Scientist, Vol. 82, No. 5, pp. 446-455, 1994.

[3] Bas, E., Tekalp, M. e Salman, F. S., *Automatic vehicle counting from video for traffic flow analysis*, 2007 IEEE Intelligent Vehicles Symposium, pp. 392-397, 2007.

[4] Cao, J., Wang, W. e Liang, Y., *An improved detection algorithm of moving vehicles based on computer vision*, IEEE International Conference on Intelligent Computing and Intelligent Systems 2009, ICIS 2009, pp. 20-24, 2009.

[5] Chen, T., Lin, Y., Chen, T., *Intelligent vehicle counting method based on blob analysis in traffic surveillance*, 2007 Second International Conference on Innovative Computing, Information and Control, ICICIC'07, 2007.

[6] DNIT, *Plano Nacional de Contagem de Trânsito*, Disponível em: http://www1.dnit.gov.br/rodovias/contagem/plano_contagem.htm, Acessado em: 04/11/2011.

[7] Du, Y., Liu, C. e Shi, W., *A real-time vehicle recognition method based on video sequence images*, The Ninth International Conference on Electronic Measurement & Instruments, ICEMI'2009, 2009.

[8] Goldner, L. G., *Engenharia de Tráfego – 1º Módulo*, Material didático, Departamento de Engenharia Civil, Universidade Federal de Santa Catarina, Santa Catarina, 2008.

[9] Gonzalez, R. C. e Woods, R. E., *Digital Image Processing, 2/E*, Prentice Hall, New Jersey, 2002.

[10] Huang, L. e Barth, M., *Tightly-coupled LIDAR and computer vision integration for vehicle detection*, 2009 IEEE Intelligent Vehicles Symposium, pp. 604-609, 2009.

[11] KadewTraKuPong, P. e Bowden, R., *An improved adaptive background mixture model for real-time tracking with shadow detection*, Proceedings 2nd European Workshop on Advanced Video-Based Surveillance Systems, 2001.

[12] Lei, M., Lefloch, D., Gouton, P., Madani, K., *A video-based real-time vehicle counting system using adaptive background method*, Proceedings 2008 IEEE International Conference on Signal Image Technology and Internet Based Systems, SITIS'08, pp. 523-528, 2008.

[13] OpenCV: Open Source Computer Vision. Disponível em: <http://opencv.willowgarage.com/wiki/> Acessado em 04/11/2011

[14] Pang, C. C. C., Lam, W. W. L. e Yung, N. H. C., *A method for vehicle count in the presence of multiple-vehicle occlusions in traffic images*, IEEE Transactions on Intelligent Transportation Systems 8, pp. 441-459, Tokyo, 2007.

[15] Pornpanomchai, C., Liamsanguan, T. e Vannakosit, V., *Vehicle detection and counting from a video frame*, International Conference on Wavelet Analysis and Pattern Recognition 2008, ICWAPR '08, pp. 356-361, 2008.

[16] Resende, P. T. e de Sousa, P. R., “Mobilidade urbana nas grandes cidades brasileiras: um estudo sobre os impactos do congestionamento”, Anais do XII Simpósio de Administração da Produção, Logística e Operações Internacionais, São Paulo, 2009.

[17] Roess, R. P.; Prassas, E. S.; McShane, W. R., *Traffic Engineering, 3/E*, Pearson Prentice Hall, New Jersey, 2004.

[18] Stauffer, C. e Grimson, W. E. L., *Adaptive background mixture models for real-time tracking*, Proceedings 1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No PR00149), IEEE Comput. Soc. Part vol. 2, 1999.

[19] Stopher, P. R., *Reducing road congestion: a reality check*, Transport Policy, Volume 11, Issue 2, pp. 117-131, Elsevier, 2004.

[20] Sun, Z., Bebis, G. e Miller, R., *On-road vehicle detection: A review*, IEEE Transactions on Pattern Analysis and Machine Intelligence 28, pp. 694-711, 2006.

[21] Szasz, P. A. e Pereira, A. L. S., “NT 066/81: Métodos para contagem volumétrica abreviada”, CET, São Paulo, 1981.

[22] Tamersoy, B., Aggarwal, J. K., *Counting vehicles in highway surveillance videos*, 2010 20th International Conference on Pattern Recognition, ICPR, pp. 3631-3635, 2010.

[23] Vasconcelos, E. A., *Circular é preciso, viver não é preciso*, Annablume: FAPESP, São Paulo, 2003.

[24] Wang, G., Xiao, D. e Gu, J., *Review on vehicle detection based on video for traffic surveillance*, 2008 IEEE International Conference on Automation and Logistics, ICAL 2008, pp. 2961-2966, Qingdao, 2008.

[25] Wang, W., *Reach on Sobel operator for vehicle recognition*, International Joint Conference on Artificial Intelligence 2009, ICAI '09, 2009.

[26] Wu, J. e Gu, C., *The design and implementation of real-time automatic vehicle detection and counting system*, International Conference on Information Engineering and Computer Science 2009, ICIECS 2009, pp. 1-4, 2009.