



Rastreamento e contagem de pedestre em tempo real por meio de imagens digitais

Alexssandro Ferreira Cordeiro
Bolsista

Centro Latino-americano de Tecnologias Abertas (CELTAB) Centro Latino-americano de Tecnologias Abertas (CELTAB)
Foz do Iguaçu, Paraná
Email: alexssandrofc@gmail.com

Cristian A. U. Ojeda
Bolsista

Centro Latino-americano de Tecnologias Abertas (CELTAB)
Foz do Iguaçu, Paraná
Email: urunagacrsthian@gmail.com

Pedro L. P. Filho
Departamento de Computação

Universidade Tecnológica Federal do Paraná (UTFPR) Centro Latino-americano de Tecnologias Abertas (CELTAB)
Medianeira, Paraná
plpf2004@gmail.com

Gustavo R. Valiati
Departamento de Computação

Centro Latino-americano de Tecnologias Abertas (CELTAB)
Foz do Iguaçu, Paraná
gustavo.valiati@pti.org.br

Resumo—Este trabalho apresenta o uso de dois modelos neurais trabalhando em conjunto a fim de efetuar a tarefa de detecção de pedestres, rastreamento e contagem por meio de imagens digitais. A ideia é poder mensurar a quantidade de pessoas em um determinado local. Foi utilizado uma rede de classificação para efetuar a detecção dos pedestre e repassado as coordenadas das detecções para uma rede de rastreamento, possibilitando definir sua trajetória e assim efetuar a contagem referente a entrada ou saída do pedestre de um determinado local.

Keywords—Rastreamento; Contagem de pedestre; Detecção pedestre; Redes convolucionais; Deep Learning;

I. INTRODUÇÃO

Atualmente, as aplicações baseadas na detecção e contagem de pessoas em um espaço representam grande interesse em diferentes aplicações [11]. A estimativa do número de pessoas presentes em uma área pode ser uma informação extremamente útil por várias razões, dentre elas segurança e otimização, como uma mudança anômala no número de pessoas, podendo ser a causa ou o efeito de um evento perigoso, ou a otimização do cronograma de um sistema de transporte público com base no número de passageiros. Assim, vários trabalhos nas áreas de análise de vídeo e vigilância por vídeo inteligente abordaram essas tarefas [1] [2] [3] [4].

A contagem de pessoas é uma das principais técnicas de vigilância por vídeo. Essa tarefa geralmente encontra muitos desafios em ambientes populosos, como oclusão pesada, baixa resolução, variabilidade de ponto de vista de imagem [12]. Motivado pelo sucesso de redes neurais convolucionais na detecção de objetos [10], este trabalho busca a implementação de um aplicativo para contagem de pessoas por meio da detecção e rastreamento, executando a fase de detecção a cada N quadros, possibilitando rastrear o objeto até alcançar o N-ésimo quadro, desta forma se tem

um ciclo de detecção e rastreamento de objeto otimizando a carga computacional.

II. MATERIAIS E MÉTODOS

Em análise ao problema proposto, o qual deseja mensurar a quantidade de pessoas em um determinado local por meio de imagens, foram levantados os requisitos bem como as tecnologias compatíveis a fim de desenvolver um software capaz de solucionar o problema. A abordagem se deu na divisão do problema em três etapas. Primeiramente foi necessário definir a metodologia de detecções de pedestres em uma imagem digital, após isso, efetuar o rastreamento e por fim a contagem dos pedestres conforme demonstrado no fluxograma da Figura 1.

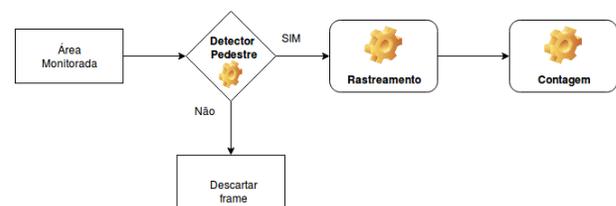


Figura 1. Ilustração do fluxo do processo de detecção, rastreamento e contagem de pedestres por frames.

Conforme demonstrado no fluxograma da Figura 1, a rede neural responsável pela detecção ficará analisando as imagens enviadas à rede, caso houver detecções e forem pertencentes a classe *person* (pedestre), os dados referentes as coordenadas dos pedestres detectados serão enviados para a rede neural responsável pelo rastreamento, a qual irá mapear os movimentos dos pedestres de forma individual, desta maneira permitindo a contagem referente a entrada ou saída de um determinado local.

Para o desenvolvimento, validação e testes do sistema, foi

utilizado uma câmara IP de vigilância modelo Axis 210¹ e um notebook com um processador Intel Core I7-6700HQ, 16 GB de memória RAM DDR3, 250GB de armazenamento em disco e uma placa de vídeo NVIDIA 960M com 640 CUDA Cores e 4GB GDDR5.

Os tópicos a seguir apresentam as três etapas que constituem a solução. O tópico Detecção de pedestre apresenta a metodologia utilizada para efetuar a classificação dos objetos detectados pela rede neural a fim de obter somente a classe alvo. O tópico Rastreamento de pedestre apresenta a metodologia utilizada para efetuar o rastreamento dos pedestres detectados pela etapa anterior e o tópico Contagem de pedestres descreve como é efetuado o processo de contagem utilizando os dados recebidos da etapa de rastreamento.

A. Detecção de pedestre

O processo de detecção de pedestres ficou a cargo da rede YOLO (You-Only-Look-Once) [5], que utiliza a rede Darknet-53 como backbone. O YOLO, na sua terceira versão (YOLOv3), apresenta resultados competitivos com trabalhos no estado da arte de desafios como o COCO [13].

YOLO é uma rede neural pré treinada capaz de detectar mais de 200 classes e prever mais de 9000 categorias de objetos conforme [6], ela permite obter acurácia e desempenho em tempo real, desta maneira permitindo seu uso no projeto, pois uma das classes de conhecimento da rede é a classe *person*, classe alvo deste projeto.

De acordo com [7], a rede YOLO realiza a detecção por meio do uso de *Bouding Boxes* (BB), caixas delimitadoras que são espalhadas pela imagem a fim de obter características locais do objeto a ser reconhecido, assim obtendo probabilidade de tais características pertencerem as classes informadas previamente no treinamento, conforme demonstrado na Figura 2. Com o uso dos BB a YOLO é capaz de detectar múltiplos objetos em uma imagem.

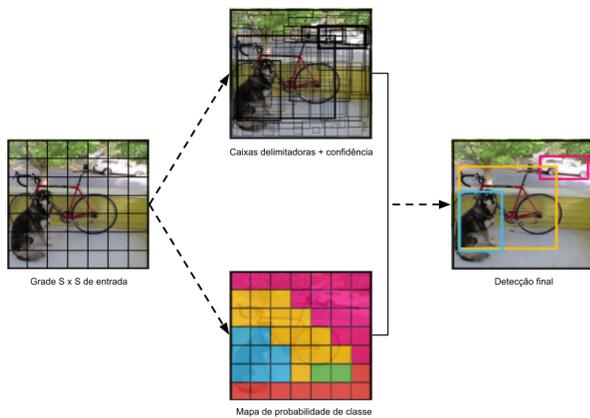


Figura 2. Demonstração do funcionamento da rede Yolo com o uso de bounding box. Fonte [7]

¹<https://www.axis.com/pt-br/products/axis-210>

A rede YOLO divide as imagens de entrada em uma grade $S \times S$, onde cada grade prediz M BB, e cada BB consiste de 5 predições, sendo x , y , w , h e a confiança dada pela intersecção sobre a união (IOU) conforme demonstrado na Equação 1, onde as coordenada (x,y) representam o centro do BB em relação aos limites da grade, a largura e altura (w,h) em relação a imagem inteira e a confiança representada entre a caixa de predição da rede e qualquer caixa definida como correta passada para o treino (*ground truth*), vale destacar que o treinamento de rede YOLO é supervisionado.

$$IOU = \frac{area(B_p \cap B_{(gt)})}{area(B_p \cup B_{(gt)})} \quad (1)$$

A YOLO é compatível com a tecnologia CUDA² proprietária da NVIDIA³ para computação massivamente paralela, essa tecnologia visa aumentar o processamento paralelo, proporcionando uma significativa queda no tempo de treinamento e utilização da rede por meio de Unidade de Processamento Gráfico (GPU).

Para exibição das imagens/vídeos processadas pelo YOLO, utilizou-se a biblioteca Open Source Computer Vision Library (OpenCV)⁴, o qual provê uma infraestrutura para aplicações de visão computacional, facilitando o tratamento das imagens.

No estudo foi utilizada a rede YOLO padrão versão 3 acelerada com CUDA versão 10 e utilizando a infraestrutura do OpenCV versão 4. A linguagem de programação utilizada para o desenvolvimento da solução foi o Python na versão 3.7.

Após o processamento, a rede YOLO devolve a localização do elemento detectado na imagem, bem como a probabilidade deste elemento pertencer a uma determinada classe de conhecimento da rede. Para o problema proposto, qualquer detecção não correspondente a classe *person* é ignorada, caso contrário, as coordenadas do BB do pedestre detectado são repassadas para a rede de rastreamento.

B. Rastreamento de pedestre

A rede de rastreamento utilizada é a proposta por [9], onde utiliza-se uma abordagem por meio da aprendizagem de filtros de correlação discriminativos, utilizando uma estimativa de escala genérica, assim possibilitando sua aplicação em diversos métodos de rastreamento.

Conforme demonstrado na Figura 3, a rede necessita de uma informação de entrada, o qual será fornecida pela detecção do YOLO. Com base nas coordenadas do BB é possível a rede efetuar o rastreamento de múltiplos pedestres.

Para cada detecção será criado um identificador, desta forma é possível rastrear cada pedestre de forma independente e permitindo o rastreamento de múltiplos pedestres concomitantemente.

²<https://developer.nvidia.com/cuda-zone>

³<https://www.nvidia.com/>

⁴<https://opencv.org/>

Entrada :
 Imagem I_t
 Posicao previa do alvo S_{t-1} e escala S_{t-1}
 Translacao do modelo A_{t-1}^{trans} , B_{t-1}^{trans} e a escala do modelo A_{t-1}^{escala} , B_{t-1}^{escala}

Saída :
 Estimar a posicao do alvo P_t e escala S_t
 Atualizar a translacao do modelo A_t^{trans} , B_t^{trans} e a escala do modelo A_t^{escala} , B_t^{escala}

Figura 3. (Adaptado) “Algoritmo” do funcionamento da rede de rastreamento. Fonte [9]

Para auxiliar no desenvolvimento do software foi utilizado a biblioteca DLIB⁵, o qual implementa o algoritmo de rastreamento por filtros de correlação, evitando trabalho em duplicidade e acrescentando agilidade no desenvolvimento.

C. Contagem de pedestres

A contagem de pedestre elaborada neste trabalho tem como objetivo mensurar a quantidade de pessoas em um determinado local, logo a ideia é identificar quantas pessoas entraram ou saíram.

A vantagem de se utilizar uma contagem de pessoas por meio de imagens, é a possibilidade de efetuá-la sem a necessidade de interação com as pessoas. Logo, o uso de sensores, catracas e outros dispositivos os quais interagem com os usuários e alteram o fluxo está descartada. Desta maneira foi posicionada a câmera na porta, permitindo um fluxo controlando de pessoas de forma natural, prevenindo oclusões e permitindo melhora na acurácia e desempenho do software.

Para definir qual ação está acontecendo durante a filmagem (entrada ou saída do pedestre), criou-se uma linha vertical no meio da imagem conforme Figura 4.



Figura 4. Demonstração do cenário utilizado na contagem de pedestres.

Na primeira detecção efetuada do pedestre, o software obtém o local de origem (esquerda ou direita), assim caso ultrapasse a linha é possível saber sua ação (entrada ou saída). A Figura 5 apresenta o fluxograma da contagem de pedestre efetuada pelo software.

Conforme demonstrado no fluxograma da Figura 5, ao tocar na linha o software efetua a atualização da origem do pedestre e a contagem com base na sua origem anterior, assim, caso não saia completamente da imagem e deseja

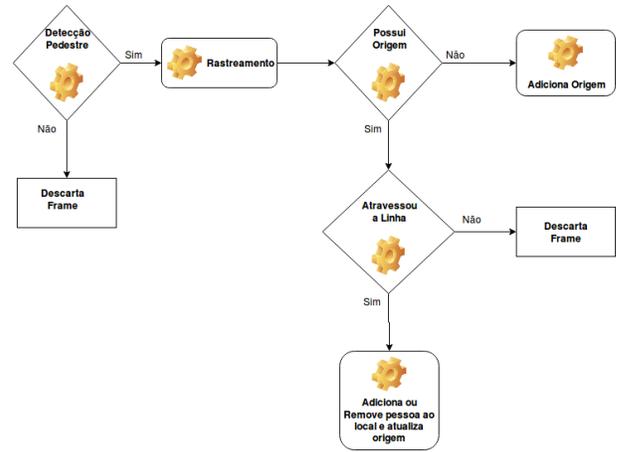


Figura 5. Fluxograma da contagem de pedestres.

retornar será contado novamente e o ciclo se repetirá até que saia completamente da imagem.

Caso o pedestre seja detectado, porém, retorne sem atravessar a linha, o software não irá computar nenhuma ação, pois nenhum pedestre atravessou a linha.

III. RESULTADOS PRELIMINARES

Os resultados preliminares dos testes sem oclusão dos pedestres foram satisfatórios, efetuando a contagem de todos os pedestres que passaram na área monitorada conforme pode ser observado na Figura 6.



Figura 6. Demonstração da contagem de pedestres.

Foram detectados problemas quando há oclusões de pedestres, em momentos que há multidões, ocorre uma perda na taxa de detecção de aproximadamente 50% conforme demonstrado na Figura 7, onde a rede YOLO apresenta falhas na detecção de pedestres com oclusão e consequentemente efetua marcações incorretas de BB, por sua vez esse erro é repassado para a rede de rastreamento, ocasionando falhas no rastreamento dos pedestres, assim não permitindo mensurar com exatidão a quantidade de pessoas em um determinado local. Acredita-se que é possível melhorar a acurácia por meio de ajustes no posicionamento da câmera, a fim de evitar ou suavizar as oclusões.

⁵<http://dlib.net/>

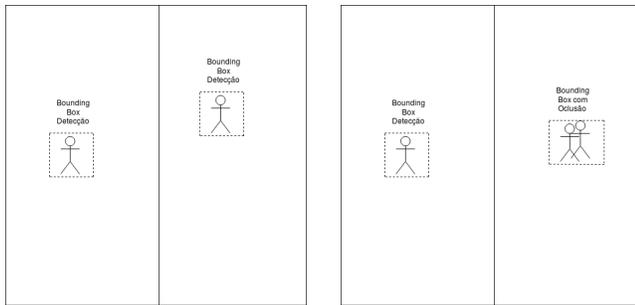


Figura 7. Imagem a esquerda demonstra detecções de pessoas onde não há oclusão, apresentando de forma eficaz as marcações dos BB nas detecções das pessoas e permitindo o rastreamento de forma correta. A imagem a esquerda apresenta momentos em que há oclusão de pessoas nas imagens, a YOLO começa a efetuar marcações incorretas, marcando 2 pessoas como uma detecção somente (um BB), assim, afetando a acurácia da rede de rastreamento.

Para aliviar a carga de processamento no computador, foi definido após testes, que somente será processado 1 frame a cada 6, em um sistema de captação de imagens de 30 Frames Por Segundo (FPS), o qual percebeu-se que não afetou a acurácia do sistema e evitando sobrecarga no processador e nos CUDA cores da GPU.

IV. CONCLUSÃO

Os Resultados preliminares foram satisfatórios, porém, ainda é necessário efetuar ajustes no sistema de aquisição de imagens bem como uma análise de possíveis correções no processamento das imagens, as quais servem de entrada na rede neural a fim evitar ou suavizar os problemas relacionados a oclusão, possibilitando uma melhora na acurácia.

O desempenho apresentado pela rede em tempo real foi satisfatório, não apresentando travamento e ocupando uma carga de 75% de processamento em média dos CUDA cores e 70% de um dos 8 núcleos do processador, porém acredita-se que é possível melhorar ainda mais o desempenho com estudos em torno do sistema de aquisição de imagens e da forma como está o processamento das imagens.

AGRADECIMENTOS

Gostaríamos de agradecer ao programa de Mestrado da Universidade Tecnológica Federal do Paraná (UTFPR) campus Medianeira, ao Parque Tecnológico Itaipu (PTI) e ao Centro Latino-americano de Tecnologias Abertas (CELTAB) pelo apoio ao projeto.

REFERÊNCIAS

[1] Xu, Zheng and Hu, Chuanping and Mei, Lin, *Video structured description technology based intelligence analysis of surveillance videos for public security applications*, Multimedia Tools and Applications, p. 12155-12172, V. 75, 2016. Disponível em <<https://link.springer.com/article/10.1007/s11042-015-3112-5>>

[2] O. Sidla and Y. Lypetsky and N. Brandle and S. Seer, *Pedestrian Detection and Tracking for Counting Applications in Crowded Situations*, IEEE International Conference on Video and Signal Based Surveillance, p. 70, 2006. Disponível em <<https://ieeexplore.ieee.org/document/4020729>>

[3] A. Broggi and P. Cerri and S. Ghidoni and P. Grisleri and H. G. Jung, *A New Approach to Urban Pedestrian Detection for Automatic Braking*, IEEE Transactions on Intelligent Transportation Systems, p. 594-605, V. 10, 2009. Disponível em <<https://ieeexplore.ieee.org/document/5290131>>

[4] A. Velastin and B. A. Boghossian and B. P. L. Lo and Jie Sun and M. A. Vicencio-Silva, *PRISMATICA: toward ambient intelligence in public transport environments*, IEEE Transactions on Systems, p. 164-182, V. 35, 2005. Disponível em <<https://ieeexplore.ieee.org/document/1369353>>

[5] Joseph Redmon and Ali Farhadi, *YOLOv3: An Incremental Improvement*, CoRR, Vol. abs/1804.02767, 2018. Disponível em <<https://arxiv.org/abs/1804.02767>>

[6] Joseph Redmon and Ali Farhadi, *YOLO9000: Better, Faster, Stronger*, CoRR, Vol. abs/1612.08242, 2016. Disponível em <<http://arxiv.org/abs/1612.08242>>

[7] Joseph Redmon and Ali Farhadi, *You Only Look Once: Unified, Real-Time Object Detection*, CoRR, Vol. abs/1506.02640, 2016. Disponível em <<http://arxiv.org/abs/1506.02640>>

[8] Wang, Naiyan and Yeung, Dit-Yan *Learning a deep compact image representation for visual tracking*.

[9] Martin Danelljan and Gustav Hager and Fahad Shahbaz Khan and Michael Felsberg, *Accurate Scale Estimation for Robust Visual Tracking*, British Machine Vision Conference, 2014. Disponível em <<http://www.bmva.org/bmvc/2014/files/paper038.pdf>>

[10] Girshick, Ross and Donahue, Jeff and Darrell, Trevor and Malik, Jitendra *Rich feature hierarchies for accurate object detection and semantic segmentation*.

[11] Dollar, Piotr and Wojek, Christian and Schiele, Bernt and Perona, Pietro. *Pedestrian detection: An evaluation of the state of the art*.

[12] Wang, Naiyan and Yeung, Dit-Yan. *Learning a deep compact image representation for visual tracking*.

[13] Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., & Zitnick, C. L. (2014, September). *Microsoft coco: Common objects in context*. In *European conference on computer vision* (pp. 740-755). Springer, Cham.