

Modelo estimativo do fluxo de embarque e desembarque de passageiros em sistemas de monitoramento metroferroviário com base em técnicas de visão computacional

Wanderson Gomes de Souza, Iguatemi E. da Fonseca, Francisco A. Belo

Resumo - A falta de monitoramento adequado do fluxo de usuários nos sistemas de transporte público metroviários, pode comprometer à segurança e o bem-estar dos passageiros. Nesse artigo, é proposto um modelo estimativo em tempo real, automático e contínuo, do fluxo de passageiros do sistema metroviário da CBTU (Companhia Brasileira de Trens Urbanos) de Recife, utilizando técnicas de visão computacional. Para este fim, foi proposto um modelo que envolve análise e reconhecimento de padrões de cabeças e ombros de passageiros. Pretende-se, em uma implementação futura, detectar e prever pontos críticos de multidões, de forma a garantir a segurança, gerar prognósticos para o planejamento estratégico operacional e evitar futuras aglomerações.

Palavras-Chave – Visão Computacional, Modelo Estimativo de Passageiros, Trens Urbanos, Reconhecimento de Padrões.

Abstract— The lack of adequate monitoring of the flow of users in subway-railroad transport systems, can compromise the safety and well being of passengers. In this paper, an automatic and continuous real-time estimation model of the passenger flow of the CBTU (Companhia Brasileira de Trens Urbanos) metro system of Recife, using computational vision techniques, is proposed. Towards this end, a model has been proposed that involves analysis and recognition of patterns of heads and shoulders of passengers. It is intended, in a future implementation, to detect and predict critical points of crowds in order to ensure safety, generate forecasts for strategic operational planning and avoid future agglomerations.

Keywords— Computer Vision, Passenger Estimate Model, Urbans trains, Pattern Recognition.

I. INTRODUÇÃO

Atualmente o número de pessoas que vivem nas grandes cidades está aumentando em ritmo acelerado. Segundo os dados das Nações Unidas [1], espera-se que até 2050 a população mundial, que residem nas cidades, cresça 66%. Nesse mesmo ritmo, os serviços de transporte público necessitam de mudanças para suprir a alta demanda de passageiros, em especial os sistemas de transporte metroferroviário. Diante desta realidade, faz-se necessário implementar métodos confiáveis que visem a estimar a quantidade e monitorar o fluxo de usuários, de modo a intensificar ações preventivas contra acidentes, garantir conforto aos passageiros e o pleno funcionamento dos trens. Diversos trabalhos têm-se desenvolvido utilizando métodos para estimar contagem de passageiros, tais como: rede de sensores wireless

[2], sensores infravermelho [3] ou RFID de longo alcance [4]. No entanto, câmeras convencionais de segurança se tornaram mais acessíveis e o desenvolvimento de novas técnicas de visão computacional para detecção de passageiros receberam uma atenção significativa nos últimos anos.

Recentemente, muitos métodos de processamento de imagem baseados em contagem de pessoas em sistemas de transporte público foram propostos. Cong *et al.* [5] propuseram um algoritmo baseado na estimativa do fluxo de velocidade para contar o número de pedestres que atravessam uma linha de detecção ou uma região específica. Hejin [6] apresentou um método para estimar o fluxo de passageiros de ônibus baseado em trajetória de *clustering*. Mukherjee *et al.* [7] desenvolveram um *framework* para contagem de passageiros em uma estação ferroviária. O *framework* possui três componentes: detecção de pessoas, rastreamento e validação.

Lengvenis *et al.* [8] apresentaram um sistema de contagem de passageiros baseado em visão computacional. Dentre os quatro algoritmos desenvolvidos para estimar a quantidade de passageiros, o algoritmo de simulação de barreiras por zonas (ABSZ) obteve menores taxas de falso positivos e se mostrou eficaz na contagem de pessoas. Dewei *et al.* [9] implementaram um sistema capaz de estimar o fluxo de passageiros em estações metroviárias usando a convolução por filtro de Sobel. Yan-Yan *et al.* [10] apresentaram um método para contagem automática de usuários do sistema de metrô usando câmeras de monitoramento. Neste artigo, o passageiro é detectado e reconhecido utilizando o detector de característica *Haar-like* e o classificador *AdaBoost*. Zhang *et al.* [11] propuseram uma arquitetura de Rede Neural Convolutiva de Múltiplas Colunas (MCNN) simples e eficaz para mapear a densidade de multidões. Zhao *et al.* [12] implementaram uma rede neural convolutiva profunda para contar pessoas que cruzam uma linha de interesse (LOI). O modelo CNN proposto decompõe o problema inicial em dois sub-problemas mais simples: estimativa do mapa de densidade e velocidade da multidão. Essa decomposição visa facilitar a aprendizagem de representações com características mais discriminativas. Li *et al.* [13] apresentaram um *framework* unificado para detecção e classificação simultânea de pedestres e ciclistas em veículos inteligentes. O método de detecção proposto é baseado no modelo UB-MPR. Os resultados experimentais apontam que o método proposto atinge mais de 10% de melhorias em relação aos métodos baseados em *Fast R-CNN*. Escolano *et al.* [14] desenvolveram um sistema de visão computacional para contagem de passageiros de ônibus usando

Wanderson Gomes de Souza, Francisco Antônio Belo (Programa de Pós-Graduação em Engenharia Mecânica) e Iguatemi E. da Fonseca (Centro de Informática), Universidade Federal da Paraíba (UFPB), João Pessoa-PB, Brasil, E-mails: wandersonsouza.info@gmail.com, belo@pq.cnpq.br e iguatemi@ci.ufpb.br.

fluxo ótico. Esses dados são transmitidos para uma central de despachos, em que um controlador de lógica fuzzy determina o horário do envio de um novo ônibus.

O objetivo principal desse trabalho é propor um modelo estimativo em tempo real, de forma automática e contínua do fluxo de passageiros do sistema metroviário da Companhia Brasileira de Trens Urbanos (CBTU) - Recife, reunindo diversas técnicas de visão computacional, tais como: Características *Haar-like*, Normalização de amostras, Imagem integral, algoritmo de aprendizagem de máquina *AdaBoost* e Cascata de classificadores.

II. VISÃO GERAL DO SISTEMA

O sistema aqui proposto é dividido em duas etapas: treinamento e reconhecimento/contagem. O fluxograma deste processo é mostrado na Figura 1.

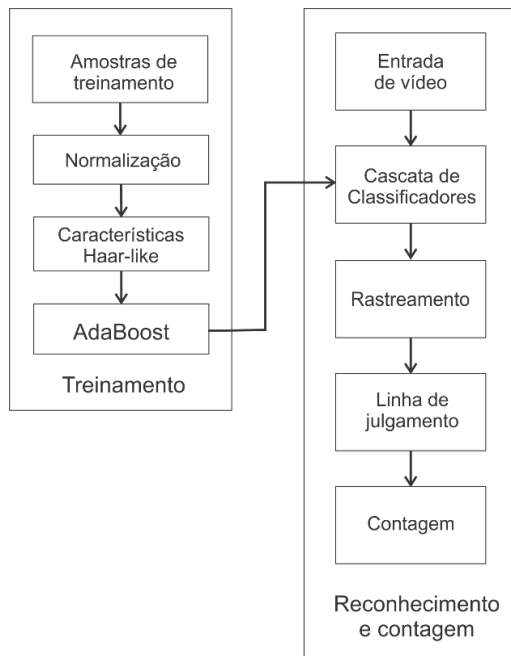


Fig. 1 - Fluxograma de detecção de passageiros

Na etapa de treinamento serão colhidas as amostras de treinamento positivas e negativas. O processo de normalização visa padronizar o tamanho das amostras, recorta-las e convertê-las na escala de cinza. A etapa de características *haar-like* é responsável por extrair informações relevantes de acordo com o formato do objeto a ser detectado. Logo após a extração das características, o algoritmo de aprendizagem *AdaBoost* [15] foi escolhido, devido a sua execução rápida, para treinar a cascata de classificadores.

O principal objetivo da segunda etapa é reconhecer características e estimar a contagem dos passageiros. Dado o fluxo de entrada de vídeo, é iniciada imediatamente a cascata de classificadores, cuja função é detectar a presença de características de objetos previamente treinados e descartar as informações irrelevantes. Após a etapa de classificação, o próximo passo é rastrear os objetos detectados em cena. E, finalmente, os passageiros são contabilizados ao cruzarem a linha de julgamento.

III. MÉTODOS UTILIZADOS

Nesta seção, serão apresentados os conceitos desenvolvidos por [16], que servirão como base para a construção do sistema aqui proposto. Este método possui três principais contribuições

que o torna computacionalmente eficiente, tais como: a imagem integral, o algoritmo de aprendizagem *AdaBoost* e a combinação de classificadores fracos baseado em características *Haar-like*, dando ênfase às regiões mais promissoras da imagem.

Para facilitar a compreensão, subdividimos o sistema em duas etapas: treinamento e reconhecimento/contagem.

A. Treinamento

Para iniciar o processo de treinamento do classificador, é necessário preparar um conjunto de amostras contendo objetos dos quais se desejam detectar (imagens positivas) e amostras contendo tudo o que não se deseja detectar (imagens negativas).

Afim de se reduzir o esforço computacional, a normalização é uma importante fase do pré-processamento que visa eliminar os efeitos ou sinais da imagem que não são relevantes ou que retratem particularidades indesejadas inerentes a todo o processo imperfeito de aquisição de dados. Essas variabilidades são geradas mediante o processo impreciso de aquisição ou por limitações dos instrumentos de captura de imagem utilizados, afirma [17].

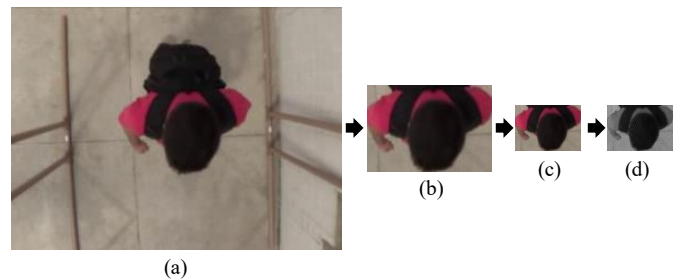


Fig. 2 - Normalização de imagens: (a) imagem original, (b) recorte, (c) escala e (d) descoloração

No contexto deste trabalho, o processo de normalização será realizado em três etapas: recorte, escala e descoloração. A etapa de recorte visa preservar apenas informações e características relevantes da imagem; a etapa de escala, é responsável pela redução e padronização do tamanho das amostras e, a descoloração, contribui para redução do esforço computacional substituindo o sistema que trabalha com três canais de cores (RGB), por uma única camada em escala de cinza. A Figura 2 ilustra o processo de normalização.

O próximo passo é criar uma rotina para detecção baseada nas principais características de um objeto. Tais características representam diferenças de intensidade entre áreas da imagem. O valor correspondente à característica de *haar* é obtida subtraindo o somatório da intensidade dos pixels da região escura do total acumulado pela intensidade dos pixels da região clara. Este cálculo poderá ser realizado utilizando a representação de imagem integral que será abordado na equação (1).

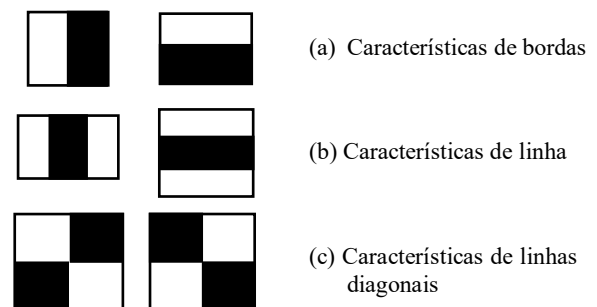


Fig. 3 - Características básicas de Haar.

As características de *haar* da Figura 3 (a) são responsáveis por detectar bordas verticais e horizontais, enquanto que na Figura 3 (b) o formato possibilita a detecção de características em linhas verticais e horizontais, na Figura 3 (c) o formato é responsável por detectar linhas diagonais.

Com intuito de otimizar o cálculo de tais características retangulares, Viola & Jones [16] propuseram a utilização de uma representação intermediária da imagem conhecida como imagem integral. Assim, cada ponto (x, y) contém o somatório da matriz de pixels que inicia em sua origem até a sua localização atual. Observe a Figura 4.

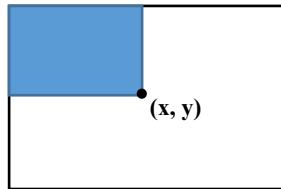


Fig. 4 - Área retangular delimitada para o cálculo da Imagem Integral.

A representação matemática pode ser verificada na equação abaixo, onde o resultado de (x, y) pode ser calculado mediante a uma única varredura na imagem original.

$$ii(x, y) = \sum_{x' \leq x, y' \leq y} i(x', y') \quad (1)$$

em que, $ii(x, y)$ é a integral da imagem e $i(x', y')$ é a imagem original.

Após a extração de características do objeto, o próximo passo é treinar o sistema utilizando o algoritmo *AdaBoost*. O *AdaBoost* (*Adaptive Boosting*) é um algoritmo meta-heurístico de aprendizagem supervisionada do tipo *boost*, proposto por [15], utilizado tanto para selecionar características quanto para treinar o classificador. A técnica consiste em combinar um conjunto de classificadores fracos, derivado de um conjunto de características básicas de *haar*, para formar um classificador forte, conforme mostrado na equação que se segue:

$$H(x) = \alpha_1 h_1 + \alpha_2 h_2 + \dots + \alpha_n h_n(x) \quad (2)$$

em que, $H(x)$ é um classificador forte e α_i é o peso associado ao classificador h_i .

B. Reconhecimento e contagem

A estrutura de cascata de classificadores foi originalmente descrita por [16] como um tipo de árvore de decisão degenerativa. A ideia de utilizar uma cascata de classificadores com N estágios é possibilitar a otimização do reconhecimento de objetos. Os estágios são criados através de uma junção de funções de classificação anteriormente formadas pelo uso do algoritmo de aprendizado *AdaBoost*.

A Figura 5 representa os N estágios de uma cascata de classificadores. A técnica consiste em descartar rapidamente o maior número de regiões nos primeiros estágios que não contém o objeto, afim de reduzir o esforço computacional de outras subáreas da imagem original.

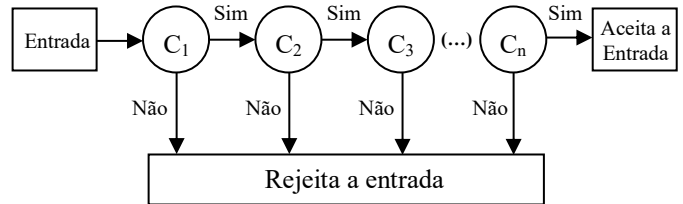


Fig. 5 - Representação de uma cascata de classificadores

O resumo de todo o processo é ilustrado na Figura 6, desde a aquisição das amostras positivas e negativas, passando pela seleção de características, treinamento da cascata de classificadores até a classificação dos elementos.

Após a etapa de classificação, o próximo passo é rastrear os objetos detectados em cena. A linha de julgamento será disposta horizontalmente, próximo à entrada da porta do TUE (Trem Unidade Elétrico).

O processo de rastreamento iniciará gerando uma lista de rastreamento vazia. Se um objeto for detectado, esse será um candidato a ser adicionado na lista de rastreamento de acordo com as seguintes condições:

$$CO = \begin{cases} 0, & (y_{co} < y_{ls} \parallel y_{co} > y_{li}) \\ candidato, & caso contrário \end{cases} \quad (3)$$

em que, CO indica a região da cabeça/ombro, o y_{co} representa a coordenada central da posição do objeto no eixo y , y_{ls} e y_{li} indicam o respectivo valor do limite superior e inferior da região de contagem, disposta paralelamente em torno da linha de julgamento. Conforme mostra o esboço na Figura 7.

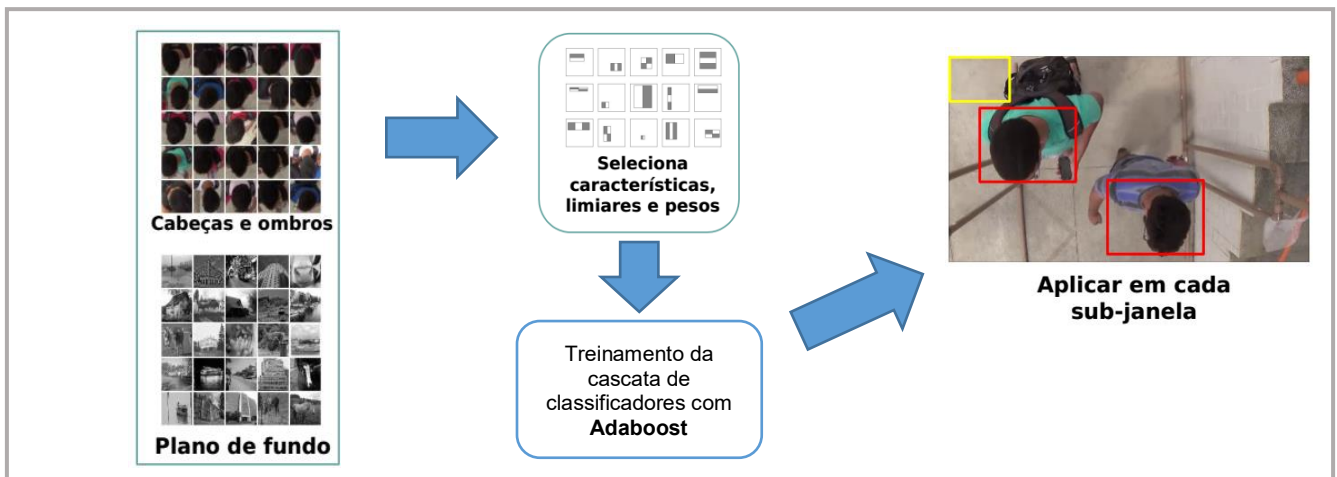


Fig. 6 - Visão geral do processo de treinamento e reconhecimento do sistema.

Caso o objeto esteja na região de contagem, será analisado a possibilidade desse objeto já fazer parte da lista de rastreamento ou se tratar de um novo elemento, de acordo com a seguinte regra:

$$CO = \begin{cases} 0, & Area_{sob}(CO, CO_{lista}) > L \\ \text{novο objeto}, & \text{caso contrário} \end{cases} \quad (4)$$

em que, CO_{lista} são os elementos presentes na lista de rastreamento, $Area_{sob}(CO, CO_{lista})$ corresponde à área de sobreposição entre CO e CO_{lista} , e L representa o limiar para medir o nível de sobreposição.

Após identificar os possíveis candidatos que integram a lista rastreamento, faz-se necessário descobrir o sentido do deslocamento do alvo, seguindo as condições abaixo:

$$D_{CO} = \begin{cases} CB, & y_{co} > y_{co} - 1 \\ BC, & y_{co} < y_{co} - 1 \end{cases} \quad (5)$$

em que, CB é o sentido de cima para baixo e BC o sentido inverso, o D_{CO} é a direção do alvo, e y_{co} representa a posição atual do objeto no eixo y , $y_{co} - 1$ corresponde posição anterior do deslocamento do objeto.

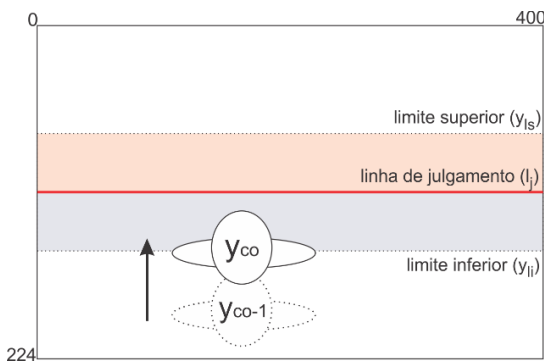


Fig. 7 - Rastreamento do objeto

Após determinarmos o sentido do deslocamento do objeto, será possível realizar a contagem utilizando a posição atual do alvo, seguindo as condições abaixo:

$$Cont = \begin{cases} ContEntrada, & D_{co} = CB \text{ e } y_{co} > l_j \parallel y_{co} < y_{li} \\ ContSaída, & D_{co} = BC \text{ e } y_{co} < l_j \parallel y_{co} > y_{li} \end{cases} \quad (6)$$

em que, y_{co} representa a coordenada atual do eixo y , juntamente o sentido do deslocamento D_{CO} , o l_j corresponde ao posicionamento da linha de julgamento. Se o alvo atender às condições citadas na Equação (6), o contador incrementa uma unidade de entrada ou saída; caso contrário, os alvos serão descartados.

IV. MODELO ESTIMATIVO DE EMBARQUE E DESEMBARQUE DE PASSAGEIROS COM VISÃO COMPUTACIONAL

A. Treinamento

Devido à falta de conjunto de dados para detecção de cabeças e ombros vistos de cima, foram coletadas inicialmente 400 imagens positivas (cabeças e ombros) a partir de frames de vídeos e 4500 amostras negativas (sem a presença de cabeças e ombros) com diferentes tipos e tamanhos, provenientes do banco de imagens [18].

O processo de normalização reduziu as imagens positivas originais em 50% do seu tamanho original que equivale a 60 x 45 px de resolução. Nas amostras negativas as imagens foram reduzidas a uma resolução de 100x100 px. Em seguida, ambas as imagens foram convertidas em escala de cinza.

Em nosso processo de treinamento do classificador, foram considerados 20 estágios, com taxa de acerto mínimo de cada estágio de 0,999, taxa de falso alarme máxima de cada estágio equivalente a 0,5 e a janela de deslizamento com resolução de 20x15 px.

B. Testes qualitativos

Considerando a realidade dos sistemas de transportes metroferroviário, foi desenvolvido, em laboratório, uma estrutura formada por tubos de PVC para simular o embarque/desembarque dos passageiros em três níveis de densidade: baixa, média e alta densidade, conforme mostra a Figura 8. A estrutura possui 1,5 metros de largura e pode ser ajustada em três níveis de altura: 3,50 m, 3,75 m e 4,00 m. O motivo da escolha desses valores, dar-se pela largura da porta do TUE e devido à altura média do pé direito das plataformas de embarque das estações da CBTU/METROREC.

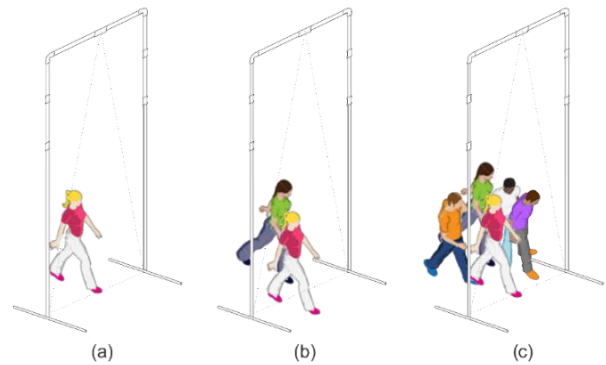


Fig. 8 - Níveis de densidade de passageiros: (a) baixa, (b) média e (c) alta densidade

Os testes foram realizados mediante à participação voluntária dos alunos do curso de computação da Universidade Federal da Paraíba (UFPB). Os procedimentos foram executados em várias etapas, inicialmente, separando-os em grupos de acordo com os níveis de densidade de passageiros. As imagens em vídeo foram capturadas e armazenadas, servindo como base para treinamento e testes posteriores.

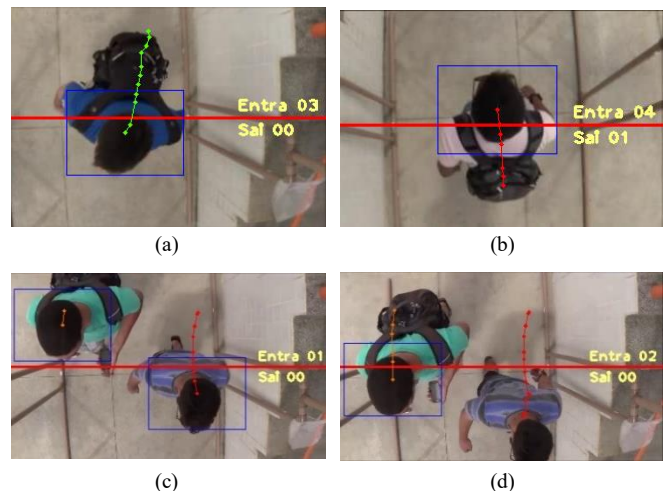




Fig. 9 - Simulação de contagem de passageiros em diferentes níveis de densidades

Os primeiros testes em laboratório foram considerados favoráveis. De acordo com a Figura 9 (a)-(f) é importante salientar que a contagem será apenas concretizada se o centro da cabeça e dos ombros ultrapassarem a linha de julgamento disposta no eixo horizontal. Na Figura 9 (a) e (b) é possível observar o deslocamento e a contagem bi-direcional dos alunos envolvidos nos testes. Nesse mesmo exemplo, é retratado a situação de baixa densidade de passageiros, onde, na maioria das vezes, o grau de complexidade de classificação é baixo. O nível de densidade média é ilustrado na Figura 9 (c) e (d), onde o número de pedestres é considerado intermediário e variam entre duas ou três pessoas próximas. Na situação da Figura 9 (d), a detecção desaparece quando a cabeça é posta parcialmente fora de cena. Na Figura 9 (e) e (f), o nível de espaço interpessoal é reduzido, possivelmente causando incômodo e desconforto aos usuários do sistema de transporte metroviários. Nota-se também que o grau de complexidade de detecção é maior, devido a fatores inerentes à oclusão.

Em alguns casos foi constatado que a mochila que os alunos carregam nas costas foram confundidas com amostras positivas (cabeças/ombros), para isso, fez-se necessário refazer o treinamento, incluindo nas amostras negativas, todos os elementos considerados próximos ao formato do objeto a ser detectado.

Nos próximos testes, serão avaliadas as taxas de acerto e o nível de acurácia do sistema.

V. CONCLUSÕES

Este trabalho tem como principal contribuição desenvolver um modelo estimativo em tempo real, automático e contínuo do fluxo de passageiros do sistema metroferroviário da Companhia Brasileira de Trens Urbanos (CBTU) de Recife. O modelo teórico foi elaborado de forma abstrata para desempenhar estratégias funcionais com o intuito de prever pontos críticos de multidões, de forma a melhorar a segurança, gerar relatórios e prognósticos para o planejamento operacional (aumentar a frota e/ou aperfeiçoar a dinâmica de despacho dos trens), evitando, no futuro, o surgimento de novas aglomerações. Com os avanços obtidos neste estudo, a implementação futura do controle de fluxo de passageiros com visão computacional, em sistemas de transporte metroferroviário, é viável, contribuindo assim para os avanços qualitativos dos serviços de transporte público em geral.

O estudo empírico foi iniciado mediante testes experimentais com a participação voluntária de alunos do curso de Computação do Centro de Informática da Universidade Federal da Paraíba (UFPB). O próximo passo é prover a instalação dos equipamentos e realizar levantamento preliminar dos dados coletados na estação Coqueiral, localizada na linha centro do Metrô de Recife - METROREC. O trabalho será validado comparando os estudos experimentais com os resultados obtidos a partir do monitoramento observatório (manual) de passageiros.

AGRADECIMENTOS

Agradecimento à CAPES e ao CNPq pelo apoio financeiro, ao Laboratório de Redes (LAR), Laboratório de Energia Solar (LES) da Universidade Federal da Paraíba e ao METROREC pelo incentivo e oportunidade de realizar esta pesquisa.

REFERÊNCIAS

- [1] United Nation, "World Urbanization Prospects: The 2014 Revision, Highlights," *Technical Report ST/ESA/SER.A/352*, p. 1, 2014.
- [2] Y. Yuan, J. Zhao, C. Qiu e W. Xi, "Estimating Crowd Density in an RF-Based Dynamic Environment.," *IEEE SENSORS JOURNAL*, vol. 13, nº 10, pp. 3837-3845, 2013.
- [3] F. Zhu, X. Yang, J. Gu e R. Yang, "A New Method for People-Counting Based on Support Vector.," em *2009 Second International Conference on Intelligent Networks and Intelligent Systems*, 2009.
- [4] C. Oberli, M. Torres-Torriti e D. Landau, "Performance Evaluation of UHF RFID Technologies for Real-Time Passenger Recognition in Intelligent Public Transportation Systems.," *IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS*, vol. 11, nº 30, pp. 748-753, 2010.
- [5] Y. Cong, H. Gong, S. Zhu e Y. Tang, "Flow Mosaicking: Real-time Pedestrian Counting without Scene-specific Learning.," em *2009 IEEE Conference on Computer Vision and Pattern Recognition*, 2009.
- [6] Y. Hejin, "A Bus Passenger Flow Estimation Method Based on Feature Point's Trajectory Clustering.," em *2010 IEEE International Conference on Intelligent Computing and Intelligent Systems*, 2010.
- [7] S. Mukherjee, B. Saha, R. Jamal, R. Leclerc e N. Ray, "Anovel Framework for Automatic Passenger Counting.," em *2011 18th IEEE International Conference on Image Processing*, 2011.
- [8] P. Lengvenis, R. Simutis, V. Vaitkus e R. Maskeliunas, "Application Of Computer Vision Systems For Passenger Counting In Public Transport.," *Elektronika Ir Elektrotechnika*, vol. 19, nº 3, pp. 69-72, 2013.
- [9] L. Dewei, Z. Lin e S. Yuxing, "Crowd Estimation Using Image Processing Based on Gray Scale Edge Histogram in Subway Station of Beijing.," em *2013 Fifth International Conference on Measuring Technology and Mechatronics Automation*, 2013.
- [10] C. Yan-Yan, C. Ning, Z. Yu-yang, W. Ke-han e Z. Wei-Wei, "Pedestrian Detection and Tracking for Counting Applications in Metro Station.," *Discrete Dynamics in Nature and Society*, vol. 2014, pp. 1-11, 2014.
- [11] Y. Zhang, D. Zhou, S. Chen, S. Gao e Y. Ma, "Single-Image Crowd Counting via Multi-Column Convolutional Neural Network.," em *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.
- [12] Z. Zhao, H. Li, R. Zhao e X. Wang, "Crossing-line Crowd Counting with Two-phase Deep Neural Networks.," em *European Conference On Computer Vision*, 2016.
- [13] X. Li, L. Li, F. Flohr, J. Wang, H. Xiong, M. Bernhard, S. Pan, D. Gavrilu e K. Li, "A Unified Framework for Concurrent Pedestrian and Cyclist Detection.," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, nº 2, pp. 269 - 281, 2016.
- [14] C. O. Escolano, R. K. C. Billones, E. Sybingco, A. D. Fillone e E. P. Dadios, "Passenger demand forecast using optical flow passenger counting system for bus dispatch scheduling.," em *2016 IEEE Region 10 Conference (TENCON)*, 2016.
- [15] Y. Freund e R. E. Schapire, "A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting.," *Journal of Computer and System Sciences*, vol. 55, nº 1, pp. 119-139, 1997.
- [16] P. Viola e M. Jones, "Rapid object detection using a boosted cascade of simple features.," *Computer Vision and Pattern Recognition*, vol. 1, pp. 511-518, 2001.
- [17] V. Amaral, C. Figaro-Garcia, G. J. Gattas e C. Thomaz, "Normalização espacial de imagens frontais de face em ambientes controlados e não-controlados.," *FaSci-Tech, FATEC São Caetano*, vol. 1, 10 2009.
- [18] ImageNet, "ImageNet," Stanford Vision Lab, Stanford University, Princeton University, [Online]. Available: <http://www.image-net.org>. [Acesso em 17 03 2017].