XXIX Congresso de Iniciação Científica UNICAMP 2021

Contagem de Pessoas em Multidões Utilizando Aprendizado de Máquina Profundo

Christian Massao Konishi*, Helio Pedrini*

*Instituto de Computação, Universidade de Campinas (UNICAMP), Campinas, SP, Brasil, 13083-852
Palavras-chave: contagem de pessoas, aprendizado de máquina, redes neurais convolucionais, análise de imagens

***Resumo*—A contagem de multidões por meio de imagens é um campo de pesquisa de grande interesse por suas diversas aplicações, como monitoramento de imagens de câmeras de segurança, planejamento urbano, além da possibilidade de usar esses modelos para a contagem de outros objetos, em outros domínios de problemas. Neste trabalho, um modelo é proposto baseado em Redes Adversárias Generativas (GANs) com custo Wasserstein e em redes neurais de múltiplas colunas, para obter melhores estimativas da quantidade de pessoas. Os resultados obtidos apresentam ganhos de acima de 30% no erro médio absoluto, em relação ao modelo original de múltiplas colunas.**

I. Introdução

Obter uma estimativa adequada da quantidade de pessoas presentes em uma imagem possui diversas aplicações práticas. A contagem de algumas dezenas de indivíduos é simples para ser efetuada manualmente, entretanto, em grandes multidões, como em manifestações, eventos musicais e esportivos, utilizar um modelo de contagem de pessoas em multidões pode ser, não raramente, a única opção viável, permitindo um melhor planejamento urbano, de eventos e a vigilância de aglomerações.

Além disso, modelos capazes de lidar com a contagem de pessoas em multidões densas têm potencial de serem aplicados em outros domínios de problemas [1], tais como microscopia celular, contagem de animais e pesquisas ambientais.

Uma forma intuitiva de modelar um contador de objetos é treinar um detector e, com ele, determinar a quantidade presente na imagem [2, 3]. Contudo, esses modelos não conseguem lidar adequadamente com grandes densidades de pessoas [1], por dependerem de reconhecer alguma parte do corpo, como cabeça e ombros, que pode estar parcialmente oclusa em uma multidão. Outros modelos [4, 5] não buscam detectar e localizar a posição de cada pessoa, eles procuram calcular a quantidade de objetos em uma imagem pela estimativa da densidade em determinada região da figura.

Uma dificuldade nesses modelos é lidar com variações de condições da imagem, tais como iluminação, densidade e tamanho das pessoas. Utilizar uma rede neural convolucional com filtros de diferentes tamanhos, como uma Rede Neural de Múltiplas Colunas (MCNN) [4] é uma alternativa para esses cenários, visto que pode lidar com variações no tamanho das pessoas em uma única imagem e com variações ocasionadas pelas diferentes dimensões das imagens. Por outro lado, uma limitação da MCNN está no fato de que sua saída é um mapa de densidade de altura e largura menores do que a imagem original, o que ocasiona uma perda de informação inerente ao próprio modelo.

Neste trabalho, foram propostas modificações na MCNN, tanto em nível de arquitetura quanto em nível de treinamento, com o objetivo de obter eficácias maiores, convergências mais estáveis e mapas de densidade mais fiéis aos mapas *ground truth*. Para isso, além da rede neural que estima a densidade de pessoas na imagem, foi adicionada uma segunda rede, cujo papel é avaliar a saída da primeira quando comparada com as densidades reais. Essa abordagem é uma aplicação das Redes Adversárias Generativas (GANs) [6], mais precisamente, a Wasserstein-GAN [7], no contexto de contagem de pessoas em multidões por mapas de densidade. O modelo proposto para o estimador parte de uma MCNN, mas introduz uma série de modificações para melhorar a qualidade da saída, recuperando a dimensão da imagem original e adicionando mais conexões possíveis entre os vários níveis da rede.

Este texto está organizado como segue. Na Seção II, os principais conceitos nos quais os testes se apoiaram são introduzidos. Trabalhos relacionados ao tema sob investigação são brevemente apresentados na Seção III. Na Seção IV, os principais procedimentos realizados neste trabalho são descritos em detalhes, tal que possam ser reproduzidos nas mesmas condições. Além disso, as métricas de avaliação dos modelos são apresentadas. Os resultados obtidos pelos modelos testados são apresentados na Seção V. Considerações finais e perspectivas para trabalhos futuros são descritas na Seção VI.

II. Conceitos

A tarefa de contagem de multidões consiste em estimar a quantidade de pessoas presentes em uma imagem ou um vídeo. Apesar de existirem outras abordagens (detecção de objetos, regressão), os modelos mais modernos têm se baseado em Redes Totalmente Convolucionais (FCN) [1], uma classe de Redes Neurais Convolucionais (CNN) que não apresenta camadas densamente conectadas.

A. Mapas de Densidade

Uma Rede Totalmente Convolucional é capaz de estimar a quantidade de pessoas em uma imagem de uma multidão produzindo um mapa de densidade cuja soma de todos os elementos resulta na quantidade de pessoas (Figura 4). Para treinar uma rede capaz de gerar esses mapas, é necessário produzir valores *ground truth* para as imagens de treinamento.

Dada uma imagem I, de dimensões $M \times N$, com k pessoas, a posição de cada indivíduo é aproximada para um único ponto, tal que as posições das pessoas sejam $P = \{(x_1, y_2), (x_2, y_2), (x_3, y_3), \ldots, (x_k, y_k)\}$. Com base nisso, é definido uma mapa H com as posições das pessoas.

$$H(x,y) = \begin{cases} 1, \text{ se } (x,y) \in P \\ 0, \text{ caso contrário} \end{cases} \quad (1)$$

O mapa de densidade é obtido por meio de uma convolução, aplicando em H um filtro Gaussiano, o tamanho do filtro, assim como o valor de σ devem ser definidos, podendo variar ou não.

B. Redes Adversárias Generativas

As Redes Adversárias Generativas (GANs) foram propostas em 2014 por Goodfellow et al. [6] como uma estrutura capaz de criar modelos generativos utilizando duas redes adversárias: (i) uma rede generativa G é responsável por gerar imagens por meio de *perceptrons* alimentados por ruído e (ii) uma rede discriminativa D alimentada com a base de dados e exemplares gerados por G. As redes devem ser então treinadas de forma a D conseguir discriminar imagens falsas de verdadeiras, ao passo que G é penalizada se D conseguir evitar os erros.

Melhorias em relação ao modelo original de GANs foram feitas, entre elas, a Wasserstein-GAN [7], que, ao propor uma função de custo diferente (Seção IV-E), diminui a necessidade de manter um equilíbrio cuidadoso entre as redes Generativas e Discriminativas (também referida como Crítica).

C. Aumento de Dados

Utilizar bases de dados mais robustas é a melhor forma de um modelo ser mais generalista. Entretanto, nem sempre conseguir mais amostras para o banco de dados é uma solução viável. Uma alternativa para isso é a introdução de novos exemplos de dados por meio de transformações automatizadas dos dados já existentes.

Para imagens, transformações como rotações, translações e distorções são fáceis de serem aplicadas e resultam em ganhos consideráveis [8]. Outra medida eficiente é a adição de ruído nas entradas das redes, que costuma resultar em modelos que melhor generalizam [9].

III. Trabalhos Relacionados

O trabalho conduzido por Quispe et al. [10] estudou a utilização de diversas redes neurais de múltiplos canais e de diferentes filtros gaussianos, de tamanho fixo e variável, propondo o próprio método para definir o filtro gaussiano utilizado para gerar os valores *ground truth* para o treinamento.

As Redes Generativas Adversárias (GANs) foram propostas por Goodfellow et al. [6] como um modelo para gerar imagens sintéticas a partir de ruído, este trabalho inicial utilizava redes neurais densamente conectadas e apresentava um treinamento muito sensível ao equilíbrio entre as redes Generativa e Discriminativa. Já o trabalho de Radford et al. [11] apresentou arquiteturas de redes totalmente convolucionais para as GANs, permitindo criar modelos mais complexos, a arquitetura da rede Discriminativa de Radford será utilizada neste trabalho também. Por fim, o trabalho de Arjovsky et al. [7], ao propor a Wasserstein-GAN, diminuiu a necessidade de manter um equilíbrio no treinamento das redes Generativa e Discriminativa.

A criação de novos modelos de contadores de pessoas em multidões depende da existência de bases de dados diversas com imagens apropriadamente anotadas, entre elas:

- **UCSD** [12] é a primeira base de dados para contagem de multidões [1] composta por imagens de câmeras de calçadas, o que limita a variedade de perspectiva das imagens.
- **UCF_CC_50** [13] é uma base de dados desafiadora, contendo imagens com pessoas em diferentes escalas e regiões com extrema densidade de indivíduos, limitada a apenas 50 figuras.
- **Shanghai Tech** [4] é um base de dados dividida em duas partes, uma coletada na Internet, outra coletada de uma rua de Shanghai. A base é numerosa, conta com 1198 imagens anotadas.
- **JHU-CROWD++** [14]: é uma base de dados maior ainda, com 4372 imagens em condições diversas de densidade, iluminação, perspectiva e ambiente.

IV. Metodologia

Nesta seção, os métodos empregados nos experimentos realizados neste trabalho são apresentados, contendo informações a respeito das bases de dados, algoritmos, e arquiteturas empregadas nos testes.

A. Bases de Dados

A base de dados de contagem de pessoas em multidões utilizada neste trabalho é a UCF-CC-50 [13]. A base apresenta 50 imagens de multidões de densidades variáveis, podendo ser extremamente densas (Figura 1), com anotações para a posição de cada pessoa.

Figura 1: Exemplo de uma imagem da UCF-CC-50. Pode-se observar como a imagem apresenta variação de densidade, com regiões de concentração extrema.

Devido à escassez na quantidade de imagens, a estratégia de validação cruzada *5-fold* foi utilizada, dividindo o conjunto original em 5 grupos, tomando 4 grupos para treinamento e 1 para teste, repetindo o procedimento para cada divisão possível de treinamento e teste. Os resultados apresentados compreendem a média das eficácias em cada uma das 5 partições (*folds*), assim como o desvio padrão.

B. Aumento de Dados

Processos de aumento de dados foram empregados para expandir a quantidade de dados disponível para treinamento em cada *fold*. As estratégias empregadas visaram, em um primeiro momento, reproduzir os resultados obtidos por Quispe et al. [10], sendo mantidas para testes posteriores, permitindo a comparação direta de modelos.

Três formas de aumento de dados foram adotadas, sendo elas: (i) utilização de uma janela deslizante de 256×256 píxeis, que percorre as imagens com um passo de 70 píxeis, recortando as novas figuras; (ii) adição ruído gaussiano (média zero e variância 0,1) em metade das imagens geradas na etapa anterior e de ruído impulsivo (com 4% de probabilidade de afetar um pixel) na outra metade, duplicando a quantidade de amostras; (iii) alterações nas

condições de iluminação (Equação 2) das imagens disponíveis após (ii), dobrando novamente a quantidade de imagens. As imagens originais não são utilizadas

$$f' = \begin{cases} f_i + 10, \text{ se i é par} \\ 1,25\ f_i - 50, \text{ caso contrário} \end{cases} \quad (2)$$

sendo f' a imagem de saída e f_i a i-ésima imagem disponível após a etapa (ii).

C. Arquiteturas das Redes Neurais

O modelo utilizado possui duas redes neurais profundas — assim como em um modelo padrão de GAN —, sendo elas a rede crítica (Figura 2) e uma rede neural de múltiplos canais, que será referida como MCNN-U (Figura 3), análoga à rede generativa de uma Wasserstein-GAN, mas cuja entrada é uma imagem monocromática e não um ruído, além de utilizar convoluções transpostas e *skip connections*, diferenciando-a de modelos anteriores [10].

A MCNN empregada é composta por 4 colunas – referidas como colunas-U – de diferentes tamanhos de filtros (Figura 3). Cada coluna apresenta operações de convolução e *max pooling* que reduzem a dimensão da ativação do bloco, seguidas de operações de convolução transposta, que recuperam o tamanho original do mapa (*upscale*). Além disso, *skip connections* foram adicionadas conectando as camadas convolutivas e as camadas de *upscale*, permitindo que a arquitetura possa combinar ativações de diferentes profundidades da rede.

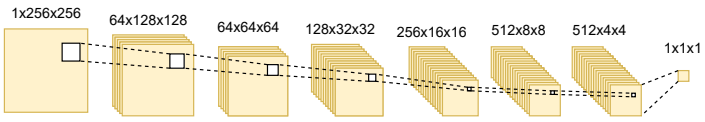


Figura 2: Arquitetura da rede crítica. A função de ativação utilizada é a *leaky* ReLU ($\alpha = 0,02$) e as convoluções têm parâmetros 4, 2 e 1 para o tamanho do *kernel*, *stride* e *padding*, respectivamente. A camada de saída não tem função de ativação e a convolução tem parâmetros 4, 1 e 0.

D. Mapas de Densidade

A tarefa da MCNN-U é produzir um mapa de densidade cujo somatório corresponda à quantidade de pessoas na imagem. Para produzir os valores *ground truth*, dada uma imagem de dimensões $M \times N$, uma matriz nula de mesmo tamanho é criada. Na posição de cada pessoa, o valor 1 é colocado na matriz e, por fim, é feita uma convolução com um filtro gaussiano (Figura 4) — dimensões 15×15, $\sigma = 15$ —.

E. Função de Custo

A função de custo utilizada para o treinamento das redes neurais descritas pode ser dividida em duas partes, uma que corresponde ao objetivo básico de diminuir a distância entre a saída da MCNN-U — a rede MCNN-U será denotada G, logo a saída de G para uma imagem I é denotada $G(I)$ — e do valor *ground truth* (gt); e outra parte que diz respeito ao custo *Wasserstein* do modelo de redes adversárias. A rede crítica será denotada C.

A distância entre os mapas de distribuição de densidade se dá pelo erro médio quadrático dos valores da matriz:

$$L_{\text{MSE}} = \frac{1}{m}\sum_{i=1}^{m}\frac{1}{MN}\sum_{x=1}^{M}\sum_{y=1}^{N}\left[G(I_i)(x,y) - gt(x,y)\right]^2 \quad (3)$$

sendo M a largura e N a altura do mapa de densidade, enquanto m é o tamanho do *batch*.

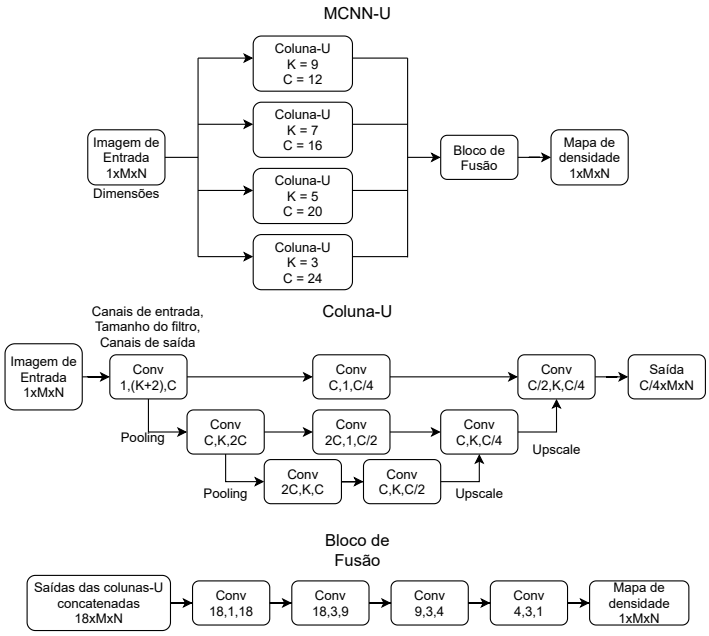


Figura 3: Descrição da MCNN-U. A dimensão dos tensores é representada no formato <canais>x<largura>x<altura>, já as as convoluções, no formato <canais de entrada>, <tamanho do *kernel*>, <canais de saída>; cada convolução é seguida de uma função de ativação ReLU, com exceção da camada de saída. A operação de *max pooling* é utilizada para reduzir as dimensões de altura e largura pela metade, já o *upscale* recupera a dimensão original com convoluções transpostas com tamanho de kernel K e stride $= 2$. Convoluções 1x1 foram utilizadas nas *skip connections* para diminuir a quantidade de canais transmitidos.

Figura 4: Visualização do mapa de distribuição de densidade em forma de mapa de calor, sobreposta com sua imagem original, para a base UCF-CC-50.

Já o custo *Wasserstein* para a rede generativa é dado por:

$$L_W = -\frac{1}{m}\sum_{i=1}^{m} C(G(I_i)) \quad (4)$$

Combinando os dois custos, tem-se o custo da rede generativa:

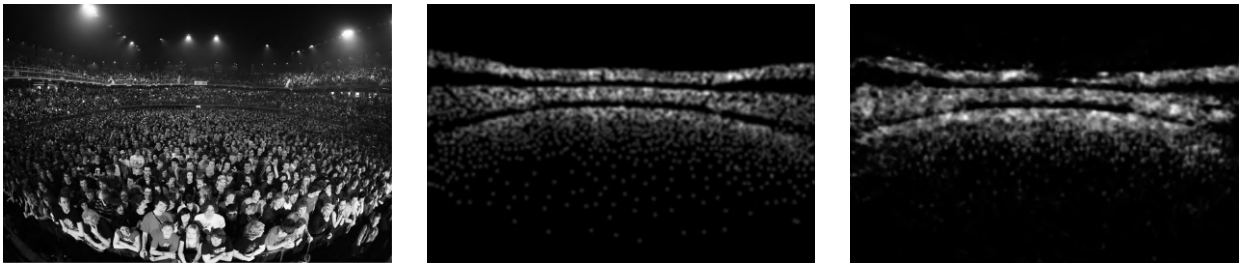
$$L_G = L_{\text{MSE}} + \alpha L_W \quad (5)$$

sendo α um hiperparâmetro a ser decidido.

Já o custo para a rede crítica é dado por:

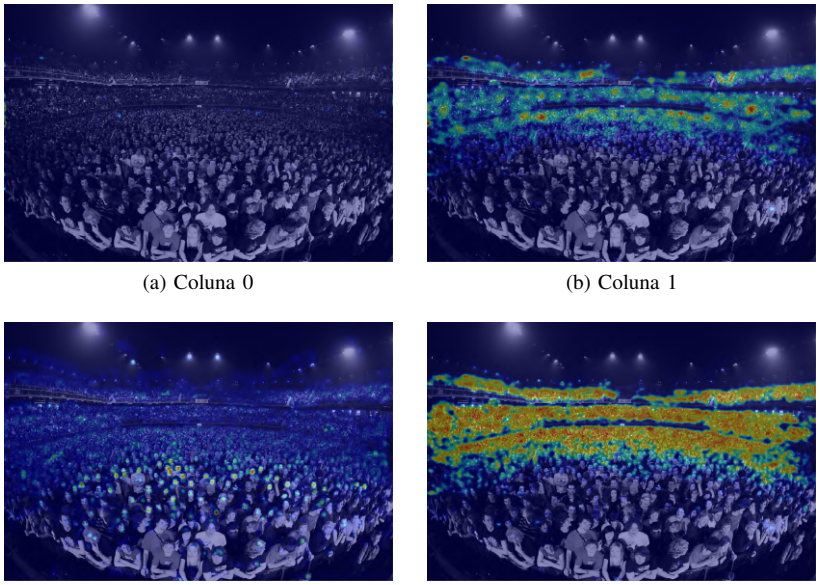
$$L_C = \frac{1}{m}\sum_{i=1}^{m} C(gt_i) - \frac{1}{m}\sum_{i=1}^{m} C(G(I_i)) \quad (6)$$

O objetivo do método é minimizar o valor de L_G e maximizar o de L_C.



(a) *Imagem original* (b) *Ground truth* (c) MCNN-U (*fold* 2)

Figura 5: Visualização dos mapas de densidade produzidos por meio dos valores *ground truth* e pela MCNN-U.



(a) Coluna 0 (b) Coluna 1

(c) Coluna 2 (d) Coluna 3

Figura 6: Visualização dos mapas de ativação de cada uma das colunas da MCNN-U, sendo a coluna 0 (a), a de maior tamanho de filtro, e a coluna 3 (d), a de menor.

F. Configuração do Teste

Esta subseção é responsável por apresentar a forma como os treinamentos e avaliações de modelos foram conduzidas. Diferentes versões do modelo, com variações nos hiperparâmetros foram avaliadas, mas apenas a versão final será apresentada neste resumo.

O custo *Wasserstein* foi aplicado utilizando o valor de $\alpha = 3.500$ (Equação 5), além disso, os mapas de densidade foram multiplicados por 16.000. Para cada um dos conjuntos de treinamento e teste da UCF-CC-50, mil épocas foram executadas, com um *batch size* de 32 imagens.

Foi adotado o otimizador Rectified Adam [15], com taxa de aprendizado $lr = 1.\ 10^{-4}$, $\beta_1 = 0,9$ e $\beta_2 = 0,999$. O valor de *ncritic*, isto é, a quantidade de vezes que a rede crítica recebe dados e é otimizada para cada passagem pela rede generativa, foi definido em 3.

Os algoritmos foram todos executados por meio de máquinas virtuais do *Google Colaboratory*. As máquinas disponibilizam, em geral, um núcleo de um Intel Xeon (modelo variável), cerca de 12 GB de memória RAM e uma placa gráfica (do inglês, *graphics processing unit* - GPU) que pode variar entre uma Nvidia Tesla K80, uma Nvidia Tesla P100 ou uma Nvidia Tesla T4.

G. Métricas de Desempenho

Para medir o desempenho (eficácia) das redes testadas, o somatório do mapa de densidade da saída da rede generativa é comparado com o do mapa gerado por meio do filtro gaussiano. Para quantizar a diferença entre as duas contagens, duas medidas foram empregadas, sendo elas:

- Erro médio absoluto:

$$\text{MAE} = \frac{1}{N}\sum_{i=1}^{N}|cnt_i - cnt'_i| \tag{7}$$

- Raiz do erro médio quadrático:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(cnt_i - cnt'_i)^2} \tag{8}$$

em que N a quantidade de imagens no conjunto de teste, cnt_i a contagem correta de pessoas da imagem i e cnt'_i a contagem feita a partir do resultado da rede neural. Note que esses valores apenas avaliam o resultado da contagem e não o próprio conteúdo dos mapas de distribuição de densidade.

Para o caso da UCF-CC-50, as duas métricas são calculadas para cada um dos 5 *folds*, e o resultado final é dado pela média, mas o desvio padrão também foi calculado, por representar o quão homogênea foi a eficácia do modelo para cada *fold*.

V. Resultados Experimentais

É possível visualizar os mapas de densidade *ground truth* e os produzidos pela MCNN-U para uma mesma imagem, colocados lado a lado para fins de comparação (Figura 5). Além disso, o comportamento de cada coluna pôde ser observado de maneira independente, por meio da visualização de seu mapa de ativação pelo algoritmo LayerCAM [16] (Figura 6). É notável que cada coluna foi mais (ou menos) ativada em diferentes regiões da imagem, devido à variação de densidade de pessoas.

Tabela I: Eficácias obtidas pelo modelo MCNN-U.

Modelo	*Fold* 1	*Fold* 2	*Fold* 3	*Fold* 4	*Fold* 5	Média	Desvio padrão
MAE	424,5566	204,7662	197,8882	229,6904	225,2515	256,4306	84,9117
RMSE	709,003	294,2144	323,6123	321,3106	345,6961	398,7673	155,9756

Os resultados obtidos após o treinamento da MCNN-U, para os hiperparâmetros definidos na Subseção IV-F foram compilados na Tabela I, separados por *fold*.

A eficácia do modelo foi relativamente constante em todos os *folds*, com exceção do primeiro. Esse padrão se repetiu ao longo dos modelos anteriores, sendo essa a partição mais desafiadora da base de dados, possivelmente pela alta densidade das imagens de teste.

Para poder avaliar a eficácia obtida pela MCNN-U, as métricas de desempenho foram comparadas com outros resultados da literatura (Tabela II).

Tabela II: Comparação da eficácia atingida pela MCNN-U em relação a outros modelos da literatura.

Modelo	MAE médio	RMSE médio
MCNN [4]	377,6	509,1
Quispe et al. [10] ($MSNN_3$, *Face*)	374,0	554,6
CP-CNN [17]	295,8	320,9
MCNN-U	256,4	398,8
CAN [18]	212,2	243,7

A eficácia obtida representa um salto quando comparada com a MCNN original, demonstrando que as modificações aplicadas foram de fato apropriadas ao modelo. Em relação aos trabalhos da literatura, os resultados foram competitivos, mas abordagens mais complexas que buscam avaliar o contexto da imagem obtiveram eficácias superiores, o que pode, por outro lado, resultar em treinamentos mais complexos.

VI. Conclusões e Trabalhos Futuros

A eficácia obtida pela MCNN-U foi consideravelmente superior em relação à MCNN original, as alterações propostas no modelo foram testadas de maneira incremental, havendo ainda espaço para mais modificações que podem, por sua vez, resultarem em eficácias ainda mais elevadas. De qualquer forma, os resultados atingidos pelo modelo atual já se mostram competitivos.

Outro aspecto que deve ser avaliado futuramente é como a MCNN-U se comporta com imagens de outras bases de dados ou com outros tipos de contagens, além de cenários de multidões já testados.

Agradecimentos

Os autores são gratos ao Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) pela concessão de bolsa de iniciação científica.

Referências

[1] G. Gao, J. Gao, Q. Liu, Q. Wang, and Y. Wang, "CNN-based Density Estimation and Crowd Counting: A Survey," *arXiv preprint arXiv:2003.12783*, pp. 1–25, 2020.

[2] M. Li, Z. Zhang, K. Huang, and T. Tan, "Estimating the Number of People in Crowded Scenes by Mid based Foreground Segmentation and Head-shoulder Detection," in *19th International Conference on Pattern Recognition*. IEEE, 2008, pp. 1–4.

[3] Sheng-Fuu Lin, Jaw-Yeh Chen, and Hung-Xin Chao, "Estimation of Number of People in Crowded Scenes using Perspective Transformation," *IEEE Transactions on Systems, Man, and Cybernetics - Part A: Systems and Humans*, vol. 31, no. 6, pp. 645–654, 2001.

[4] Y. Zhang, D. Zhou, S. Chen, S. Gao, and Y. Ma, "Single-image Crowd Counting via Multi-column Convolutional Neural Network," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 589–597.

[5] V. Lempitsky and A. Zisserman, "Learning to Count Objects in Images," *Advances in Neural Information Processing Systems*, vol. 23, pp. 1324–1332, 2010.

[6] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," *Advances in neural information processing systems*, vol. 27, pp. 2672–2680, 2014.

[7] M. Arjovsky, S. Chintala, and L. Bottou, "Wasserstein GAN," *1701.07875*, pp. 1–32, 2017.

[8] P. Y. Simard, D. Steinkraus, and J. C. Platt, "Best Practices for Convolutional Neural Networks Applied to Visual Document Analysis," in *International Conference on Document Analysis and Recognition*, vol. 3, 2003, pp. 1–6.

[9] J. Sietsma and R. J. Dow, "Creating Artificial Neural Networks that Generalize," *Neural Networks*, vol. 4, no. 1, pp. 67–79, 1991.

[10] R. Quispe, D. Ttito, A. Rivera, and H. Pedrini, "Multi-Stream Networks and Ground Truth Generation for Crowd Counting," *International Journal of Electrical and Computer Engineering Systems*, vol. 11, no. 1, pp. 33–41, 2020.

[11] A. Radford, L. Metz, and S. Chintala, "Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks," *arXiv preprint arXiv:1511.06434*, pp. 1–16, 2015.

[12] A. B. Chan, Z.-S. J. Liang, and N. Vasconcelos, "Privacy Preserving Crowd Monitoring: Counting People without People Models or Tracking," in *IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2008, pp. 1–7.

[13] H. Idrees, I. Saleemi, C. Seibert, and M. Shah, "Multi-source Multi-scale Counting in Extremely Dense Crowd Images," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 2547–2554.

[14] V. Sindagi, R. Yasarla, and V. M. M. Patel, "Jhu-crowd++: Large-scale crowd counting dataset and a benchmark method," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–17, 2020.

[15] L. Liu, H. Jiang, P. He, W. Chen, X. Liu, J. Gao, and J. Han, "On the Variance of the Adaptive Learning Rate and Beyond," in *Eighth International Conference on Learning Representations (ICLR)*, April 2020, pp. 1–14.

[16] P.-T. Jiang, C.-B. Zhang, Q. Hou, M.-M. Cheng, and Y. Wei, "LayerCAM: Exploring Hierarchical Class Activation Maps for Localization," *IEEE Transactions on Image Processing*, vol. 30, pp. 5875–5888, 2021.

[17] V. A. Sindagi and V. M. Patel, "Generating High-Quality Crowd Density Maps Using Contextual Pyramid CNNs," in *IEEE International Conference on Computer Vision (ICCV)*, 2017, pp. 1879–1888.

[18] W. Liu, M. Salzmann, and P. Fua, "Context-Aware Crowd Counting," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019, pp. 5099–5108.