Generalização de Modelo de Rede Convolucional Multi-Coluna em Contagem de Multidão

Lucas C. Favaro e Rodrigo S. Couto

***Resumo*— A contagem de multidão é um problema de visão computacional com diversas aplicações, como monitoramento de espaços públicos e gestão de tráfego. Apesar da grande quantidade de modelos propostos na literatura, poucos estudos abordam a generalização dos modelos na contagem de multidão. Este trabalho busca analisar a generalização da contagem de multidão utilizando um modelo de rede convolucional multi-coluna. Os resultados indicam dificuldade na generalização. Nos experimentos com o *dataset* de imagens de câmeras de segurança de shopping, nos cenários em que o conjunto de testes é de um *dataset* diferente daquele utilizado durante o treinamento, o desempenho é duas vezes pior no melhor caso e 60 vezes pior no pior caso.**

***Palavras-Chave*— Contagem de multidão, Generalização, CNN, Redes Neurais Convolucionais, Visão Computacional, Mapa de Densidades.**

***Abstract*— Crowd counting is a computer vision problem with many applications, such as monitoring public spaces and traffic management. Many models have been proposed in the literature, but few studies address the generalization of these models. This work seeks to analyze the generalization of crowd counting using a multi-column convolutional network model. The results indicate difficulty in generalization. For the *dataset* of shopping mall images, when the test set is from a different *dataset* than the one used for training, performance is twice as bad in the best case and 60 times worse in the worst case.**

***Keywords*— Crowd Counting, Generalization, CNN, Convolutional Neural Networks, Computer Vision, Density Map.**

I. Introdução

A contagem de multidão é uma tarefa na qual deseja-se obter a quantidade de pessoas presentes em uma determinada imagem por meio de um modelo computacional. Tal modelo pode ser utilizado em aplicações como monitoramento de vias públicas, controle de tráfego e gestão de lotação de estabelecimentos [1]. Além disso, a contagem de multidão pode ser usada para garantir a segurança em eventos com grande fluxo de pessoas.

Diversos modelos foram propostos para contagem de multidão utilizando técnicas de visão computacional [1]. Os primeiros modelos se baseavam em técnicas clássicas de aprendizado de máquina, como *Support Vector Machines* [2] e modelos probabilísticos [3], [4], aliadas a técnicas de processamento de imagens. Porém, com o sucesso de Redes Neurais Convolucionais (CNNs - *Convolutional Neural Networks*) em tarefas de visão computacional, as CNNs passaram a ser amplamente utilizadas em modelos de contagem de multidão [5], [6], [7]. Além desses modelos, recentemente, alguns estudos sugerem o uso de *Transformers* como alternativa viável às CNNs [8].

Apesar de ser uma tarefa amplamente estudada e com diversos modelos propostos, a pesquisa sobre generalização de tais modelos ainda é incipiente [1]. Poucos trabalhos abordaram a generalização dos modelos e os que fizeram se restringiram a cenários fixos e limitados a poucos *datasets* [6], [9], [10]. Isso dificulta a adoção prática de ferramentas automáticas para contagem de pessoas em grandes cidades e eventos.

Este trabalho visa então analisar a generalização de um dos modelos mais citados e considerados estado-da-arte, o modelo de CNNs-multicoluna (*Multi-column Convolutional Neural Network*-MCNN) [5], utilizando cinco *datasets* diversos entre si. Para avaliar a generalização, o modelo é treinado com cada *dataset* e avaliado com os demais.

Os resultados apontam que o modelo estudado não generaliza bem para imagens diferentes daquelas presentes no domínio de seu conjunto de treinamento. Os experimentos mostram que o desempenho para um *dataset* específico reduz consideravelmente quando o modelo é treinado utilizando um *dataset* diferente. No caso do *dataset* de imagens de câmeras de segurança de shopping, o desempenho é duas vezes pior no melhor caso e 60 vezes pior no pior caso, comparando com o cenário no qual são utilizados os dados do próprio *dataset*.

O restante deste trabalho está organizado da seguinte forma. A Seção II apresenta uma breve revisão dos trabalhos relacionados. A Seção III traz em detalhes a metodologia utilizada nos experimentos e na avaliação dos resultados. A Seção IV expõe e discute os resultados obtidos nos experimentos. Por fim, a Seção V apresenta as considerações finais e possíveis direções para trabalhos futuros.

II. Trabalhos Relacionados

Redes neurais convolucionais alcançaram grande sucesso em tarefas de visão computacional [11]. Esse sucesso resultou na adoção de arquiteturas baseadas em CNNs na contagem de multidão. Consequentemente, modelos de CNNs atingiram o status de estado-da-arte e se tornaram amplamente adotados para essa tarefa [5], [6], [7]. A abordagem mais utilizada nos trabalhos de contagem de multidão consiste em utilizar esses modelos para gerar um mapa de densidade da imagem de interesse. A integral desse mapa de densidade dá a quantidade total de pessoas presentes na imagem.

Zhang et al. [5] propõem um modelo de CNNs multi-colunas (MCNN), no qual três colunas de CNNs são treinadas

Lucas Favaro, GTA/PEE/COPPE, UFRJ Universidade Federal do Rio de Janeiro (UFRJ), RJ-RJ, e-mail: favaro@gta.ufrj.br; Rodrigo Couto, GTA/PEE/COPPE, Universidade Federal do Rio de Janeiro (UFRJ), RJ-RJ, e-mail: rodrigo@gta.ufrj.br. Este trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001, CNPq, FAPERJ e FAPESP.

em paralelo com filtros diferentes entre si e a saída é a média das previsões individuais de cada coluna. Dada essa arquitetura multi-colunar, variações no tamanho das pessoas causadas pela perspectiva e diferenças na resolução em partes da imagem podem ser aprendidas adaptativamente. Como a rede utiliza diferentes filtros em cada coluna, as características extraídas pelas colunas de CNNs conseguem capturar essas variações. Essa adaptação às variações também permite que as imagens tenham tamanho arbitrário. Apesar de eficiente em detectar tais variações, a adição de múltiplas colunas de CNN resulta em um aumento de complexidade computacional.

Liu et al. [12] propõem uma CNN que codifica a escala da informação contextual necessária para prever corretamente a densidade de multidões. Essa rede consciente do contexto (CAN - *Context Aware Network*) busca aprender a explorar o contexto de cada parte de cada imagem. O modelo combina características obtidas através de múltiplos tamanhos de campos receptivos nas CNNs e aprende a importância de cada uma dessas características em cada parte da imagem.

Para resolver problemas de imagens com regiões superlotadas e vazias simultaneamente, Jiang et al. [13] propõem um modelo denominado ASNet. A rede convolucional proposta aprende fatores de escala para ajustar automaticamente a estimativa de densidade de cada sub-região correspondente. A ASNet gera várias estimativas de densidade local que são agregadas para formar o mapa de densidade final.

O aprendizado por transferência (*Transfer Learning*) pode ser utilizado para melhorar a precisão de modelos pequenos de contagem de multidão em imagens com alta densidade de pessoas. Li et al. [14] propõem o CSRNet, um modelo composto por um *front-end* de CNNs utilizando as primeiras 10 camadas do modelo VGG-16, treinado no conjunto de dados ImageNet e um *back-end* composto por CNNs dilatadas. Uma CNN dilatada tem seu *kernel* expandido por meio de lacunas entre seus elementos, permitindo que cada unidade cubra uma área maior da entrada sem aumentar os parâmetros. Conforme observado por Khan et al. [1], o CSRNet foi o primeiro modelo utilizando aprendizagem por transferência a apresentar um aumento significativo de desempenho em relação ao estado-da-arte.

Mais recentemente, Yan et al. [6] desenvolveram um modelo de contagem de multidão em vários domínios (DCANet) como uma alternativa para melhorar a generalização sem degradação do desempenho no domínio de origem. Os autores desenvolveram uma rede para lidar com a contagem de multidões em vários domínios usando um único modelo profundo, que mistura dados de diferentes domínios. Esse trabalho avalia o desempenho de outros modelos em domínios fixos não vistos, porém se limita a analisar a generalização considerando apenas três *datasets* para treinamento, sendo dois deles as partes distintas do mesmo conjunto, as partes A e B do ShaghaiTech [5].

Du et al. [9] propõem um modelo visando resolver o problema da generalização de domínio em cenários não vistos. A proposta consistem em um arcabouço no qual o domínio de origem é dividido em diversos sub-domínios que são usados numa abordagem de *meta-learning*. A rede proposta conta com dois módulos distintos: um para extrair características específicas dos domínios e um para extrair características comuns a todos os domínios. Os experimentos sobre generalização são conduzidos considerando apenas as duas partes do ShanghaiTech e sua combinação como domínios de origem, não explorando o treinamento utilizando outros *datasets*.

Este trabalho busca estudar a generalização no problema de contagem de multidão por meio do modelo MCNN, um dos modelos mais citados em trabalhos de contagem de multidão e considerado um dos modelos de estado-da-arte, analisando seu desempenho em cenários diversos. Para tal, consideram-se cinco conjuntos de dados diferentes para treinamento e avalia-se o desempenho de cada um desses cinco *datasets* em cenários distintos daqueles vistos durante as fases de treinamento do modelo.

Dos trabalhos relacionados, apenas dois consideram a generalização, mas utilizam poucos *datasets*. A Tabela II mostra os *datasets* utilizados neste trabalho e nos relacionados. Assim, do ponto de vista da análise da generalização, este trabalho possui um escopo maior de resultados, considerando *datasets* com diferenças significativas entre si. O MCNN é utilizado por ser o modelo que introduziu uma arquitetura de grande sucesso na literatura (CNN de múltiplas colunas) e por permitir que as entradas tenham tamanho arbitrário.

TABELA I
Datasets DOS TRABALHOS QUE CONSIDERAM GENERALIZAÇÃO.

Trabalho	Datasets de treinamento
Yan et al. [6]	ShanghaiTech, UCF_QNRF [15]
Du et al. [9]	ShanghaiTech
Este trabalho	ShanghaiTech, UCF_CC_50, Mall, UCSD, VisDrone

III. METODOLOGIA

Para avaliar a generalização do modelo em cenários nos quais o *dataset* de testes é diferente do *dataset* utilizado para o treinamento, neste trabalho o modelo é treinado com os dados de cada um dos cinco *datasets* separadamente e avaliado com os conjuntos de testes dos demais. O desempenho é avaliado utilizando o erro absoluto médio (*Mean Absolute Error* - MAE), que é a métrica mais utilizada na literatura [1]. Cada experimento é repetido dez vezes e são calculados intervalos de confiança pelo método da distribuição *t-Student* considerando um nível de confiança de 95%. O código em Pytorch [16] utilizado como base para implementação dos experimentos com o MCNN pode ser encontrado em `https://github.com/CommissarMa/MCNN-pytorch/blob/master/train.py`. Todos os códigos utilizados nos experimentos deste trabalho estão disponíveis em `https://github.com/GTA-UFRJ/CrowdCountingGeneralization`.

A. Datasets

Este trabalho utiliza cinco *datasets*. Quatro desses datasets estão entre os mais utilizados em problemas de contagem de multidão: UCSD [17], UCF_CC_50 (UCF_50) [15], ShaghaiTech - parte A (SHA) [5] e Mall [18]. Além disso, utiliza-se um dataset com imagens de drones, denominado

VisDrone [19], para introduzir um cenário completamente diferente dos demais. Para ilustrar as diferenças entre as imagens, a Figura 1 traz uma imagem esparsa (isto é, com poucas pessoas na imagem) capturada por *drones*, enquanto a Figura 2 representa uma imagem densa capturada por uma câmera livre.

Cada um dos *datasets* foi divido em conjuntos de treinamento e teste gerados aleatoriamente, com 75% das amostras no conjunto de treinamento e 25% no conjunto de teste, sem interseção de imagens entre os dois conjuntos. Esse processo foi repetido em cada uma das dez iterações.

Fig. 1. Exemplo de imagem esparsa encontrada no *dataset* VisDrone.

Fig. 2. Exemplo de imagem densa encontrado no UCF_50. Imagem alterada para preservar a identidade das pessoas.

O UCSD consiste em um *dataset* com 2.000 imagens de camêras de segurança de resolução 238×158 retratando pedestres, com uma média de 25 pessoas por imagem. O UCF_CC_50 possui 50 imagens de fontes e resoluções variadas, com média de 1.279 pessoas por imagem. O Shanghai Tech - parte A (SHA) é um *dataset* com média de 501 pessoas por imagem, composto por 482 imagens de diferentes fontes e resoluções. O Mall é um conjunto de 2.000 imagens de câmeras de segurança de um shopping com resolução de 640×480, com média de 31 pessoas por imagem. O VisDrone é um dataset composto por 3.360 imagens de *drone* com uma média de 145 pessoas por imagem e resolução de 1920×1080.

A Tabela III-A resume as principais características de cada *dataset*.

TABELA II
CARACTERÍSTICAS DOS DATASETS.

Dataset	Contagem média de pessoas	Resolução	Total de Imagens
SHA	501	-	482
UCSD	25	238×158	2000
UCF_CC_50	1279	-	50
Mall	31	640×480	2000
VisDrone	145	1920×1080	3360

B. Mapa de densidade

Todos os *datasets* considerados neste trabalho possuem anotações que representam os pontos da imagem identificados como pessoas. Essas anotações são utilizadas para gerar mapas de densidade de cada imagem. Os mapas são utilizados no treinamento das redes e a integral de cada mapa corresponde ao total de pessoas em cada imagem.

Cada pessoa em uma imagem pode ser representada por um impulso unitário centrado no pixel anotado:

$$\delta(\boldsymbol{x} - \boldsymbol{x_i}), \tag{1}$$

onde $\boldsymbol{x_i}$ é um vetor de duas dimensões que representa a i-ésima pessoa em uma imagem.

Dessa forma, é possível representar uma imagem como

$$H(\boldsymbol{x}) = \sum_{i=1}^{N} \delta(\boldsymbol{x} - \boldsymbol{x_i}), \tag{2}$$

onde N é o número total de pessoas na imagem. O mapa de densidade é gerado fazendo a convolução de $H(\boldsymbol{x})$ com um *kernel* Gaussiano G_σ [20]. O parâmetro σ é um parâmetro de escala que define o espalhamento do mapa de densidade. A abordagem mais simples consiste em fixar o valor desse parâmetro. Porém, em casos mais complexos, como o caso de variações nas escalas das imagens, é necessário usar um mapa com *kernel* adaptativo.

Para gerar um mapa que considere variações nas escala, este trabalho utiliza a técnica de filtros adaptativos à geometria, proposta por Zhang et al. [5], na qual o mapa é dado por

$$F(\boldsymbol{x}) = \sum_{i=1}^{N} \delta(\boldsymbol{x} - \boldsymbol{x_i}) * G_{\sigma_i}, \tag{3}$$

com $\sigma_i = \beta \bar{m}_i$, sendo β um parâmetro de escala e $\bar{m}_i$ a média das distâncias do i-ésimo ponto para seus k vizinhos mais próximos.

Todos os mapas de densidade deste trabalhao são gerados utilizando o kernel adaptativo da Equação 3, considerando os 4 vizinhos mais próximos para geração de $\bar{m}_i$ e $\beta = 0.3$, como sugerido pelos autores [5]. A Figura III-B ilustra o mapa de densidade da Figura 2, gerado utilizando esse processo. É possível observar que áreas mais densas são representadas por pontos com maior valor (pontos com cores mais próximas do amarelo)

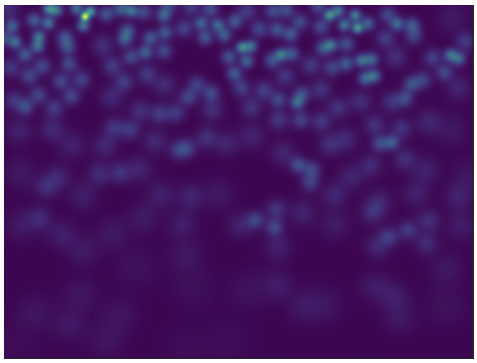

Fig. 3. Mapa de densidade gerado a partir da Figura 2.

C. Métrica de avaliação

O número de pessoas em cada imagem é dado pela integral do mapa de densidade. Para avaliação do desempenho do modelo é utilizado o MAE, definido como:

$$MAE = \frac{1}{P}\sum_{i=1}^{P}|z_i - \hat{z}_i|, \tag{4}$$

onde P é o número total de imagens, z_i é o número de pessoas na i-ésima imagem e $\hat{z}_i$ é o número de pessoas previsto pelo modelo para a i-ésima imagem.

D. Modelo

Este trabalho utiliza o MCNN por ser um dos modelos considerados estado-da-arte mais citados na literatura e por ter implementação disponível abertamente. Além disso, o modelo permite que as entradas sejam de tamanho arbitrário, o que reduz distorções nos mapas de densidade causadas pelo redimensionamento das imagens.

Como citado anteriormente, o MCNN é formado por três colunas de CNNs que são treinadas em paralelo com filtros diferentes entre si. A saída da rede é a média das previsões individuais de cada coluna. Essa arquitetura permite que variações de escala intra-imagem e diferenças de resolução sejam aprendidadas de maneira adaptativa.

A Figura III-D ilustra a estrutura da rede convolucional utilizada no modelo. As estruturas de cada uma das colunas só difere no tamanho e na quantidade de filtros [5]. A função de ativação utilizada nas camadas convolucionais é a ReLU. A função perda utilizada é o erro médio quadrático (MSE), definida por:

$$L(\mathbf{\Omega}) = \frac{1}{N}\sum_{i=1}^{N}(F(\mathbf{\Omega}, \boldsymbol{X_i}) - F_i)^2, \tag{5}$$

onde F_i representa o mapa de densidade para a i-ésima imagem, $F(\mathbf{\Omega}, \boldsymbol{X_i})$ representa o mapa de densidade previsto pela rede a partir dos parâmetros $\mathbf{\Omega}$ da rede e da i-ésima imagem $\boldsymbol{X_i}$, e N representa o total de imagens utilizadas no treinamento. Como as imagens são de tamanho arbitrário, é utilizado um *batch* de tamanho um.

IV. Experimentos e Resultados

Os experimentos consistem na geração de cinco cenários nos quais o modelo é treinado com os dados de um *dataset* e avaliado com os conjuntos de testes dos demais. Em cada um dos cenários, são realizadas 100 épocas de treinamento com *learning rate* de 10^{-8}. O *learning rate* foi definido por meio de uma otimização de hiper-parâmetros utilizando o conjunto de treinamento do SHA. Não foi utilizado conjunto de validação devido às restrições de tamanho dos *datasets*.

A Tabela IV mostra o MAE obtido para cada cenário. Cada coluna da tabela representa um *dataset* utilizado para teste. Nos casos dos *datasets* mais densos (SHA, UCF_50, Drone), isto é, com maior contagem de pessoas por imagem, o valor do MAE na coluna é mais elevado pois existem mais pessoas na multidão. Nos demais casos, espera-se um MAE menor, dado que existem menos pessoas nas imagens. Os valores em destaque representam os casos no qual os conjuntos de treino e teste pertencem ao mesmo *dataset*, respeitando o princípio de que não há interseção entre os dois conjuntos.

Analisando os resultados expostos na Tabela IV, é possível notar que o desempenho do modelo nos *datasets* SHA e UCF_50 apresenta grande variância, o que indica que o modelo apresentou dificuldade na generalização. Mesmo no caso em que o *dataset* utilizado é o mesmo nos conjuntos de treinamento e teste, o modelo apresentou essa dificuldade, o que indica um possível sobreajuste. Esse comportamento pode ser explicado pelo fato de que esses *datasets* possuem poucas imagens, o que pode prejudicar o treinamento.

Para os demais *datasets*, é possível notar que há uma perda de desempenho considerável nos casos em que o conjunto de testes não pertence ao mesmo *dataset* do conjunto de treinamento. Para o Mall, nota-se que o MAE do segundo melhor modelo é mais de 2 vezes maior do que no cenário no qual o modelo é treinado com seus dados, e no pior caso o erro é 60 vezes maior. Para o UCSD, os erros ficam em média 5,19 vezes maiores nos casos em que o modelo não é treinado com as imagens desse *dataset*. E para o VisDrone, o erro no segundo melhor cenário 2 vezes maior que o do melhor cenário.

V. Conclusões e Trabalhos Futuros

Neste trabalho, foi analisada a generalização do modelo estado-da-arte MCNN no problema de contagem de multidão. Para tal, utilizaram-se cinco *datasets* distintos. Os experimentos mostram que há uma perda de desempenho significativa quando o modelo é confrontado com dados pertencentes a conjuntos de imagens diferentes daqueles usados para o treinamento.

Um caminho para trabalhos futuros é estender a análise para diferentes modelos, considerando modelos mais recentes com outras arquiteturas, e incluir *datasets* ainda mais densos.

Referências

[1] M. A. Khan, H. Menouar, and R. Hamila, "Revisiting crowd counting: State-of-the-art, trends, and future perspectives," *Image and Vision Computing*, vol. 129, p. 104597, 2023. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0262885622002268

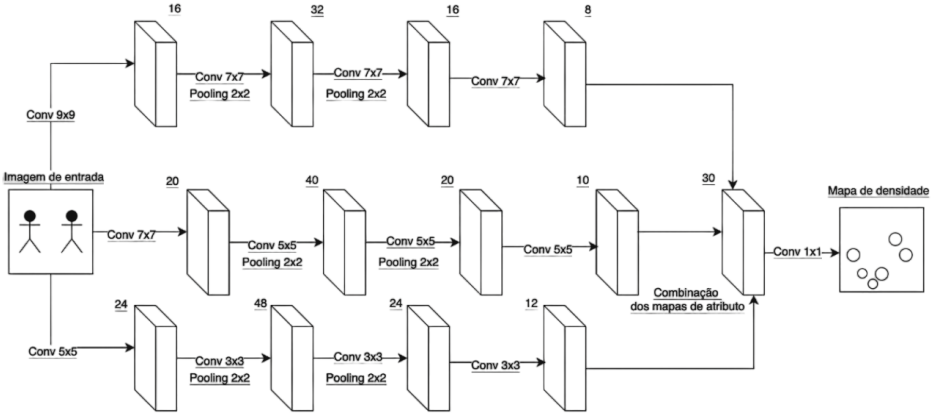

Fig. 4. Estrutura da rede do modelo MCNN. Adaptado de [5].

TABELA III

MAE PARA CADA CENÁRIO COM INTERVALO DE CONFIANÇA DE 95%.

	Teste				
Treinamento	SHA	Mall	UCSD	UCF_50	VisDrone
SHA	**294,96±57,15**	88,58±37,24	24,60±4,28	698,61±148,46	996,64±332,36
Mall	325,06±38,51	**4,34±0,24**	31,48±1,08	1131,62±184,61	140,00±15,26
UCSD	342,68±125,58	262,93±139,66	**5,66±0,06**	916,74±157,09	1832,90±854,52
UCF_50	537,24±137,11	216,14±125,74	34,85±9,72	**808,57±110,06**	1642,87±594,04
VisDrone	373,97±35,90	9,59±1,58	26,51±0,19	1212,75±189,11	**70,28±2,87**

[2] Z. Lin and L. S. Davis, "Shape-based human detection and segmentation via hierarchical part-template matching," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 4, pp. 604–618, 2010.

[3] B. Leibe, E. Seemann, and B. Schiele, "Pedestrian detection in crowded scenes," in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)*, vol. 1, 2005, pp. 878–885 vol. 1.

[4] I. S. Topkaya, H. Erdogan, and F. Porikli, "Counting people by clustering person detector outputs," in *2014 11th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*, 2014, pp. 313–318.

[5] Y. Zhang, D. Zhou, S. Chen, S. Gao, and Y. Ma, "Single-image crowd counting via multi-column convolutional neural network," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 589–597.

[6] Z. Yan, P. Li, B. Wang, D. Ren, and W. Zuo, "Towards learning multi-domain crowd counting," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 33, no. 11, pp. 6544–6557, 2023.

[7] Y. Pang, Z. Ni, and X. Zhong, "Federated learning for crowd counting in smart surveillance systems," *IEEE Internet of Things Journal*, vol. 11, no. 3, pp. 5200–5209, 2024.

[8] D. Liang, X. Chen, W. Xu, Y. Zhou, and X. Bai, "Transcrowd: Weakly-supervised crowd counting with transformer," *CoRR*, vol. abs/2104.09116, 2021. [Online]. Available: https://arxiv.org/abs/2104.09116

[9] Z. Du, J. Deng, and M. Shi, "Domain-general crowd counting in unseen scenarios," in *Proceedings of the Thirty-Seventh AAAI Conference on Artificial Intelligence and Thirty-Fifth Conference on Innovative Applications of Artificial Intelligence and Thirteenth Symposium on Educational Advances in Artificial Intelligence*, ser. AAAI'23/IAAI'23/EAAI'23. AAAI Press, 2023. [Online]. Available: https://doi.org/10.1609/aaai.v37i1.25131

[10] X. Hou, J. Xu, J. Wu, and H. Xu, "Cross domain adaptation of crowd counting with model-agnostic meta-learning," *Applied Sciences*, vol. 11, no. 24, 2021. [Online]. Available: https://www.mdpi.com/2076-3417/11/24/12037

[11] N. Sharma, V. Jain, and A. Mishra, "An analysis of convolutional neural networks for image classification," *Procedia Computer Science*, vol. 132, pp. 377–384, 2018, international Conference on Computational Intelligence and Data Science. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S1877050918309335

[12] W. Liu, M. Salzmann, and P. Fua, "Context-aware crowd counting," in *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019, pp. 5094–5103.

[13] X. Jiang, L. Zhang, M. Xu, T. Zhang, P. Lv, B. Zhou, X. Yang, and Y. Pang, "Attention scaling for crowd counting," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

[14] Y. Li, X. Zhang, and D. Chen, "Csrnet: Dilated convolutional neural networks for understanding the highly congested scenes," 06 2018, pp. 1091–1100.

[15] H. Idrees, I. Saleemi, C. Seibert, and M. Shah, "Multi-source multi-scale counting in extremely dense crowd images," in *2013 IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 2547–2554.

[16] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, A. Desmaison, A. Kopf, E. Yang, Z. DeVito, M. Raison, A. Tejani, S. Chilamkurthy, B. Steiner, L. Fang, J. Bai, and S. Chintala, "Pytorch: An imperative style, high-performance deep learning library," in *Advances in Neural Information Processing Systems 32*. Curran Associates, Inc., 2019, pp. 8024–8035.

[17] A. B. Chan, Z.-S. J. Liang, and N. Vasconcelos, "Privacy preserving crowd monitoring: Counting people without people models or tracking," in *2008 IEEE Conference on Computer Vision and Pattern Recognition*, 2008, pp. 1–7.

[18] K. Chen, C. C. Loy, S. Gong, and T. Xiang, "Feature mining for localised crowd counting," in *British Machine Vision Conference*, 2012.

[19] P. Zhu, L. Wen, D. Du, X. Bian, H. Fan, Q. Hu, and H. Ling, "Detection and tracking meet drones challenge," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 11, pp. 7380–7399, 2021.

[20] V. S. Lempitsky and A. Zisserman, "Learning to count objects in images," in *Neural Information Processing Systems*, 2010.