SELF-SUPERVISED DOMAIN ADAPTATION IN CROWD COUNTING

Pha Nguyen[1]*, Thanh-Dat Truong*[1]*, Miaoqing Huang*[1]*, Yi Liang*[2]*, Ngan Le*[1]*, Khoa Luu*[1]

[1] Department of CSCE, University of Arkansas, Fayetteville, AR, USA
[2] Department of Biological & Agricultural Engineering, University of Arkansas, Fayetteville, AR, USA
{panguyen, tt032, mqhuang, yliang, thile, khoaluu}@uark.edu

2022 IEEE International Conference on Image Processing (ICIP) | 978-1-6654-9620-9/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICIP46576.2022.9897440

ABSTRACT

Self-training crowd counting has not been attentively explored though it is one of the important challenges in computer vision. In practice, the fully supervised methods usually require an intensive resource of manual annotation. In order to address this challenge, this work introduces a new approach to utilize existing datasets with ground truth to produce more robust predictions on unlabeled datasets, named domain adaptation, in crowd counting. While the network is trained with labeled data, samples without labels from the target domain are also added to the training process. In this process, the entropy map is computed and minimized in addition to the adversarial training process designed in parallel. Experiments on Shanghaitech, UCF_CC_50, and UCF-QNRF datasets prove a more generalized improvement of our method over the other state-of-the-arts in the cross-domain setting.

Index Terms— Crowd Counting, Domain Adaptation, Entropy Minimization, Adversarial Learning.

1. INTRODUCTION

Crowd counting has recently been one of the popular tasks in computer vision. Recent developed methods [1, 2, 3] and datasets [4, 5, 6] have been introduced to tackle the counting task with thousands of targets. However, in real-world scenarios, these supervised methods usually learn to count through a training process that requires an extensive annotation of densely populated points in thousands of images. Directly employing models that are trained on existing datasets to a new dataset suffers from a significant performance decrease due to the domain gap.

Therefore, in addition to semantic scene understanding [7] and video temporal modeling [8, 9, 10, 11], some self-training methods appear to utilize existing datasets with labels, i.e. source domain, and perform counting on more open-set scenarios, i.e. target domain, [12, 13] by transfer learning and domain adaptation techniques. Liu et al. [13] enable knowledge distillation between both regression-based and detection-based models by formulating the mutual transformation of outputs. Xu et al. [14] enhance the generalization over density variance by categorizing image patches into several density levels. While general self learning methods improve the generalization capability by attempting to estimate pseudo ground-truths or distillation learning from a teacher network, a few approaches investigate a new direction to narrow the domain shift from entropy feedback of the target domain, especially in the semantic segmentation task [15].

In this paper, we introduce a new training approach to the crowd counting task toward a domain adaptation setting where the crowd counter utilizes the entropy minimization and adversarial learning to alleviate the distributional discrepancy between the source domain and the target domain. Particularly, our contributions can be summarized as follows:

- Reformulate the crowd counting problem from normally estimating density map to directly predicting target points in images, inspired by anchor-based and offset-based approaches.
- Utilize the Shannon entropy formula as a loss objective function to maximize the prediction certainty.
- Design an adversarial learning scheme to motivate the network to produce similar distributional predictions over the source domain and the target domain.
- Evaluate the proposed method with cross-domain settings to demonstrate its substantial generalization compared against the previous crowd counting methods and further perform estimating on a new chicken counting dataset.

2. DOMAIN ADAPTATION FOR CROWD COUNTING

2.1. Point Proposal Network

Far apart from prior approaches that normally learn to predict a density map [2, 16], this work designs a network to estimate head points directly. Given an RGB image $\mathbf{x} \in \mathcal{X}$, the training source domain, the deep feature extracted from the backbone network $\mathcal{F}$ can be denoted as $\mathcal{F}(\mathbf{x})$ and its output size is $W \times H \times D$. $\mathcal{F}(\mathbf{x})$ involves a hyper-parameter s that is the backbone's downscale stride. In particular, each cell on the feature map $\mathcal{F}(\mathbf{x})$ basically is correspondence to a window size

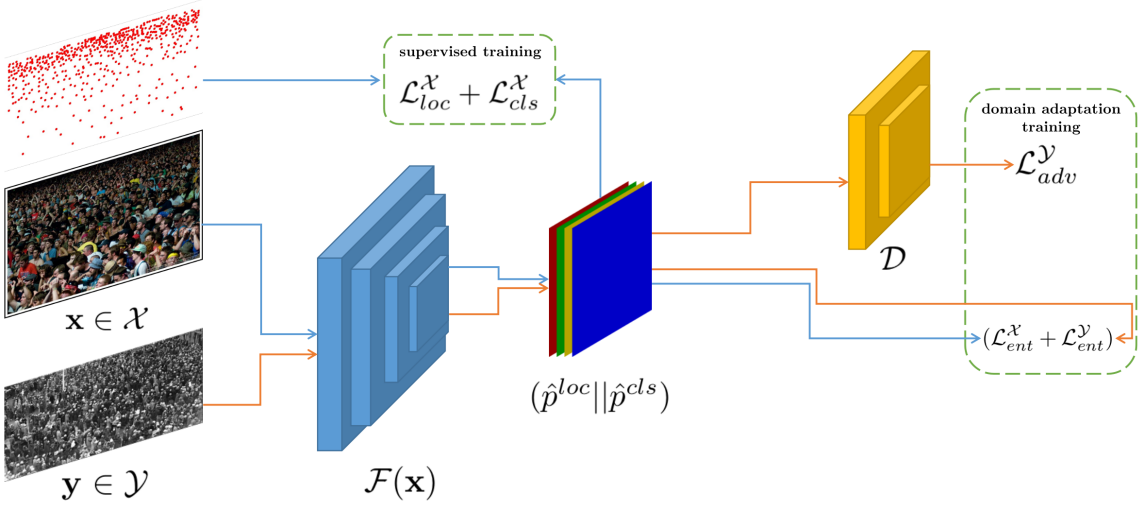

Fig. 1. Our overall framework: Given an image sample, the deep network first extracts $\mathcal{F}(\mathbf{x})$ feature, then estimates location offset map and classification map $(\hat{p}^{loc}, \hat{p}^{cls})$. With source domain sample $\mathbf{x} \in \mathcal{X}$, since label is available, supervised L2 Distance $\mathcal{L}_{loc}^{\mathcal{X}}$ loss and Cross Entropy $\mathcal{L}_{cls}^{\mathcal{X}}$ loss can be effortlessly calculated and they are used to guide the network. On the other hand, since sample on target domain $\mathbf{y} \in \mathcal{Y}$ does not have label, $\mathcal{L}_{ent}^{\mathcal{X}}, \mathcal{L}_{ent}^{\mathcal{Y}}, \mathcal{L}_{adv}^{\mathcal{Y}}$ loss functions are emloyed to additionally teach the domain adaptation learning process. Blue arrows indicate source sample's learning flow, while orange arrows indicate the learning flow of target sample.

$s \times s$ on the original input $\mathbf{x}$. The maximum number of points that can exist in the window is D (point's index is denoted as $k, k \in [0, D-1]$). Then, given the processed feature map $\mathcal{F}(\mathbf{x})$, two network branches are adopted to predict the point coordinate (denoted as $\hat{p}^{loc}$) and background-foreground classification (denoted as $\hat{p}^{cls}$). From the location (i, j) where the pixel is located in the feature map $\mathcal{F}(\mathbf{x})$, the regression branch learns to estimate $2 \times k$ offset values $(\delta_{ik}, \delta_{jk})$ in the range $[-1, 1]$. The point location $\hat{p}_{i,j,k}^{loc} = (\hat{x}_k, \hat{y}_k)$ is computed as follows:

$$\begin{aligned} \hat{x}_k &= s(i + \delta_{ik}) \\ \hat{y}_k &= s(j + \delta_{jk}) \end{aligned} \tag{1}$$

In the classification task, two predicted scores belong to positive class pos_k (object's point) and negative class neg_k (background). The Softmax function is employed to normalize two confident scores $\hat{p}_{i,j,k}^{cls} = (\hat{cls}_k^{pos}, \hat{cls}_k^{neg})$ that follow a probability distribution whose total sums up to one:

$$\begin{aligned} \hat{cls}_k^{pos} &= \frac{e^{pos_k}}{e^{pos_k} + e^{neg_k}} \\ \hat{cls}_k^{neg} &= \frac{e^{neg_k}}{e^{pos_k} + e^{neg_k}} \end{aligned} \tag{2}$$

Supervised Training Losses. On the source domain $\mathcal{X}$ where labels are provided, the supervised training losses on both branches are formulated as the standard ones. The ℓ_2 distance and Cross Entropy losses are adopted for the regression branch and the classification branch, respectively. Denoting $p_i^{loc}, cls_i^{pos}, cls_i^{neg}$ as corresponding ground-truth values of $\hat{p}_i^{loc}, \hat{cls}_i^{pos}, \hat{cls}_i^{neg}$, those loss functions are defined as follows:

$$\mathcal{L}_{loc}(\mathbf{x}) = \frac{1}{|N|} \sum_{i=1}^{|N|} ||\hat{p}_i^{loc} - p_i^{loc}||_2 \tag{3}$$

$$\mathcal{L}_{cls}(\mathbf{x}) = -\frac{1}{|M|} \sum_{i=1}^{|M|} (cls_i^{pos} \log \hat{cls}_i^{pos} + cls_i^{neg} \log \hat{cls}_i^{neg}) \tag{4}$$

where N is the set of points of the ground truth and M is the set of proposals containing both negative and positive pixel points. M can be obtained from a one-to-one matching strategy (i.e. Hungarian algorithm [17, 18, 3]). Finally, the fully supervised training loss can be obtained as follows:

$$\mathcal{L}_{loc}^{\mathcal{X}} + \mathcal{L}_{cls}^{\mathcal{X}} \tag{5}$$

where $\mathcal{L}^{\mathcal{X}}$ denotes a particular loss calculated on all samples from the source domain $\mathcal{X}$.

2.2. Entropy Minimization on Target Domain

On the target domain $\mathcal{Y}$, where labels are not available, while some approaches utilize output from a teacher model as a pseudo-label with lower confidence to guide the learning process [19, 20, 21], entropy minimization is a more preferable principle in self-training semantic segmentation demonstrated through a number of research works [15, 22, 23]. By formulating the point's head classification similar to the semantic

segmentation problem, the Shannon entropy formulation [24] can be adopted to be a loss function in order to encourage the deep network to produce a higher confidence score. Given an RGB image $\mathbf{y} \in \mathcal{Y}$ on the target domain, the classification per pixel entropy can be formulated as follows:

$$\mathcal{E}(\mathbf{y})_{i,j,k} = \frac{-1}{\log 2}(\hat{cls}_k^{pos} \log \hat{cls}_k^{pos} + \hat{cls}_k^{neg} \log \hat{cls}_k^{neg}) \quad (6)$$

And the self-training entropy loss can be defined as:

$$\mathcal{L}_{ent}(\mathbf{y}) = \frac{1}{W \times H \times D} \sum_{i}^{W} \sum_{j}^{H} \sum_{k}^{D} \mathcal{E}(\mathbf{y})_{i,j,k} \quad (7)$$

2.3. Distribution Discrepancy Minimization by Adversarial Learning

To further narrow the domain gap, we utilize a discriminator $\mathcal{D}$, which is a fully convolutional neural network classifier, to motivate the network to extract similar distribution output over both domains. This discriminator tries to determine which domain the input belongs to by learning domain classification $(\mathcal{D}_{\mathcal{X}}, \mathcal{D}_{\mathcal{Y}})$, while the main network tries to make the discriminator produce fault predictions. Given the concatenation of offset and category maps from the network $(\hat{p}^{loc}||\hat{p}^{cls})$, the loss function of the discriminator can be formulated as follows,

$$\mathcal{L}_{dis}(\hat{p}^{loc}||\hat{p}^{cls}) = -\sum_{i}^{W}\sum_{j}^{H}[(1-z)\log \mathcal{D}_{\mathcal{X}}(\hat{p}^{loc}||\hat{p}^{cls}) + z \log \mathcal{D}_{\mathcal{Y}}(\hat{p}^{loc}||\hat{p}^{cls})] \quad (8)$$

where $z = 0$ if $\hat{p} \equiv \mathcal{F}(\mathbf{x})$ or $z = 1$ if $\hat{p} \equiv \mathcal{F}(\mathbf{y})$, which $\mathbf{x} \in \mathcal{X}, \mathbf{y} \in \mathcal{Y}$, and $(.||.)$ is the tensor concatenation operation.

Additionally, to narrow the produced distributions of source domain and the target domain, we add an adversarial loss in the main network's training process:

$$\mathcal{L}_{adv}(\mathbf{y}) = -\sum_{i}^{W}\sum_{j}^{H}[\log \mathcal{D}_{\mathcal{X}}(\hat{p}^{loc}_{\mathbf{y}}||\hat{p}^{cls}_{\mathbf{y}})] \quad (9)$$

More specifically, the adversarial loss is designed to maximize the probability of the discriminator predicting source domain class given target domain samples $\mathbf{y} \in \mathcal{Y}$.

To summarize, the learning process of the main point proposal network involves Eqn. 3, 4, 7 and 9 loss functions:

$$\lambda_{loc}\mathcal{L}^{\mathcal{X}}_{loc} + \lambda_{cls}\mathcal{L}^{\mathcal{X}}_{cls} + \lambda_{ent}(\mathcal{L}^{\mathcal{X}}_{ent} + \mathcal{L}^{\mathcal{Y}}_{ent}) + \lambda_{adv}\mathcal{L}^{\mathcal{Y}}_{adv} \quad (10)$$

whereλ_{loc}, λ_{cls}, λ_{ent}, λ_{adv} are weighted parameters to balance corresponding objective functions, $\mathcal{L}^{\mathcal{X}}$ and $\mathcal{L}^{\mathcal{Y}}$ denote particular losses calculated on all samples from domain $\mathcal{X}$ and $\mathcal{Y}$, respectively. In parallel, the discriminator $\mathcal{D}$ learns with the guidance of Eqn. 8:

$$\mathcal{L}^{\mathcal{X}}_{dis} + \mathcal{L}^{\mathcal{Y}}_{dis} \quad (11)$$

The entire training procedure is depicted as in Fig. 1.

Table 1. Error rates comparison among loss components. Numbers in italic indicate error rates on source domain, while underlined numbers are results on adapted domain.

Components	SHTechA		SHTechB	
	MAE	MSE	MAE	MSE
$\mathcal{L}^{\mathcal{X}}_{ent}$	*54.32* 162.78	*90.39* 289.47	25.36 *7.92*	39.14 *11.53*
$\mathcal{L}^{\mathcal{Y}}_{ent}$	*60.76* 105.48	*95.34* 164.36	22.03 *10.43*	34.27 *15.60*
$\mathcal{L}^{\mathcal{X}}_{ent} + \mathcal{L}^{\mathcal{Y}}_{ent}$	*54.04* 87.76	*89.37* 126.53	21.58 *8.03*	30.84 *11.98*
$\mathcal{L}^{\mathcal{Y}}_{adv}$	*62.83* 174.59	*107.42* 302.87	28.39 *15.57*	47.58 *27.38*
$\mathcal{L}^{\mathcal{X}}_{ent} + \mathcal{L}^{\mathcal{Y}}_{ent} + \mathcal{L}^{\mathcal{Y}}_{adv}$	*57.67* **69.21**	*93.71* **95.36**	**18.29** *8.72*	**26.21** *12.53*

Table 2. Error rates comparison between our approach with other domain adaptation (DA) and supervised methods. Numbers in italic indicate error rates on source domain, while underlined numbers are results on adapted domain.

Method	DA	SHTechA		SHTechB	
		MAE	MSE	MAE	MSE
DM-Count [1]	✗	*60.04* 142.00	*96.01* 241.02	22.91 *7.33*	34.69 *11.87*
UEPNet [2]	✗	*55.26* -	*91.94* -	24.36 *6.38*	37.22 *10.88*
P2P [3]	✗	*53.02* 158.30	*88.48* 267.51	21.91 *6.55*	33.86 *9.50*
ConvNets [12]	✓	*73.5* 140.4	*112.3* 226.1	49.1 *18.7*	99.2 *26.0*
SPN+L2SM [14]	✓	*64.2* 126.8	*98.4* 203.9	21.2 *7.2*	38.7 *11.1*
RDBT [13]	✓	- 112.24	- 218.18	**13.38** -	29.25 -
Ours	✓	*57.67* **69.21**	*93.71* **95.36**	18.29 *8.72*	**26.21** *12.53*

3. EXPERIMENTAL RESULTS

3.1. Ablation Study

To illustrate the effectiveness of each proposed objective loss in our method, we conduct the ablative experiments as shown in Tab. 1. We slightly add and remove our training strategies on top of the original supervised approach. The experimental results have shown that our proposed losses have achieved significant improvement.

3.2. Comparison against SOTA Methods on Public Datasets

Shanghaitech Dataset [4] consists of two parts: Part-A and Part-B and it contains totally 1,198 images of 330,165 peo-

Table 3. Error rates comparison between our approach with other domain adaptation (DA) and supervised methods.

Method	DA	UCF_CC_50		UCF-QNRF	
		MAE	MSE	MAE	MSE
DM-Count [1]	✗	427.16	638.92	315.94	542.23
ConvNets [12]	✓	364.0	545.8	-	-
SPN+L2SM [14]	✓	332.4	425.0	227.2	405.2
RDBT [13]	✓	368.01	518.92	175.02	294.76
Ours	✓	**305.57**	**400.62**	**154.73**	**237.84**

ple. We use these two parts to take turns as source and target domains as shown in Tab. 2. In each method, the first row is using SHTechA for the source domain, SHTechB for the target domain, and the second row is trained in reversed order. The results show that, with domain adaptation learning, our method can be aware of the target's distribution, and yields better quantitative results on its samples (69.21/95.36 vs 112.24/218.18 of RDBT [13] on SHTechA), while the performance on source domain is not hurt very much (57.67/93.71 vs 53.02/88.48 on SHTechA and 8.72/12.53 vs 6.55/9.50 on SHTechB of P2P [3]).

UCF_CC_50 dataset [5] and **UCF-QNRF dataset** [6] have a large variant number of head counts. While the former only contains 50 images but the number of head points varies from 94 to 4,543, the latter consists of 1,535 images with 1,251,642 point heads in total. We use Shanghaitech Part-A for the source domain to adapt on these two datasets. The results also prove our method with domain adaptation perform superior quantitative results on target domain as shown in Tab. 3 (305.57/400.62 vs 332.4/425.0 of SPN+L2SM [14] on UCF_CC_50) and (154.73/237.84 vs 227.2/405.2 of SPN+L2SM [14] on UCF-QNRF).

3.3. Qualitative Result on Chicken Counting

We want to evaluate the proposed training method on our chicken dataset collected in farm scenes which have not been annotated as shown in Fig. 2. The dataset will be annotated and soon publicly release a test set for quantitative evaluation. We train the SHTech dataset as the source domain and try different domain adaptation training strategies on this dataset.

The first row is the training process with entropy minimization on the target domain. Since the network is mainly guided to learn the localization and classification tasks from the human dataset, the network finds it difficult to recognize chickens as positive class and the result mostly returns false negatives. The second row is the training process with adversarial loss. While the distribution gap is more narrow resulting in more densely populated prediction, the network produces more false positives by trying to map the dense distribution of the source domain. The final training process balances those loss functions with weighted parameters and refines better results. However, it still does not yield optimal predictions and there are some missing counts caused by different lighting conditions (i.e. darker and brighter areas in top-left and bottom-left corners).

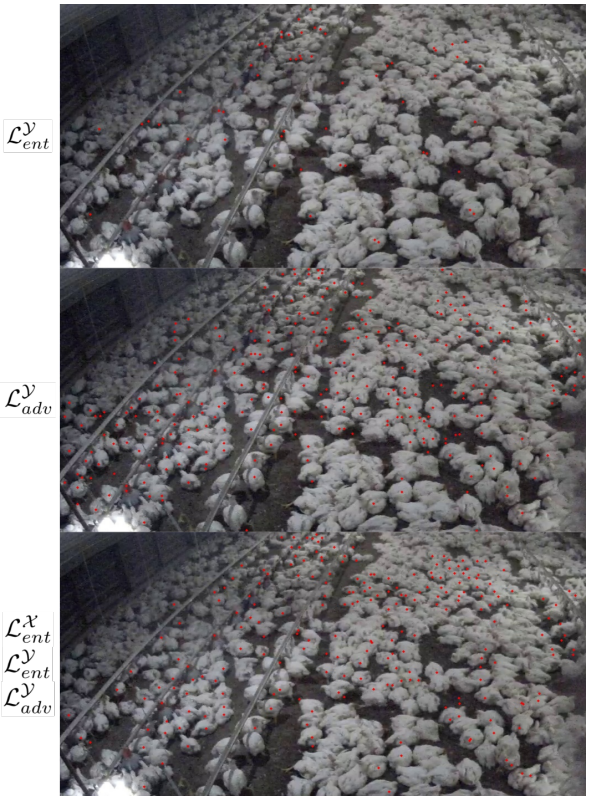

Fig. 2. Our qualitative result on our chicken dataset with different domain adaptation training strategies (from top to bottom: entropy minimization loss; adversarial loss; both the losses). Best viewed in color and zoom in.

4. CONCLUSION

In this paper, we have proposed a domain adaptation training scheme for the crowd counting task. Our method is designed to minimize the domain gap between the source domain and the target domain through the entropy loss and the adversarial loss. The entropy minimization is computed on both domains while the adversarial objective minimizes the distribution discrepancy on target samples. As a result, our proposed method shows better results on the target domain than recent self-training learning methods, while maintaining nearly the same error rates on the source domain. Furthermore, we show qualitative estimation on our chicken dataset which is used as the target domain. However, there are still some false negative counts on chickens, due to the lighting condition problem which is not fully addressed in this work. The dataset will be released and the limitation will be studied more in future work.

Acknowledgement This work is supported by NSF Data Science, Data Analytics that are Robust and Trusted (DART) and the Chancellor's Innovation and Collaboration Fund from University of Arkansas Fayetteville.

5. REFERENCES

[1] Boyu Wang, Huidong Liu, Dimitris Samaras, and Minh Hoai, "Distribution matching for crowd counting," in *Advances in Neural Information Processing Systems*, 2020. 1, 3, 4

[2] Changan Wang, Qingyu Song, Boshen Zhang, Yabiao Wang, Ying Tai, Xuyi Hu, Chengjie Wang, Jilin Li, Jiayi Ma, and Yang Wu, "Uniformity in heterogeneity: Diving deep into count interval partition for crowd counting," 2021. 1, 3

[3] Qingyu Song, Changan Wang, Zhengkai Jiang, Yabiao Wang, Ying Tai, Chengjie Wang, Jilin Li, Feiyue Huang, and Yang Wu, "Rethinking counting and localization in crowds: A purely point-based framework," 2021. 1, 2, 3, 4

[4] Yingying Zhang, Desen Zhou, Siqin Chen, Shenghua Gao, and Yi 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. 1, 3

[5] Haroon Idrees, Imran Saleemi, Cody Seibert, and Mubarak 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. 1, 4

[6] Haroon Idrees, Muhmmad Tayyab, Kishan Athrey, Dong Zhang, Somaya Al-Maadeed, Nasir Rajpoot, and Mubarak Shah, "Composition loss for counting, density map estimation and localization in dense crowds," in *Computer Vision – ECCV 2018*, Vittorio Ferrari, Martial Hebert, Cristian Sminchisescu, and Yair Weiss, Eds., Cham, 2018, pp. 544–559, Springer International Publishing. 1, 4

[7] T. Hoang Ngan Le, Kha Gia Quach, Khoa Luu, Chi Nhan Duong, and Marios Savvides, "Reformulating level sets as deep recurrent neural network approach to semantic segmentation," *TIP*, 2018. 1

[8] Chi Nhan Duong, Kha Gia Quach, Khoa Luu, T Hoang Ngan Le, Marios Savvides, and Tien D Bui, "Learning from longitudinal face demonstration—where tractable deep modeling meets inverse reinforcement learning," *IJCV*, 2019. 1

[9] Chi Nhan Duong, Khoa Luu, Kha Gia Quach, Nghia Nguyen, Eric Patterson, Tien D. Bui, and Ngan Le, "Automatic face aging in videos via deep reinforcement learning," in *CVPR*, 2019. 1

[10] Kha Gia Quach, Pha Nguyen, Huu Le, Thanh-Dat Truong, Chi Nhan Duong, Minh-Triet Tran, and Khoa Luu, "Dyglip: A dynamic graph model with link prediction for accurate multi-camera multiple object tracking," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2021, pp. 13784–13793. 1

[11] Thanh-Dat Truong, Quoc-Huy Bui, Chi Nhan Duong, Han-Seok Seo, Son Lam Phung, Xin Li, and Khoa Luu, "Direcformer: A directed attention in transformer approach to robust action recognition," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2022, pp. 20030–20040. 1

[12] Zenglin Shi, Le Zhang, Yun Liu, Xiaofeng Cao, Yangdong Ye, Ming-Ming Cheng, and Guoyan Zheng, "Crowd counting with deep negative correlation learning," in *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2018, pp. 5382–5390. 1, 3, 4

[13] Yuting Liu, Zheng Wang, Miaojing Shi, Shin'ichi Satoh, Qijun Zhao, and Hongyu Yang, "Towards unsupervised crowd counting via regression-detection bi-knowledge transfer," 2020. 1, 3, 4

[14] Chenfeng Xu, Kai Qiu, Jianlong Fu, Song Bai, Yongchao Xu, and Xiang Bai, "Learn to scale: Generating multipolar normalized density maps for crowd counting," 10 2019, pp. 8381–8389. 1, 3, 4

[15] Tuan-Hung Vu, Himalaya Jain, Maxime Bucher, Mathieu Cord, and Patrick Pérez, "Advent: Adversarial entropy minimization for domain adaptation in semantic segmentation," in *CVPR*, 2019. 1, 2

[16] Zhi-Qi Cheng, Jun-Xiu Li, Qi Dai, Xiao Wu, and Alexander Hauptmann, "Learning spatial awareness to improve crowd counting," 2019. 1

[17] Zijun Wei, Boyu Wang, Minh Hoai, Jianming Zhang, Xiaohui Shen, Zhe Lin, Radomír Měch, and Dimitris Samaras, "Sequence-to-segments networks for detecting segments in videos," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 43, no. 3, pp. 1009–1021, 2021. 2

[18] Russell Stewart, Mykhaylo Andriluka, and Andrew Y. Ng, "End-to-end people detection in crowded scenes," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 2325–2333. 2

[19] Vishwanath A. Sindagi, Rajeev Yasarla, Deepak Sam Babu, R. Venkatesh Babu, and Vishal M. Patel, "Learning to count in the crowd from limited labeled data," in *Computer Vision – ECCV 2020*, Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, Eds., Cham, 2020, pp. 212–229, Springer International Publishing. 2

[20] Yaqi Liu, Lingqiao Liu, Peng Wang, Pingping Zhang, and Yinjie Lei, "Semi-supervised crowd counting via self-training on surrogate tasks," *ArXiv*, vol. abs/2007.03207, 2020. 2

[21] Yanda Meng, Hongrun Zhang, Yitian Zhao, Xiaoyun Yang, Xuesheng Qian, Xiaowei Huang, and Yalin Zheng, "Spatial uncertainty-aware semi-supervised crowd counting," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2021, pp. 15549–15559. 2

[22] Fei Pan, Inkyu Shin, Francois Rameau, Seokju Lee, and In So Kweon, "Unsupervised intra-domain adaptation for semantic segmentation through self-supervision," in *IEEE Conference on Computer Vision and Pattern Recoginition (CVPR)*, 2020. 2

[23] Thanh-Dat Truong, Chi Nhan Duong, Ngan Le, Son Lam Phung, Chase Rainwater, and Khoa Luu, "Bimal: Bijective maximum likelihood approach to domain adaptation in semantic scene segmentation," in *International Conference on Computer Vision*, 2021. 2

[24] C. E. Shannon, "A mathematical theory of communication," *The Bell System Technical Journal*, vol. 27, no. 3, pp. 379–423, 1948. 3