Cross-View Cross-Scene Multi-View Crowd Counting Supplemental

Qi Zhang[1], Wei Lin[2], Antoni B. Chan[1]
[1] Department of Computer Science, City University of Hong Kong, Hong Kong SAR, China
{qzhang364-c@my., abchan@}cityu.edu.hk
[2]School of Computer Science and School of Artificial Intelligence,
Northwestern Polytechnical University, Xi'an, Shaanxi, P. R. China.
elonlin24@gmail.com

A. Additional ablation studies

Ablation study on combining MVMS+NoiseV. The ablation study results of combining MVMS [3] and noise view regularization are shown in Table 1. In general, combining MVMS with noise view regularization improves the performance of MVMS, but not as much as our CVCS, which includes the camera selection and noise-view regularization.

Single-view counting method CSRnet for multi-view counting task. We next test a single-view counting method, CSRnet [1], on the multi-view counting task. The CSRnet is trained on the synthetic data. The results are presented in Table 2 (left column). The ground-truth count is the people number covered by the corresponding N-cameras. Single-view counting performs poorly on the multi-view counting task, and the counting error increases as the counting region becomes larger (the region is covered by more cameras). The reason is that the scenes are large and wide, and cannot be fully covered by a single camera. Note the CSRnet we adopted performs normally on single-view counting (N=1); the MAE/NAE is 10.77/0.187 with ground-truth count number around 90-180, which is similar to the result reported by [1] on ShanghaiTech B (MAE of 10.6 with average GT number of 123.6).

Multi-view counting with CSRnet and density map weighting. We next test CSRnet [1] with the traditional multi-view counting method based on weighting density maps from different views (denoted by Dmap_weighted) [3, 2]. In particular, density maps predicted by CSRnet on the camera-views are fused into a scene-level count using weight maps, which are based on how many views can see a particular pixel.

The results on the synthetic dataset are presented in Table 2. Dmap_weighted fusion improves the performance, compared to using only a single camera. However, the performance of Dmap_weighted is still much worse than the proposed CVCS multi-view counting method. This shows that the simple multi-view fusion method cannot well handle the multi-view counting task even though the single-view counting method is sophisticated.

Table 1: Ablation study combining MVMS and noise view regularization.

Model	MAE	NAE
Backbone	14.13	0.115
+MVMS	9.30	0.080
+MVMS+NoiseV (Type D)	8.80	0.075
+MVMS+NoiseV (Type E)	8.78	0.074
+MVMS+NoiseV (Type F)	9.07	0.077
+MVMS+NoiseV (Type G)	9.00	0.076
Backbone+CamSel	8.63	0.074
CVCS (Backbone+CamSel+NoiseV)	**7.22**	**0.062**

B. Neural network setting

The layer setting details of the each neural networks module are shown in the Table 3-6.

C. Visualizations

The full-size visualization results on the synthetic and real datasets are presented in Fig. 1 and 2, respectively.

References

[1] Yuhong Li, Xiaofan Zhang, and Deming Chen. Csrnet: Dilated convolutional neural networks for understanding the highly congested scenes. In *CVPR*, pages 1091–1100, 2018. 1

[2] David Ryan, Simon Denman, Clinton Fookes, and Sridha Sridharan. Scene invariant multi camera crowd counting. *Pattern Recognition Letters*, 44(8):98–112, 2014. 1

[3] Qi Zhang and Antoni B Chan. Wide-area crowd counting via ground-plane density maps and multi-view fusion cnns. In *Computer Vision and Pattern Recognition*, pages 8297–8306, 2019. 1

Table 2: Ablation study on different numbers of input camera views. The ground-truth counting number is the people count covered by the N-cameras.

	1-camera		**N-cameras**					
	CSRnet		CSRnet+Dmap_wtd		Backbone		CVCS	
N Views	MAE	NAE	MAE	NAE	MAE	NAE	MAE	NAE
1	10.77	0.187	-	-	-	-	-	-
3	48.78	0.469	23.60	0.231	14.28	0.130	**7.24**	**0.071**
5	63.45	0.537	30.02	0.258	14.13	0.115	**7.22**	**0.062**
7	68.66	0.551	34.72	0.281	14.35	0.113	**7.07**	**0.058**
9	72.30	0.562	36.38	0.285	14.56	0.112	**7.04**	**0.056**
11	74.86	0.575	37.73	0.291	15.15	0.115	**7.00**	**0.055**

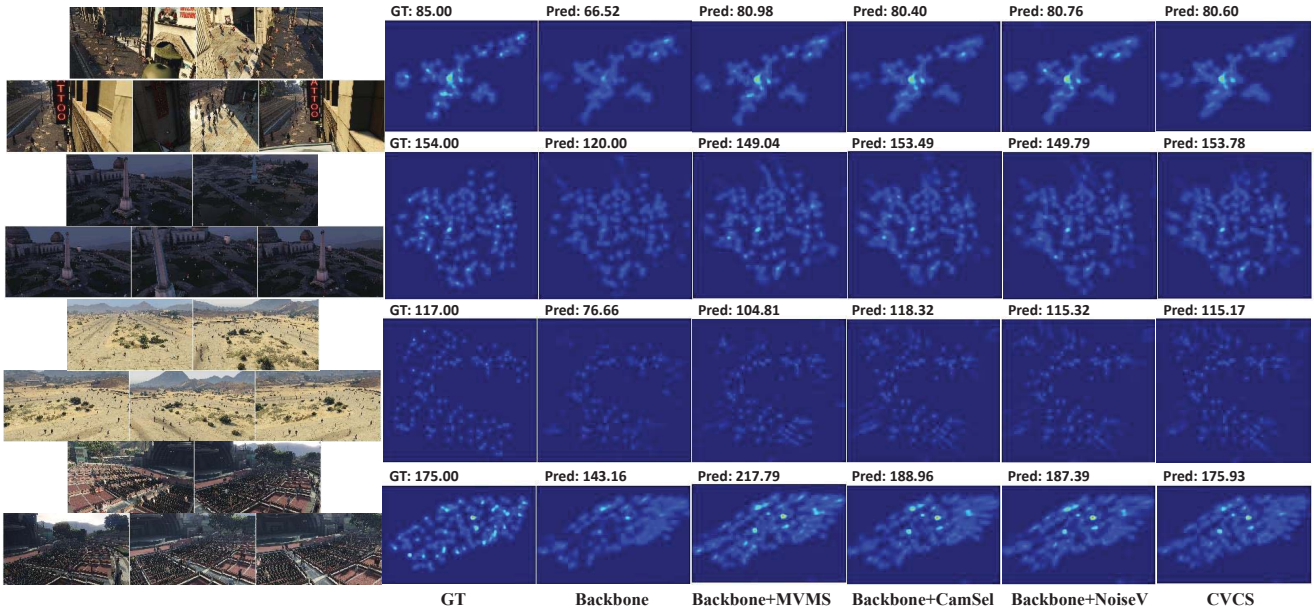

Figure 1: The results of CVCS variations on the synthetic dataset. Using camera selection and/or noise-view regularization (CVCS, Backbone+CamSel, Backbone+NoiseV) are more accurate than the backbone or backbone with MVMS (Backbone+MVMS).

Table 3: The layer settings for the single-view feature extraction. The filter dimensions are output channels, input channels, and filter size $w_0 \times h_0$.

Single-view feature extraction	
Layer	Filter
conv 1	$64\times3\times3\times3$
conv 2	$64\times64\times3\times3$
pooling	2×2
conv 3	$128\times64\times3\times3$
conv 4	$128\times128\times3\times3$
pooling	2×2
conv 5	$256\times128\times3\times3$
conv 6	$256\times256\times3\times3$
conv 7	$256\times256\times3\times3$

Table 4: The layer settings for the multi-view decoder.

Multi-view decoder	
Layer	Filter
conv 1	$512\times256\times3\times3$
conv 2	$512\times512\times3\times3$
conv 3	$512\times512\times3\times3$
conv 4	$256\times512\times3\times3$
conv 5	$128\times256\times3\times3$
conv 6	$64\times128\times3\times3$
conv 7	$1\times64\times3\times3$

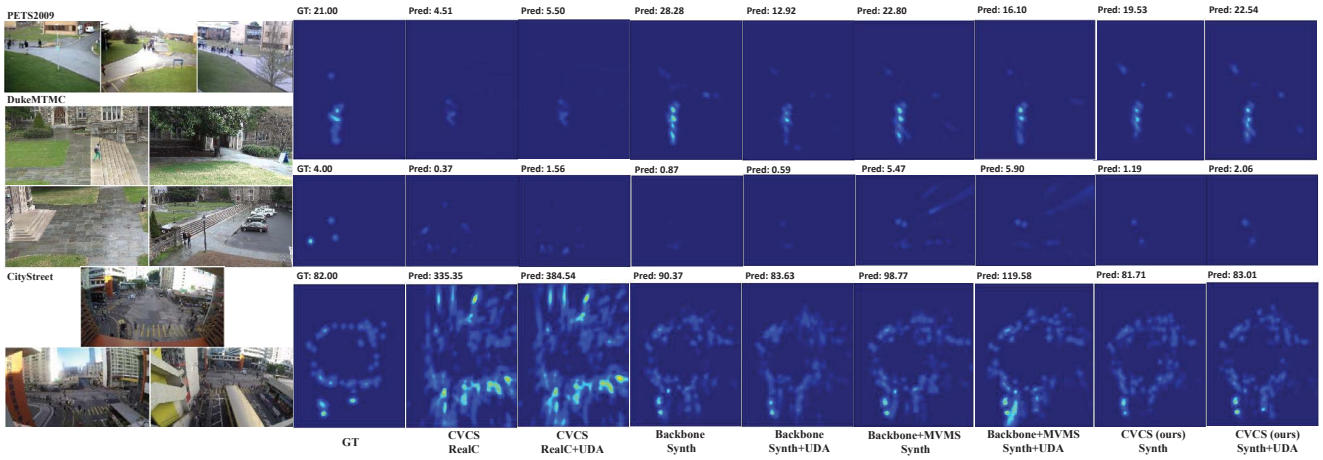

Figure 2: The cross-view cross-scene results on real datasets. Our CVCS model trained on the synthetic data shows better performance than CVCS trained on real data. Applying unsupervised domain adaptation (UDA) to our CVCS improves the performance.

Table 5: The layer settings for the camera selection module.

1 conv	
Layer	Filter
conv 1	$1\times1\times1\times1$
Initialization	k=1, b=0
Activation	None

3 conv	
Layer	Filter
conv 1	$64\times1\times3\times3$
conv 2	$16\times64\times3\times3$
conv 3	$1\times16\times3\times3$
Initialization	k=1, b=0
Activation	tanh

Table 6: The layer settings for the Discriminator for domain adaptation.

Discriminator	
Layer	Filter
conv 1	$64\times256\times3\times3$, stride=(2, 2)
conv 2	$64\times64\times3\times3$, stride=(1, 1)
Flatten layer	
Fully-connected layer	