

Rethinking Cell Counting Methods: Decoupling Counting and Localization

Zixuan Zheng^{1*}, Yilei Shi^{1*}, Chunlei Li¹, Jingliang Hu¹, Xiao Xiang Zhu², and Lichao Mou¹✉

¹ MedAI Technology (Wuxi) Co. Ltd., Wuxi, China
lichao.mou@medimagingai.com

² Technical University of Munich, Munich, Germany

Abstract. Cell counting in microscopy images is vital in medicine and biology but extremely tedious and time-consuming to perform manually. While automated methods have advanced in recent years, state-of-the-art approaches tend to increasingly complex model designs. In this paper, we propose a conceptually simple yet effective decoupled learning scheme for automated cell counting, consisting of separate counter and localizer networks. In contrast to jointly learning counting and density map estimation, we show that decoupling these objectives surprisingly improves results. The counter operates on intermediate feature maps rather than pixel space to leverage global context and produce count estimates, while also generating coarse density maps. The localizer then reconstructs high-resolution density maps that precisely localize individual cells, conditional on the original images and coarse density maps from the counter. Besides, to boost counting accuracy, we further introduce a global message passing module to integrate cross-region patterns. Extensive experiments on four datasets demonstrate that our approach, despite its simplicity, challenges common practice and achieves state-of-the-art performance by significant margins. Our key insight is that decoupled learning alleviates the need to learn counting on high-resolution density maps directly, allowing the model to focus on global features critical for accurate estimates. Code is available at <https://github.com/MedAITech/DCL>.

Keywords: cell counting · decoupling · message passing

1 Introduction

Counting cells in microscopy is of paramount importance in medicine and biology [1]. Traditionally, this is done manually, but it is extremely tedious, labor-intensive, and time-consuming. Automated cell counting is therefore vital to many medical and research fields.

In this direction, research has made rapid advances during the past years, driven primarily by the development of counting algorithms in computer vision.

* Equal contribution.

Given an input image, they predominantly learn to predict a full-resolution density map that encodes the quantity and spatial distribution of objects (e.g., people and cells) [2,3], that is to say, these methods couple counting and localization via the density map. Albeit successful, current state-of-the-art methods tend to increasingly complex designs as diverse as more network modules and additional losses [4,5,6,7].

In this paper, we break this trend and propose a simple yet effective method for microscopy cell counting. Specifically, we decouple counting and localization, and devise an architecture consisting of two deep networks. The first is a counter that takes an image as input, estimates the number of cells, and generates a coarse, low-resolution density map as a by-product. The second is a localizer that learns to reconstruct a fine-grained localization map conditioned on the original image and the generated coarse density map which implies the estimated cell count. For the counting network, in contrast to prior work performing counting on high-resolution density maps, we propose to do so on intermediate feature representations. Our key insight is that this allows our model to more easily capture global features, without being overwhelmed by fine, low-level details that may not be as relevant. The intermediate representations appear more amenable to learning key attributes needed to produce accurate count estimates.

In addition, since convolution operations in CNNs have limited receptive fields, upon counting touching or overlapping cells, they lack sufficient context to robustly differentiate individual cells. Thus, we propose a global message passing module leveraging patterns and cues across the full image to improve counting accuracy.

We conduct extensive experiments to compare the proposed decoupled learning scheme with other counting methods that jointly learning counting and localization. The competitors include conventional approaches and recent, carefully designed, and more complex models (e.g., using sophisticated network units and losses). From our extensive study across four cell counting datasets, DCC, ADI, MBM, and VGG, we make the following intriguing observations:

- We find that decoupling counting and localization has surprising results, which challenges the current common practice in cell counting.
- It is advantageous in cell counting to make use of global context. For this reason, we propose a plug-and-play global message passing module.
- By applying our decoupled learning scheme, we achieve significantly higher accuracy than well established state-of-the-art methods on multiple cell counting benchmark datasets.

2 Methodology

2.1 Decoupling Counting and Localization

Counting and localization are two highly interrelated yet conflicting tasks in cell counting. The former is a more coarse-grained problem that requires richer

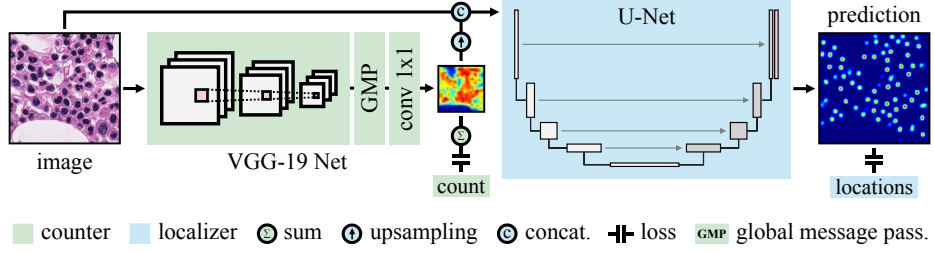


Fig. 1. The pipeline of the proposed method for cell counting. Unlike prior works, we decouple counting and localization, and design an architecture with separate counter and localizer to enable each to specialize on its target task. Moreover, we introduce a global message passing module into the counting network to model long-range spatial dependencies between image regions, enriching feature representations.

semantic context, while the latter is more fine-grained and demands detailed information. To address this conflict, we apply decoupled networks. Fig. 1 shows our method.

Counter We use a VGG-19 network [8], removing its last pooling and fully connected layers, as the backbone for our counting network. Given an image, the output of the backbone is upsampled by a factor of 2 using bilinear interpolation, and then passed to the proposed global message passing module for context modeling. Subsequently, a 1×1 convolutional layer is utilized to generate a single-channel feature map. We make the ℓ_1 norm of the flattened feature map $\mathbf{z}$ as close as possible to ground truth count y using the following loss:

$$\mathcal{L} = |||\mathbf{z}|_1 - y|. \quad (1)$$

With the designed network architecture and loss, we find that the counter learns a weak, inherent localization ability (see Fig. 2).

Localizer The counting network provides an estimate of the total number of cells, but lacks precise spatial information to localize individual cells. To enable localization, we employ a UNet-based network. By leveraging UNet’s capacity for dense prediction, we can reconstruct a fine-grained localization map. Our UNet-based localizer is trained conditioned on both the input image and the predicted coarse density map from the counting network. Learning with this auxiliary guidance allows the localizer to spatially distribute the estimated cell count. We optimize the localization network using mean square error loss. Note that ground truth localization maps are generated via convolving ground truth dot maps with a Gaussian kernel.

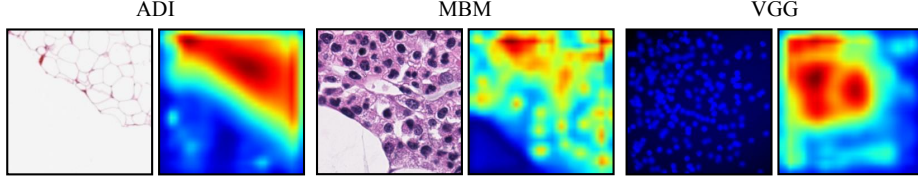


Fig. 2. Visualization of the learned single-channel feature map in the counting network. Note that the counter manages to localize cells despite having no cell location annotations at training, just cell count labels.

2.2 Global Message Passing

The proposed global message passing module consists of sampling and feature aggregation, and its goal is to enrich feature representations with long-range dependencies.

Sampling Each feature vector of the convolved image $\mathbf{x}$ is associated with a spatial index $\mathbf{p} = (x, y)$. For a query feature vector $\mathbf{x}_{\mathbf{p}}$, we consider learning a sampling operating across grids, in order to sample relevant context features to enable longer-range modeling. Specifically, a sampling position $\mathbf{s}$ can be calculated by applying the following equation:

$$\mathbf{s} = \mathbf{p} + \vec{\omega}(\mathbf{p}) = (x, y) + \vec{\omega}|_{(x,y)}, \quad (2)$$

where $\vec{\omega} = (u, v)$ is a moving vector. Since we would like to learn a task-driven sampling to gather useful context, we define a learnable $\vec{\omega}$ as follows:

$$\vec{\omega} = (u(\mathbf{x}_{\mathbf{p}}), v(\mathbf{x}_{\mathbf{p}})). \quad (3)$$

With $u(\cdot)$ and $v(\cdot)$, the sampled position can be predicted, conditioned on the feature of the current position. For the sake of simplicity and more efficient computation, we consider them in the form of linear embeddings, i.e.,

$$u(\mathbf{x}_{\mathbf{p}}) = \mathbf{w}_u \mathbf{x}_{\mathbf{p}}, \quad (4)$$

$$v(\mathbf{x}_{\mathbf{p}}) = \mathbf{w}_v \mathbf{x}_{\mathbf{p}}, \quad (5)$$

where $\mathbf{w}_u$ and $\mathbf{w}_v$ are learnable weights and can be implemented as 1×1 convolutions.

Feature Aggregation This step aims at aggregating sampled features and generate new feature representations that can facilitate the subsequent cell counting tasks.

We first revisit feature aggregation in self-attention models, which is considered the following equation:

$$\mathbf{a}_{\mathbf{p}} = \sum_{\forall \mathbf{q}} \frac{1}{C} w_{pq} \mathbf{x}_{\mathbf{q}}. \quad (6)$$

Here $\mathbf{p}$ is a query position whose response $\mathbf{a}_{\mathbf{p}}$ is to be calculated, and $\mathbf{q}$ indicates all possible positions. $w_{\mathbf{p}\mathbf{q}}$ represents the relationship between $\mathbf{p}$ and $\mathbf{q}$. Moreover, C is a normalization constant. Eq. (6) is a weighted sum of all feature-map vectors, but learning pairwise relations $\mathbf{w} = \{w_{\mathbf{p}\mathbf{q}}\}$ is computationally expensive.

In order to reduce the computational overhead, in this work, we perform feature aggregation at zero parameters as follows:

$$\mathbf{a}_{\mathbf{p}} = \sum_{\mathbf{s} \in \mathcal{V}(\mathbf{p})} \frac{1}{|\mathcal{V}(\mathbf{p})|} \mathbf{x}_{\mathbf{s}}. \quad (7)$$

where $\mathcal{V}(\mathbf{p})$ is a set of sampled positions, conditioned on the position $\mathbf{p}$. As compared to Eq. (6), our method has two changes: (1) $\forall \mathbf{q} \rightarrow \mathbf{s} \in \mathcal{V}(\mathbf{p})$; (2) $\frac{1}{C}w_{\mathbf{p}\mathbf{q}} \rightarrow \frac{1}{|\mathcal{V}(\mathbf{p})|}$. By doing so, we achieve feature aggregation in an efficient way.

3 Experiments

3.1 Experimental Setups

Datasets We evaluate the proposed method on four public benchmarks: the Dublin cell counting (DCC) dataset [9], the human subcutaneous adipose tissue (ADI) dataset [10], the modified bone marrow (MBM) dataset [11], and the synthetic fluorescence microscopy (VGG) dataset [12]. To achieve more compelling results given the limited size of these datasets, we partition each into training, test, and validation sets with an approximate ratio of 10:9:1.

The DCC dataset comprises images of various cell types, including embryonic mice stem cells, human lung adenocarcinoma, and human monocytes. Image sizes range from 306×322 to 798×788 to increase diversity.

Sampled from high-resolution histology slides using a 1700×1700 sliding window, the ADI dataset contains images with a resolution of 150×150 . Adipocytes within vary dramatically in size and represent a difficult test case given that they are densely packed adjoining cells with few gaps.

The MBM dataset consists of 11 1200×1200 images of bone marrow from 8 healthy individuals, cropped to 600×600 . The standard staining procedure depicts cell nuclei in blue and other constituents in shades of pink and red.

The VGG dataset is composed of 200 images of size 256×256 containing simulated bacterial cells from fluorescence microscopy. The images include overlapping cells at various focal distances, simulating real-life microscopy. Each image contains 174 ± 64 cells.

Table 1 provides an overview of the four datasets and their features.

Evaluation Metrics To quantify the cell counting performance of different methods, we employ the widely used mean absolute error (MAE) and mean square error (MSE) metrics [13,14], which measure the discrepancy between the estimated and ground truth cell counts. The MAE is calculated as follows:

$$\text{MAE} = \frac{1}{N} \sum_{i=1}^N |\hat{y}_i - y_i|, \quad (8)$$

Table 1. Overview of four public datasets and their features.

	Image Size	Cell Count	Type
DCC	306×322 to 798×788	34 ± 22	real
ADI	150×150	165 ± 44	real
MBM	600×600	126 ± 33	real
VGG	256×256	174 ± 64	synthetic

Table 2. Quantitative comparison with state-of-the-art methods on four public datasets. Performance is measured by MAE and MSE.

	DCC		ADI		MBM		VGG	
	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE
MCNN	5.4	6.4	25.8	35.7	3.2	4.3	20.9	25.3
FCRN	5.6	7.3	20.6	28.3	2.8	3.7	17.7	21.5
CSRNet	2.2	2.9	13.5	18.3	2.2	2.9	7.9	10.2
SFCN	2.7	3.7	16.0	22.3	2.4	3.1	13.8	17.9
DMCount	2.6	3.8	9.4	13.5	2.6	3.5	6.0	8.0
SASNet	8.9	12.0	9.0	12.2	3.9	5.2	4.9	6.8
DQN	3.5	4.6	9.7	13.1	3.1	4.2	5.5	7.3
OrdinalEntropy	3.2	4.3	9.1	12.0	2.9	3.8	5.7	7.8
DiffuseDenoiseCount	2.8	3.7	8.8	11.9	2.9	3.9	5.5	7.0
Ours	0.8	1.3	8.4	11.7	1.4	2.1	4.1	5.9
improvement	63.6%	55.2%	4.5%	1.7%	36.4%	27.6%	16.3%	13.2%

where $\hat{y}_i$ denotes the estimated cell count for the i -th test image, and N is the number of test images. MAE is the most commonly used metric in counting tasks. However, a limitation of MAE is its robustness to outliers (i.e., large counting errors). Thus, we additionally report MSE, which is more sensitive to outliers:

$$\text{MSE} = \frac{1}{N} \sum_{i=1}^N (\hat{y}_i - y_i)^2. \quad (9)$$

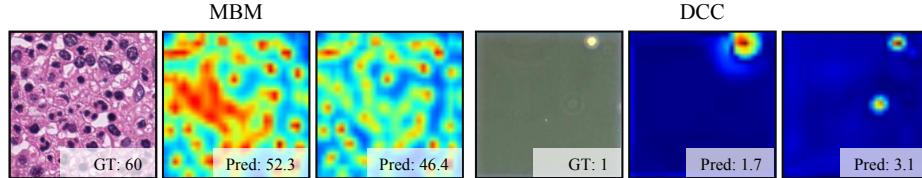
Implementation Details Due to diverse image sizes across datasets, we first preprocess all images as follows. For the DCC dataset, which contains images of different sizes, for each image, we pad the shorter image side to match the scale of the longer side. Images are then resized to the nearest multiple of 256 and divided into non-overlapping 256×256 patches organized in a $k \times k$ grid. ADI images are directly resized to 256×256 . For the MBM dataset, we divide each 1200×1200 image into four 300×300 patches before resizing to 256×256 .

For data augmentation during training, we utilize horizontal flipping, vertical flipping, and 90-degree clockwise/counter-clockwise rotations.

We optimize models using Adam with a learning rate of $1e-4$ and a batch size of 8. We also employ a cosine learning rate decay scheme with warm restarts [15]. Our model is implemented in PyTorch and runs on an NVIDIA RTX 4090 GPU.

Table 3. Ablation study results quantifying the impact of the proposed global message passing module on four datasets.

	DCC		ADI		MBM		VGG	
	MAE	MSE	MAE	MSE	MAE	MSE	MAE	MSE
w/o GMP	1.2	1.8	8.9	12.1	1.9	2.6	5.0	6.8
Full model	0.8	1.3	8.4	11.7	1.4	2.1	4.1	5.9
improvement	33.3%	27.8%	5.6%	3.3%	26.3%	19.2%	18.0%	13.2%

**Fig. 3.** Visualization of learned single-channel feature maps from counting networks with and without the proposed global message passing module, along with the estimated cell counts.

3.2 Comparison with State-of-the-Art Approaches

We compare against state-of-the-art methods including MCNN [16], FCRN [17], CSRNet [18], SFCN [19], DMCount [20], SASNet [21], DQN [22], OrdinalEntropy [23], and DiffuseDenoiseCount [24]. Quantitative results are given in Table 2.

On DCC, our model achieves significant improvements over prior arts, reducing MAE and MSE by 63.6% and 55.2%, respectively. For ADI, we lower state-of-the-art MAE and MSE by 4.5% and 1.7%. On MBM, our method reduces MAE by 36.4% and MSE by 27.6% over previous best. Finally, on VGG our approach decreases MAE and MSE by 16.3% and 13.2% over state-of-the-art. The consistent advancements across datasets validate the efficacy and generalization of our proposed method.

3.3 Ablation Study

We validate the efficacy of our proposed global message passing module via an ablation study, comparing models with and without this component. Removing the module consistently degrades performance across all datasets, as measured by MAE and MSE. For instance, we observe significant drops in MAE and MSE on DCC (33.3% and 27.8%) and MBM (26.3% and 19.2%) when removing the module. Although smaller, the degradation on ADI (5.6% MAE, 3.3% MSE) and VGG (18.0% MAE, 13.2% MSE) highlights the module’s contribution to precise cell counting. These indicate the module’s importance for modeling global context to derive enhanced features. Fig. 3 presents a qualitative ablation study on the effect of this module.

4 Conclusion

In this paper, we present a decoupled learning scheme for automated cell counting in microscopy images. Our approach deviates from the common practice of jointly learning counting and density map estimation, instead decoupling these objectives into separate counter and localizer networks. The counter operates on intermediate feature maps to leverage global context and produce coarse density maps along with count estimates, while the localizer reconstructs high-resolution density maps to localize individual cells, conditioned on the input images and coarse density maps. To further enhance counting accuracy, we introduced a global message passing module to integrate cross-region patterns.

Through extensive experiments on four datasets, we demonstrate that our decoupled learning approach challenges the common practice and achieves state-of-the-art performance by significant margins, despite its simplicity.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this paper.

References

1. Guo, Y., Stein, J., Wu, G., Krishnamurthy, A.: SAU-Net: A universal deep network for cell counting. In: ACM Conference on Bioinformatics, Computational Biology and Biomedicine, pp. 299–306. (2019)
2. Liu, W., Salzmann, M., Fua, P.: Context-aware crowd counting. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5099–5108. (2019)
3. Cheng, Z.Q., Li, J.X., Dai, Q., Wu, X., Hauptmann, A.G.: Learning spatial awareness to improve crowd counting. In: IEEE/CVF International Conference on Computer Vision, pp. 6152–6161. (2019)
4. Wang, Z., Yin, Z.: Cell counting by a location-aware network. In: Machine Learning in Medical Imaging, pp. 120–129. (2021)
5. Idughayiq, B., Ashfaq, F., Jhanjhi, N.Z., Humayun, M.: YOLOv5-FPN: A robust framework for multi-sized cell counting in fluorescence images. *Diagnostics* **13**(13), 2280 (2023)
6. Ma, Y., Sanchez, V., Guha, T.: FusionCount: Efficient crowd counting via multiscale feature fusion. In: IEEE International Conference on Image Processing, pp. 3256–3260. (2022)
7. Lin, W., Chan, A.B.: Optimal transport minimization: Crowd localization on density maps for semi-supervised counting. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 21663–21673. (2023)
8. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
9. Marsden, M., McGuinness, K., Little, S., Keogh, C.E., O’Connor, N.E.: People, penguins and petri dishes: Adapting object counting models to new visual domains and object types without forgetting. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 8070–8079. (2018)
10. Cohen, J.P., Boucher, G., Glastonbury, C.A., Lo, H.Z., Bengio, Y.: Count-ception: Counting by fully convolutional redundant counting. In: IEEE International Conference on Computer Vision Workshops, pp. 18–26. (2017)

11. Kainz, P., Urschler, M., Schuster, S., Wohlhart, P., Lepetit, V.: You should use regression to detect cells. In: International Conference on Medical Image Computing and Computer Assisted Intervention, pp. 276–283. (2015)
12. Lempitsky, V., Zisserman, A.: Learning to count objects in images. In: Advances in Neural Information Processing Systems, pp. 1324–1332. (2010)
13. Sam, D.B., Surya, S., Babu, R.V.: Switching convolutional neural network for crowd counting. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 5744–5752. (2017)
14. Liu, J., Gao, C., Meng, D., Hauptmann, A.G.: DecideNet: Counting varying density crowds through attention guided detection and density estimation. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 5197–5206. (2018)
15. Loshchilov, I., Hutter, F.: SGDR: Stochastic gradient descent with warm restarts. arXiv preprint arXiv:1608.03983 (2016)
16. Zhang, Y., Zhou, D., Chen, S., Gao, S., Ma, Y.: Single-image crowd counting via multi-column convolutional neural network. In: IEEE Conference on Computer Vision and Pattern Recognition, pp. 589–597. (2016)
17. Xie, W., Noble, J.A., Zisserman, A.: Microscopy cell counting and detection with fully convolutional regression networks.: Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization **6**(3), 283–292 (2018)
18. Li, Y., Zhang, X., Chen, D.: CSRNet: Dilated convolutional neural networks for understanding the highly congested scenes. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 1091–1100. (2018)
19. Wang, Q., Gao, J., Lin, W., Yuan, Y.: Learning from synthetic data for crowd counting in the wild. In: IEEE/CVF Conference on Computer Vision and Pattern Recognition, pp. 8198–8207. (2019)
20. Wang, B., Liu, H., Samaras, D., Nguyen, M.H.: Distribution matching for crowd counting. In: Advances in Neural Information Processing Systems, pp. 1595–1607. (2020)
21. Song, Q., Wang, C., Wang, Y., Tai, Y., Wang, C., Li, J., Wu, J., Ma, J.: To choose or to fuse? Scale selection for crowd counting. In: AAAI Conference on Artificial Intelligence, pp. 2576–2583. (2021)
22. Lu, H., Liu, L., Wang, H., Cao, Z.: Counting crowd by weighing counts: A sequential decision-making perspective. IEEE Transactions on Neural Networks and Learning Systems, **35**(4), 5141–5154 (2022)
23. Zhang, S., Yang, L., Mi, M.B., Zheng, X., Yao, A.: Improving deep regression with ordinal entropy. arXiv preprint arXiv:2301.08915 (2023)
24. Ranasinghe, Y., Nair, N.G., Bandara, W.G.C., Patel, V.M.: Diffuse-Denoise-Count: Accurate crowd-counting with diffusion models. arXiv preprint arXiv:2303.12790 (2023)