Where are the People? A Multi-Stream Convolutional Neural Network for Crowd Counting via Density Map from Complex Images

Darwin Ttito*, Rodolfo Quispe*, Adín Ramírez Rivera, Helio Pedrini

Institute of Computing, University of Campinas (UNICAMP)
Campinas, SP, Brazil, 13083-852
{darwin.ttito, quispe}@liv.ic.unicamp.br, {adin, helio}@ic.unicamp.br

***Abstract*—Crowd counting is a challenging task that aims to compute the number of people present in a single image. The problem has a significant impact on various applications, for instance, urban planning, forensic science, surveillance and security, among others. In this work, we propose and evaluate a Multi-Stream Convolutional Neural Network that receives an image as input, generates a density map as output that represents the spatial distribution of people in an end-to-end fashion, then we estimate the number of people in the image from the density map. The network architecture employs receptive fields with different size filters for each stream in order to deal with extremely unconstrained scale and perspective changes, which are complex issues in the crowd counting context. Although simple, the proposed architecture achieves effective results on the two challenging `UCF_CC_50` and `ShanghaiTech` datasets.**

I. Introduction

A rush-hour stampede at a railway station in Mumbai, India, left more than 20 people dead and dozens more injured in September 2017. This and other similar tragedies could be prevented or their consequences reduced through monitoring systems capable of measuring the quantity of people in public places.

The problem of crowd counting aims to estimate the amount of people present in high density scenes. Crowd counting has applications in several domains, such as forensic search, city planning, virtual environments, safety monitoring, and disaster management [1], [2], [3], [4], [5]. The techniques can also be applied to other tasks, such as counting cells or bacteria in microscopic images [6], [7].

Figure 1 illustrates examples of scenarios for crowd counting. Perspective changes are a critical issue in the crowd counting context since they create different shapes of people. In some cases, people can be seen as points, while in others, deformability of human body gains importance. Occlusion is another issue and classical pedestrian detection methods are not feasible. Furthermore, illumination variation can create small differences between background and crowd. These factors make crowd counting a challenging task.

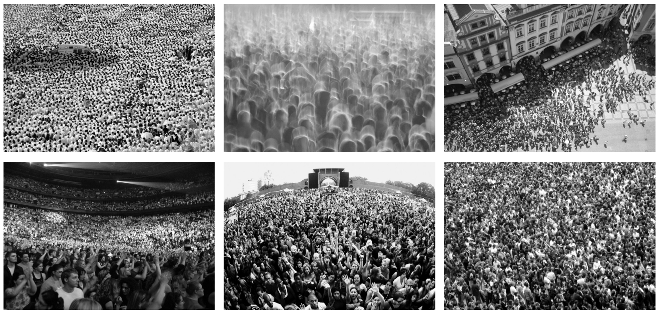


Figure 1: Image samples for crowd counting from the `UCF_CC_50` dataset. The unconstrained scenarios (different settings of the cameras, scaling, view points, etc.) make the task challenging.

A common approach to addressing the crowd counting problem is to generate a density map that models the probability distribution of people heads present in the images. Then, the head count is estimated by calculating the accumulated distribution of the density map. Our method has similarities with the approach developed by Zhang et al. [8], where a Convolutional Neural Network (CNN) with multiple streams was used to combine filters with receptive fields of different sizes to be invariant to dramatic scale changes that usually exist in crowd counting. However, our network consists of more streams to handle several scale changes, our definition of density map is simpler yet more effective, we do not need a complex training process for each stream individually, and a hard negative mining process is introduced for an effective data augmentation scheme. In addition to these contributions, we conducted extensive experiments on two challenging image sets, `UCF_CC_50` and `ShanghaiTech` dataset, achieving competitive results.

The remainder of the paper is organized as follows. Section II briefly describes related work on estimating crowd density. Section III introduces the details of our work. We describe the experiments and results in Section IV. Finally, we conclude with some final remarks and future work in

[*] These authors contributed equally to this work.

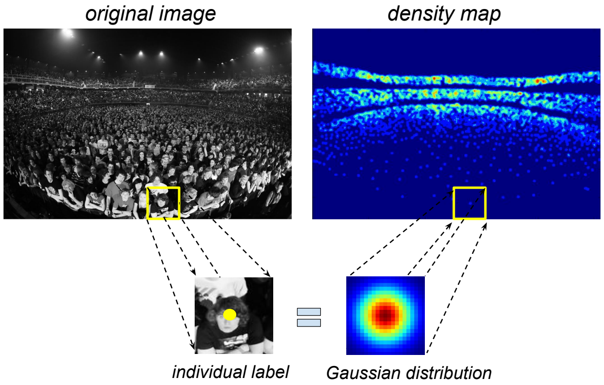


Figure 2: Fixed Gaussian kernel for density map construction. Each head position is converted into a Gaussian distribution (overlapped distributions are summed).

Section V.

II. Related Work

Although several methods for crowd counting [3], [9], [10] have been developed over the last years, it remains an open problem. Early approaches focused only on determining the total amount of people in the scene. Recently, the use of density maps has gain popularity [3]. A density map is an intuitive representation of the spatial distribution of people in an image, and it is more usable for real life applications, such as security, since it gives a notion of the people spatial distribution (as shown on the left of Figure 2).

Previous approaches used to count directly the heads on the image, by detecting them. For instance, Idrees et al. [10] proposed to obtain a head count by mixing several features. They used a combination of handcrafted fractured Fourier analysis, Histogram-of-Oriented-Gradient based head detection, and interest-points based counting, and post-processed the resulting features with multi-scale Markov Random Field. However, handcrafted features often suffer a drop in accuracy when subject to perspective distortion, severe occlusion and variation in illumination.

To overcome the limitations of handcrafted methods, recent research focuses on creating neural networks that estimate the body parts. Boominathan et al. [11] proposed a deep learning framework, where they combined deep and shallow fully convolutional networks to predict a density map. Such combination was used to capture both high-level semantic (face and body detectors) information and low-level fractures (blob detectors). Furthermore, to deal with the small amount of data, they used a multi-scale data augmentation technique. However, they did not supervise the type of images generated with their technique (for instance, images without or very few people).

Onoro et al. [12] proposed two neural networks, named Counting CNN (CCNN), and Hydra CNN (HCNN). The CCNN was the first network formulated as a regression model. It learns a mapping between the appearance of the image patches to their corresponding object density maps. However, the scale is still a major issue. Thus, the HCNN learns a multi-scale non-linear regression model that uses a pyramid of image patches extracted at multiple scales to perform the final density prediction. However, similar to the work developed by Boominathan et al. [11], they did not supervise the type of images generated through the data augmentation technique.

Walach et al. [13] proposed a general method for object counting based on density maps using CNNs with layered boosting and selective sampling. An important aspect of their method is the improvement in training time based on thresholding the samples used for training. Our work uses a similar idea, however, we focus on the number of people for the threshold.

Sam et al. [5] used switching CNN architectures that employ patches from a grid within a crowd scene to create independent CNN regressors based on a switch classifier. Each independent CNN regressor is chosen with different receptive fields and field-of-view as a multi-stream network in order to be invariant to scale changes. Similar to the work by Zhang et al. [8], the ground truth was generated with adaptive Gaussian kernels. In our work, we use a simpler, yet more effective, kernel for creating the density maps.

Sindagi et al. [14] proposed an end-to-end network that aims to learn two related tasks, crowd count classification and density map estimation in a cascaded fashion. In the training stage, they augmented data by creating patches without supervising the number of people in the images. In our work, we instead consider a minimum number of people per patch.

III. Proposed Crowd Counting Method

We propose a neural network to estimate a density map of the head locations in a crowd for a given image. Our model learns several kernels to identify head positions at different scales. Then, we fuse the responses of scale-aware detectors to estimate the density of each detection. We train our model by minimizing the deviation of a true density of the given image.

Since we train the network in an end-to-end fashion, we need to create the ground truth estimates. First, we present the process to estimate this true density from the impulse responses of annotated images. Then, we present our Multi-Stream Convolutional Neural Network (MC-CNN) that estimates its density map given an input image. Finally, we detail our training and data augmentation process.

A. Density Map Construction

Crowd counting datasets provide images and positions (usually located in the heads) of each person. Based on these labels, we create a density map since it has been demonstrated [5], [8], [12], [13], [14] that such representation is simple, yet effective to predict the number of people present in the scenes. The purpose of the density maps is to describe the density distribution of people in a given image (Figure 2).

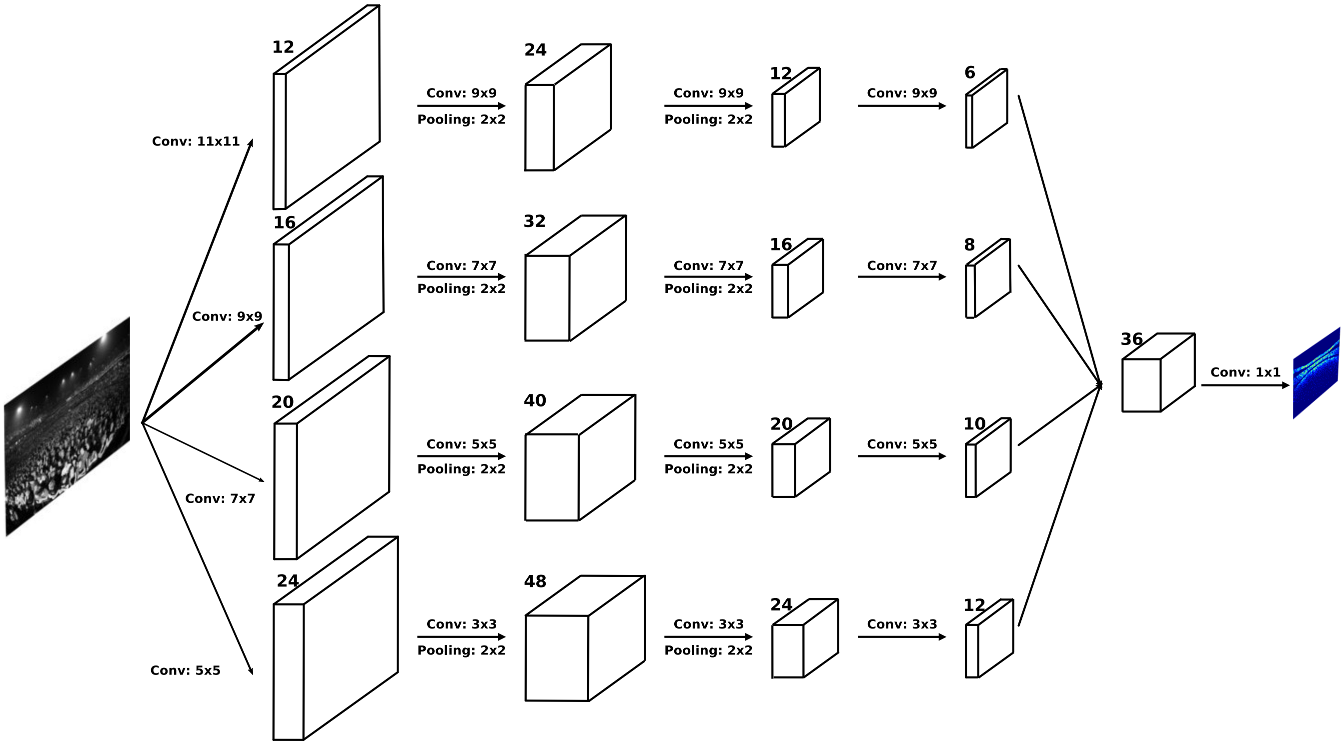


Figure 3: Proposed multi-stream neural network for crowd counting, where each stream (set of layers at a particular scale) aims to learn head detectors at a certain scale. The detection results are fused and converted into a density map that represents the probability of a head for that pixel.

Following the work by Zhang et al. [8], if a person is located at pixel x_i, then an image labeled with y heads is represented through the accumulation of y impulse functions, such as

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

where the $\delta(\cdot)$ function is defined as

$$\delta(x - a) = \begin{cases} 1 & \text{if } x = a, \\ 0 & \text{otherwise.} \end{cases} \tag{2}$$

To convert such image to a continuous domain, we convolve it with a Gaussian kernel (with standard deviation σ)

$$F(x) = H(x) * G_\sigma(x). \tag{3}$$

Originally, Zhang et al. [8] proposed to use a geometry-dependent parameter σ_i for each person's Gaussian kernel. This parameter σ_i was defined by the mean distance $\overline{d}^i$ to the k closest persons and a regularization parameter β. Then, the density map was defined in the same fashion (3) with $\sigma = \sigma_i = \beta \overline{d}^i$.

We changed this idea and used a fixed kernel with a single $\sigma = 4$. This simplification proves to obtain better results, as shown in Section IV. Since we are creating fixed responses at different scales, our model needs to only predict heads, and then just estimate their distributions, instead of predicting their scale as well.

Finally, the predicted number of people in a density map is computed as

$$y' = \sum_{x} F(x). \tag{4}$$

We use these density maps as ground truth for training our MC-CNN.

B. Multi-Stream Convolutional Neural Network

As mentioned previously, crowd counting is subject to unconstrained perspective changes. Thus, images typically contain people at different scales. Classic deep learning approaches using filters with receptive fields of uniform size suffer from capturing features at different scales. Consequently, a multi-stream convolutional neural network is an intuitive solution since it can learn features to predict heads and, consequently, their distributions at multiple scales.

Inspired by the work of Zhang et al. [8], we propose a convolutional neural network that learns head detectors at four different scales and that converts the fused detection results into a density map, as illustrated in Figure 3.

The architecture contains four parallel CNNs (streams) with different sizes of filters which are joined to predict the final density map of the people. Each stream has the same network structure (i.e., convolution-pooling-convolution-pooling), however, with different size and number of filters. Max pooling is applied to each 2×2 region, whereas rectified linear unit (ReLU) is used as the activation function.

The size of filters is smaller for the streams with more channels. This technique is adopted to reduce the number of parameters to be optimized. The final output (density map) is obtained through 1×1 filters to combine the responses of each stream for a determined pixel, and to further convert it into its probability. It is worth observing that the architecture can handle images of any dimension, which is useful to validate the method.

To train the network, we find the optimal parameters θ^* (for the network) that minimize the error between the estimated and ground truth density through

$$\theta^* = \arg\min_{\theta} L(\theta), \tag{5}$$

where the loss function is

$$L(\theta) = \frac{1}{2N} \sum_{i=1}^{N} \left\| \mathcal{F}(X_i; \theta) - F_i \right\|_2^2, \tag{6}$$

where $\mathcal{F}$ is the function approximated through our network, θ is a set of learnable parameters in the multi-stream neural network, X_i is the i-th input image and F_i its ground truth density map (3), N is the number of training images, and $\|\cdot\|_2$ is the Euclidean distance (although any other ℓ_2 norm can be used).

C. Training and Data Augmentation

The loss function (6) is optimized via back-propagation and batch-based stochastic gradient descent. Differently from the work described by Zhang et al. [8], we do not train each stream independently. Due to the two pooling layers, the size of the output is a quarter of the original size, then we resize the ground truth images in order to compare them with the output.

In order to train both datasets, we use the following parameter configuration: 2000 epochs, learning rate of 0.00001, momentum of 0.9, and batch size of 500 images.

Since the availability of data for training is small, we consider an effective data augmentation step. Although it is similar to other existing approaches, we add a hard negative mining that improves the final results.

We perform an extensive data augmentation of the training dataset by creating images with a sliding window of 200×300 pixels and a displacement of 10 pixels in each iteration, such that the number of people in the image is larger than a threshold t. After applying this data augmentation technique, the size of training set is increased from 3 to 37 times, depending on the evaluated dataset.

In our experiments, we consider $t = 200$ and $t = 0$ for `UCF_CC_50` dataset and `ShanghaiTech` dataset, respectively. Such decision is due to the characteristics present in the datasets: the average number of people in the `UCF_CC_50` dataset is greater than in `ShanghaiTech` dataset. Therefore, the network learns to count larger crowds with higher values of t and, analogously, smaller crowds with lower values of t.

IV. Results

In order to evaluate the performance of the proposed crowd counting method, we compute the Mean Absolute Error (MAE) and Root Mean Squared Error (RMSE) metrics, defined respectively as

$$\text{MAE} = \frac{1}{N} \sum_{i=1}^{N} |y_i - y_i'|, \tag{7}$$

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

where N is number of test samples, y_i is the ground truth count, and y_i' is the estimated count corresponding to the i-th sample. Note that previous works [8] denote RMSE (8) as Mean Squared Error (MSE), instead.

We conduct experiments on the `UCF_CC_50` and `ShanghaiTech` datasets to demonstrate the effectiveness of our method. Results are compared to state-of-the-art approaches and baselines of the proposed datasets. Since the source code[1] of the proposed approach is available, the experimental procedure can be reproduced.

A. `UCF_CC_50` Experiment

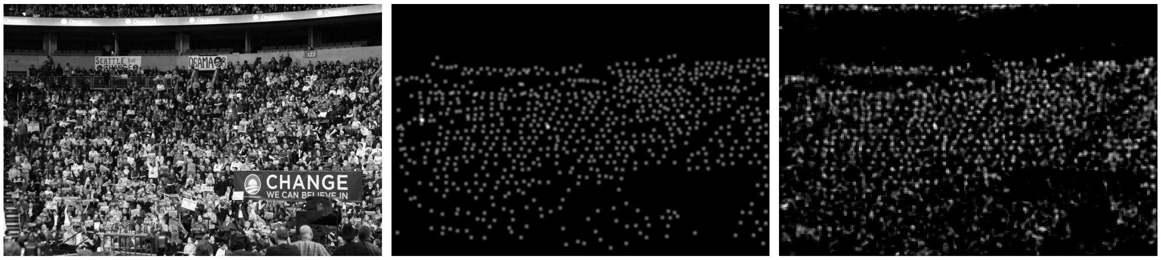
The `UCF_CC_50` dataset is an extremely challenging dataset introduced by Idrees et al. [10]. It contains 50 images of different resolutions and aspect ratios extracted from the Internet, where the number of people varies from 94 to 4543. Following the original standard protocol, we show the results using a 5-fold cross-validation.

As shown in the comparative results of Table I, our method achieves state-of-the-art results in terms of MAE, but not in RMSE. By analyzing each fold prediction, we found that our method has difficulty in dealing with a specific case where the ground truth has over 4000 people, as it predicts a count below 3000. This result is due to the size of the kernel that incorporates several heads within a single region. Thus, the method does not count overlapping people as different persons, i.e., two close persons are considered one due to kernel size.

Despite the incorrect estimation in such situations, our method is capable of reaching an assertive prediction of people. An example is illustrated in Figure 4, where there is a mislabeled ground truth image, but the proposed network finds the heads that are not labeled. This demonstrates the capability of the proposed method and the complexity of the problem, since the precise interpretation of crowed scenes is challenging even for human beings.

Qualitative results of the proposed method on this dataset are shown in the first three rows of Figure 5. The overlay on the figures shows the accuracy of the proposed method. A failure case is shown in the third row of the figure, where the method has trouble in detecting the blurred heads.

[1] https://github.com/RQuispeC/multi-stream-crowd-counting



(a) Crowd image (b) Ground truth (c) Estimated density map

Figure 4: Comparison of density map results. The proposed network is capable of handling ground truth labeling error (top part of the images) in UCF_CC_50 dataset.

Table I: Comparison of several method results for UCF_CC_50 dataset.

Method	MAE	RMSE
Idrees et al. [10]	419.5	541.6
Zhang et al. [8]	355.6	487.1
Boominathan et al. [11]	—	452.0
Onoro et al. [12]	333.7	425.2
Walach et al. [13]	364.4	341.4
Switching-CNN [5]	318.1	439.2
Cascaded-MTL [14]	322.8	341.4
CP-CNN [15]	295.8	**320.9**
Proposed	**295.0**	443.7

Table II: Comparison of several method results for ShanghaiTech dataset.

	Part A		Part B	
Method	MAE	RMSE	MAE	RMSE
Zhang et al. [8]	110.2	173.2	26.4	41.3
Cascaded-MTL [14]	101.3	152.4	20.0	31.1
Switching-CNN [5]	90.4	135.0	21.6	33.4
CP-CNN [15]	**73.6**	**106.4**	20.1	30.1
Proposed	96.1	150.5	**17.8**	**27.7**

B. *ShanghaiTech Experiment*

ShanghaiTech dataset was introduced by Zhan et al. [8] and it is among the largest ones in terms of labeled people, with 330 164 annotated heads. The dataset was created to boost the research in crowd counting using deep learning approaches. It is divided into two parts. Part A is composed of 482 images randomly taken from the Internet, which have different sizes and contain between 501 and 3139 people. Part B is composed of 716 images taken from a busy street of the metropolitan area of Shanghai, containing between 123 and 578 people. The crowd density varies significantly between the two subsets, making an accurate crowd estimation more challenging than other datasets. Both parts have predefined training and testing sets: Part A has 300 images for training and the remaining for testing, whereas Part B has 400 images for training and 316 for testing.

For Part A, our results are superior than those obtained by Zhang et al. [8] and Cascaded-MTL [14]. Our network architecture, data augmentation scheme, and density map definition were crucial to improve the results. On the other hand, Switching-CNN [5] and CP-CNN [15] use much more complex architectures and can handle the challenging ShanghaiTech dataset. For Part B, our method achieves state-of-the-art. Although the number of people present in Part B is smaller than in Part A, the network estimated the distribution of people accurately, as shown in Figure 5.

V. Conclusions and Future Work

In this work, we extended the architecture proposed by Zhang et al. [8] for crowd counting, as well as incorporated an effective density map construction and a data augmentation scheme. Our method was able to achieve state-of-the-art results on the UCF_CC_50 dataset for MAE metric, competitive results on ShanghaiTech-part A and state-of-the-art in ShanghaiTech-part B datasets.

As directions for future work, we intend to evaluate methods that automatically set the best size for the receptive fields of layer filters. Moreover, we plan to evaluate the creation of synthetic data for training purpose.

VI. Acknowledgments

The authors are grateful to São Paulo Research Foundation (FAPESP grants #2014/12236-1 and #2016/19947-6) and Brazilian National Council for Scientific and Technological Development (CNPq grants #305169/2015-7, #307425/2017-7 and #309330/2018-1) for their financial support.

References

[1] Y. Hu, H. Chang, F. Nian, Y. Wang, and T. Li, "Dense Crowd Counting from Still Images with Convolutional Neural Networks," *Journal of Visual Communication and Image Representation*, vol. 38, pp. 530–539, 2016.

[2] Y. Wang, R. Chen, and D.-C. Wang, "A Survey of Mobile Cloud Computing Applications: Perspectives and Challenges," *Wireless Personal Communications*, vol. 80, no. 4, pp. 1607–1623, 2015.

[3] V. A. Sindagi and V. M. Patel, "A Survey of Recent Advances in CNN-based Single Image Crowd Counting and Density Estimation," *Pattern Recognition Letters*, vol. 107, pp. 3–16, 2018.

[4] M. Zhao, J. Zhang, F. Porikli, C. Zhang, and W. Zhang, "Learning a Perspective-embedded Deconvolution Network for Crowd Counting," in *IEEE International Conference on Multimedia and Expo*. IEEE, 2017, pp. 403–408.

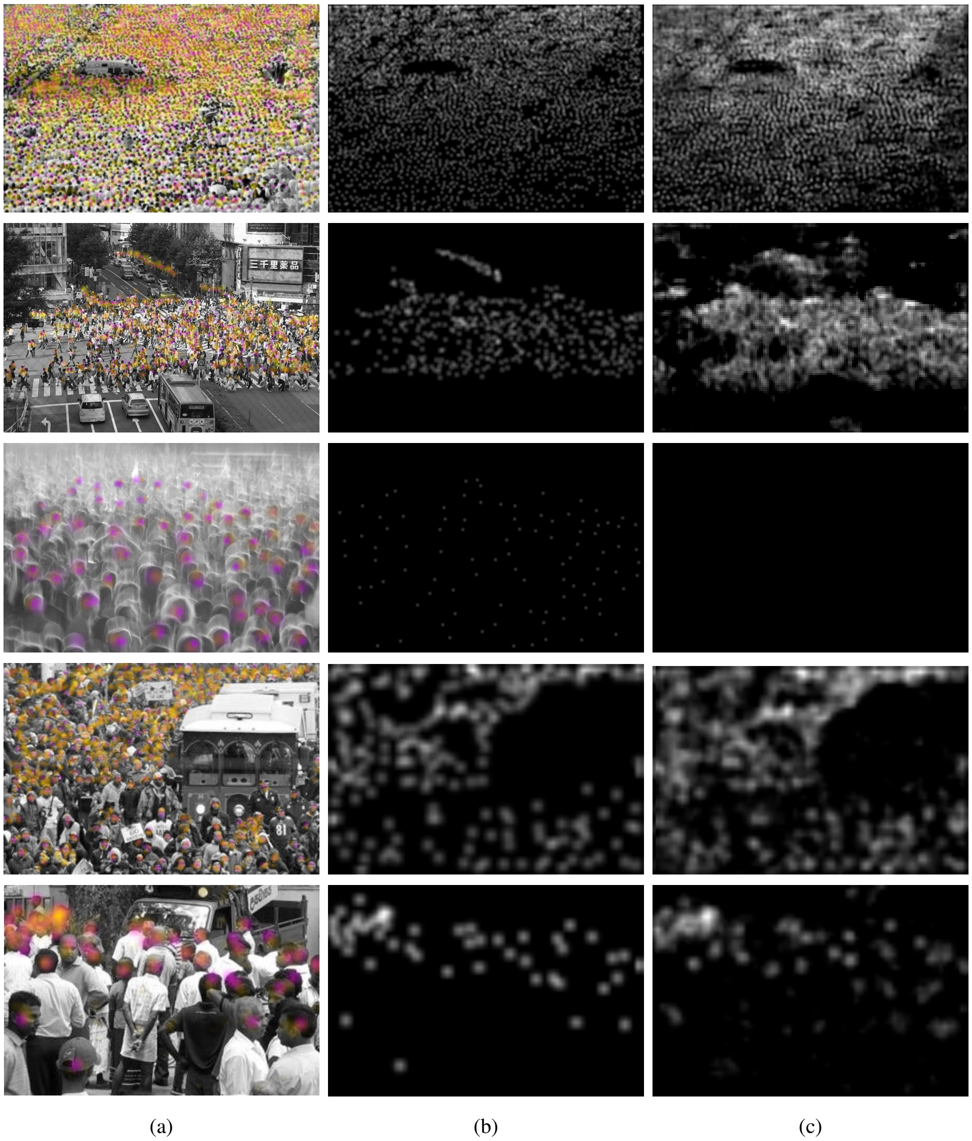


Figure 5: Qualitative results of the estimated density maps on the `UCF_CC_50` (first three rows) and `ShanghaiTech` (last two rows) datasets. (a) The original image overlaid with the ground truth (on purple) and the estimated (on yellow) densities. We also show the (b) ground truth and (c) estimated densities alone for clarity purposes.

[5] D. B. Sam, S. Surya, and R. V. Babu, “Switching Convolutional Neural Network for Crowd Counting,” in *IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, no. 3, 2017, pp. 1–6.

[6] W. Xie, J. A. Noble, and A. Zisserman, “Microscopy Cell Counting and Detection with Fully Convolutional Regression Networks,” *Computer Methods in Biomechanics and Biomedical Engineering: Imaging & Visualization*, pp. 1–10, 2016.

[7] M. Marsden, K. McGuiness, S. Little, and N. E. O’Connor, “Fully Convolutional Crowd Counting On Highly Congested Scenes,” in *12th International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications*, 2017.

[8] 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.

[9] M. Xu, Z. Ge, X. Jiang, G. Cui, B. Zhou, C. Xu *et al.*, “Depth Information Guided Crowd Counting for Complex Crowd Scenes,” *Pattern Recognition Letters*, 2019.

[10] 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.

[11] L. Boominathan, S. S. Kruthiventi, and R. V. Babu, “CrowdNet: A Deep Convolutional Network for Dense Crowd Counting,” in *ACM on Multimedia Conference*. ACM, 2016, pp. 640–644.

[12] D. Onoro-Rubio and R. J. López-Sastre, “Towards Perspective-Free Object Counting with Deep Learning,” in *European Conference on Computer Vision*. Springer, 2016, pp. 615–629.

[13] E. Walach and L. Wolf, “Learning to Count with CNN Boosting,” in *European Conference on Computer Vision*. Springer, 2016, pp. 660–676.

[14] V. A. Sindagi and V. M. Patel, “CNN-based Cascaded Multi-Task Learning of High-Level Prior and Density Estimation for Crowd Counting,” in *14th IEEE International Conference on Advanced Video and Signal Based Surveillance*. IEEE, 2017, pp. 1–6.

[15] ——, “Generating High-Quality Crowd Density Maps using Contextual Pyramid CNNs,” in *IEEE International Conference on Computer Vision*, Oct. 2017, pp. 1879–1888.