Multi-Stream Networks and Ground Truth Generation for Crowd Counting

Original Scientific Paper

Rodolfo Quispe
University of Campinas, Institute of Computing
Av. Albert Einstein 1251, Campinas, SP - Brazil, 13083-852
e-mail: quispe@liv.ic.unicamp.br

Darwin Ttito
University of Campinas, Institute of Computing
Av. Albert Einstein 1251, Campinas, SP - Brazil, 13083-852
e-mail: ttito@liv.ic.unicamp.br

Adín Rivera
University of Campinas, Institute of Computing
Av. Albert Einstein 1251, Campinas, SP - Brazil, 13083-852
e-mail: adin@ic.unicamp.br

Helio Pedrini
University of Campinas, Institute of Computing
Av. Albert Einstein 1251, Campinas, SP - Brazil, 13083-852
e-mail: helio@ic.unicamp.br

Abstract *– Crowd scene analysis has received a lot of attention recently due to a wide variety of applications, e.g., forensic science, urban planning, surveillance and security. In this context, a challenging task is known as crowd counting [1–6], whose main purpose is to estimate the number of people present in a single image. A multi-stream convolutional neural network is developed and evaluated in this paper, which receives an image as input and produces a density map that represents the spatial distribution of people in an end-to-end fashion. In order to address complex crowd counting issues, such as extremely unconstrained scale and perspective changes, the network architecture utilizes receptive fields with different size filters for each stream. In addition, we investigate the influence of the two most common fashions on the generation of ground truths and propose a hybrid method based on tiny face detection and scale interpolation. Experiments conducted on two challenging datasets, UCF-CC-50 and ShanghaiTech, demonstrate that the use of our ground truth generation methods achieves superior results.*

Keywords *– crowd counting, deep learning, density maps, multi-stream network*

1. INTRODUCTION

The task of crowd counting is to estimate the number of people from a single RGB (red-green-blue) image. The problem has a significant impact on several applications, for instance, urban planning, forensic science, surveillance and security [2, 7–10]. The main challenge in this task is aggressive variation in the scale and perspective of people in the images. Therefore, it can be complicated to differentiate between the background and the people (Figure 1).

Initial approaches used more classical people detection algorithms to directly count people in the image. For instance, Idrees et al. [16] proposed to obtain a headcount by mixing several features. They used a combination of head detection based on a histogram of oriented gradient, handcrafted Fourier analysis and interest-point based counting, then processed the resulting features with a multi-scale Markov random field. Similarly to other tasks in computer vision, handcrafted features often suffer from a decrease in accuracy when subjected to heavy variation in illumination, scale, severe occlusion, perspective and distortion.

To overcome the limitations of handcrafted methods, the seminal work of Zhang et al. [17] proposed a multiple stream neural network (MSNN) to estimate density maps. A density map represents the spatial distribution of people in an image, and it is more suitable for real-life applications since it gives a notion of the people spatial distribution. The popularity of density maps has grown in deep learning methods [3, 11–16], such that they have become the default option for the prediction of deep networks [3].

The main idea behind the MSNN is to specialize each stream at a specific person's scale. Thus, each stream follows similar architecture, yet with different filter sizes. Therefore, the active field of each stream differs according to the scale it focuses on. Following the work developed by Zhang et al. [17], many other MSNN variations have been proposed [2, 7–10]. Moreover, various types of ground truth generation from density maps have been proposed, leading to a lack of consensus on which method is best. These generation methods can be categorized into fixed and variable kernel methods, as explained in Section 2.1.

Onoro et al. [8] proposed a variation of the MSNN, named the hydra CNN (HCNN). This network makes each stream more powerful by stacking more convolutional layers than in the MSNN. The HCNN is based on the three-stream counting CNN (CCNN). The HCNN learns a mapping between image patches to their corresponding density maps, which differs from the MSNN since this is fully convolutional, such that it can handle random size images. The authors of the HCNN designed the network to be scale-aware. Thus, the HCNN is fed with a pyramid of patches extracted at multiple scales, where each level of the pyramid is processed by a stream. Then, fully connected layers are used to join information of all streams. Finally, the prediction is a density map for the patch on top of the pyramid. To define the ground truth, they followed a fixed kernel fashion.

Another way to tackle the scale problem is to improve the network to use various active fields. Thus, Boominathan et al. [7] proposed a deep learning framework with two streams, one with deep architecture and the other with shallow architecture. The idea behind is that a deep stream and a shallow stream were used to capture both high-level semantic (face and body detectors) information and low-level fractures (blob detectors), respectively. Finally, they join the streams with a 1x1 convolution and upsample the images using bilinear interpolation, such that the output of the network has the same size as the input. Furthermore, they proposed a multi-scale data augmentation technique to increase the training size. Their framework follows the ground truth with a fixed kernel fashion. It is worth mentioning that the number of streams decreased compared to other works [8, 17].

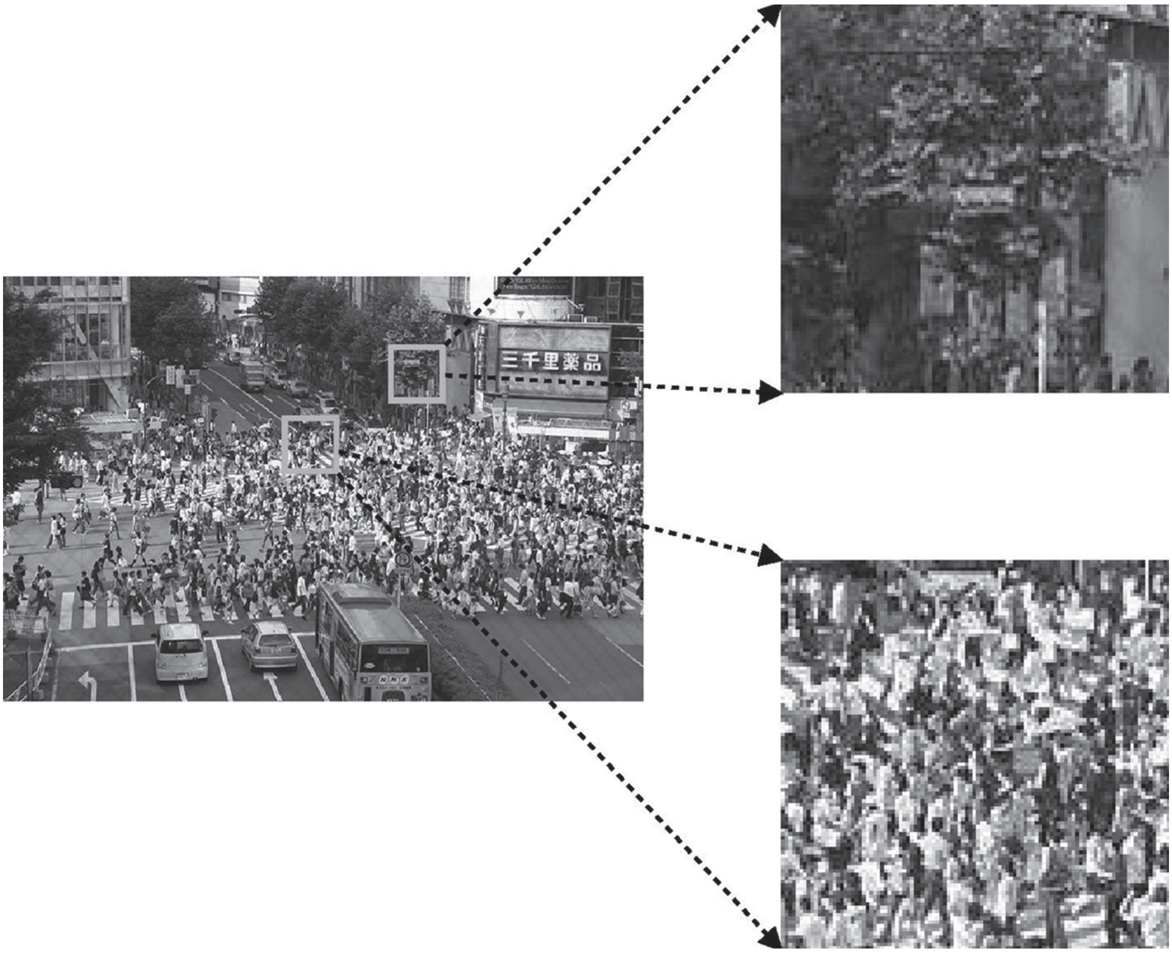

Fig. 1. Illustration of the scaling problem in crowd counting. The upper patch is a zoom of the background, whereas the lower patch is a zoom of the people. A challenging task is to differentiate the background from the people due to overlap and scale of the people.

Onoro et al. [8] proposed a variation of the MSNN, named the hydra CNN (HCNN). This network makes each stream more powerful by stacking more convolutional layers than in the MSNN. The HCNN is based on the three-stream counting CNN (CCNN). The HCNN learns a mapping between image patches to their corresponding density maps, which differs from the MSNN since this is fully convolutional, such that it can handle random size images. The authors of the HCNN designed the network to be scale-aware. Thus, the HCNN is fed with a pyramid of patches extracted at multiple scales, where each level of the pyramid is processed by a stream. Then, fully connected layers are used to join information of all streams. Finally, the prediction is a density map for the patch on top of the pyramid. To define the ground truth, they followed a fixed kernel fashion.

Another way to tackle the scale problem is to improve the network to use various active fields. Thus, Boominathan et al. [7] proposed a deep learning framework with two streams, one with deep architecture and the other with shallow architecture. The idea behind is that a deep stream and a shallow stream were used to capture both high-level semantic (face and body detectors) information and low-level fractures (blob detectors), respectively. Finally, they join the streams with a 1x1 convolution and upsample the images using bilinear interpolation, such that the output of the network has the same size as the input. Furthermore, they proposed a multi-scale data augmentation technique to increase the training size. Their framework follows the ground truth with a fixed kernel fashion. It is worth mentioning that the number of streams decreased compared to other works [8, 17].

Sam et al. [9] proposed a three-stream network with a switching module to decide which stream is better for the input images, named the switching CNN. Similarly to the HCNN, it uses patches from the image as input. Then, a stream classifier chooses the best stream to process the patch. Each independent stream is a CNN regressor with different receptive fields and fields of view, such that it focuses on a specific scale. The granularity of input patches is important since it is desirable that each patch has a uniform scale distribution. However, this method may create some more specialized streams than others due to unbalanced scale data.

To the best of our knowledge, no previous work based on the MSNN has analyzed the effects of the number of streams. Moreover, it is not common practice to evaluate various ground truth generation methods. In this paper, we aim to extend our previous work [2] by doing a comprehensive ablation study of the MSNN, specifically by studying the effects of the number of streams. We also evaluate three different methods for density map generation from ground truth, two of them being the most common methods used previously. In addition, we introduce a new method based on face detection and scale interpolation.

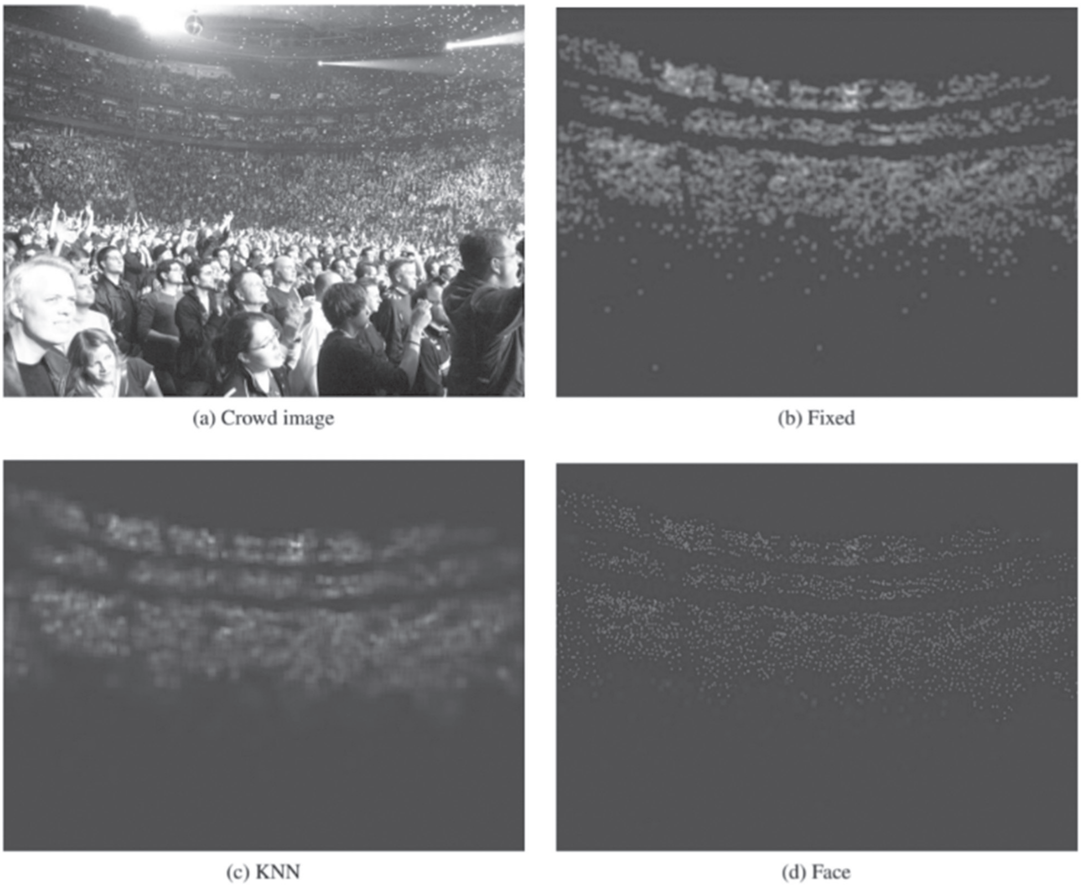

Fig. 2. Comparison of ground truth generation using three different methods.

2. RESEARCH METHOD

In this paper, we evaluate (i) ground truth construction from people's positions, and (ii) MSNN variations with different numbers of streams. These two stages are explained in the following sections.

2.1. GROUND TRUTH DENSITY MAP CONSTRUCTION

Crowd counting datasets provide images and positions (usually located in the heads) of each person. Based on these labels, a density map is created since it has been demonstrated [8–10, 17, 18] that such representation is simple, yet effective to predict the number of people present in the scenes when using deep networks. The purpose of density maps is to describe the density distribution of people in a given image (Figure 2).

Following the work developed by Zhang et al. [17], we assume that P people are located in the image. Given that the i-th person is at the pixel coordinate x_i, for the sake of simplicity, we use x_i to express both row and column positions. The image composed of pixel coordinates x is labeled with P heads through the accumulation of many impulse functions, such as:

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

where the $\delta(.)$ function is defined as:

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

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

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

Current literature uses two variations for the size of the Gaussian kernel σ: (i) a fixed value (for instance, σ=4), or (ii) a variable value for each person $\sigma_i = \beta d_i$, where d_i is defined as the mean distance to the *k* closest people and β is a regularization parameter.

Authors who employ variable σ [17] argue that ground truth created in this way simulates people's scale better such that the network can learn a scale-aware model. On the other hand, authors who employ fixed σ [2] argue that using k closest people introduces errors in poorly crowded regions with small people's scale. Moreover, they have shown equal or better results in similar setups. We consider that both fashions have valid arguments. Thus, we propose a new approach that combines these two methods, i.e., *σi* will be a fixed value for very crowded regions of the image and a variable value otherwise. Let $B=\{b_1, b_2, \dots, b_{|B|}\}$ be a list of bounding boxes of detected faces (we used an algorithm for tiny face detection [19] to find them). Each bounding box b_i is axis-aligned and defined by a centroid, height, and width. It is expected that the face detection method will not detect all people in the images. Thus, we use B to interpolate missing bounding boxes.

First, for each person *i* at pixel x_i, we must determine if it is inside a crowded region. We initially define an overlap region r_i, which is an axis-aligned rectangle centered at x_i. To count the number of people around person i, due to scale changes, we use a weighted average using the bounding boxes B, defined as:

$$r_i = \frac{\sum_{j=1}^{|B|} w_{ij}^{overlap} b_j}{\sum_{k=1}^{|B|} w_{ik}^{overlap}}, \tag{4}$$

where $w_{ij}^{overlap}$ is defined by the inverse of the l_2 distance between person *i* and centroid c_j of the bounding box b_j:

$$w_{ij}^{overlap} = \frac{1}{\| x_i - c_j \|_2}. \tag{5}$$

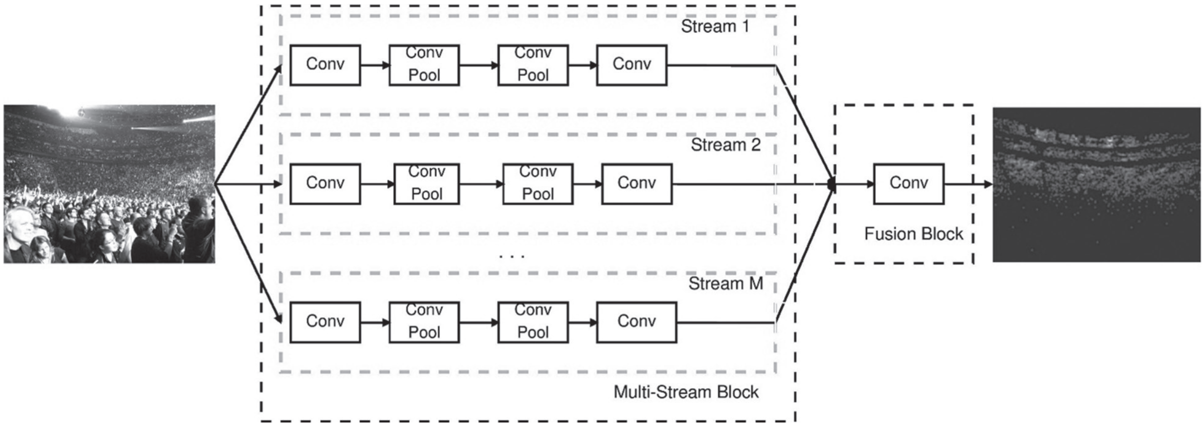

Fig. 3. Generalization of the multi-stream neural network for crowd counting. Each stream gathers information from different scales. Then, the fusion block merges the information to produce the estimated density map.

It is worth mentioning that Equation 4 weighs the bounding boxes. We use this form to express that the equation is applied independently for heights and widths of r_i and B.

Second, for every r_i, we estimate how many other bounding boxes overlap. If this number is greater than a threshold $T_{overlaps}$, then chances are that person *i* is at a heavily crowded region. Therefore, we use a fixed predefined size bounding box for it. Otherwise, x_i is not at a heavily crowded region and we can interpolate the bounding boxes by the following weight definition:

$$w_{ij}^{bb} = \frac{1}{\| x_i - c_j \|_2^{10}}. \qquad (6)$$

The only difference between equations 5 and 6 is the power of 10. In the case of Equation 5, our aim is to have an accurate estimation of the density context for each person *i*. On the other hand, with Equation 6, we quickly decrease the importance of distant elements as they create noise. Thus, bounding box d_i for person *i* is defined as:

$$d_i = \frac{\sum_{j=1}^{|B|} w_{ij}^{bb} b_j}{\sum_{k=1}^{|B|} w_{ik}^{bb}}. \qquad (7)$$

Finally, we use height and width of d_i for the Gaussian kernels $\sigma_i = d_i$.

Table 1. Architecture details for the evaluated MSNN versions.

	MSNN₁	*MSNN₂*	
	Stream 1	**Stream 1**	**Stream 2**
Multi-Stream Block	Conv 3x3x24	Conv 3x3x24	Conv 7x7x20
	Conv 3x3x48	Conv 3x3x48	Conv 5x5x40
	Pool 2x2	Pool 2x2	Pool 2x2
	Conv 3x3x24	Conv 3x3x24	Conv 5x5x20
	Pool 2x2	Pool 2x2	Pool 2x2
	Conv 3x3x12	Conv 3x3x12	Conv 5x5x10
Fusion Block	Conv 1x1x1	Conv 1x1x1	

	MSNN₃		
	Stream 1	**Stream 2**	**Stream 3**
Multi-Stream Block	Conv 3x3x24	Conv 7x7x20	Conv 9x9x20
	Conv 3x3x48	Conv 5x5x40	Conv 7x7x32
	Pool 2x2	Pool 2x2	Pool 2x2
	Conv 3x3x24	Conv 5x5x20	Conv 7x7x16
	Pool 2x2	Pool 2x2	Pool 2x2
	Conv 3x3x12	Conv 5x5x10	Conv 7x7x8
Fusion Block	Conv 1x1x1		

	MSNN4			
	Stream 1	**Stream 2**	**Stream 3**	**Stream 4**
Multi-Stream Block	Conv 3x3x24	Conv 7x7x20	Conv 9x9x20	Conv 11x11x12
	Conv 3x3x48	Conv 5x5x40	Conv 7x7x32	Conv 9x9x24
	Pool 2x2	Pool 2x2	Pool 2x2	Pool 2x2
	Conv 3x3x24	Conv 5x5x20	Conv 7x7x16	Conv 9x9x12
	Pool 2x2	Pool 2x2	Pool 2x2	Pool 2x2
	Conv 3x3x12	Conv 5x5x10	Conv 7x7x8	Conv 9x9x6
Fusion Block	Conv 1x1x1			

2.2. MULTI-STREAM NEURAL NETWORKS

Since the seminal work of Zhang et al. [17], diverse variations of MSNN have been proposed [2, 7–10]. In this paper, we generalize the original MSNN and evaluate the number of streams and their behavior with various ground truth generation methods.

A generalization of MSNN architecture is shown in Figure 3. The image is fed to the multi-stream block that has various parallel sequential convolutional layers named streams. Each stream learns how to detect people on a certain scale. Then, the fusion block combines feature maps of each stream to create the final estimation of the crowd.

We propose to evaluate MSNN using one, two, three and four streams. In order to create the MSNN with fewer streams, we iteratively remove streams with larger convolutional kernels and change the fusion block according to the new setup (details for each version are shown in Table 1). To train the network, we find optimal parameters θ^* (for the network) that minimize the error between the estimated and the ground truth density:

$$\theta = arg_\theta \min L(\theta), \qquad (8)$$

where the loss function is:

$$L(\theta) = \frac{1}{2|T|}\sum_{i=1}^{|T|} \| \mathcal{F}(X_i,\theta) - F_i \|_2^2, \qquad (9)$$

where *F* is the function approximated by 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 (Equation 3), |T| is the number of training images, and $\|.\|_2$ is the Euclidean distance.

2.3. TRAINING AND DATA AUGMENTATION

The loss function (Equation 9) is optimized via backpropagation and batch-based stochastic gradient descent. In contrast to the work described by Zhang et al.

[17], we do not train each stream independently. Due to two pooling layers, the size of the output is a quarter of the original size. Then we resize the ground truth images to compare them with the output. Parameter configuration for training in each crowd counting dataset, *UCF-CC-50* and ShanghaiTech, is reported in Table 2.

Table 2. Hyperparameters for training in each crowd counting dataset.

	UCF-CC-50	SHANGHAITECH
OPTIMIZER	Adam	Adam
LEARNING RATE	0.00001	0.00001
BATCH SIZE	32	64
EPOCHS	1000	200

We perform extensive data augmentation of the training dataset by creating images with a sliding window of 256×256 pixels and displacement of 70 pixels in each iteration. Further, we add Gaussian and bright/contrast noise. For the *UCF-CC-50* dataset, the augmentation process generates 10032, 10172, 9920, 9724 and 10248 images for five folds, respectively. For the *ShanghaiTech* dataset, the augmentation process generates 65341 and 140801 for part A and B, respectively. Unlike our previous work [2], we kept training and tuning of hyperparameters simple, as we intended to avoid the effects of these factors on the results and have a fair comparison of different network and density map generation setups.

3. RESULTS AND DISCUSSION

To evaluate the quality of ground truth generation methods and the MSNN, we use two challenging datasets, summarized as follows.

(1) The *ShanghaiTech* dataset was introduced by Zhan et al. [17]. It was created to encourage research in crowd counting using deep learning approaches. The dataset has 1198 annotated images with a total of 330,164 people with their head positions annotated. It is made up of 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. There are 300 images for training and 182 for testing. Part B is composed of 716 images taken from a busy street of the metropolitan area of Shanghai, containing between 123 and 578 people. There are 400 images for training and 316 for testing. Unlike other datasets [16], the crowd density varies significantly among the two subsets, making accurate crowd estimation more challenging.

(2) The *UCF-CC-50* dataset was introduced by Idrees et al. [16]. It is a very challenging dataset due to its extreme changes in scale and the number of people that varies from 94 to 4543. It contains 50 images extracted from the Internet with different aspect ratios and resolutions. Following the original standard protocol, we report results using a 5-fold cross-validation.

To evaluate the quantitative performance of the MSNN and ground truth methods, we compute mean absolute error (MAE) and root mean squared error (RMSE) metrics, defined as:

$$\mathrm{MAE} = \frac{1}{N}\sum_{i=1}^{N} |y_i - y_i'|, \tag{10}$$

$$\mathrm{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N} (y_i - y_i')^2}, \tag{11}$$

where *N* is the number of test samples, y_i is the ground truth count, and y_i' is the estimated count corresponding to the *i*-th sample.

Initially, we make a qualitative assessment of different density map generation methods. Then, we show and analyze results for comparing four different MSNN setups and three ground truth methods.

3.1. DENSITY MAPS

We compare the quality of maps generated using the fixed kernel method, denoted as *Fixed*, the variable kernel method, denoted as *K-NN*, and the proposed hybrid method, denoted as *Face*. Figure 2 shows generated maps for an image with large scale variations.

Consider people located far away from the camera with a tiny scale. In this case, the *Fixed* method seems to have a good representation; however, it is possible to observe that the size of the Gaussian kernel has a significant effect. If a huge value is used, the background will be labeled as people.

Analogously, consider the person on the lower left side of the image. It has a large scale, but the label generated by the *Fixed* method only considers its tiny part, then creating artifacts. The *K-NN* method introduces more artifacts for tiny scales due to the large Gaussian kernel size. This is because it is difficult to find proper hyperparameters *β* and *k* that are suitable for all crowd scenarios.

It is also possible that people with a large scale are almost invisible in the *K-NN* density maps. This same effect appears in the Face method. This effect occurs because the Gaussians are normalized to sum one before added to the density map; therefore, larger kernel sizes generate small values. In addition to this effect, the *Face* method is able to have a better estimation of people with tiny and medium scale.

To better understand the quality of the Face method, we analyze the interpolated face scales overlaid on the images (Figure 4). Our proposed ground truth methods are more general, so that decreases artifacts because we deal with dramatic changes in scales.

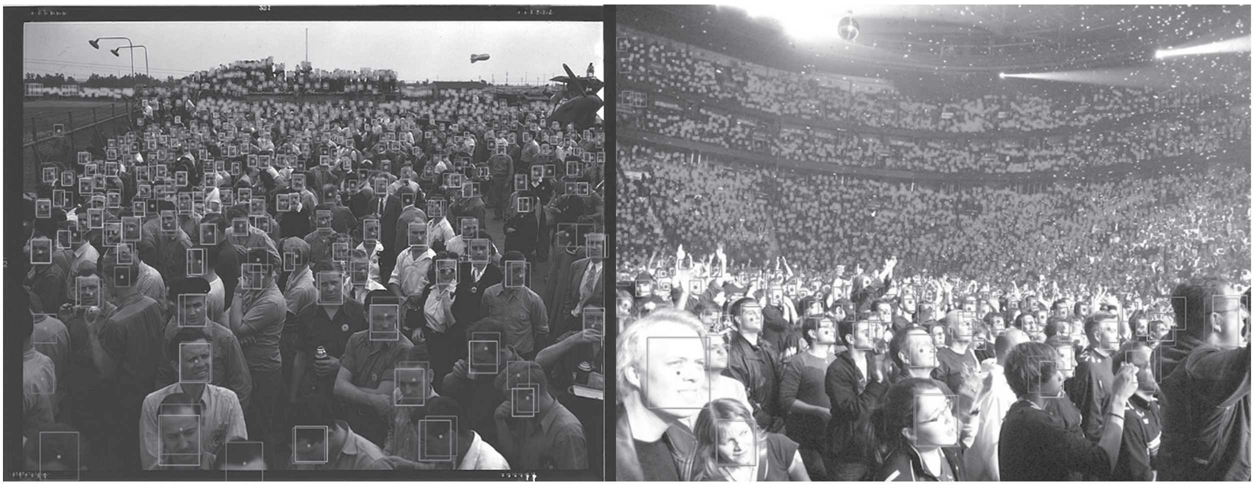

Figure 4. Results of face scale interpolation using the proposed algorithm. The bounding boxes detected by the tiny face detection algorithm [19] are shown in cyan, whereas the interpolated bounding boxes are shown in pink.

Table 3. Results for the proposed multi-stream network (lower scores are better).

DATASET	GROUND TRUTH METHODS							
UCF-CC-50	Fixed		KNN		Face		Average	
	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
MSNN$_1$	517.47	729.83	539.67	741.32	514.99	732.89	524.04	734.68
MSNN$_2$	429.54	652.48	421.58	605.41	385.72	588.69	412.28	615.53
MSNN$_3$	373.96	568.79	398.74	590.16	374.01	554.56	382.24	571.17
MSNN$_4$	409.31	619.35	368.13	614.51	379.61	556.27	385.68	596.71
AVERAGE	432.57	642.61	432.03	637.85	413.58	608.10		
DATASET	GROUND TRUTH METHODS							
SHANGHAITech	Fixed		KNN		Face		Average	
PART A	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
MSNN$_1$	193.37	279.95	191.64	275.23	187.56	273.02	190.86	276.07
MSNN$_2$	161.77	251.33	162.85	247.99	165.20	252.34	163.27	250.55
MSNN$_3$	161.61	245.39	170.45	252.16	160.80	246.40	164.29	247.98
MSNN$_4$	163.26	246.31	173.95	265.09	163.38	242.66	166.86	251.35
AVERAGE	170.00	255.74	174.72	260.12	169.23	253.60		
DATASET	GROUND TRUTH METHODS							
SHANGHAITech	Fixed		KNN		Face		Average	
PART B	MAE	RMSE	MAE	RMSE	MAE	RMSE	MAE	RMSE
MSNN$_1$	47.16	76.09	49.42	72.45	47.18	77.76	47.92	75.44
MSNN$_2$	40.62	67.13	41.72	66.56	40.75	67.80	41.03	67.16
MSNN$_3$	37.98	64.35	40.77	68.68	38.76	66.26	39.17	66.43
MSNN$_4$	36.89	63.18	39.96	65.06	34.54	57.73	37.13	61.99
AVERAGE	40.66	67.69	42.97	68.19	40.31	67.39		

3.2. CROWD COUNTING

Results for the *UCF-CC-50* dataset are reported in Table 3. It is possible to notice that for the *MSNN*$_1$ the best results for MAE were obtained with ground truth generated by the Face method and for RMSE the *Fixed* method achieved the best results with a difference of 3.06 in comparison to the *Face* method. For *MSNN*$_2$, the *Face* method achieved the best results for MAE and RMSE with a significant difference over the second-best method (for instance, 35.86 for MAE and 16.72 for RMSE). For *MSNN*$_3$, the *Fixed* method generated the best results for MAE; however, the *Face* method was really close with a tiny difference of 0.05. On the other hand, the *Face* method yielded the best results for RMSE with a difference of 14.23 in comparison with the second-best score. Unexpectedly, the *K-NN* method created the best results for *MSNN*$_4$ with MAE. However, the Face method obtained the best results for RMSE. To determine which ground truth method was the best for *UCF-CC-50*, we averaged the results over the four MSNN setups. It is possible to observe that the Face method was superior in terms of MAE and RMSE values.

From the relationship between the number of streams and result quality, the MAE and RMSE values decreased using up to three streams. However, using four streams, it decreased only for MAE and the *K-NN* method, whereas it grew in the remaining ones, even for the *Fixed* method, which increased by 35.35 and 50.56 for MAE and RMSE, respectively.

Considering the average between the ground truth generation methods, the best results in MAE and RMSE were obtained with $MSNN_3$. This may be related to the fact that all networks have the same simple fusion layer. For the MSNN with more streams, the fusion layer must learn more complex functions to map larger tensors to density maps. Overall, the best results for MAE were obtained with $MSNN_4$ and the *K-NN* method, whereas for RMSE, the best results were obtained with $MSNN_3$ and the *Face* method.

Results for the *ShanghaiTech Part A* dataset are reported in Table 3. For $MSNN_1$, the best results were obtained with the Face method for MAE and the Fixed method for RMSE. For $MSNN_2$, the best results for both MAE and RMSE were obtained with the *Face* method. In this case, compared with the second-best method, the difference was large, 35.86 and 16.72 for MAE and RMSE, respectively. For $MSNN_3$, the best results for MAE were obtained with the *Fixed* method; however, the *Face* method achieved a small difference of 0.05 and the best results for RMSE.

For $MSNN_4$, the best results for MAE were obtained with the *K-NN* method; it achieved a difference of 11.48 in relation to the second-best approach, that is, the *Face* method. Nonetheless, for RMSE, the *Face* method achieved the best results with a difference of 58.24 in comparison to the second best. Unlike the results for the *UCF-CC-50* dataset, the best number of streams for MAE is two, but the difference with three streams is 0.99. However, for RMSE, the results followed the same behavior as on the *UCF-CC-50* dataset: the results improved from one to three and were worse with four streams.

Considering the average, for the *ShanghaiTech Part B* dataset, the best results with one and two streams with MAE are obtained with the *Fixed* method; however, the *Face* method has a small difference of 0.02 and 013, respectively. For RMSE, however, the *KNN* gives the best results. For three streams, the *Fixed* method has the best results for MAE and RMSE, followed by *Face*. Up to this point, it seems that the Face method does not replicate the results shown in the previous datasets; however, with four streams, *Face* is clearly better and, considering the average performance between all network configurations, it is better but the *Fixed* method has similar results.

Considering the average between ground truth methods, we can see that the performance improves with the number of streams and overall the best results are obtained with four streams and the *Face* method.

4. CONCLUSIONS AND FUTURE WORK

In this paper, we evaluated the influence of the number of streams in multi-stream networks for the crowd counting problem. Furthermore, we evaluated the two most common strategies for generating ground truth and proposed a new hybrid method based on tiny face detection and scale interpolation.

Extensive experiments demonstrated that using three streams is better on average; however, there are some scenarios where using four streams overcomes other setups. Moreover, experiments show that the use of the proposed hybrid ground truth generation method is, in fact, better than other widely used schemes.

As directions for future work, we intend to evaluate the creation of synthetic data for the training purpose, the generation of higher definition density maps, and more accurate estimations.

ACKNOWLEDGEMENTS

The current archival periodical article is based on the conference presentation [2]. The authors are grateful to São Paulo Research Foundation (FAPESP grants #2014/12236-1 and #2016/19947-6) and the Brazilian National Council for Scientific and Technological Development (CNPq grants #305169/2015-7, #307425/2017-7 and #309330/2018-1) for their financial support.

5. REFERENCES

[1] X. Cao, Z. Wang, Y. Zhao, F. Su, "Scale Aggregation Network for Accurate and Efficient Crowd Counting", Proceedings of the 15th European Conference on Computer Vision, Munich, Germany, 8-14 September 2018, pp. 734-750.

[2] D. Ttito, R. Quispe, A. Ramírez Rivera, H. Pedrini, "Where are the People? A Multi-Stream Convolutional Neural Network for Crowd Counting via Density Map from Complex Images", Proceedings of the 26th International Conference on Systems, Signals and Image Processing, Osijek, Croatia, 5-7 June 2019, pp. 241–246.

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

[4] L. Zhang, M. Shi, Q. Chen, "Crowd Counting via Scale-Adaptive Convolutional Neural Network", Proceedings of the IEEE Winter Conference on Applications of Computer Vision, Lake Tahoe, NV, USA, 12-15 March 2018, pp. 1113–1121.

[5] M. Shami, S. Maqbool, H. Sajid, Y. Ayaz, S.-C. S. Cheung, "People Counting in Dense Crowd Images using Sparse Head Detections", IEEE Transactions on Circuits and Systems for Video Technology, Vol. 29, No. 9, 2018, pp. 2627–2636.

[6] C. Shang, H. Ai, Y. Yang, "Crowd Counting via Learning Perspective for Multi-Scale Multi-View Web Images", Frontiers of Computer Science, Vol. 13, No. 3, 2019, pp. 579–587.

[7] L. Boominathan, S. S. Kruthiventi, R. V. Babu, "CrowdNet: A Deep Convolutional Network for Dense Crowd Counting", Proceedings of the 24th ACM International Conference on Multimedia, Amsterdam, The Netherlands, 15-19 October 2016, pp. 640–644.

[8] D. Onoro-Rubio, R. J. López-Sastre, "Towards Perspective-Free Object Counting with Deep Learning", Proceedings of the 14th European Conference on Computer Vision, Amsterdam, The Netherlands, 11-14 October 2016, pp. 615–629.

[9] D. B. Sam, S. Surya, R. V. Babu, "Switching Convolutional Neural Network for Crowd Counting", Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21-26 July 2017, pp. 4031–4039.

[10] E. Walach, L. Wolf, "Learning to Count with CNN Boosting", Proceedings of the 14th European Conference on Computer Vision, Amsterdam, The Netherlands, 11-14 October 2016, pp. 660–676.

[11] W. Liu, M. Salzmann, P. Fua, "Context-Aware Crowd Counting", Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16-20 June 2019, pp. 5099–5108.

[12] M. Zhao, J. Zhang, C. Zhang, W. Zhang, "Leveraging Heterogeneous Auxiliary Tasks to Assist Crowd Counting", Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16-20 June 2019, pp. 12736–12745.

[13] Q. Wang, J. Gao, W. Lin, Y. Yuan, "Learning from Synthetic Data for Crowd Counting in the Wild", Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Long Beach, CA, USA, 16-20 June 2019, pp. 8198–8207.

[14] Z. Shen, Y. Xu, B. Ni, M. Wang, J. Hu, X. Yang, "Crowd Counting via Adversarial Cross-Scale Consistency Pursuit", Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18-23 June 2018, pp. 5245–5254.

[15] M. Xu, Z. Ge, X. Jiang, G. Cui, B. Zhou, C. Xu, "Depth Information Guided Crowd Counting for Complex Crowd Scenes", Pattern Recognition Letters, Vol. 125, 2019, pp. 563–569.

[16] H. Idrees, I. Saleemi, C. Seibert, M. Shah, "Multi-Source Multi-Scale Counting in Extremely Dense Crowd Images", Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23-28 June 2013, pp. 2547–2554.

[17] Y. Zhang, D. Zhou, S. Chen, S. Gao, Y. Ma, "Single-Image Crowd Counting via Multi-Column Convolutional Neural Network," Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27-30 June 2016, pp. 589–597.

[18] V. A. Sindagi, V. M. Patel, "CNN-based Cascaded Multi-Task Learning of High-Level Prior and Density Estimation for Crowd Counting", Proceedings of the 14th IEEE International Conference on Advanced Video and Signal Based Surveillance, Lecce, Italy, 29 August - 1 September 2017, pp. 1–6.

[19] P. Hu, D. Ramanan, "Finding Tiny Faces", Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21-26 July 2017, pp. 951–959.