Composition Loss for Counting, Density Map Estimation and Localization in Dense Crowds

Haroon Idrees[1], Muhmmad Tayyab[5], Kishan Athrey[5], Dong Zhang[2], Somaya Al-Maadeed[3], Nasir Rajpoot[4], and Mubarak Shah[5]

[1] Robotics Institute, Carnegie Mellon University
[2] NVIDIA Inc.
[3] Computer Science Department, Faculty of Engineering, Qatar University
[4] Department of Computer Science, University of Warwick, UK
[5] Center for Research in Computer Vision, University of Central Florida

Abstract. With multiple crowd gatherings of millions of people every year in events ranging from pilgrimages to protests, concerts to marathons, and festivals to funerals; visual crowd analysis is emerging as a new frontier in computer vision. In particular, counting in highly dense crowds is a challenging problem with far-reaching applicability in crowd safety and management, as well as gauging political significance of protests and demonstrations. In this paper, we propose a novel approach that simultaneously solves the problems of counting, density map estimation and localization of people in a given dense crowd image. Our formulation is based on an important observation that the three problems are inherently related to each other making the loss function for optimizing a deep CNN decomposable. Since localization requires high-quality images and annotations, we introduce UCF-QNRF dataset that overcomes the shortcomings of previous datasets, and contains 1.25 million humans manually marked with dot annotations. Finally, we present evaluation measures and comparison with recent deep CNN networks, including those developed specifically for crowd counting. Our approach significantly outperforms state-of-the-art on the new dataset, which is the most challenging dataset with the largest number of crowd annotations in the most diverse set of scenes.

Keywords: Crowd Counting · Localization · Convolution Neural Networks · Composition Loss

1 Introduction

Counting dense crowds is significant both from socio-political and safety perspective. At one end of the spectrum, there are large ritual gatherings such as during pilgrimages that typically have large crowds occurring in known and pre-defined locations. Although they generally have passive crowds coming together for peaceful purposes, disasters have known to occur, for instance, during Love Parade [9] and Hajj [1]. For active crowds, such as expressive mobs in demonstrations and protests, counting is important both from political and safety standpoint. It is very common for different sides to claim divergent numbers for crowd gathering, inclined towards their political standing on the concerned issue. Beyond subjectivity and preference for certain political or

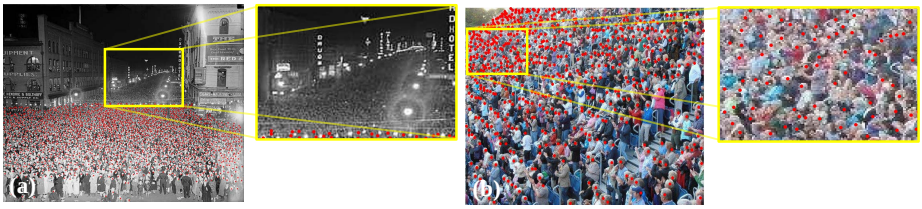

Fig. 1: This figure highlights the problems due to low resolution images from two existing dense crowd datasets: (a) shows a case where the annotations were not done on parts of the images as it is virtually impossible to distinguish heads of neighboring people, while (b) shows a case where some of the locations / counts are erroneous and therefore not suitable for localization. The UCF-QNRF dataset proposed in this paper overcomes such issues.

social outcomes, the disparate counting estimates from opposing parties have a basis in numerical cognition as well. In humans, the results on subitizing [21] suggest that once the number of observed objects increases beyond four, the brain switches from the exact Parallel Individuation System (PIS) to the inaccurate but scalable Approximate Number System (ANS) to count objects [11]. Thus, computer vision based crowd counting offers alternative fast and objective estimation of the number of people in such events. Furthermore, crowd counting is extendable to other domains, for instance, counting cells or bacteria from microscopic images [17,27], animal crowd estimates in wildlife sanctuaries [2], or estimating the number of vehicles at transportation hubs or traffic jams [19].

In this paper, we propose a novel approach to crowd counting, density map estimation and localization of people in a given crowd image. Our approach stems from the observation that these three problems are very interrelated - in fact, they can be decomposed with respect to each other. Counting provides an estimate of the number of people / objects without any information about their location. Density maps, which can be computed at multiple levels, provide weak information about location of each person. Localization does provide accurate location information, nevertheless, it is extremely difficult to estimate directly due to its very sparse nature. Therefore, we propose to estimate all three tasks simultaneously, while employing the fact that each is special case of another one. Density maps can be 'sharpened' till they approximate the localization map, whose integral should equal to the true count.

Furthermore, we introduce a new and the largest dataset to-date for training and evaluating **dense** crowd counting, density map estimation and localization methods, particularly suitable for training very deep Convolutional Neural Networks (CNNs). Though counting has traditionally been considered the primary focus of research, density map estimation and localization have significance and utility beyond counting. In particular, two applications are noteworthy: initialization / detection of people for tracking in dense crowds [13]; and rectifying counting errors from an automated computer vision algorithm. That is, a real user or analyst who desires to estimate the exact count for a real image *without any error*, the results of counting alone are insufficient. The

Dataset	Number Images	Number Annotations	Average Count	Maximum Count	Average Resolution	Average Density
UCF_CC_50 [12]	50	63,974	1279	4633	2101×2888	2.02×10^{-4}
WorldExpo'10 [29]	3980	225,216	56	334	576×720	1.36×10^{-4}
ShanghaiTech_A [30]	482	241,677	501	3139	589×868	9.33×10^{-4}
UCF-QNRF	**1535**	**1,251,642**	**815**	**12865**	$\mathbf{2013 \times 2902}$	$\mathbf{1.12 \times 10^{-4}}$

Table 1: Summary of statistics of different datasets. UCF_CC_50 (44MB); WorldExpo'10 (325MB); ShanghaiTech_A (67MB); and the proposed UCF-QNRF Dataset (4.33GB).

single number for an entire image makes it difficult to assess the error or the source of the error. However, the localization can provide an initial set of dot locations of the individuals, the user then can quickly go through the image and remove the false positives and add the false negatives. The count using such an approach will be much more accurate and the user can get 100% precise count for the query image. This is particularly important when the number of image samples are few, and reliable counts are desired.

Prior to 2013, much of the work in crowd counting focused on low-density scenarios. For instance, UCSD dataset [4] contains $2,000$ video frames with $49,885$ annotated persons. The dataset is low density and low resolution compared to many recent datasets, where train and test splits belong to a single scene. WorldExpo'10 dataset [29], contains 108 low-to-medium density scenes and overcomes the issue of diversity to some extent. UCF dataset [12] contains 50 different images with counts ranging between 96 and $4,633$ per image. Each image has a different resolution, camera angle, and crowd density. Although it was the first dataset for dense crowd images, it has problems with annotations (Figure 1) due to limited availability of high-resolution crowd images at the time. The ShanghaiTech crowd dataset [30] contains $1,198$ annotated images with a total of $330,165$ annotations. This dataset is divided into two parts: Part A contains 482 images and Part B with 716 images. The number of training images are 300 and 400 in both parts, respectively. Only the images in Part A contain high-density crowds, with 482 images and 250K annotations.

Table 1 summarizes the statistics of the multi-scene datasets for dense crowd counting. The proposed UCF-QNRF dataset has the most number of high-count crowd images and annotations, and a wider variety of scenes containing the most diverse set of viewpoints, densities and lighting variations. The resolution is large compared to WorldExpo'10 [29] and ShanghaiTech [30], as can be seen in Fig. 2(b). The average density, i.e., the number of people per pixel over all images is also the lowest, signifying high-quality large images. Lower per-pixel density is partly due to inclusion of background regions, where there are many high-density regions as well as zero-density regions. Part A of Shanghai dataset has high-count crowd images as well, however, they are severely cropped to contain crowds only. On the other hand, the new UCF-QNRF dataset contains buildings, vegetation, sky and roads as they are present in realistic scenarios captured in the wild. This makes this dataset more realistic as well as difficult. Similarly, Figure 2(a) shows the diversity in counts among the datasets. The distribution of proposed dataset is similar to UCF_CC_50 [12], however, the new dataset is 30 and

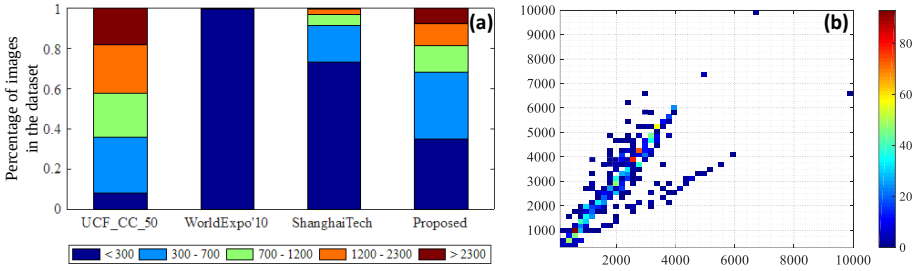

Fig. 2: (a) This graph shows the relative distribution of image counts among the four datasets. The proposed UCF-QNRF dataset has a fair number of images from all five count ranges. (b) This graph shows a 2D histogram of image resolution for all the images in the new dataset. The x-axis shows the number of rows, while y-axis is the number of columns. Each bin (500×500 pixels) is color-coded with the number of images that have the corresponding resolution.

20 times larger in terms of number of images and annotations, respectively, compared to UCF_CC_50 [12]. We hope the new dataset will significantly increase research activity in visual crowd analysis and will pave way for building deployable practical counting and localization systems for dense crowds.

The rest of the paper is organized as follows. In Sec. 2 we review related work, and present the proposed approach for simultaneous crowd counting, density map estimation and localization in Sec. 3. The process for collection and annotation of the UCF-QNRF dataset is covered in Sec. 4, while the three tasks and evaluation measures are motivated in Sec. 5. The experimental evaluation and comparison are presented in Sec. 6. We conclude with suggestions for future work in Sec. 7.

2 Related Work

Crowd counting is active an area of research with works tackling the three aspects of the problem: counting-by-regression [23], [17], [12], [4], [28], density map estimation [17], [7], [29], [20], [30] and localization [18], [22].

Earlier regression-based approaches mapped global image features or a combination of local patch features to obtain counts [15], [5], [12], [6]. Since these methods only produce counts, they cannot be used for density map estimation or localization. The features were hand-crafted and in some cases multiple features were used [4], [12] to handle low resolution, perspective distortion and severe occlusion. On the other hand, CNNs inherently learn multiple feature maps automatically, and therefore are now being extensively used for crowd counting and density map estimation.

CNN based approaches for crowd counting include [16], [29], [30], [19], [2]. Zhang *et al.* [29] train a CNN alternatively to predict density map and count in a patch, and then average the density map for all the overlapping patches to obtain density map for the entire image. Lebanoff and Idrees [16] introduce a normalized variant of the Euclidean loss function in a deep network to achieve consistent counting performance

across all densities. The authors in [30] use three column CNN, each with different filter sizes to capture responses at different scales. The count for the image is obtained by summing over the predicted density map. Sindagi and Patel [26] presented a CNN-based approach that incorporates global and local contextual information in an image to generate density maps. The global and local contexts are obtained by learning to classify the input image patches into various density levels, later fused with the output of a multi-column CNN to obtain the final density map. Similarly, in the approach by Sam *et al*. [24], image patches are relayed to the appropriate CNN using a switching mechanism learnt during training. The independent CNN regressors are designed to have different receptive fields while the switch classifier is trained to relay the crowd scene patch to the best CNN regressor.

For localization in crowded scenes, Rodriguez *et al*. [22] use density map as a regularizer during the detection. They optimize an objective function that prefers density map generated on detected locations to be similar to predicted density map [17]. This results in both better precision and recall. The density map is generated by placing a Gaussian kernel at the location of each detection. Zheng *et al*. [18] first obtain density map using sliding window over the image through [17], and then use integer programming to localize objects on the density maps. Similarly, in the domain of medical imaging, Sirinukunwattana *et al*. [27] introduced spatially-constrained CNNs for detection and classification of cancer nuclei. In this paper, we present results and analysis for simultaneous crowd counting, density map estimation, and localization using Composition Loss on the proposed UCF-QNRF dataset.

3 Deep CNN with Composition Loss

In this section, we present the motivation for decomposing the loss of three interrelated problems of counting, density map estimation and localization, followed by details about the deep Convolutional Neural Network which can enable training and estimation of the three tasks simultaneously.

3.1 Composition Loss

Let $\mathbf{x} = [x, y]$ denote a pixel location in a given image, and N be the number of people annotated with $\{\mathbf{x}_i : i = 1, 2, \dots N\}$ as their respective locations. Dense crowds typically depict heads of people as they are the only parts least occluded and mostly visible. In localization maps, only a single pixel is activated, i.e., set to 1 per head, while all other pixels are set to 0. This makes localization maps extremely sparse and therefore difficult to train and estimate. We observe that successive computation of 'sharper' density maps which are relatively easier to train can aid in localization as well. Moreover, all three tasks should influence count, which is the integral over density or localization map. We use the Gaussian Kernel and adapt it for our problem of simultaneous solution for the three tasks.

Due to perspective effect and possibly variable density of the crowd, a single value of bandwidth, σ, cannot be used for the Gaussian kernel, as it might lead to well-defined separation between people close to the camera or in regions of low density, while excess

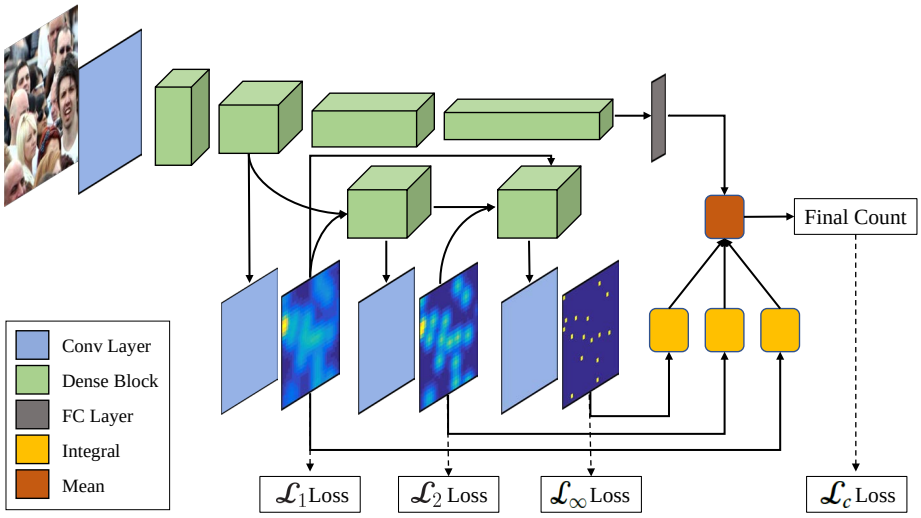

Fig. 3: The figure shows the proposed architecture for estimating count, density and localization maps simultaneously for a given patch in an image. At the top is the base DenseNet which regresses only the counts. The proposed Composition Loss is implemented through multiple dense blocks after branching off the base network. We also test the effect of additional constraint on the density and localization maps (shown with amber and orange blocks) such that the count after integral in each should also be consistent with the groundtruth count.

blurring in other regions. Many images of dense crowds depict crowds in their entirety, making automatic perspective rectification difficult. Thus, we propose to define σ_i for each person i as the minimum of the ℓ_2 distance to its nearest neighbor in spatial domain of the image or some maximum threshold, τ. This ensures that the location information of each person is preserved precisely irrespective of default kernel bandwidth, τ. Thus, the adaptive Gaussian kernel is given by,

$$D(\mathbf{x},\, f(\cdot)) = \sum_{i=1}^{N} \frac{1}{\sqrt{2\pi} f(\sigma_i)} \exp\left(-\frac{(x - x_i)^2 + (y - y_i)^2}{2 f(\sigma_i)^2} \right), \tag{1}$$

where the function f is used to produce a successive set of 'sharper' density maps. We define $f_k(\sigma) = \sigma^{1/k}$. Thus, $D_k = D(\mathbf{x},\, f_k(\cdot))$. As can be seen when $k = 1$, D_k is a very smoothed-out density map using nearest-neighbor dependent bandwidth and τ, whereas as $k \longrightarrow \infty$, D_k approaches the binary localization map with a Dirac Delta function placed at each annotated pixel. Since each pixel has a unit area, the localization map assumes a unit value at the annotated location. For our experiments we used three density levels with last one being the localization map. It is also interesting to note that the various connections between density levels and base CNN also serve to provide intermediate supervision which aid in training the filters of base CNN towards counting and density estimation early on in the network.

Hypothetically, since integral over each estimated $\hat{D}_k$ yields a count for that density level, the final count can be obtained by taking the mean of counts from the density and localization maps as well as regression output from base CNN. This has two potential advantages: 1) the final count relies on multiple sources - each capturing count at a different scale. 2) During training the mean of four counts should equal the true count, which implicitly enforces an additional constraint that $\hat{D}_k$ should not only capture the density and localization information, but that each of their counts should also sum to the groundtruth count. For training, the loss function of density and localization maps is the mean square error between the predicted and ground truth maps, i.e. $\boldsymbol{\mathcal{L}}_k = \text{MSE}(\hat{D}_k, D_k)$, where $k = 1, 2,$ and ∞, and regression loss, $\boldsymbol{\mathcal{L}}_c$, is Euclidean loss between predicted and groundtruth counts, while the final loss is defined as the weighted mean all four losses.

3.2 DenseNet with Composition Loss

We use DenseNet [10] as our base network. It consists of 4 Dense blocks where each block has a number of consecutive 1×1 and 3×3 convolutional layers. Each dense block (except for the last one) is followed by a Transition layer, which reduces the number of feature-maps by applying 1×1 convolutions followed by 2×2 average pooling with stride 2. In our experiments we used DenseNet-201 architecture. It has $\{6, 12, 48, 32\}$ sets of 1×1 and 3×3 convolutional layers in the four dense blocks, respectively.

Layer	Output Size	Filters
	$512 \times 28 \times 28$	
Density Level 1	$1 \times 28 \times 28$	1×1 conv
Density Level 2	$641 \times 28 \times 28$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 4$
	$1 \times 28 \times 28$	1×1 conv
Density Level ∞	$771 \times 28 \times 28$	$\begin{bmatrix} 1 \times 1 \text{ conv} \\ 3 \times 3 \text{ conv} \end{bmatrix} \times 4$
	$1 \times 28 \times 28$	1×1 conv

Table 2: This table shows the filter dimensions and output of the three density layer blocks appended to the network in Fig. 3.

For density map estimation and localization, we branch out from DenseBlock2 and feed it to our Density Network (see Table 2). The density network introduces 2 new dense blocks and three 1×1 convolutional layers. Each dense block has features computed at the previous step, concatenated with all the density levels predicted thus far as input, and learns features aimed at computing the current density / localization map. We used 1×1 convolutions to get the output density map from these features. Density Level 1 is computed directly from DenseBlock2 features.

We used Adam solver with a step learning rate in all our experiments. We used 0.001 as initial learning rate and reduce the learning rate by a factor of 2 after every 20 epochs. We trained the entire network for 70 epoch with a batch size of 16.

4 The UCF-QNRF Dataset

Dataset Collection. The images for the dataset were collected from three sources: Flickr, Web Search and the Hajj footage. The Hajj images were carefully selected so

that there are multiple images that capture different locations, viewpoints, perspective effects and times of the day. For Flickr and Web Search, we manually generated the following queries: CROWD, HAJJ, SPECTATOR CROWD, PILGRIMAGE, PROTEST CROWD and CONCERT CROWD. These queries were then passed onto the Flickr and Google Image Search APIs. We selected desired number of images for each query to be 2000 for Flickr and 200 for Google Image Search. The search sorted all the results by RELEVANCE incorporating both titles and tags, and for Flickr we also ensured that only those images were downloaded for which original resolutions were permitted to be downloaded (through the URL_O specifier). The static links to all the images were extracted and saved for all the query terms, which were then downloaded using the respective APIs. The images were also checked for duplicates by computing image similarities followed by manual verification and discarding of duplicates.

Initial Pruning. The initial set of images were then manually checked for desirability. Many of the images were pruned due to one or more of the following reasons:

- Scenes that did not depict crowds at all or low-density crowds
- Objects or visualizations of objects other than humans
- Motion blur or low resolution
- Very high perspective effect that is camera height is similar to average human height
- Images with watermarks or those where text occupied more than 10% of the image

In high-density crowd images, it is mostly the heads that are visible. However, people who appear far away from the camera become indistinguishable beyond a certain distance, which depends on crowd density, lighting as well as resolution of the camera sensor. During pruning, we kept those images where the heads were separable visually. Such images were annotated with the others, however, they were cropped afterwards to ensure that regions with problematic annotations or those with none at all due to difficulty in recognizing human heads were discarded.

We performed the entire annotation process in two stages. In the first stage, unannotated images were given to the *annotators*, while in the second stage, the images were given to *verifiers* who corrected any mistakes or errors in annotations. There were 14 annotators and 4 verifiers, who clocked $1,300$ and 200 hours respectively. In total, the entire procedure involved $2,000$ human-hours spent through to its completion.

Statistics. The dataset has $1,535$ jpeg images with $1,251,642$ annotations. The train and test sets were created by sorting the images with respect to absolute counts, and selecting every 5th image into the test set. Thus, the training and test set consist of 1201 and 334 images, respectively. The distribution of images from [Flickr, Web, Hajj] for the train and test are [1078, 84, 39] and [306, 21, 7], respectively. In the dataset, the minimum and maximum counts are 49 and $12,865$, respectively, whereas the median and mean counts are 425 and 815.4, respectively.

5 Definition and Quantification of Tasks

In this section, we define the three tasks and the associated quantification measures.

Counting: The first task involves estimation of count for a crowd image i, given by $\mathbf{c}_i$. Although this measure does not give any information about location or distribution of people in the image, this is still very useful for many applications, for instance, estimating size of an entire crowd spanning several square kilometers or miles. For the application of counting large crowds, Jacob's Method [14] due to Herbert Jacob is typically employed which involves dividing the area $\mathbf{A}$ into smaller sections, finding the average number of people or density $\mathbf{d}$ in each section, computing the mean density $\bar{\mathbf{d}}$ and extrapolating the results to entire region. However, with automated crowd counting, it is now possible to obtain counts and density for multiple images at different locations, thereby, permitting the more accurate integration of density over entire area covered by crowd. Moreover, counting through multiple aerial images requires cartographic tools to map the images onto the earth to compute ground areas. The density here is defined as the number of people in the image divided by ground area covered by the image. We propose to use the same evaluation measures as used in literature for this task: the Mean Absolute Error (C-MAE), Mean Squared Error (C-MSE) with the addition of Normalized Absolute Error (C-NAE).

Density Map Estimation amounts to computing per-pixel density at each location in the image, thus preserving spatial information about distribution of people. This is particularly relevant for safety and surveillance, since very high density at a particular location in the scene can be catastrophic [1]. This is different from counting since an image can have counts within safe limits, while containing regions that have very high density. This can happen due to the presence of empty regions in the image, such as walls and sky for mounted cameras; and roads, vehicles, buildings and forestation in aerial cameras. The metrics for evaluating density map estimation are similar to counting, except that they are per-pixel, i.e., the per-pixel Mean Absolute Error (DM-MAE) and Mean Squared Error (DM-MSE). Finally, we also propose to compute the 2D Histogram Intersection (DM-HI) distance after normalizing both the groundtruth and estimated density maps. This discards the effect of absolute counts and emphasizes the error in distribution of density compared to the groundtruth.

Localization: The ideal approach to crowd counting would be to detect all the people in an image and then count the number of detections. But since dense crowd images contain severe occlusions among individuals and fewer pixels per person for those away from the camera, this is not a feasible solution. This is why, most approaches to crowd counting bypass explicit detection and perform direct regression on input images. However, for many applications, the precise location of individuals is needed, for instance, to initialize a tracking algorithm in very high-density crowd videos.

To quantify the localization error, estimated locations are associated with the ground truth locations through 1-1 matching using greedy association, followed by computation of Precision and Recall at various distance thresholds ($1, 2, 3, \ldots, 100$ pixels). The overall performance of the localization task is then computed through area under the Precision-Recall curve, L-AUC.

6 Experiments

Next, we present the results of experiments for the three tasks defined in Section 5.

Fig. 4: This figure shows pairs of images where the left image in the pair has the lowest counting error while the right image has the highest counting error with respect to the four components of the Composition Loss.

6.1 Counting

For counting, we evaluated the new UCF-QNRF dataset using the proposed method which estimates counts, density maps and locations of people simultaneously with several state-of-the-art deep neural networks [3], [8], [10] as well as those specifically developed for crowd counting [30], [25], [24]. To train the networks, we extracted patches of sizes $448, 224$ and 112 pixels at random locations from each training image. While deciding on image locations to extract patch from, we assigned higher probability of selection to image regions with higher count. We used mean square error of counts as the loss function. At test time, we divide the image into a grid of 224×224 pixel cells - zero-padding the image for dimensions not divisible by 224 - and evaluate each cell using the trained network. Final image count is given by aggregating the counts in all cells. Table 3 summarizes the results which shows the proposed network significantly outperforms the competing deep CNNs and crowd counting approaches. In Figure 4, we show the images with the lowest and highest error in the test set, for counts obtained through different components of the Composition Loss.

Method	C-MAE	C-NAE	C-MSE
Idrees *et al*. [12]*	315	0.63	508
MCNN [30]	277	0.55	426
Encoder-Decoder [3]	270	0.56	478
CMTL [25]	252	0.54	514
SwitchCNN [24]	228	0.44	445
Resnet101 [8]*	190	0.50	277
Densenet201 [10]*	163	0.40	226
Proposed	**132**	**0.26**	**191**

Table 3: We show counting results obtained using state-of-the-art methods in comparison with the proposed approach. Methods with '*' regress counts without computing density maps.

6.2 Density Map Estimation

For density map estimation, we describe and compare the proposed approach with several methods that directly regress crowd density during training. Among the deep learning methods, MCNN [30] consists of three columns of convolution networks with different filter sizes to capture different head sizes and combines the output of all the columns to make a final density estimate. SwitchCNN [24] uses a similar three column network; however, it also employs a switching network that decides which column should exclusively handle the input patch. CMTL [25] employs a multi-task network that computes a high level prior over the image patch (crowd count classification) and density estimation. These networks are specifically designed for crowd density estimation and their results are reported in first three rows of Table 4. The results of proposed approach are shown in the bottom row of Table 4. The proposed approach outperforms existing approaches by an order of magnitude.

Method	DM-MAE	DM-MSE	DM-HI
MCNN [30]	0.006670	0.0223	0.5354
SwitchCNN [24]	0.005673	0.0263	0.5301
CMTL [25]	0.005932	0.0244	0.5024
Proposed	**0.00044**	**0.0017**	**0.9131**

Table 4: Results for Density map estimation: We show results on Histogram intersection (HI), obtained using existing state-of-the-art methods compared to the proposed approach.

6.3 Localization

For the localization task, we adopt the same network configurations used for density map estimation to perform localization. To get the accurate head locations, we post-process the outputs by finding the local peaks / maximums based on a threshold, also known as non-maximal suppression. Once the peaks are found, we match the predicted location with the ground truth location using 1-1 matching, and compute precision and recall. We use different distance thresholds as the pixel distance, i.e., if the detection is within the a particular distance threshold of the groundtruth, it is treated as True Positive, otherwise it is a False Positive. Similarly, if there is no detection within a groundtruth location, it becomes a False Negative.

The results of localization are reported in Table 5. This table shows that DenseNet [10] and Encoder-Decoder [3] outperform ResNet [8] and MCNN [30], while the proposed approach is superior to all the compared methods. The performance on the localization task is dependent on post-processing, which can alter results. Therefore, finding optimal strategy for localization from neural network output or incorporating the post-processing into the network is an important direction for future research. We also show some qualitative results of localization in Figure 5. The red dots represent the groundtruth while yellow circles are the locations estimated by the our approach.

6.4 Ablation Study

We performed an ablation study to validate the efficacy of composition loss introduced in this paper, as well as various choices in designing the network. These results are

Method	Av. Precision	Av. Recall	L-AUC
MCNN [30]	59.93%	63.50%	0.591
ResNet74 [8]	61.60%	66.90%	0.612
DenseNet63 [10]	70.19%	58.10%	0.637
Encoder-Decoder [3]	71.80%	62.98%	0.670
Proposed	**75.8%**	**59.75%**	**0.714**

Table 5: This table shows the localization results averaged over four distance thresholds for different methods. We show Average Precision, Average Recall and AUC metrics.

Fig. 5: Two examples of localization using the proposed approach. Ground truth is depicted in red and predicted locations after threshold are shown in yellow.

shown in Table 6. Next, we describe and provide details for the experiment corresponding to each row in the table.

BaseNetwork: This row shows the results with base network of our choice, which is DenseNet201. A fully-connected layer is appended to the last layer of the network followed by a single neuron which outputs the count. The input patch size is 224 $\times$ 224.

DenseBlock4: This experiment studies the effect of connecting the Density Network (Table 2) containing the different density levels with DenseBlock4 of the base DenseNet instead of DenseBlock2. Since DenseBlock4 outputs feature maps of size 7 $\times$ 7, we therefore used deconvolution layer with stride 4 to upsample the features before feeding in to our Density Network.

DenseBlock3: This experiment is similar to **DenseBlock4**, except that we connect our Density Network to Denseblock3 of the base network. DenseBlock3 outputs feature maps which are 14 $\times$ 14 in spatial dimensions, whereas we intend to predict density maps of spatial dimension 28 $\times$ 28, so we upsample the feature maps by using deconvolution layer before feeding them to the proposed Density Network.

D_1 **only**: This row represents the results if we use Density Level 1 only in the Density Network along with regression of counts in the base network. The results are much

Experiment	Count			D_∞			D_2			D_1		
	MAE	MSE	NAE	MAE	MSE	NAE	MAE	MSE	NAE	MAE	MSE	NAE
BaseNetwork	163	227	0.395	-	-	-	-	-	-	-	-	-
DenseBlock4	148	265	0.385	382	765	0.956	879	1235	3.892	2015	4529	4.295
DenseBlock3	144	236	0.363	295	687	0.721	805	1159	3.256	1273	2936	3.982
D_1 only	141	233	0.261	-	-	-	-	-	-	1706	2496	5.677
D_1 & D_2 only	137	208	0.251	-	-	-	691	1058	2.459	1887	3541	6.850
Concatenate	139	223	0.264	258	508	0.634	718	1096	3.570	1910	4983	6.574
Mean	150	341	0.271	405	710	1.135	1015	2099	2.916	1151	3170	3.283
Proposed	132	191	0.258	236	408	0.506	682	922	2.027	1629	3600	4.396

Table 6: This table shows the results of ablation study. D_∞ corresponds to the results of counting using localization map estimation, while D_2 and D_1 represent results from the two density maps, respectively.

worse compared to the proposed method which uses multiple levels in the Composition Loss.

D_1 **and** D_2 **only**: Similar to D_1 **only**, this row represents the results if we use Density Levels 1 and 2 and do not use the D_∞ in the Density Network. Incorporation of another density level improves results slightly in contrast to a single density level.

Concatenate: Here, we take the sum of the two density and one localization map to obtain 3 counts. We then concatenate these counts to the output of fully-connected layer of the base network to predict count from the single neuron. Thus, we leave to the optimization algorithm to find appropriate weights for these 3 values along with the rest of 1920 features of the fully-connected layer.

Mean: We also tested the effect of using equal weights for counts obtained from the base network and three density levels. We take sum of each density / localization map and take the mean of 4 values (2 density map sums, one localization sum, and one count from base network). We treat this mean value as final count output - both during training and testing. Thus, this imposes the constraint that not only the density and localization map correctly predict the location of people, but also their counts should be consistent with groundtruth counts irrespective of predicted locations.

Proposed: In this experiment, the Density Network is connected with the DenseBlock2 of base network, however, the Density Network simply outputs two density and one localization maps, none of which are connected to count output (see Figure 3).

In summary, these results show that the Density Network contributes significantly to performance on the three tasks. It is better to branch out from the middle layers of the base network, nevertheless the idea of multiple connections back and forth from the base network and Density Network is an interesting direction for further research. Furthermore, enforcing counts from all sources to be equal to the groundtruth count slightly worsens the counting performance. Nevertheless, it does help in estimating better density and localization maps. Finally, the decrease in error rates from the right to left in Table 6 highlights the positive influence of the proposed Composition Loss.

7 Conclusion

This paper introduced a novel method to estimate counts, density maps and localization in dense crowd images. We showed that these three problems are interrelated, and can be decomposed with respect to each other through Composition Loss which can then be used to train a neural network. We solved the three tasks simultaneously with the counting performance benefiting from the density map estimation and localization as well. We also proposed the large-scale UCF-QNRF dataset for dense crowds suitable for the three tasks described in the paper. We provided details of the process of dataset collection and annotation, where we ensured that only high-resolution images were curated for the dataset. Finally, we presented extensive set of experiments using several recent deep architectures, and show how the proposed approach is able to achieve good performance through detailed ablation study. We hope the new dataset will prove useful for this type of research, with applications in safety and surveillance, design and expansion of public infrastructures, and gauging political significance of various crowd events.

Acknowledgment: This work was made possible in part by NPRP grant number NPRP 7-1711-1-312 from the Qatar National Research Fund (a member of Qatar Foundation). The statements made herein are solely the responsibility of the authors.

References

1. A history of hajj tragedies. The Guardian (2006), `http://www.guardian.co.uk/world/2006/jan/13/saudiarabia`. [Accessed: July 1, 2013]
2. Arteta, C., Lempitsky, V., Zisserman, A.: Counting in the wild. In: European Conference on Computer Vision. pp. 483–498. Springer (2016)
3. Badrinarayanan, V., Kendall, A., Cipolla, R.: Segnet: A deep convolutional encoder-decoder architecture for image segmentation. arXiv preprint arXiv:1511.00561 (2015)
4. Chan, A., Liang, Z., Vasconcelos, N.: Privacy preserving crowd monitoring: Counting people without people models or tracking. In: CVPR (2008)
5. Chen, K., Loy, C., Gong, S., Xiang, T.: Feature mining for localised crowd counting. In: BMVC (2012)
6. Chen, K., Gong, S., Xiang, T., Change Loy, C.: Cumulative attribute space for age and crowd density estimation. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 2467–2474 (2013)
7. Fiaschi, L., Köthe, U., Nair, R., Hamprecht, F.A.: Learning to count with regression forest and structured labels. In: Pattern Recognition (ICPR), 2012 21st International Conference on. IEEE (2012)
8. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 770–778 (2016)
9. Helbing, D., Mukerji, P.: Crowd disasters as systemic failures: analysis of the love parade disaster. EPJ Data Science **1**(1), 1–40 (2012)
10. Huang, G., Liu, Z., Weinberger, K.Q., van der Maaten, L.: Densely connected convolutional networks. arXiv preprint arXiv:1608.06993 (2016)
11. Hyde, D.C.: Two systems of non-symbolic numerical cognition. Frontiers in human neuroscience **5** (2011)

12. Idrees, H., Saleemi, I., Seibert, C., Shah, M.: Multi-source multi-scale counting in extremely dense crowd images. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2013)
13. Idrees, H., Warner, N., Shah, M.: Tracking in dense crowds using prominence and neighborhood motion concurrence. Image and Vision Computing **32**(1), 14–26 (2014)
14. Jacobs, H.: To count a crowd. Columbia Journalism Review **6**, 36–40 (1967)
15. Kong, D., Gray, D., Tao, H.: A viewpoint invariant approach for crowd counting. In: Pattern Recognition, 2006. ICPR 2006. 18th International Conference on. vol. 3, pp. 1187–1190. IEEE (2006)
16. Lebanoff, L., Idrees, H.: Counting in dense crowds using deep learning (2015)
17. Lempitsky, V., Zisserman, A.: Learning to count objects in images. In: NIPS (2010)
18. Ma, Z., Yu, L., Chan, A.B.: Small instance detection by integer programming on object density maps. In: CVPR (2015)
19. Onoro-Rubio, D., López-Sastre, R.J.: Towards perspective-free object counting with deep learning. In: European Conference on Computer Vision. Springer (2016)
20. Pham, V.Q., Kozakaya, T., Yamaguchi, O., Okada, R.: Count forest: Co-voting uncertain number of targets using random forest for crowd density estimation. In: Proceedings of the IEEE International Conference on Computer Vision (2015)
21. Piazza, M., Mechelli, A., Butterworth, B., Price, C.J.: Are subitizing and counting implemented as separate or functionally overlapping processes? Neuroimage **15**(2), 435–446 (2002)
22. Rodriguez, M., Sivic, J., Laptev, I., Audibert, J.Y.: Density-aware person detection and tracking in crowds. In: ICCV (2011)
23. Ryan, D., Denman, S., Fookes, C., Sridharan, S.: Crowd counting using multiple local features. In: Digital Image Computing: Techniques and Applications, 2009. DICTA'09. (2009)
24. Sam, D.B., Surya, S., Babu, R.V.: Switching convolutional neural network for crowd counting. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. vol. 1 number=3, pages=6, year=2017
25. Sindagi, V.A., Patel, V.M.: Cnn-based cascaded multi-task learning of high-level prior and density estimation for crowd counting. In: Advanced Video and Signal Based Surveillance (AVSS), 2017 14th IEEE International Conference on. pp. 1–6. IEEE (2017)
26. Sindagi, V.A., Patel, V.M.: Generating high-quality crowd density maps using contextual pyramid cnns. In: IEEE International Conference on Computer Vision (2017)
27. Sirinukunwattana, K., Raza, S.E.A., Tsang, Y.W., Snead, D.R., Cree, I.A., Rajpoot, N.M.: Locality sensitive deep learning for detection and classification of nuclei in routine colon cancer histology images. IEEE transactions on medical imaging **35**(5), 1196–1206 (2016)
28. Wang, C., Zhang, H., Yang, L., Liu, S., Cao, X.: Deep people counting in extremely dense crowds. In: Proceedings of the 23rd ACM international conference on Multimedia. ACM (2015)
29. Zhang, C., Li, H., Wang, X., Yang, X.: Cross-scene crowd counting via deep convolutional neural networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2015)
30. Zhang, Y., Zhou, D., Chen, S., Gao, S., Ma, Y.: Single-image crowd counting via multi-column convolutional neural network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (2016)